

www.konferenciaonline.org.ua

**Міжнародна наукова
інтернет-конференція**

**Інформаційне суспільство:
технологічні, економічні
та технічні аспекти становлення**

(випуск 62)

ISSN 2522-932X

12 жовтня 2021 р.

Тернопіль
2021

УДК 001 (063)
ББК 72я431

Міжнародна наукова інтернет-конференція "Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення (випуск 62)" / Збірник тез доповідей: випуск 62 (м. Тернопіль, 12 жовтня 2021 р.). –Тернопіль. – 2021. – 99 с.

Збірник тез доповідей підготовлено за матеріалами Міжнародної наукової інтернет-конференції (випуск 62) від 12 жовтня 2021 р.

Оргкомітет:

Патряк Олександра Тарасівна, кандидат економічних наук,
Західноукраїнський національний університет;
Огінська Анастасія Юріївна, кандидат економічних наук, Think Global
Ternopil;
Ященко Василь Миколайович, кандидат педагогічних наук;
Русенко Святослав Ярославович, здобувач Університету митної справи та
фінансів.

Тексти матеріалів конференції подаються в авторській редакції.
Відповідальність за точність, достовірність і зміст поданих матеріалів
несуть автори.

Всі права захищено. У разі будь-якого використання опублікованих
матеріалів посилання на джерело є обов'язковим.

Наша адреса: Оргкомітет МНІК "Конференція онлайн"
а/с 797, м. Тернопіль 46005
тел. моб. 068 366 0 525
e-mail: inetkonf@ukr.net

URL Інтернет-конференції: <http://www.konferenciaonline.org.ua/>

ISSN 2522-932X

Секція 1. Інформаційні системи і технології

*Vash Y., postgraduate, department assistant, Department of software systems,
Uzhhorod National University, Uzhhorod;*

*Rol M., assistant, Department of software systems,
Uzhhorod National University, Uzhhorod*

USING OF THE UTILITY SYSTEMS WITH BEHAVIOR TREES

The concept of utility-based systems [6] has been developed primary for mathematically depended areas such as economics. Through time utility concepts also impact other areas. In past years was explored by AI programmers primally in the game development area. By nature, utility systems are easy to map on behavior patterns. Basic idea behind utility systems is rather making complicated preconstructed task selection logic with manually determined priorities the developers are going to specify set of tasks and conditions (a.k.a. considerations [3]) for picking most suitable task depend on the AI agent and world properties. Every condition provides normalized float value from 0 to 1. The task with larger weight will be selected. For example, we can specify custom attack enemy task that will depend on a few conditions:

1. Do we have a weapon?
2. Do we have enough ammo?
3. How much health do we have?

Each of those conditions return normalized value, each value multiplied to provide single weight for task. The system which will pick most preferable task called – Reasoner. The reasoner can be implemented as class which contains list of tasks and additional info about priorities. Reasoners main purpose is to determine task to execute.

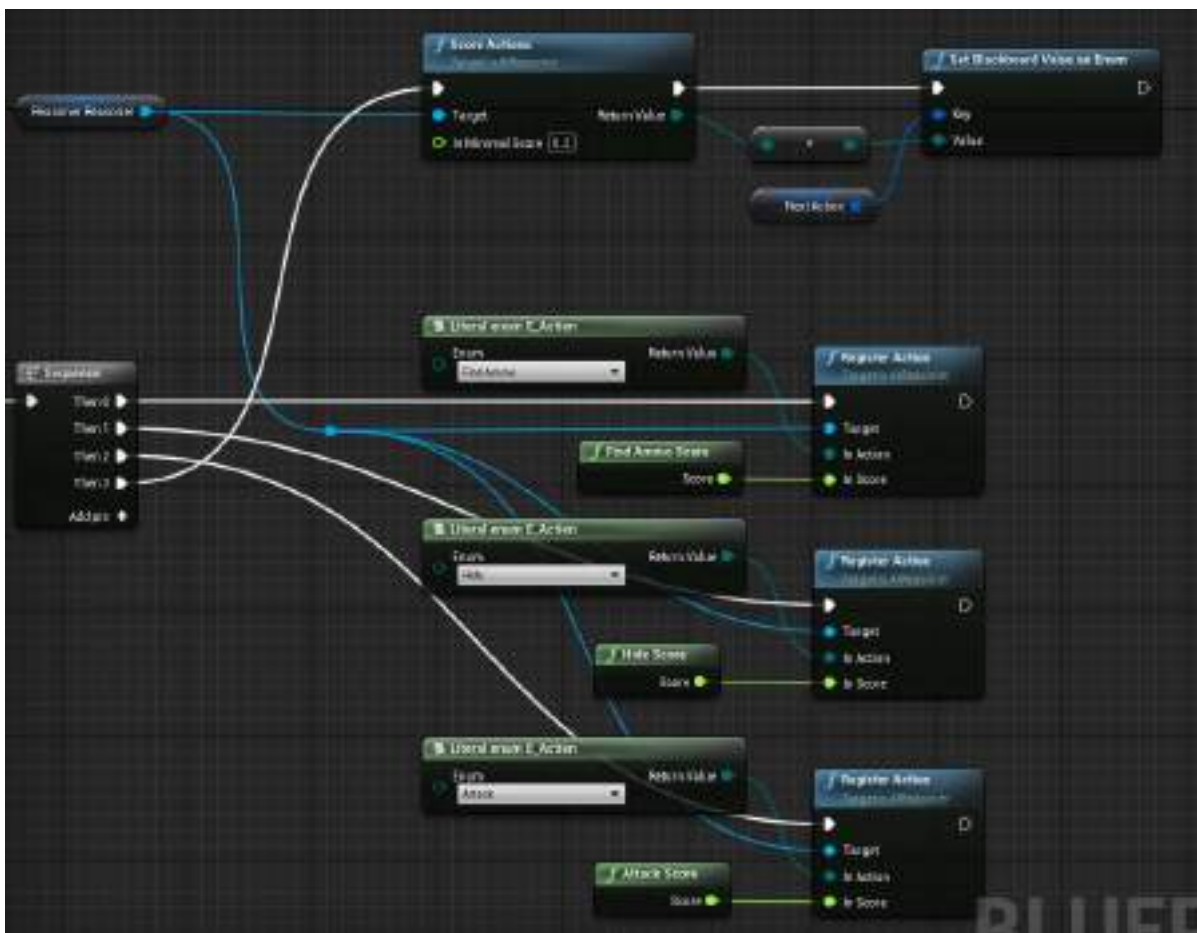
Most AI systems have their own pros and cons. For example, behavior trees by their nature are more static and require from the developer to specify priorities in sequence to check certain conditions. When it comes to large scale behavior trees with separated variative logic it can be tricky to configure. Most of the modern games try to avoid predictable AI agent behaviors. Here comes utility-based ai systems. It can help to improve behavior selection so the ai agent will behave less predictable. Most often behavior trees and utility systems are going along side by side. Combining those architectures bring developers to systems with the pros from both behavior trees and utility.

One of the commonly used way to combine those architecture is to make custom selector systems based on utility ai while rest logic is done using behavior tree. For example, game engines like Unreal Engine 4 provides powerful support of behavior trees [1] where users can specify their own services, decorators or even make custom engine modification to create game-specific sequence or selector nodes. Despite behavior tree system Unreal Engine 4 [2] provides tools for curve editing which can be used to make custom curves for utility ai score evaluations.

Implementation of utility ai integrated into behavior tree can be different depend on the project and requirements of the software. Most of the systems contains few common abstractions: tasks, conditions, reasoner. Applying utility ai to behavior tree developers can operate without task abstractions and create hybrid approach based on behavior tree services. For example, reasoner can be separated as standalone class which determine generic task id which has the highest priority. It means that we can abstract from direct involvement into the agent behavior logic and write more generic code.

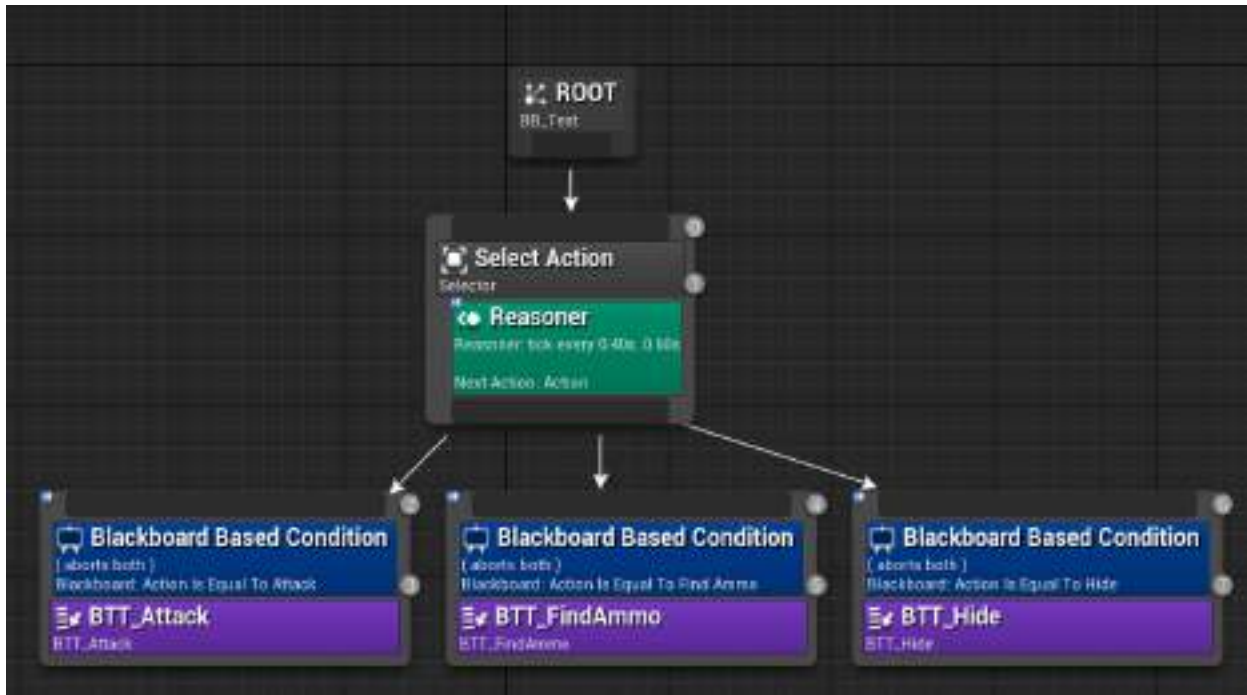
Reasoner operates with enumerations rather task objects. Each enumeration can be casted to uint8 data type. This allow us to reuse same reasoning logic across different behavior tree leaves. This class can be used inside behavior tree service to determine enumeration variable that can be used by behavior tree selector to switch between different tasks. Also, reasoner can support minimal score value as a threshold. When evaluation return value that is not completely satisfying our needs, we can pick the default action instead.

To make reasoner to pick more unpredictable actions we can use weighted randomization based on action scores. Reasoner can be placed inside behavior tree service to determine next action. Here is example of behavior tree service written in blueprints using Unreal Engine 4:



Pic 1. Usage of reasoner inside behavior tree service

Reasoner can be placed inside behavior tree service to determine next action. Here is example of behavior tree service written in blueprints inside behavior tree:



Pic 2. Example of behavior tree using utility ai

Utility AI provides nice way to describe variable behavior for tree-based AI architectures. Utility systems are easy to write, easy to debug and easy to integrate with behavior tree services or AI controllers.

References:

1. Behaviour Tree Overview – [Electronic Resource] – Access Mode: <https://docs.unrealengine.com/en-US/InteractiveExperiences/ArtificialIntelligence/BehaviorTrees/BehaviorTreesOverview/index.html>
2. Unreal Engine 4 Documentation – [Electronic Resource] – Access Mode: <https://docs.unrealengine.com/en-US/index.html>
3. LEWIS, Mike. Choosing effective utility-based considerations. In: Game AI Pro 360. CRC Press, 2019. p. 361-372.
4. MERRILL, Bill. Building utility decisions into your existing behavior tree. In: Game AI Pro 360. CRC Press, 2019. p. 81-90.
5. HANLON, Sebastian; WATTS, Cody. Behavior Decision System: Dragon Age Inquisition's Utility Scoring Architecture. In: Game AI Pro 360. CRC Press, 2019. p. 73-82.
6. MARK, Dave; DILL, Kevin. Improving ai decision modeling through utility theory. In: Game Developers Conference. 2010.

*Vash Y., postgraduate, department assistant, Department of software systems,
Uzhhorod National University, Uzhhorod;
Rol M., assistant, Department of software systems,
Uzhhorod National University, Uzhhorod*

REVISITING ENVIRONMENT QUERY SYSTEM

The AI systems in modern games requires useful features and tools from the game engine software. There are many architectural patterns and algorithms to solve variety of AI-related problems, like: FSM, Behavior Tree, Path finding algorithms. Modern games require realistic agent behavior which tends to inventing new features. Such feature was made by Epic Games in Unreal Engine 4 [1] called Environment Query System (EQS) [2].

Environment Query System solve problem of collecting, filtering and scoring information about environment around the AI agent. EQS consists of few based abstractions: generators, tests and contexts.

Generators represent logic that has set of rules which collect data in certain area. For example, generators can provide points around fixed locations sorted in grid or circle. Generators can apply additional logic (pre-pass filtering) to provide list of locations that meets the requirements.

Test, in case of EQS, represent a simple rule which can determine the score of the location (or object) in environment. For example, there are distance tests which provide score to certain location or object within the environment to decide how far location is, or is the location beyond certain distance threshold. There are variety of tests inside Unreal Engine 4 EQS system like: distance, trace, project, dot, overlap, pathfinding. Test can contain information about curve which will be used to determine score for each location, minimum and maximum values which will be considered inside scoring algorithm. Additionally, each test can have purpose option which indicates the algorithm of processing the location or object. For instance, the same distance test can evaluate locations closer to context or filter out all location that will be beyond certain distance value.

Context in terms of EQS represent set of data which describes the certain location, actor or set of locations and actors. Contexts are used inside both generators and tests to provide additional parameters. For example, context of the trace test provides information about trace start location. It can be represented by actor (game player or AI agent) or specified location vector.

Each EQS query requires Querier context that represent caller or initial data for EQS generators. Additionally, each EQS query can return single best matching result, all results or specific percent of success results. By default, return values describes in vector or object data types.

Environment Query System provides high-quality mechanism to gather information about surroundings and objects, pick useful data which can be used to help make more clear and dynamic behavior logic in general. Unreal Engine 4 have strong integration of EQS with Behavior Trees [3] and EQS editor with graphics user interface to increase programming and debugging speed.

References:

1. Unreal Engine 4 Documentation – [Electronic Resource] – Access mode: <https://docs.unrealengine.com/en-US/index.html>
2. Environment Query System – [Electronic Resource] – Access mode: <https://docs.unrealengine.com/4.26/en-US/InteractiveExperiences/ArtificialIntelligence/EQS/>
3. Behaviour Tree Overview – [Electronic Resource] – Access mode: <https://docs.unrealengine.com/4.26/en-US/InteractiveExperiences/ArtificialIntelligence/BehaviorTrees/>

*Бичковський В.О., к.т.н, доцент, кафедра радіотехнічних систем,
Національний технічний університет України «Київський політехнічний
інститут імені Ігоря Сікорського», м. Київ;
Реутська Ю.Ю., кафедра радіотехнічних систем, Національний технічний
університет України «Київський політехнічний інститут імені
Ігоря Сікорського», м. Київ*

ФОРМАЛІЗАЦІЯ ЗАДАЧІ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ ПРОЦЕСУ ПОШУКУ ОБ'ЄКТІВ

В сучасних умовах значна увага приділяється організації та забезпеченню пошуку об'єктів [1, 2]. Вирішення задачі пошуку об'єктів не представляється можливим без достатнього рівня інформаційного забезпечення процесу пошуку. Таким чином, формалізацію задачі інформаційного забезпечення процесу пошуку об'єктів необхідно розглядати як актуальну.

Нехай S_0 — площа, у межах якої може знаходитися об'єкт. Внаслідок використання різноманітної інформації I від передбачених джерел інформації площа, у межах якої відбувається пошук, постійно зменшується. Отже, за рахунок збільшення у часі $I(t)$ забезпечується умова $S < S_0$.

Нехай K — константа швидкості зменшення S завдяки надходженню інформації I . Тоді можна записати

$$dS = -KSdI. \quad (1)$$

Прийmemo до уваги, що $I = \ln N$, де N — інформаційна спроможність джерел інформації [3]. Приймаючи до уваги залежність (1), знаходимо

$$\frac{dS}{S} = -K \frac{dN}{N}. \quad (2)$$

Врахуємо початкові умови $S = S_0$, $N = 1$ (тобто $I = 0$). Тоді після інтегрування лівої та правої частин рівняння (2) визначаємо

$$\frac{S}{S_0} = \frac{1}{N^K}. \quad (3)$$

Аналіз залежності (3) показує, що для встановлення результативності пошуку об'єктів необхідно визначити N . Для розв'язання поставленої задачі

прийнемо до уваги, що одним із ефективних методів макроскопічного аналізу та прогнозування є метод аналогій [4].

Скористуємося процедурою макроскопічного аналізу та математичною моделлю Т. Мальтуса

$$\frac{dy}{dt} = ry, \quad (4)$$

де y — чисельність популяції, r — мальтузіанський коефіцієнт [5]. Розглянемо швидкість зміни кількості інформації $C_1 = dI/dt$. Оскільки $I = \ln N$, то можна записати

$$\frac{dN}{dt} = C_1 N. \quad (5)$$

Аналіз залежностей (4), (5) показує, що мальтузіанському коефіцієнту r відповідає C_1 у інформаційному аналогу. Таким чином, можна перейти до використання математичної моделі Кетле-Ферхюльста, яка враховує фактор самолімітування

$$\frac{dy}{dt} = ry \left(1 - \frac{y}{y_{cm}} \right), \quad (6)$$

$$y_{cm} = \lim_{t \rightarrow \infty} y(t).$$

Зауважимо, що $I = \ln N$. Отже, відлік N починається від одиниці. Отже, у праву частину аналогу рівняння (6) необхідно ввести складову $N-1$. Тоді можна записати

$$\frac{dN}{dt} = C_1 N \left(1 - \frac{N-1}{M-1} \right), \quad (7)$$

$$M = \lim_{t \rightarrow \infty} N(t).$$

Перепишемо рівняння (7) у вигляді

$$\frac{dN}{N(M-N)} = \frac{C_1 dt}{M-1}. \quad (8)$$

Після інтегрування лівої та правої частин рівняння (8) за початкових умов $N=1$, $t=0$, знаходимо

$$N = \frac{M}{1 + (M-1) \exp\left(-\frac{MC_1 t}{M-1}\right)}. \quad (9)$$

На підставі рівнянь (3), (9) визначаємо

$$\frac{S}{S_0} = \left[\frac{1}{M} + \left(1 - \frac{1}{M} \right) \exp\left(-\frac{MC_1 t}{M-1}\right) \right]^K. \quad (10)$$

Аналіз співвідношення (10) показує, що за умови $M \gg 1$ можна записати

$$\frac{S}{S_0} \approx \exp(-KC_1 t). \quad (11)$$

Таким чином, на підставі залежностей (10), (11) можна визначити результативність інформаційного забезпечення пошуку об'єктів. Отримані результати доцільно використовувати в процесі прогнозування ефективності пошуку об'єктів в умовах обмежувальних факторів.

Література:

1. Абчук В. А. Поиск объектов / В. А. Абчук, В. Г. Суздаль. — М.: Сов. радио, 1977. — 336 с.
2. Горбунов В. А. Эффективность обнаружения целей. — М.: Воениздат, 1979. — 160 с.
3. Новицкий П. В. Основы информационной теории измерительных устройств. — М.: Энергия, 1968. — 248 с.
4. Кузнецов Ю. М. Прогнозування розвитку технічних систем / Ю. М. Кузнецов, Р. А. Склярів. — К.: ТОВ «ЗМОК» - ПП «ГНОЗИС», 2004. — 323 с.
5. Долгий Ю. Ф. Математические модели динамических систем с запаздыванием: учебн. пособие / Ю. Ф. Долгий, П. Г. Сурков. — Изд-во Урал. Ун-та, 2012. — 122 с.

***Бровина О.І., студент,**
кафедра моделювання та програмного забезпечення,
Криворізький національний університет, м. Кривий Ріг;
Іосінов Р.К., студент,
кафедра моделювання та програмного забезпечення,
Криворізький національний університет, м. Кривий Ріг*

ВИКОРИСТАННЯ PyCairo У РОБОТІ З ВЕКТОРНИМИ ЗОБРАЖЕННЯМИ

У сьогоденному світі завжди є необхідність роботи з даними, велика частина яких складається із зображень. Для роботи з ними потрібна їх обробка – процес аналізу і роботи з цифровим зображенням, спрямований на поліпшення якості картинки або ж вилучення певної інформації. Завдяки доступності мови Python і її зростаючій популярності, всередині цієї мови з'явилося безліч першокласних бібліотек для обробки зображень.

Існує два види комп'ютерної графіки: векторна і растрова. У растровій графіці зображення представлено набором пікселів, а у векторній графіці – у вигляді геометричних примітивів: точок, ліній, кривих, полігонів. Дані примітиви формуються за допомогою математичних рівнянь. Обидва види графіки мають свої плюси і мінуси, наприклад, декілька переваг векторної графіки над растровою: маленький розмір, можливість масштабування до нескінченності, при переміщенні, масштабуванні і заповненні якість зображення не втрачається.

Одною з таких бібліотек є PyCairo, вона являє собою набір прив'язок коду для графічної бібліотеки Cairo, яка, в свою чергу, є 2D-бібліотекою для відтворення векторної графіки.

Бібліотека Cairo написана на мові програмування C, але також працює у зв'язці з Python, Perl, C++, C#, Java, та є мультиплатформною бібліотекою, що працює з такими ОС, як Linux, BSDs, Windows, OSX.

Для створення зображення в PyCairo, насамперед треба створити контекст малюнка (drawing context). Він включає в себе такі параметри, як ширина лінії, колір, параметри поверхні і багато іншого. Це дозволяє використовувати меншу кількість аргументів для спрощення інтерфейсу.

Контур (path) представлений набором точок для створення таких геометричних фігур, як лінії, криві, дуги. Контури існують відкриті і закриті, у останніх початкові і кінцеві точки не перетинаються. У PyCairo ми починаємо з порожнього контуру, визначаємо або заповнюємо його. Після кожного рядка stroke() або fill() він стає порожнім, і ми можемо сформувану новий контур. Якщо нам потрібно зберегти вже існуючий, можна використати команди stroke_preserve() і fill_preserve().

Source (вихідний матеріал) – інструменти, що ми використовуємо при малюванні. Існує чотири види базових початкових кодів: кольори, градієнти, візерунки, зображення.

Surface (поверхня) – простір, де власне і виконується малювання. Для малювання безпосередньо на платформі застосовуються поверхні Win32 і Xlib. За необхідністю використовується mask(маска) у якості фільтра – вона визначає, де буде чи не буде застосовуватися вихідний матеріал. Її непрозорі частини дозволяють його копіювати, а прозорі, навпаки – забороняють.

Pattern (шаблон) – відображає вихідні дані під час малювання на поверхні. У PyCairo його також можна використовувати в якості маски при малюванні.

Далі розглянемо декілька базових примітивів у PyCairo, скориставшись операціями заливки і обведення, пунктиром, закінченнями ліній та їх вигинами.

Лінія є базовим векторним об'єктом. Щоб намалювати лінію, використовуємо два методи: початок визначається методом move_to(), а кінець – методом line_to(). Візьмемо приклад, де кожне натискання лівою кнопкою миші у вікні (у випадковому порядку) зберігається у списку. При натисканні правою кнопкою миші, усі крапки з'єднуються з точками, зазначеними у списку, повторне натискання правою кнопкою очищає вікно. Створимо власний клас, щоб позначити клавіші миші ідентифікаторами. Деякі дії не зазначені за замовчуванням, тому нам потрібно вказати дії, викликані натисканням миші за допомогою методу set_event().

```
self.darea.set_events(Gdk.EventMask.BUTTON_PRESS_MASK)
self.darea.connect("button-press-event", self.on_button_press)
```

За допомогою коду ми отримуємо реакції на дії, викликані натисканням миші. Задаємо колір ліній та їх товщину, та пов'язуємо кожну точку в списку з усіма іншими. Метод stroke() малює зазначену лінію: при натисненні лівої кнопки миші ми додаємо координати x та y у список координат, а при

натисненні правої кнопки ми викликаємо метод `queue_draw()`, який перемальовує область. Всі точки тепер з'єднані лініями.

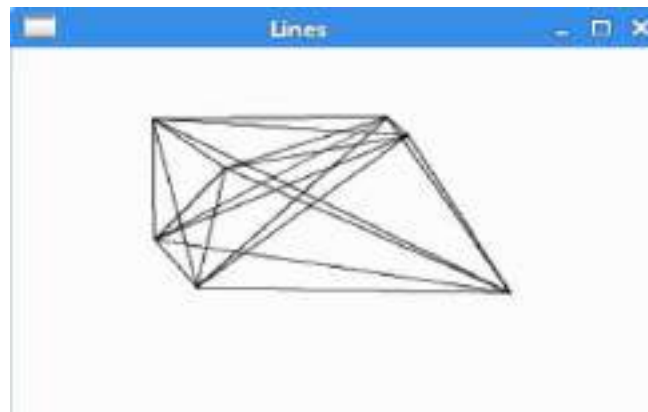


Рисунок 1 - З'єднані точки

Окрім малювання базових примітивів, PyCairo має такі функції, як створення базових та складних фігур за допомогою методів, їх заливка, градієнт та візерунки; можливість досягти ефекту прозорості за допомогою альфа-компоненту; перетворення, обертання, масштабування або зсув фігур; є можливість обробки тексту та текстових даних; відтворення зображення у форматах PNG і JPEG у вікні GTK, та безліч інших функцій.

Cairo популярний у співтоваристві `opensource`, так як забезпечує перенесення та якісне малювання двовірних зображень. GTK+ повністю підтримує Cairo. Починаючи з 2005 року GTK+ використовує Cairo для відтворення більшості графічних керуючих елементів, а починаючи з версії 3.0 малює через цю бібліотеку абсолютно все. Проект Mozilla задіяв Cairo в браузерному движку `Gecko`.

Література:

1. Drawing the outline in PyCairo [Електронний ресурс] / Режим доступу: <https://www.geeksforgeeks.org/pycairo-drawing-the-outline/>
2. Basic drawing in PyCairo [Електронний ресурс] / Режим доступу: <https://www.pythoninformer.com/python-libraries/pycairo/basic-drawing/>
3. Recursive functions in PyCairo [Електронний ресурс] / Режим доступу: <https://franklin.dyer.me/post/169>

Науковий керівник: Рибальченко Олена Геннадіївна, старший викладач, кафедра моделювання та програмного забезпечення, Криворізький національний університет, м. Кривий Ріг

*Бухарова Л.Д., студентка,
кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка,
кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

ОГЛЯД НАЙПОПУЛЯРНІШИХ МЕТОДІВ ШИФРУВАННЯ

На сьогоднішній день на ринку існує значна кількість VPN-провайдерів, що надають величезну кількість різних типів шифрування. Шифрування – це процес кодування інформації, в результаті якого вона стає недоступною для третіх осіб [1].

Існують різні технології VPN з різним ступенем шифрування. Наприклад, протокол тунелювання «точка-точка» (PPTP) працює швидко, але набагато менш безпечний, ніж інші протоколи, такі як IPSec або OpenVPN, який використовує SSL / TLS (Secure Sockets Layer / Transport Layer Security). Крім того, при використанні VPN на основі TLS також важливі тип алгоритму шифрування і довжина ключа [2].

Хоча OpenVPN підтримує безліч комбінацій шифрів, протоколів обміну ключами і алгоритмів хешування, найбільш поширеною реалізацією, запропонованою постачальниками послуг VPN для з'єднань OpenVPN, є шифрування AES з обміном ключами RSA і сигнатурами SHA. Рекомендованими параметрами є шифрування AES256 з ключем RSA довжиною не менше 2048 біт і криптографічною хеш-функцією SHA-2 (SHA256) замість SHA-1.

Проте, розглянувши протокол OpenVPN можна сказати, що незважаючи на те, що він заснований на відкритих джерелах, він вважається одним з найбезпечніших протоколів VPN [3]. Він стабільний і надійний, легко конфігурується для роботи на будь-якому порту, підтримує апаратне прискорення для поліпшення швидкості, здатний перетинати міжмережеві екрани і трансляцію мережевих адрес (NAT) і використовує бібліотеки OpenSSL для шифрування.

У той же час, найменш захищений протокол VPN – це протокол PPTP. Він має слабе шифрування і відносно легко блокується провайдерами.

Так протоколи тунелювання VPN пропонують різні функції і рівні безпеки, і для кожного з них є переваги і недоліки [4, 5]. Існує п'ять основних протоколів тунелювання VPN: Протокол тунелювання захищених сокетів (SSTP), Протокол тунелювання «точка-точка» (PPTP), Протокол тунелювання другого рівня (L2TP), OpenVPN і Internet Key Exchange версії 2 (IKEv2).

Варто зазначити, що шифрування може впливати на швидкість з'єднання. Вибір технології VPN і методів шифрування повинен проводитися в кожному конкретному випадку, в залежності від того, які дані будуть передаватися.

Література:

1. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:<https://doi.org/10.26906/SUNZ.2021.1.090>.
2. Ruban I.V., Churyumov G.I., Tokariiev V.V., Tkachov V.M. Structural-functional reconfiguration of computer systems with reconstruct structure. Проблеми інформатики та моделювання: тези доповідей 19-ї міжн. наук.-техн. конф., м. Одеса, 11-16 вер. 2019р. Одеса, С.71 — 72.
3. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
4. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii Tkachov, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdet'ska // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020.
5. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Pp. 25-26.

*Бухарова Л.Д., студентка,
кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка,
кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

РОЗГЛЯД VPN ТА АНОНИМАЙЗЕРІВ В ЯКОСТІ ЗАСОБІВ БЕЗПЕЧНОГО ВИКОРИСТАННЯ НЕЗАХИЩЕНИХ МЕРЕЖ

Сучасний розвиток інформаційних технологій і, зокрема, мережі Internet, призводить до необхідності захисту інформації, переданої в рамках розподіленої корпоративної мережі, що використовує мережі відкритого доступу [1]. Однак Інтернет є незахищеною мережею, тому доводиться винаходити засоби захисту конфіденційних даних, переданих по незахищеній мережі.

Існує декілька засобів, один з яких це анонімайзери або проксі-сервера, вони приховують дані про комп'ютер або користувача в локальній мережі від віддаленого сервера. Розглянувши принцип роботи проксі-серверів, можемо сказати, що анонімайзер завжди розриває прямий ланцюг зв'язку і стає посередником між веб-браузером і потрібним веб-сервером [2].

У сфері кібербезпеки анонімізуючий проксі-сервер – це інструмент, який можна використовувати для того, щоб зробити онлайн-діяльність невідстежуваною або анонімною. Ці проксі, по суті, діють як посередницькі "шлюзи" між користувачем Інтернету та їхнім місцем призначення, як і VPN.

Проте VPN – це технологія, яка об'єднує довірені мережі, вузли і користувачів через відкриті мережі, яким немає довіри. Технологія, яка набуває все більшого поширення серед не тільки технічних фахівців, а й серед звичайних користувачів, яким також потрібно захищати свою інформацію (наприклад, користувачі Internet-банків або Internet-порталів) [3].

На відміну від анонімайзерів, VPN шифрують онлайн-трафік. Анонімайзер, в свою чергу, тільки маскує вашу IP-адресу, він не може захистити вас від відстеження вашим постачальником послуг Інтернету або іншими третіми сторонами.

Проксі-сервери працюють на рівні додатків, тоді як VPN працюють на рівні операційної системи. Іншими словами, VPN може охоплювати весь Інтернет-трафік, що надходить з комп'ютера користувача, тоді як проксі-сервер покриває лише трафік, що надходить із певного браузера чи програми. Користувачі VPN можуть використовувати техніку, яка називається розділеним тунелюванням, щоб вибрати, який трафік буде маршрутизуватися через їх VPN [4,5].

Так будь-який інструмент, який перенаправляє веб-трафік клієнта для захисту його конфіденційності, швидше за все, вплине на швидкість Інтернету. Однак, оскільки VPN також шифрують дані клієнта, вони можуть бути повільнішими за анонімайзери. Компроміс полягає в тому, що VPN часто пропонують більш надійну безпеку та конфіденційність, ніж анонімайзери.

Література:

1. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
2. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Pp. 25-26.
3. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:<https://doi.org/10.26906/SUNZ.2021.1.090>.
4. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii Tkachov, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdet'ska // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020.
5. V. Tkachov, M. Bondarenko, O. Ulyanov and O. Reznichenko, "Overlay Network Infrastructure for Remote Control of Radio Astronomy Observatory," 2019

*Бухарова Л.Д., студентка,
кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка,
кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

КОНЦЕПЦІЯ ВІРТУАЛЬНОЇ ПРИВАТНОЇ МЕРЕЖІ В ПОРІВНЯННІ З МЕРЕЖАМИ КОНВЕРТОВОГО РАДІОРЕЛЕЙНОГО ЗВ'ЯЗКУ

В умовах дедалі більш глобалізованої економіки компанії починають шукати географічний розподіл, або пов'язаний з податковими стимулами, або просто з можливістю розширення. У цьому випадку співробітники потребують свободи здійснення своєї діяльності без географічних обмежень. Досягненню даної мети можуть посприяти віртуальні приватні мережі або мережі конвертового радіорелейного зв'язку [1].

Концепція віртуальної приватної мережі – VPN, з'явилася як альтернатива захищеної комунікації через громадські канали зв'язку, такі як Інтернет, і незабаром стала технологією, що широко використовується, орієнтованою на безпеку, що гарантує цілісність та конфіденційність інформації [2].

Тоді як, так звані frame-relay networks (мережі конвертового радіорелейного зв'язку) – це технологія передачі даних, яка надсилає інформацію по мережі WideArea (WAN) шляхом поділу даних на пакети або кадри, та дозволяють кільком користувачам підключатися до спільного простору ("хмари") та створювати приватні віртуальні схеми між офісами в мережі [3]. Хмари з ретрансляцією кадрів пропонують економію коштів у порівнянні з іншими альтернативами приватної мережі, оскільки їм не потрібно орендувати більш дорогі лінії «точка-точка» між офісами. Абоненти фрейм-ретрансляторів отримують вигоду від економії на масштабі завдяки спільному використанню мережевої основи.

Потрібно зазначити, що сьогодні в Інтернеті використовуються два типи технологій VPN: надійний VPN та безпечний VPN (IP VPN). Надійні VPN-мережі надаються та управляються постачальниками послуг за допомоги визначення шляхів через їхні мережі для забезпечення маршрутизації трафіку клієнтів по надійному шляху. Безпечні VPN захищають трафік і забезпечують конфіденційність, аутентифікацію та цілісність даних за допомогою криптографічних алгоритмів, і ними може керувати замовник або постачальник послуг. Надійні та безпечні VPN також можна використовувати разом у гібридній VPN [4, 5].

Так, і надійні, і безпечні VPN – це доступні альтернативи приватним мережам, таким як фрейм-ретранслятор. Мережі VPN можуть забезпечити таку

ж продуктивність та надійність кадрового ретранслятора, але з більшою гнучкістю. Проте більшої безпеки, легше досягти за допомогою ретранслятора (FR), так як він автоматично забезпечує безпеку користувача, тоді як користувач IP VPN повинен вжити додаткових заходів для підвищення безпеки.

Література:

1. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Pp. 25-26.
2. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
3. Ruban I.V., Churyumov G.I., Tokariev V.V., Tkachov V.M. Structural-functional reconfiguration of computer systems with reconstruct structure. Проблеми інформатики та моделювання: тези доповідей 19-ї міжн. наук.-техн. конф., м. Одеса, 11-16 вер. 2019р. Одеса, С.71 — 72.
4. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:<https://doi.org/10.26906/SUNZ.2021.1.090>.
5. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii Tkachov, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdet'ska // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020.

*Бухарова Л.Д., студентка,
кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка,
кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

ОСНОВИ ТУНЕЛЮВАННЯ, ЯК РЕАЛІЗАЦІЯ ТЕХНОЛОГІЇ ВІРТУАЛЬНИХ ПРИВАТНИХ МЕРЕЖ

В даний час існує безліч варіантів реалізацій технології VPN, з певними перевагами і недоліками. Одною найпоширеніших реалізацій є віртуальні приватні мережі на основі техніки тунелювання [1].

Тунелювання – це мережева технологія, яка дозволяє інкапсулювати один тип пакету протоколу в дейтаграму іншого протоколу. Наприклад, з'єднання Windows VPN можуть використовувати пакети протоколу тунелювання (PPTP)

від точки до точки, щоб інкапсулювати і відправляти приватний мережевий трафік, такий як трафік TCP / IP в загальнодоступній мережі, такий як Інтернет.

Так клієнт тунелю або сервер використовує протокол передачі даних тунелю для підготовки даних для передачі [2, 3]. Наприклад, коли клієнт тунелю відправляє корисне навантаження на тунельний сервер, він спочатку додає заголовок протоколу передачі даних тунелю в корисне навантаження. Потім клієнт відправляє отримане інкапсульоване корисне навантаження по мережі, яка направляє його на сервер тунелів. Сервер тунелів приймає пакети, видаляє заголовок протоколу передачі даних тунелю і пересилає корисне навантаження в цільову мережу.

Існує два типи тунелювання: добровільне та обов'язкове. Добровільне тунелювання – користувач або клієнтський комп'ютер може видати запит VPN для настройки і створення добровільного тунелю. Добровільне тунелювання відбувається, коли клієнтський комп'ютер або сервер маршрутизації створює віртуальне з'єднання з цільовим сервером тунелів [4].

При примусовому тунелюванні клієнтський комп'ютер розміщує виклик віддаленого доступу в NAS з підтримкою тунелювання в ISP. Наприклад, корпорація могла б укласти контракт з інтернет-провайдером для розгортання загальнонаціонального набору FER.

Таким чином, у зв'язку з популярністю Інтернету і стека TCP / IP ситуація, коли несучим протоколом транзитної мережі зазвичай виступає протокол IP, а протоколом-пасажиром – деякий каналний протокол, є дуже поширеною. Разом з тим застосовуються і інші схеми інкапсуляції, такі як інкапсуляція IP в IP, Ethernet в MPLS, Ethernet в Ethernet. Подібні схеми інкапсуляції потрібні не тільки для того, щоб узгодити транспортні протоколи, а й для інших цілей, наприклад для шифрування вихідного трафіку або для ізоляції адресного простору транзитної мережі провайдера від адресного простору для користувача мереж [5].

Література:

1. Ruban I.V., Churyumov G.I., Tokariev V.V., Tkachov V.M. Structural-functional reconfiguration of computer systems with reconstruct structure. Проблеми інформатики та моделювання: тези доповідей 19-ї міжн. наук.-техн. конф., м. Одеса, 11-16 вер. 2019р. Одеса, С.71 — 72.
2. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:<https://doi.org/10.26906/SUNZ.2021.1.090>.
3. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
4. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Рр. 25-26.

5. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii Tkachov, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdetzka // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020.

*Глюза А.П., студентка, комп'ютерна інженерія та управління,
Харківський національний університет радіоелектроніки, м. Харків*

МОСК-ОБ'ЄКТИ В ЮНІТ-ТЕСТУВАННІ

Тестування – це процес в програмуванні, що дозволяє перевірити на коректність окремі модулі вихідного коду програми.

Існує кілька підходів до визначення тестів. Перша модель – класика: спочатку розробка, а потім тестування «code first». Це означає, що спочатку відбувається написання коду, потім тестування продукту і відправка його або на доопрацювання, або перехід до наступної стадії розробки.

Інший підхід можна назвати «test first» режимом. Це означає, що тестування починається ще до написання самої функції – наприклад, створити одиничний тест або автоматично виконується набір тестів до того, як функція або якийсь шматок коду буде розроблений і впроваджений в додаток. Одним з найбільш популярних прикладів тут є Test-Driven Development.

Проте, варто згадати, що техніка «test first» не так популярна, як «code first». Це пов'язано з тим, що в більшості проектів все ще складно автоматизувати те, що ще не було розроблено. Узагальнюючи обидва згаданих вище підходу, можна зробити висновок, що немає особливої різниці і що автоматизацію тестів можна використовувати в будь-якому з варіантів.

Важливі терміни, з якими доводиться стикатися в процесі написання тестів – це стаб (stubs) і моки (mock).

Дуже часто код (функція, модуль) мають зовнішні залежності. Зовнішня залежність – це все, що робить тести не правдивими і складно-підтримуваними. Файлова система – залежність: структура каталогів може бути іншою на іншій машині. База даних – залежність, її може не бути на іншій машині. Веб-сервіс – залежність: може не бути інтернету.

Якщо на питання: «Чи буде цей компонент поводитися так само на іншій машині?», відповідь – «ні», то його необхідно «підмінити» і тут на допомогу якраз приходять стаб і моки. Але є й інша сторона медалі, коли розробник починає захоплюватися і приходять до того, що підміняє взагалі все. Відповідно тести перестають перевіряти сам додаток і починають тестувати стаб та моки. Це в корені не вірно. Якщо «живих» реалізацій в тесті немає, то цей тест не тестує нічого.

Іноді ці терміни stubs і mock плутають: різниця в тому, що стаб нічого не перевіряє, а лише імітує заданий стан. А мок – це об'єкт, у якого є очікування.

Наприклад, що даний метод класу повинен бути викликаний певне число раз. Іншими словами, тест ніколи не зламається через стаб, а ось через мока може.

З технічної точки зору це означає, що, використовуючи стаб, ми перевіряємо стан тестованого класу або результат виконаного методу. При використанні мока ми перевіряємо, чи відповідають очікування мока поведінки тестованого класу. Також краще використовувати не більше одного мока на тест. Інакше з високою ймовірністю ви порушите принцип «тестувати тільки одну річ». При цьому, в одному тесті може бути скільки завгодно стабів або ж мок і стаб.

Є вже готові фреймворки, які надають такий функціонал: Sinon, Jasmine, enzyme, Jest, testdouble.

Незважаючи на те, що існують ситуації, в яких Моки потрібні, в більшості ситуацій їх потрібно уникати. Моки занадто багато знають про те, як працює код. Будь-який тест з моками з чорного ящика перетворюється в білий ящик. Повсюдне використання моков призводить до двох речей:

- після рефакторинга доводиться переписувати тести, навіть якщо код працює правильно. Відбувається це через зав'язки на те, як конкретно працює код;
- код може перестати працювати, але тести будуть проходити, тому що вони сфокусовані не на результатах його роботи, а на тому, як він влаштований всередині.

Література:

1. Mock-объект [Електронний ресурс] – Режим доступу до ресурсу: <https://dic.academic.ru/dic.nsf/ruwiki/166069>.
2. Модуль Mock: макеты-пустышки в тестировании [Електронний ресурс]. – 2012. – Режим доступу до ресурсу: <https://habr.com/ru/post/141209/>.

*Глюза А.П., студентка, комп'ютерна інженерія та управління,
Харківський національний університет радіоелектроніки, м. Харків*

МУТАЦІЙНЕ ТЕСТУВАННЯ

Мутаційне тестування – це метод тестування, котрі базуються на всіляких зміни вихідного коду і перевірці реакції на ці зміни набору автоматичних тестів. Якщо тести після зміни коду успішно виконуються, значить або код не покритий тестами, або написані тести неефективні. Критерій, що визначає ефективність набору автоматичних тестів, називається Mutation Score Indicator (MSI).

Мета Mutation Testing – оцінити якість тестових випадків, які повинні бути досить надійними, щоб не виконувати мутантний код. Цей метод також називається стратегією тестування на основі помилок, так як він включає в себе створення помилки в програмі.

Мутація була спочатку запропонована в 1971 році, але втратила запал через високі витрати. Тепер вона знову набирає обертів і широко використовується для таких мов, як Java і XML.

У програмній інженерії тестування на мутації можна розділити на три типи: мутація операторів, мутація рішень і мутація значень.

Мутація оператора – розробник вирізає і вставляє частину коду, результатом якого може бути видалення деяких рядків.

Value Mutation – значення первинних параметрів змінені.

Мутація рішень – оператори управління повинні бути змінені

Показник мутації визначається як відсоток убитих мутантів від загальної кількості мутантів, тобто оцінка мутації = (убиті мутанти/загальна кількість мутантів)*100.

Тестові випадки відповідають мутацій, якщо оцінка складає 100%. Експериментальні результати показали, що мутаційні тестування є ефективним підходом для вимірювання адекватності тестових випадків. Але головний недолік полягає у високій вартості генерації мутантів і виконання кожного тестового прикладу для цієї програми мутантів.

Переваги мутаційного тестування:

- Це потужний підхід для досягнення високого охоплення вихідної програми.
- Це тестування здатне всебічно протестувати мутантних програму.
- Мутаційне тестування забезпечує хороший рівень виявлення помилок розробнику програмного забезпечення.
- Цей метод розкриває неоднозначність у вихідному коді і здатний виявляти всі помилки в програмі.
- Клієнти отримують вигоду від цього тестування, отримуючи саму надійну і стабільну систему.

Недоліки мутаційного тестування:

- Тестування на мутації є надзвичайно дорогим і трудомістким, так як існує безліч мутантних програм, які необхідно створити.
- Так як це забирає багато часу, можна сказати, що це тестування неможливо без інструменту автоматизації.
- У кожній мутації буде той же кількість тестових випадків, що і у вихідної програми. Таким чином, велика кількість мутантних програм, можливо, буде потрібно протестувати за допомогою оригінального набору тестів.
- Оскільки цей метод включає в себе зміни вихідного коду, він зовсім не застосуємо для тестування чорного ящика.

Мутаційне тестування – це форма тестування методом білого ящика, що має на увазі внесення в код помилкових змін (або мутантів) з подальшим виконанням набору тестів для кожного з них. Якщо тест не проходить, то мутант «вбитий». Успішне ж виконання тесту означає, що він «вижив». Чим більше убитих мутантів, тим більшу семантичну стабільність коду гарантує набір тестів. Дана техніка не тільки високоефективна, а й дорога. Тому через

великий масштаб завдання її застосування стає непрактичним, навіть для невеликих проектів ПО.

Література:

1. Witko K. Зачем нам мутационное тестирование [Електронний ресурс] / Kinga Witko. – 2020. – Режим доступу до ресурсу: <https://software-testing.ru/library/testing/other-testing/3331-why-do-you-need-mutation-testing>.
2. Мутационное тестирование на простом примере [Електронний ресурс] – Режим доступу до ресурсу: <http://getbug.ru/mutatsionnoe-testirovanie-na-prostom-primere/>.

*Гончаренко Д.В., студент, комп'ютерна інженерія та управління,
Харківський національний університет радіоелектроніки, м. Харків*

СЕРВІСИ AWS

Amazon Web Services (AWS) – це найпоширеніша в світі хмарна платформа з широкими можливостями, що надає більше 200 повнофункціональних сервісів для центрів обробки даних по всій планеті. Мільйони клієнтів, в тому числі стартапи, які стали лідерами за швидкістю зростання, найбільші корпорації і передові урядові установи, використовують AWS для зниження витрат, підвищення гнучкості і прискореного впровадження інновацій.

Деякі сервіси розглянемо детальніше.

1) Amazon S3

Amazon S3 занесено до списку послуг AWS на першому місці, тому що зберігання та отримання даних відіграє помітну роль у хмарних обчисленнях. S3 дозволяє користувачеві зберігати, завантажувати та отримувати великі файли розміром до 5 ТБ з хмари. Це масштабована, недорога та високошвидкісна веб-служба, призначена для архівування та резервного копіювання прикладних програм та даних.

2) Amazon EC2 (Elastic Compute Cloud)

AWS EC2 Amazon EC2 забезпечує масштабовані обчислювальні можливості в хмарі AWS. За допомогою Amazon EC2 можна швидко та ефективно розробляти та розгортати програми за низьку ціну. А також ви можете використовувати Amazon EC2 для запуску віртуальних серверів відповідно до ваших вимог.

3) AWS Lambda

AWS Lambda – це сервіс, який дозволяє користувачеві запускати код без будь-якого сервера. Amazon Lambda виконує код лише тоді, коли це потрібно користувачеві, і автоматично масштабує його. Користувачі платять лише за час розрахунку, не потрібно стягувати плату, коли ваш код не працює. Ця служба підтримує код, написаний на Node.js, Java, Python та мовах, які підтримуються Amazon Linux.

4) Amazon Glacier

Amazon Glacier – одна з найважливіших послуг, що надається AWS. Glacier – це онлайн-сервіс, який надає низьку вартість та ефективне зберігання з функціями безпеки для архівування та резервного копіювання даних. За допомогою Glacier можна ефективно зберігати інформацію протягом місяців, років або навіть десятиліть.

5) Amazon SNS

Amazon SNS – це веб-сервіс, що надається AWS. SNS розшифровується як служба простого сповіщення, а також цей веб-сервіс керує та доставляє повідомлення або сповіщення користувачам та клієнтам з будь-якої хмарної платформи. У SNS існує два типи клієнтів: передплатники та видавці. Видавці створюють і надсилають повідомлення до екземпляра абонента по каналах зв'язку. Абоненти отримують сповіщення від видавця щодо одного з підтримуваних протоколів, таких як Amazon SQS, HTTP і Lambda тощо.

6) Amazon EBS

AWS EBS – це служба Amazon, яка використовується для зберігання постійних даних, і це сховище на рівні блоків для використання екземплярів EC2. Можна використовувати службу EBS для переміщення даних з одного екземпляра в інший, не втрачаючи збережених даних у EBS. Ви можете змонтувати кілька томів в одному екземплярі, але кожен том можна одночасно приєднати до одного екземпляра.

7) Amazon Kinesis

AWS Kinesis використовується для обробки великих даних у режимі реального часу. Це дозволяє розробникам брати будь-який великий обсяг даних з будь-якого джерела, яке може працювати на екземплярах EC2. Він зберігає, збирає та обробляє дані з великих розподілених потоків, таких як канали соціальних мереж та події журналу. Після завершення обробки даних він одночасно розповсюджує дані споживачам.

Література:

1. Миграция с помощью AWS [Електронний ресурс]. – 2021. – Режим доступу до ресурсу: <https://softprom.com/ru/vendor/amazon-web-services/product/migratsiya-v-oblako-aws>.
2. Amazon Web Services [Електронний ресурс] – Режим доступу до ресурсу: <https://itglobal.com/ru-ru/company/glossary/amazon-web-services/>.

*Гончаренко Д.В., студент, комп'ютерна інженерія та управління,
Харківський національний університет радіоелектроніки, м. Харків*

TYPES OF CLOUD SERVICES

Let's start by trying to define what a «cloud» is. In the general case, the cloud is a tool that allows the customer's IT service to solve a problem quickly, efficiently and with minimal capital costs. Every year, businesses expect IT to increase the speed of output and provide new internal and external services.

Conventionally, all types of cloud services can be divided into three types:

- Infrastructure as a Service;
- Platform as a Service;
- Software as a Service.

As you can see, there is a prefix as a Service everywhere. This means that all types of clouds are provided on a subscription basis, i.e. they are used only when needed. The concept of Pizza-as-a-Service perfectly explains the essence of cloud services:

PIZZA AS A SERVICE

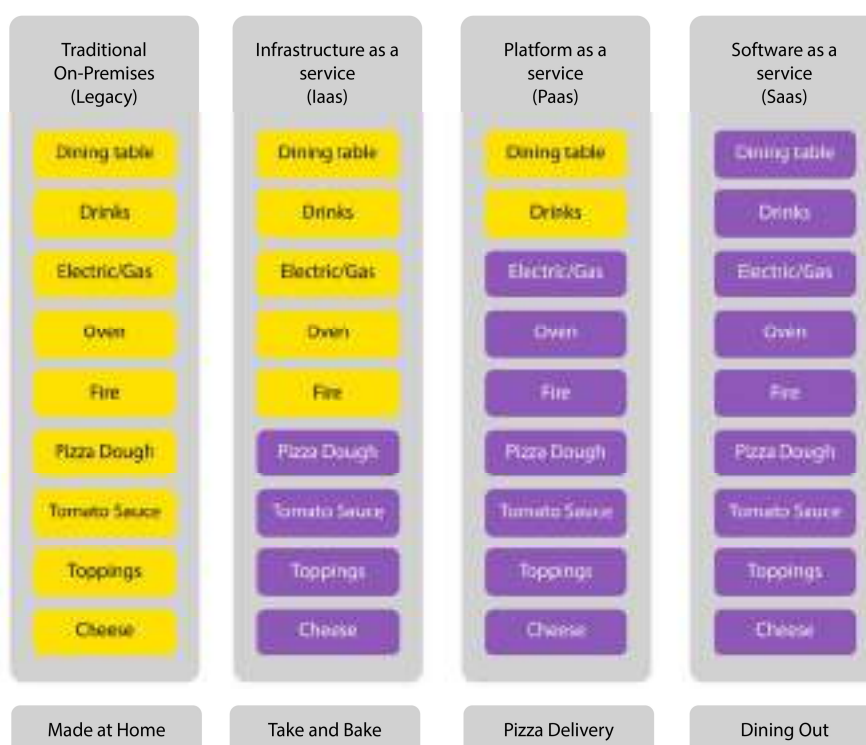


Fig.1 – The concept of Pizza-as-a-Service

So, let's start with the type of cloud technology, the representatives of which are best known – Software as a Service. This type implies the provision of a ready-made solution for the client with minimal need for settings. That is, theoretically, subscribing to such a service, it can be managed by any user with minimal involvement of the system administrator or without it at all. The most well-known representatives of such a service in the corporate environment are Office 365. If we talk about SMB, then it is worth mentioning such cloud services as Dropbox, Evernote, Trello and others.

PaaS services are designed primarily for developers. They represent sets of ready-made components for creating applications, as well as frameworks for platform management. In this case, the components will be database services, repositories, automated deployment tools, test environments, and the like. Examples of PaaS

services are Google AppEngine, VMWare Pivotal Cloud Foundry, Red Hat's OpenShift, Heroku and more.

Finally, the closest type of service to system administrators is IaaS. Infrastructure as a service in its facilities and characteristics is closest to owning your own «hardware» and virtualization. In the case of IaaS, you get cloud processors, memory, disks, and networks, from which you later create virtual routers and configure the network topology as you need.

The logic of choosing the right type of cloud service is to find a balance between setup speed and system flexibility. It is unlikely to sharpen SaaS exactly for your business processes, but it is also almost impossible to build a ready-made solution based on IaaS in a couple of hours. It is also worth noting the need to build multi-vendor solutions, which in the case of SaaS and PaaS can be very difficult.

And it's necessary to mention one more of the cloud models, without which the review would not be complete, namely VPS and Dedicated servers. Formally, this service also falls into the IaaS class, but has significant differences. The essence of the service consists in the provision by the customer of virtual or dedicated physical servers for rent at very low prices.

Література:

1 Види облаков и облачных услуг [Електронний ресурс] – Режим доступу до ресурсу: <https://www.datafort.ru/blog/the-types-of-clouds.html>.

2 Види облачных сервисов: IaaS, PaaS, SaaS и другие модели [Електронний ресурс] – Режим доступу до ресурсу: <https://oblako.kz/iaas-blog/samyepopuljarnye-oblachnye-servisy-v-mire>.

Думчиков С.А., студент 6 курсу, факультет інформаційних технологій та комп'ютерної інженерії, кафедра захисту інформації,

Вінницький національний технічний університет, м. Вінниця;

Лукічов В.В., к.т.н., старший викладач, факультет інформаційних технологій та комп'ютерної інженерії, кафедра захисту інформації,

Вінницький національний технічний університет, м. Вінниця

ВИЯВЛЕННЯ ФІШИНГОВИХ ІНФОРМАЦІЙНИХ РЕСУРСІВ

Фішинг – це популярна форма атаки соціальної інженерії, при якій зловмисник обманює жертву, видаючи себе за іншу особу або ресурс. Електронні листи і повідомлення зі шкідливими вкладеннями або скомпрометованими URL-адресами, що перенаправляють на шкідливі веб-сайти, є одними з найпоширеніших векторів атак, які використовуються при фішингу. Технологічний прогрес надав фішерам більш досконалі інструменти для запуску небезпечних і витончених атак. У звіті Phishlabs про тенденції в області фішингу за 2018 рік [1] згадується, що цілі фішингових атак перемістилися з приватних осіб на підприємства. Що ще гірше, у фішерів тепер є доступ до безкоштовних SSL-сертифікатів. Майже половина всіх фішингових

веб-сайтів в даний час використовує протокол HTTPS, який був одним з основних індикаторів легітимності веб-сайтів [2]. В іншому звіті, опублікованому APWG в першому кварталі 2019 року, говориться, що кількість фішингових атак збільшилася на 30% в порівнянні з попереднім кварталом і що основними цілями були служби «Програмне забезпечення як послуга» і веб-пошта [3].

За останнє десятиліття дослідники ідентифікували і класифікували особливості, що проявляються з векторів фішингових атак різними способами. В деяких статтях ознаки класифікуються з точки зору типу атаки, деякі класифікуються на основі того, як і де вони з'являються у векторі атаки [4], [5]. Однак, наскільки відомо, ніхто не дав систематичної класифікації, яка незалежна від підходів до виявлення і охоплює всі можливі особливості, які можуть бути добуті з векторів атак. Вектори фішингу, наприклад веб-сайти, URL-адреси, представляють собою спеціалізовані стрічки. Тому, такі категорії мови чи логіки як синтаксис, семантика і прагматика можуть бути ефективно використані для класифікації їх атрибутів. Побудова класифікації потребує детального огляду інформаційних джерел, розуміння потенційних проблем та можливостей, а також систематичного і детального підходу. Класифікація, що представлена на рисунку 1, має основні компоненти веб-сторінки в якості других рівнів. Кожний компонент далі розбивається на такі групи: синтаксис, семантика і прагматика.

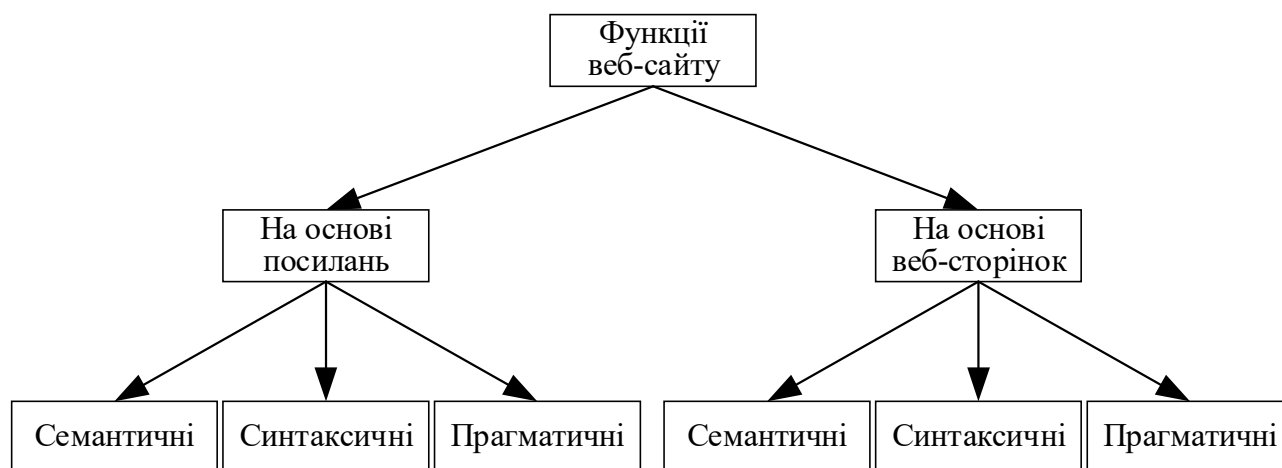


Рисунок 1 – Вигляд класифікації сайтів на основі посилань та веб-сторінок

Синтаксичні функції вектору залежать від формату і синтаксичної коректності вектору URL-адреси чи веб-сайту. Наприклад, у правильно побудованій URL-адресі домен верхнього рівня (.com, .net, .ua) з'являється лише один раз. Однак це не завжди так у випадку шкідливої URL-адреси. Ця проблема зв'язана з синтаксисом URL-адрес, тому розглядаємо позицію домену верхнього рівня як синтаксичну особливість. Досліджуючи вміст веб-сайту,

можемо підрахувати усі теги, що являється ще одною синтаксичною особливістю.

Семантичні функції зосереджені на значенні та інтерпритації текстового вмісту в URL-адресах та веб-сайтах. Прикладом семантичної функції для веб-сайту може бути значення елементів HTML, наприклад кількість прихованих об'єктів.

Прагматичні функції не мають прямого відношення до синтаксису чи семантиці URL-адреси чи веб-сайту. Наприклад, відключення натискання правої кнопки миші на веб-сайтах – це метод, що використовується зловмисниками для попередження перегляду і збереження вихідного коду користувачами. Це не має відношення ні до синтаксису вмісту HTML, ні до його семантики. Таким чином, це відноситься до категорії прагматичних функцій. Іншими прикладами прагматичних функцій є відомості про реєстрацію веб-сайту, термін дії веб-сайту і т.д.

Представлено класифікацію сайтів для виявлення фішингового вмісту. На відміну від поділу фішингового ресурсу за вектором та типом фішингової атаки, дана класифікація охоплює особливості, які можуть бути добуті з вектору атаки. Майбутня робота включає пошук способів визначення будь-якого типу фішингових сайтів якнайшвидше.

Література:

1. The 2018 Phishing Trends & Intelligence Report. URL: https://info.phishlabs.com/2018_phishing_trends_and_intelligence_report-0.
2. Patrick Nohe. HTTPS Phishing: 49% of Phishing Websites Now Sport The Green Padlock. URL: <https://www.thesslstore.com/blog/https-phishing-green-padlock/>.
3. A.-P. W. Group. Phishing Activity Trends Report-1st Quarter 2019. URL: https://docs.apwg.org/reports/apwg_trends_report_q1_2019.pdf.
4. M. Vijayalakshmi, S. Mercy Shaline, Ming Hour Yang, Raja Meenakshi U. Web phishing detection techniques: a survey on the state-of-the-art, taxonomy and future directions. URL: <https://doi.org/10.1049/iet-net.2020.0078>.
5. J. Mao, W. Tian, P. Li, T. Wei and Z. Liang. «Phishing-Alarm: Robust and Efficient Phishing Detection via Page Component Similarity» in IEEE Access, vol. 5, pp. 17020-17030, 2017, doi: 10.1109/ACCESS.2017.2743528. URL: <https://ieeexplore.ieee.org/abstract/document/8015116>.

Жакун Г.А., студент 6 курсу, факультет інформаційних технологій та комп'ютерної інженерії, кафедра захисту інформації, Вінницький національний технічний університет, м. Вінниця;
Лукічов В.В., к.т.н., старший викладач, факультет інформаційних технологій та комп'ютерної інженерії, кафедра захисту інформації, Вінницький національний технічний університет, м. Вінниця

СИСТЕМА ВИЯВЛЕННЯ АТАК З ВИКОРИСТАННЯМ HONEYROT

В сучасному світі доволі часто проводять Bruteforce атаки і взлам пароля з допомогою соціальної інженерії. Незважаючи на десятиліття досліджень і досвіду, досі все ще не вдається створити безпечні комп'ютерні системи або навіть виміряти безпеку системи.

В результаті використання нововиявлених вразливостей часто проводять атаки нульового дня. Завдяки автоматизації експлуатації та масштабному глобальному пошуку вразливостей, супротивники часто можуть піти на компроміс невдовзі після того, як вразливості стали відомі [1].

Одним із способів завчасного попередження про нові вразливості є встановлення комп'ютерних систем у мережі з очікуванням, що буде проводитися проникнення. Ці системи не мають інших законних функцій, і кожна спроба підключитися до них є підозрілою. Дану систему називають Honeyrot. Honeyrot може працювати з будь-якою операційною системою та будь з якою кількістю послуг.

Налаштовані служби визначають місця, де противник може вибрати скомпрометувати систему. Honeyrot з високою взаємодією імітує всі аспекти операційної системи, тоді як honeypot з низькою взаємодією імітують лише деякі частини, наприклад мережевий стек [2]. Також розрізняється фізичні і віртуальні honeypot. Фізичний honeypot існує як машина з відповідною IP-адресою в мережі, тоді як віртуальний honeypot розміщений на іншій машині, яка реагує на мережевий трафік, спрямований на віртуальний Honeyrot. Віртуальні Honeyrot приваблюють тим, що вони це роблять не потребує додаткових комп'ютерних систем. Використовуючи віртуальні honeypot, можна заповнити мережу з хостами, що працюють над різними операційними системами. Однак, щоб переконати супротивників у тому, що віртуальний honeypot працює під певною операційною системою, необхідно ретельно моделювати стек TCP/IP цільової операційної системи. По іншому є необхідність вміти підмінювати інструменти відбитків пальців стека TCP/IP, наприклад Xprobe [1] або Nmap [3].

Доволі часто Honeyrot дає змогу тільки направити зловмисника в заздалегідь заготовлене середовище та дослідити його атаку. Також це дає змогу зрозуміти які способи та методи атаки були використані. Тільки дана методика не дає можливості виявити зловмисника а в окремих випадках Honeyrot вдається розкрити. Тому було запропоновано об'єднати системи моніторингу для виявлення атак та Honeyrot.

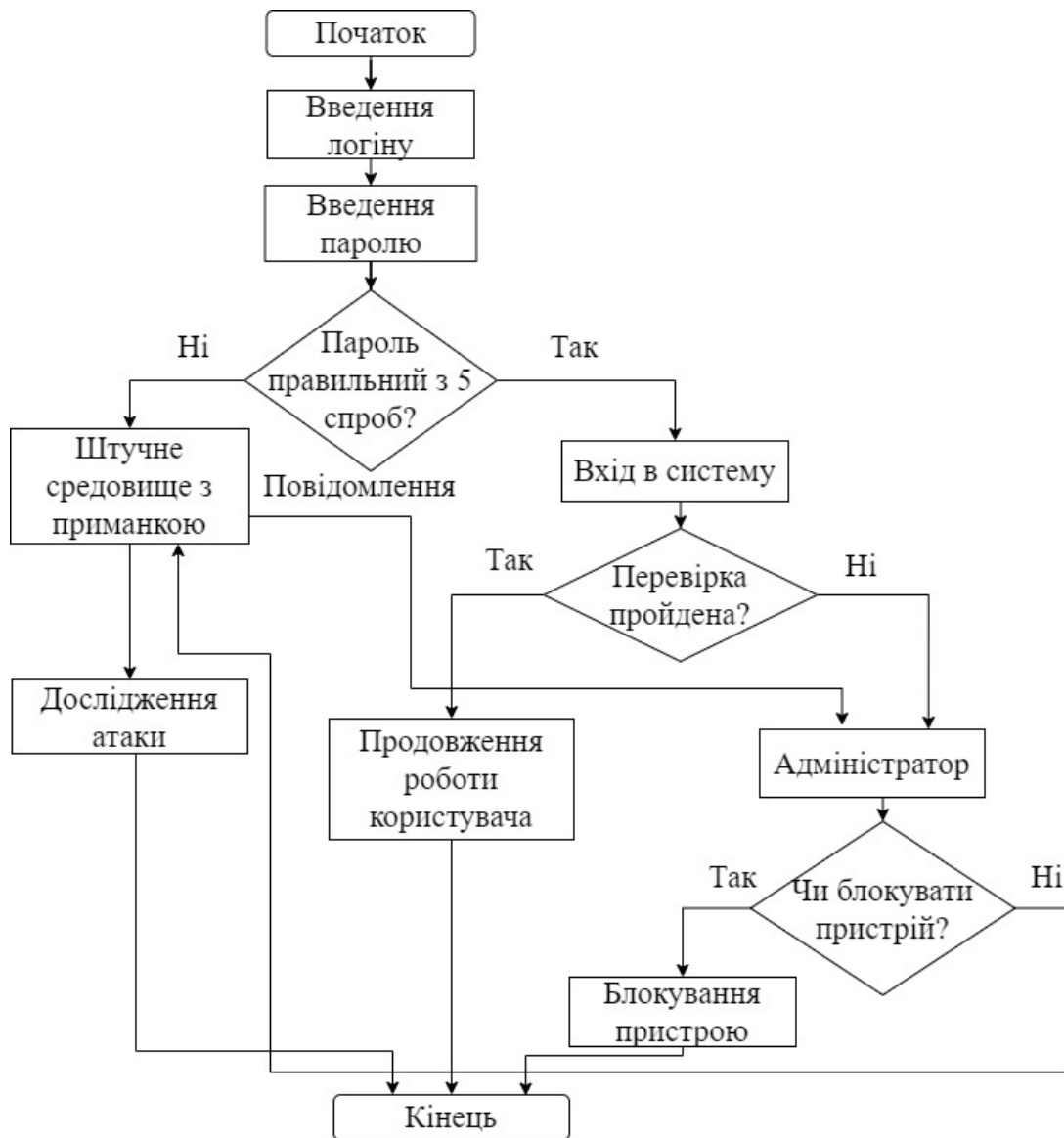


Рисунок 1 – Блок-схема виявлення атаки з використанням Honeypot.

На рисунку 1 продемонстровано програмний модуль, який об'єднує Honeypot та систему виявлення атак. Перевіркою буде дії користувача, а також спеціальні файли, які не повинні бути відкриті або змінені Honeypot, додатково будуть використовуватися дані про особливості користувача. Функція блокування буде запускатися в тому випадку, коли Адміністратор не прийме рішення за певний період часу. В такому випадку навіть якщо зловмисник проникне в систему, буде можливість його виявити та прийняти дії, Honeypot буде надавати можливість дослідити атаки. Також зловмисник який виявить honeypot не буде очікувати підміни файлів в системі.

Висновком даного рішення є можливість виявити зловмисника в системі. Також ця система має викликати у зловмисника сумніви в тому чи інформація, яка буде скопійована або розкрита чи є інформація достовірною, і чи зловмисника не було розкрито. Дана система вирішує проблему Honeypot, коли виявляється підміна середовища або можливість відкриття користувачем.

Література:

1. Lance Spitzner. Honeypots: Tracking Hackers. Addison Wesley Professional, September 2002.
2. Stuart Staniford, Vern Paxson, and Nicholas Weaver. How to Own the Internet in your Spare Time. In Proceedings of the 11th USENIX Security Symposium, August 2002.
3. Fyodor. Remote OS Detection via TCP/IP Stack Fingerprinting. <http://www.nmap.org/nmap/nmap-fingerprinting-article.html>, October 1998. 1, 6

*Кметь О.І., магістр, кафедра електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

МУРАШИНИЙ АЛГОРИТМ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧ МАРШРУТИЗАЦІЇ ТРАНСПОРТУ

В даний час йде стрімка зміна інфокомунікаційних технологій (ІКТ). Зміни, що виникають, зачіпають мережеві технології, власне – обчислювальні і комунікаційні пристрої, а також обробку даних. В результаті, інформаційні технології застосовуються у всебільшій кількості сфер життя і господарського життя людини. Однією з актуальних областей наукових досліджень є сфера життєвого оточення, яка з області Smart House розвивається в даний час в область Smart city, Smart transport system и т.п. Таким чином, у органів міського управління виникають нові завдання, які повинні не тільки вирішити цілий комплекс проблем, що виникають, а й провести кардинальну трансформацію міст. Комплекс проблем у всіх міст наступний:

- транспортні проблеми;
- екологічні проблеми;
- соціальні проблеми, пов'язані з ростом злочинності і соціальної напруженості;
- обмеження природних ресурсів;
- зникнення культурної та історичної спадщини.

Важливим моментом є детальний аналіз, розуміння даних проблем, а також можливість розгляду різних варіантів рішення. Всі перераховані проблеми, як результат активної урбанізації, є основними тригерами і вимушеними процесами розвитку міст і їх трансформації в Smart city. Рішенням цих проблем може бути застосування нової моделі розвитку міст – реалізація концепції Smart city, яка в своїй основі застосовує інфокомунікаційні технології для вирішення всіх сфер життєдіяльності населення. Було виявлено, що на поточному етапі не існує універсальної моделі Smart city і точного її визначення. Модель є сучасною стратегією об'єднання різноманітних факторів міського розвитку, спрямована на модернізацію інфраструктури з принципово новими можливостями централізованого управління, новим рівнем послуг і безпеки.

Однією з основних проблем в контексті всього міста, було виявлено цілий

ряд проблем у транспортній сфері міста: зростання заторів, проблеми в екологічній сфері, негативний ефект на економіку, низький рівень якості громадського транспорту, відсутність відповідності між транспортним і містобудівним плануванням, відсутність управління транспортними потоками і паркувальними місцями, і т.д. Один із шляхів рішень – розробка інтелектуальної транспортної системи (ІТС). Системи розумної зупинки є компонентою ІТС. Модульність системи розумної зупинки дозволяє створити базу для використання різних одиниць ІКТ в залежності від реалізованих ініціатив або ж конкретних завдань міста. Такий принцип дає можливість поєднати різне обладнання в складі єдиного апаратно-програмного комплексу.

Література:

1. Churyumov G., Tokariiev V., Tkachov V. Problem of self-organization of s-bot group movement in unorganized physical environment / G. Churyumov, V. Tokariiev, V. Tkachov // Комп'ютерні та інформаційні системи і технології: тези доповідей третьої міжн. наук.-техн. конф., 23 - 24 квіт. 2019 р. – Харків, 2019. - С.16-17.
2. Серков О.А., Князев В.В., Лазуренко Б.О., Яковенко І.В., Чурюмов Г.І., Токарев В.В. Надширокосмугові технології в задачах забезпечення електромагнітної сумісності рухомих об'єктів / О.А. Серков, В.В. Князев, Б.О. Лазуренко, І.В. Яковенко, Г.І. Чурюмов, В.В. Токарев // Проблеми електромагнітної сумісності перспективних бездротових мереж зв'язку (ЕМС-2019): збірник наукових робіт четвертої міжн. наук.-техн. конф., 24 жовт. 2019 р. - Харків, 2019. - С. 55-57.
3. Krivoulya G., Ilna I., Tokariiev V., Shcherbak V. Mathematical Model for Finding Probability of Detecting Victims of Man-Made Disasters Using Distributed Computer System with Reconfigurable Structure and Programmable Logic / G. Krivoulya, V. Tokariiev, I. Ilna, V. Shcherbak // IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology: (PIC S&T), 06-09 oct. 2020y. - Kharkiv, 2020. - P.573 - 576.

Науковий керівник: Токарев Володимир Володимирович, кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки

*Колпак М.В., студент, кафедра інформатики та програмної інженерії,
Національний технічний університет України «Київський політехнічний
інститут імені Ігоря Сікорського», м. Київ*

ПОВНОТЕКСТОВИЙ ПОШУК У ВІДСКАНОВАНИХ ДОКУМЕНТАХ В СИСТЕМАХ ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ

Використання систем електронного документообігу вимагає додавання до системи вхідної паперової документації: листів, договорів, тощо. Зазвичай їх оцифровують за допомогою сканування і в подальшому використовують ці

скановані зображення. Це зумовлює проблеми взаємодії системи з змістом цих документів, в тому числі повнотекстового пошуку по ним.

Метою даної роботи є вдосконалення повнотекстового пошуку в системах електронного документообігу та електронних архівах шляхом використання машинного розпізнавання тексту.

Для досягнення поставленої мети було виокремлено наступні завдання:

- Дослідити сучасні підходи до повнотекстового пошуку у відсканованих документах та шляхи їх інтеграції в системи електронного документообігу
- На основі аналізу обрати алгоритм розпізнавання текстового зображення для подальшого формування бази даних текстових документів
- Обрати алгоритм повнотекстового пошуку за неповними вхідними даними
- Запропонувати архітектурне рішення, яке дозволить проводити повнотекстовий пошук зі збереженням початкового інтерфейсу взаємодії з користувачем

Наразі поширеними системами розпізнавання є Azure CognitiveServices, tessnet2, Puma.net, ABBYY Cloud OCR. В той же час, більшість існуючих систем допускають артефакти, які унеможливають автоматичний перехід між форматами представлення. Водночас, отриманої точності достатньо для забезпечення пошуку. Тому в роботі пропонується архітектура, яка дозволить використати існуючі бібліотеки розпізнавання для вирішення цієї задачі (Рисунок 1).

Запропонована архітектура передбачає створення окремого мікросервісу повнотекстового пошуку по сканованих зображеннях. Зв'язок з основної частини системи електронного документообігу з мікросервісом здійснюється за допомогою транспортної шини, у якості якої може виступати наприклад RabbitMQ або Azure Service Bus. Сам сервіс складається з ізольованих модулів пошуку та розпізнавання. Модуль пошуку містить базу даних з індексованими розпізнаними документами. Може бути імплементовано декілька модулів розпізнавання, що реалізують одне АПІ, але використовують різні алгоритми; в такому випадку можна реалізувати динамічний вибір відповідного модуля, в залежності від його ефективності роботи з необхідним форматом документу, що дозволяє збільшити загальну точність розпізнавання.

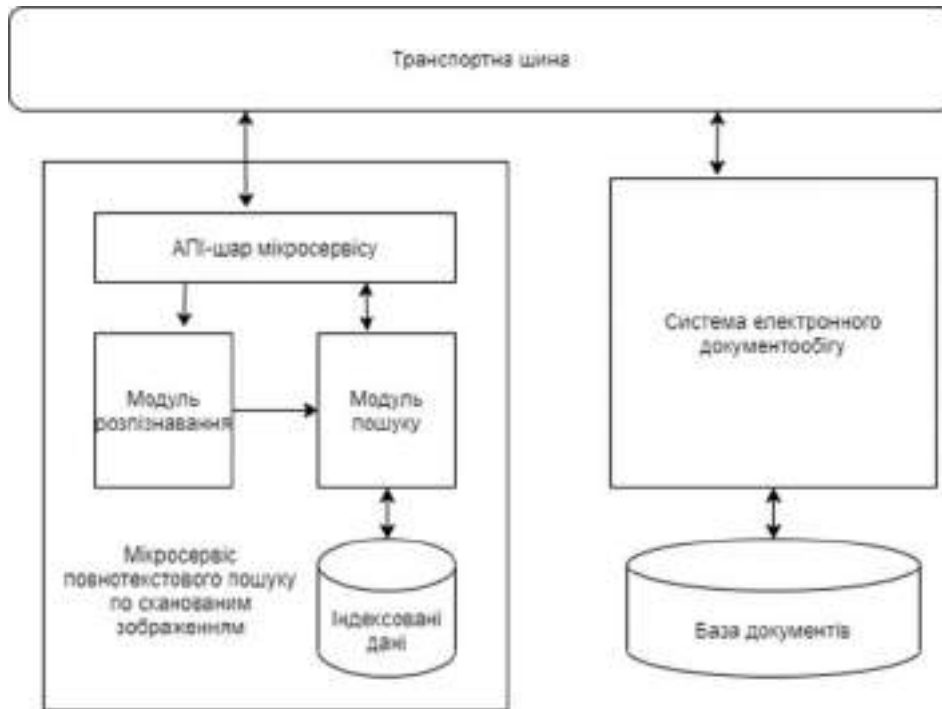


Рисунок 1 - Запропонована архітектура

Таким чином, збереження текстового представлення в окремій базі даних дозволяє уникнути повторного розпізнавання. Крім того, отримані в результаті розпізнавання документи можуть в подальшому бути використані як базис для створення текстової версії.

Література:

1. Дино Эспозито Microsoft .NET: архітектура корпоративних приложень / Дино Эспозито, Андреа Сальтарелло // Вільямс, 2016
2. Alessandro Del Sole Microsoft Computer Vision APIs Distilled / Alessandro Del Sole // - Cremona, Italy. - Appress, 2018

Науковий керівник: Мажара Ольга Олександрівна, кандидат технічних наук, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

Красовська Є.В., к.т.н., Відокремлений структурний підрозділ «Фаховий коледж інженерії та управління НАУ», м. Київ;
Красовський О.Д. студент, Відокремлений структурний підрозділ «Фаховий коледж інженерії та управління НАУ», м. Київ

СУЧАСНІ ТЕНДЕНЦІЇ ЦИФРОВОЇ ТРАНСФОРМАЦІЇ БАЗ ДАНИХ

Світова пандемія, викликана Covid-19, не може вплинути на уповільнення цифрової трансформації. Насправді, якраз навпаки: багато компаній швидко переходять в «хмару» або роблять перші кроки в ІТ-напрямку, машинного

навчання та аналізу даних. Фундаментальне питання стосується вибору баз даних, важливості «хмарних» технологій і Graph Data Science, а також нових рольових моделей в компаніях. Тут можна виділити чотири чітких тенденції.

Для цифрової трансформації дані є основною вимогою для всіх процесів і систем. Сьогодні мова йде не стільки про збір даних, скільки про пов'язування даних один з одним. Розрізнені сховища даних руйнуються, і множини даних формують ІТ-ландшафт. Ключова роль мережевих даних повільно, але вірно підриває новаторську роль реляційних баз даних (SQL). Коли справа доходить до використання даних для складних завдань, все частіше використовуються інші типи баз даних – від прогнозної аналітики й оброблення даних до процесів машинного навчання і штучного інтелекту.

Бази даних NoSQL в останні роки не лише подаються як багатообіцяюча альтернатива реляційних баз даних, але також зайняли постійне місце в ІТ-інфраструктурі компаній. Сюди входять бази даних ключових значень, бази даних документів, для мережевих даних, – бази даних графів.

Для майбутнього розвитку графових баз даних з'являються чотири основні тенденції: модель графа з властивостями стає все більш усталеною, розробники все більше переходять на роль рушійних сил інновацій, бази даних переходять в «хмару», а наука про графові бази даних стає ключем до майбутніх додатків.

1. Модель графа з властивостями. Останні роки показали, наскільки насправді важливим є аналіз мережевих даних – від пошуку вакцини проти Covid-19 до забезпечення безперебійних ланцюжків поставок і управління віддаленим робочим простором для клієнтів, співробітників і партнерів. Фактично, більшість моделей даних і предметних областей за своєю природою об'єднані в мережу. На практиці протягом довгого часу робилися спроби інтегрувати таку мережу відносин в реляційні бази даних, що витратило багато місця для зберігання, великих зусиль з точки зору продуктивності.

Модель графа з властивостями з'являється як логічна альтернатива. Він спеціально розроблений для зберігання, зв'язування і запиту мережевих даних. Це не означає, що він негайно замінить реляційні або інші типи баз даних. Моделі графів з властивостями використовуються на додаток до інших баз даних. Ізоморфна модель гарантує, що дані можуть бути перетворені з однієї моделі в іншу без втрат. Тому компаніям доведеться відповідально підходити до питання, яка модель даних підійде для яких даних і для якого додатка. В цілому, однак, модель графа з властивостями і використання технології графів все частіше прокладають шлях для інноваційних та інтелектуальних проєктів.

2. Розробники як рушійні сили інновацій. У час «хмар», контейнерних додатків стратегія зростання багатьох компаній така: розробники систем та ІТ-фахівці стають важливими особами, які приймають рішення. Саме вони стимулюють цифрову трансформацію в компаніях. Нерідко справжні інновації виходять від сотень або навіть тисяч співробітників, які стикаються з повсякденними завданнями і працюють над новими, ефективними рішеннями, заснованими на даних.

Одна з причин зміни ролей – зростаюча фрагментація середовища розробки, включаючи мови програмування і фреймворки. Ця фрагментація – проблема не тільки для сьогоденних розробників. Додатки, які впроваджуються сьогодні, також зажадають підтримки і обслуговування в майбутньому. До мов програмування майбутнього відносяться ті, які підтримують широкий спектр сценаріїв використання і платформ. Сюди входять Java і Python, а також мови, адаптовані для спеціальних типів баз даних.

3. Хмарні додатки і PaaS. Загальнодоступні «хмарні» сервіси повинні нести спільну відповідальність за інновації в області даних і аналітики, компанії повинні стимулювати зміни та інновації на основі даних, щоб мати можливість реагувати на кризи та економічні спади.

Це також стосується і PaaS. Багато компаній спочатку неохоче купували обладнання і програмне забезпечення повністю у одного постачальника і, таким чином, ставали залежними. Тим часом, однак, переваги для багатьох переважають, включаючи значно більш швидку розробку власних програм і зниження витрат. Ще одна важлива причина переходу в «хмару»: високий рівень зручності використання. Розробники можуть зосередитися на програмуванні додатків, не турбуючись про управління інфраструктурою.

На додаток до класичних сервісів баз даних, таких як DBaaS, сховище даних і безсерверні обчислення, графові бази даних також приєднуються до тенденції «як послуга» і переходять в «хмару». Для багатьох компаній це може бути першим випадком застосування графової технології. Той факт, що зараз вони роблять це в «хмарі», показує, наскільки важливо для розробників та ІТ-груп тестувати нові способи вирішення завдань з великим об'ємом даних, зберігаючи при цьому контроль над складною архітектурою графів.

4. Графові бази даних як наука для машинного навчання і штучного інтелекту. Існує ще одна тенденція щодо баз даних: графові технології як основа для сучасної аналітики даних. Наука про дані і роль фахівців з обробки баз даних, в свою чергу, набувають все більшого значення в контексті машинного навчання і штучного інтелекту. Graph Data Science (GDS) об'єднує ці дві розробки.

GDS – це тільки наступний крок в аналізі мережевих структур в складних наборах даних. Прості запити дозволяють фахівцям з обробки даних отримати початковий огляд своїх даних і їх взаємозв'язків. Однак, як правило, вони є лише відправною точкою для більш глибоких і широких досліджень. Насправді нові знахідки, приховані закономірності і невідомі відхилення можуть бути ідентифіковані тільки за допомогою графових алгоритмів. При цьому можна автоматично виявляти структури і взаємозв'язки, які в основному залишаються невикористаними або не виявленими при звичайних запитах. Залежно від пошукового запиту і завдання, алгоритми графа можуть бути згруповані за різними категоріями: пошук шляху, центральність, виявлення спільноти, прогнозування посилань і схожість.

Зазначений підхід йде ще далі з використанням так званих вкладених графів. Цей підхід використовує алгоритми інформаційних технологій, і

забезпечує машинозчитувані результати. Процеси машинного навчання працюють на цій основі і оптимізують, наприклад, точність прогнозування. Складна структура графа абстрагується, і його розмірність знижується. Простіше кажучи, Graph Data Science допомагає користувачам легше і швидше будувати прогностичні моделі. Google і Facebook – відомі компанії, які вже успішно використовують вкладені графи. Завдяки інтегрованим вкладенням графів в сучасні бази даних графів (з Neo4j GDS 1.4) процеси машинного навчання графів стають повністю доступними для всіх компаній.

Література:

1. Knowledge Graphs: Data in Context for Responsive Businesses. By Jesús Barrasa, Amy E. Hodler, and Jim Webber – [Електронний ресурс]. – Режим доступу: <https://neo4j.com/knowledge-graphs-data-in-context-for-responsive-businesses/> (2021).
2. Graph Data Science For Dummies By Amy Hodler and Mark Needham – [Електронний ресурс]. – Режим доступу: <https://neo4j.com/graph-data-science-for-dummies/> (2021).
3. Graph Databases For Dummies By Dr. Jim Webber and Rik Van Bruggen – [Електронний ресурс]. – Режим доступу: <https://neo4j.com/graph-databases-for-dummies/> (2021).
4. Graph Algorithms: Practical Examples in Apache Spark and Neo4j By Mark Needham and Amy E. Hodler – [Електронний ресурс]. – Режим доступу: <https://neo4j.com/graph-algorithms-book/> (2021).

*Крят Д.С., магістр, кафедра електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

МОДЕЛЬ МОБІЛЬНОЇ СЕНСОРНОЇ МЕРЕЖІ ДЛЯ VANET

Літаючі сенсорні мережі FUSN (Flying Ubiquitous Sensor Network) є одним з класів бездротових або всепроникаючих сенсорних мереж WSN / USN. Технологія даних мереж заснована на самоорганізаційному об'єднанні безлічі різних датчиків з низьким енергоспоживанням в мережу і їх розміщенні в важкодоступних місцях. Передача даних здійснюється за допомогою протоколів ZigBee, Bluetooth, 6LoWPAN. Зародження сенсорних мереж почалося в другій половині XX століття, а саме – в 1980 році, коли американське оборонне агентство DARPA (Defense Advanced Research Project Agency) почало дослідження за програмою «Розподілені сенсорні мережі» DSN (Distributed Sensor Networks). Основне завдання даної програми полягало в перевірці можливості застосування нового підходу для машинної взаємодії, введеного вперше в ARPANET (попередник Інтернету). Дослідникам DARPA було необхідно спроектувати мережу з зонально-розподілених датчиків, які повинні були бути недорогими, працювати автономно і обмінюватися даними

незалежно один від одного. Такі вимоги до цих пір застосовуються для розробки сучасних сенсорних мереж. У той час апаратна база була слабкою і недосконалою для реалізації сенсорних мереж, тому учасники програми DNS повинні були не тільки визначити концепцію сенсорних мереж і самих сенсорів, а й здійснювати безпосередньо саму технічну розробку. Серед найважливіших областей досліджень, на думку розробників DNS, були виявлені: обробка сигналів, розподілені обчислення і передача інформації по бездротових каналах зв'язку. Ad-Нос-мережі також почали активно впроваджуватися в усі сфери суспільства і розділилися на кілька видів рис.1:

- мобільні цільові мережі MANET (Mobile Ad-Hoc Network);
- автомобільні цільові мережі VANET;
- літаючі цільові мережі FANET.



Рис.1. Структура цільових мереж MANET, VANET и FANET

Автомобільні цільові мережі VANET є одним з видів бездротових сенсорних мереж і призначені для забезпечення і підтримки безпеки на дорогах, дозволяючи транспортним засобам обмінюватися між собою даними за допомогою V2V-комунікацій (Vehicle-to-Vehicle) або шляхом підключення до прилеглої фіксованою інфраструктури V2I (Vehicle-to-Infrastructure).

Література:

1. Churyumov G., Tokariev V., Tkachov V. Problem of self-organization of s-bot group movement in unorganized physical environment / G. Churyumov, V. Tokariev, V. Tkachov // Комп'ютерні та інформаційні системи і технології: тези доповідей третьої міжн. наук.-техн. конф., 23 - 24 квіт. 2019 р. – Харків, 2019. - С.16-17.
2. Серков О.А., Князев В.В., Лазуренко Б.О., Яковенко І.В., Чурюмов Г.І., Токарєв В.В. Надширокосмугові технології в задачах забезпечення електромагнітної сумісності рухомих об'єктів / О.А. Серков, В.В. Князев, Б.О. Лазуренко, І.В. Яковенко, Г.І. Чурюмов, В.В. Токарєв // Проблеми електромагнітної сумісності перспективних бездротових мереж зв'язку (EMC-2019):збірник наукових робіт четвертої міжн. наук.-техн. конф., 24 жовт. 2019 р. - Харків, 2019. - С. 55-57.

3. Krivoulya G., Ilina I., Tokariyev V., Shcherbak V. Mathematical Model for Finding Probability of Detecting Victims of Man-Made Disasters Using Distributed Computer System with Reconfigurable Structure and Programmable Logic / G. Krivoulya, V. Tokariyev, I. Ilina, V. Shcherbak // IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology: (PIC S&T), 06-09 oct. 2020y. - Kharkiv, 2020. - P.573 - 576.

Науковий керівник: Токаре́в Володимир Володимирович, кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки

Ланцюгов Д.К., магістр, кафедра електронних обчислювальних машин, Харківський національний університет радіоелектроніки, м. Харків

МОДЕЛЬ ДОСТАВКИ ДАНИХ З WEARABLE THINGS НА БАЗІ DRONE

Існуючі системи ідентифікації і управління інформацією в мережі засновані на класичній клієнт-серверній архітектурі рис.1. Сервер в такій системі є місцем зберігання інформації і обробки запитів від клієнтів на роботу з даною інформацією. DOA, на відміну від такого підходу прагне вирішити питання не про локалізацію, а про контекст цифрового об'єкта.



Рис.1. Приклад архітектури цифрового об'єкта

Цифровий об'єкт в цій архітектурі характеризується не тільки інформацією про своє місцезнаходження. Крім цього, існує можливість отримувати різні відомості про сам об'єкт:

- вимоги до доступу;

- аутентифікації;
- інформацію про автора та інше.

Вся ця інформація вноситься самим творцем цифрового об'єкта. Для цього в архітектуру DOA інтегрована спеціальна інфраструктура, що забезпечує необхідне шифрування і верифікацію доступу.

Основними структурними елементами DOA є цифровий об'єкт, система резолюції ідентифікатора (Handle System) та репозиторій і реєстр цифрових об'єктів. Зупинимося на принципах система резолюції докладніше.

Кожному цифровому об'єкту в описуваній архітектурі ставиться у відповідність унікальний ідентифікатор – DOI (від англ. Digital Object Identifier). Даний ідентифікатор чимось нагадує URL, на базі якого побудований сучасний Інтернет. Однак, на відміну від останнього, присвоєнні ідентифікатори залишаються постійними і не залежать від стану цифрового об'єкта. Саме система резолюції пов'язує ідентифікатор з інформацією про поточний статус цифрового об'єкта (місцезнаходження, доступ, інформація про автентичність). У класичній архітектурі DOA система резолюції є дворівневою:

- першим рівнем резолюції є глобальний реєстр (GHR, від англ. Global Handle Registry);
- другим рівнем – набір локальних реєстрів (LHR, от англ. Local Handle Registry) або локальних сервісів (LHS, від англ. Local Handle Service).

Література:

1. Churyumov G., Tokariev V., Tkachov V. Problem of self-organization of s-bot group movement in unorganized physical environment / G. Churyumov, V. Tokariev, V. Tkachov // Комп'ютерні та інформаційні системи і технології: тези доповідей третьої міжн. наук.-техн. конф., 23 - 24 квіт. 2019 р. – Харків, 2019. - С.16-17.
2. Серков О.А., Князєв В.В., Лазуренко Б.О., Яковенко І.В., Чурюмов Г.І., Токареєв В.В. Надширокосмугові технології в задачах забезпечення електромагнітної сумісності рухомих об'єктів / О.А. Серков, В.В. Князєв, Б.О. Лазуренко, І.В. Яковенко, Г.І. Чурюмов, В.В. Токареєв // Проблеми електромагнітної сумісності перспективних бездротових мереж зв'язку (EMC-2019): збірник наукових робіт четвертої міжн. наук.-техн. конф., 24 жовт. 2019 р. - Харків, 2019. - С. 55-57.
3. Krivoulya G., Ilina I., Tokariev V., Shcherbak V. Mathematical Model for Finding Probability of Detecting Victims of Man-Made Disasters Using Distributed Computer System with Reconfigurable Structure and Programmable Logic / G. Krivoulya, V. Tokariev, I. Ilina, V. Shcherbak // IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology: (PIC S&T), 06-09 oct. 2020y. - Kharkiv, 2020. - P.573 - 576.

Науковий керівник: Токареєв Володимир Володимирович, кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки

СИСТЕМА ІДЕНТИФІКАЦІЇ ОБ'ЄКТІВ В INTERNET

Система резолюції (Handle System) була створена, щоб подолати обмеження функціональності існуючих систем ідентифікації об'єктів в Internet. Резолюція – це процес, в якому ідентифікатор є запитом до мережевого сервісу на отримання актуальної інформації (даних про стан), що відноситься до обумовленої суті (найчастіше мова йде про місцезнаходження).

Система резолюції підтримує множинну резолюцію, тобто відповіддю на запит може бути розташування різних примірників об'єкта, пов'язані сервіси і будь-яка інша інформація, зазначена в метаданих об'єкта. Інформація, що повертається, необов'язково повинна вказувати на екземпляр об'єкта: наприклад, це може бути опис або стан об'єкта, деякі індикатори або вимірювання, відносини з іншими сутностями і т.д.

Handle – глобально унікальний і розв'язний ідентифікатор. Він представляється в такий спосіб: «префікс/суфікс», де префікс унікальний в межах Системи резолюції. Приклад Handle зображений на рис.1.



Рис. 1. Приклад Handle

Система резолюції надає доступ клієнта до місця розташування цифрового об'єкта. Для цього використовується ієрархічна модель обслуговування, що складається з глобального реєстратора Handle GHR, від (англ. Global Handle Registry) (GHR) і локальної системи обробки Handle LHS від (англ., Local Handle Registry) (LHR).

Кожна служба локальної дескрипції може містити свою власну ієрархію Handle сервісів: GHR містить інформацію про зіставлення префікса Handle для LHS, який обслуговує Handle для даного префікса. GHR відповідає за управління коренем ієрархії дескрипторів системи, виділяючи унікальні префікси і надаючи глобальну службу прив'язки префіксів до LHS цього префікса.

Кожному об'єкту в системі резолюції приписаний ряд обов'язкових атрибутів. Інформація, яка повертається, не обов'язково повинна вказувати власне на екземпляр об'єкта: наприклад, це може бути опис або стан об'єкта,

деякі індикатори або вимірювання, відносини з іншими сутностями і т.д. Для обробки запиту через протокол http система використовує проксі-сервери, які обробляють хендели, представлені у вигляді URL. Варіанти резолюції при цьому зберігаються в хендл-записи у вигляді xml-файлу. Таким чином, проксі-сервери, є надбудовою над системою резолюції.

Література:

1. Churyumov G., Tokariev V., Tkachov V. Problem of self-organization of s-bot group movement in unorganized physical environment / G. Churyumov, V. Tokariev, V. Tkachov // Комп'ютерні та інформаційні системи і технології: тези доповідей третьої міжн. наук.-техн. конф., 23 - 24 квіт. 2019 р. – Харків, 2019. - С.16-17.
2. Серков О.А., Князев В.В., Лазуренко Б.О., Яковенко І.В., Чурюмов Г.І., Токарев В.В. Надширокосмугові технології в задачах забезпечення електромагнітної сумісності рухомих об'єктів / О.А. Серков, В.В. Князев, Б.О. Лазуренко, І.В. Яковенко, Г.І. Чурюмов, В.В. Токарев // Проблеми електромагнітної сумісності перспективних бездротових мереж зв'язку (EMC-2019):збірник наукових робіт четвертої міжн. наук.-техн. конф., 24 жовт. 2019 р. - Харків, 2019. - С. 55-57.
3. Krivoulya G., Ilna I., Tokariev V., Shcherbak V. Mathematical Model for Finding Probability of Detecting Victims of Man-Made Disasters Using Distributed Computer System with Reconfigurable Structure and Programmable Logic / G. Krivoulya, V. Tokariev, I. Ilna, V. Shcherbak // IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology: (PIC S&T), 06-09 oct. 2020y. - Kharkiv, 2020. - P.573 - 576.

Науковий керівник: Токарев Володимир Володимирович, кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки

*Матвійчук Є.Є., студент, кафедра автоматизованих систем управління,
Національний університет «Львівська політехніка», м. Львів*

РЕАЛІЗАЦІЯ ВЕБ-ОРІЄНТОВАНОГО ПРОГРАМНОГО РІШЕННЯ ДЛЯ НАВЧАННЯ СТУДЕНТІВ ПРОГРАМУВАННЮ

Сучасні веб-орієнтовані програмні рішення за природою є клієнт-серверними системами. Клієнтом у цьому випадку є веб-застосунок, який завантажується, запускається та функціонує у браузері комп'ютера. Сервером виступає застосунок, який працює на віддаленому комп'ютері. Його роль – опрацювання запитів користувача, робота з базою даних.

Для реалізації веб-орієнтованого програмного рішення для навчання студентів програмуванню було обрано наступні програмні засоби та мови програмування.

Клієнтська частина була розроблена з використання мов HTML, CSS та TypeScript. У якості архітектури веб-додатку було використано SPA (Single Page Application) – принцип, в якому веб-сторінка завантажується один раз, а подальші дії користувача оброблюються засобами JavaScript та не потребують перезавантаження сторінки. Дані для відображення завантажуються з сервера у форматі JSON, використовуючи мережевий протокол HTTP/HTTPS та AJAX запити або спеціальний засіб сучасних браузерів – fetch api. Також було застосовано модульну архітектуру додатку та розділення на шари представлення, бізнес логіки та доступу до даних.

Для реалізації серверної частини було створено WEB API за допомогою мови програмування C# та фреймворку .NET Core 5. ASP.NET Core – вільне та відкрите програмне забезпечення для створення веб застосунків, що є частиною .NET Core [7]. Фреймворк .NET Core, створений компанією Microsoft, розроблявся для запровадження сумісності з операційними системами Linux, Windows та macOS. На основі цієї технології можна створювати портативні додатки. ASP.NET Core включає в себе фреймворк MVC, який об'єднує технології MVC, Web API та Web Pages [1]. Побудований фреймворк на принципі низької зв'язності компонент, тому функціонал готових продуктів легко розширювати користувачькими бібліотеками, які називаються NuGet-пакети.

Також було застосовано принципи побудови REST API, використано розділення коду на шари представлення, бізнес логіки та доступу до даних. Застосовані популярні шаблони проектування Dependency Injection та Inversion of Control, Repository Pattern. Використано бібліотеки AutoMapper для приведення моделей між архітектурними шарами додатку та бібліотеку EntityFrameworkCore для спрощення побудови бази даних та виконання запитів до бази даних. Для побудови моделі даних було використано підхід Code First – принцип, за яким контроль за версіями схеми бази даних здійснюється у вигляді міграцій, ідея яких полягає у представленні моделей даних у вигляді класів .Net, які накладаються на модель даних SQL, співставляються та автоматично генерується код, який нівелює різницю між ними.

Для збереження та управління даними користувачів було використано Microsoft SQL Server. Мова, що використовується для запитів – Transact-SQL. SQL Server 2005 має вбудовану підтримку .NET Framework. Завдяки цьому, процедури бази даних, що зберігаються, можуть бути написані на будь-якій мові платформи .NET з використанням повного набору бібліотек, доступних для .NET Framework. На відміну від інших процесів, .NET Framework виділяє додаткову пам'ять і будує засоби керування SQL Server, не використовуючи вбудовані засоби Windows. Це підвищує продуктивність порівняно із загальними алгоритмами Windows, оскільки алгоритми розподілу ресурсів спеціально налагоджені для використання у структурах SQL Server. [2]

На рис. 1 наведена діаграма зв'язків бази даних.

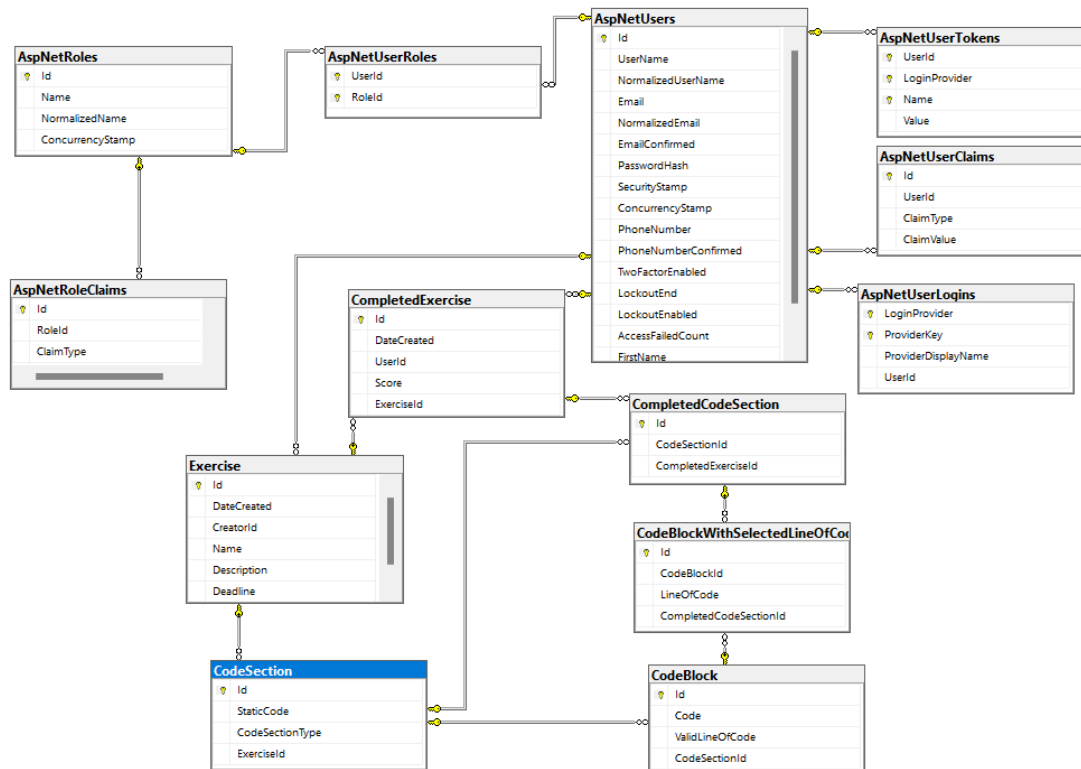


Рис. 1. Діаграма зв'язків бази даних

Література:

1. Прайс М. С# 7 и .NET Core. Кросс-платформенная разработка для профессионалов. – СПб.: Питер, 2018. – 640 с.
2. Ben-Gan I. Microsoft SQL Server 2012 T-SQL Fundamentals. – California, Microsoft Press, 2016. – 413 p.

Науковий керівник: Фабрі Людвіг Павлович, кандидат технічних наук, доцент, Національний університет «Львівська політехніка»

Нефедченко О.О., магістр, кафедра електронних обчислювальних машин, Харківський національний університет радіоелектроніки, м. Харків

МОДЕЛЬ ІНФОРМАЦІЙНОЇ ВЗАЄМОДІЇ В МЕРЕЖАХ ІоТ

В останнє десятиліття ІоТ став однією з проривних технологій, загально визнаних усіма країнами світу. ІоТ дозволяє людям і речам взаємодіяти де завгодно, коли завгодно, і в будь-яких поєднаннях при використанні інфраструктури ІоТ. Екосистема ІоТ передбачає збір даних з датчиків (або відправку команд на виконавчі пристрої), їх передачу через мережу зв'язку на хмарні платформи для подальшого аналізу з метою надання інтелектуальних послуг для людей. На рис.1 представлені ключові компоненти, необхідні для

побудови систем IoT. Згідно рисунку, датчики і пристрої знімання інформації збирають різні види даних про той чи інший об'єкт, потім ці дані можуть бути додатково оброблені та проаналізовані для отримання корисної інформації з метою надання інтелектуальних послуг. Елементи IoT зведені в одну просту формулу:

Фізичні об'єкти + контролери, сенсори, виконавчі механізми + Інтернет = IoT.

Існують різні додатки IoT, які спрямовані на вирішення конкретних завдань.

Серед типових додатків можна виділити:

- управління даними;
- аналітику;
- візуалізацію;
- управління гетерогенними мережами;
- дослідницькі цілі та ін.

Тим щонайменше, дослідження IoT все ще продовжують перебувати в зародковому стані, зважаючи на існування багатьох невирішених проблем, наприклад, проблем, пов'язаних з часом автономної роботи, простотою «легковагості» технологій передачі даних, виконанням дій в залежності від контексту того, що відбувається, питаннями ідентифікації та безпеки, вартості кінцевих пристроїв, масштабованості і гетерогенності.



Рис.1. Основні компоненти IoT

Література:

1. Churyumov G., Tokariev V., Tkachov V. Problem of self-organization of s-bot group movement in unorganized physical environment / G. Churyumov, V. Tokariev, V. Tkachov // Комп'ютерні та інформаційні системи і технології: тези доповідей третьої міжн. наук.-техн. конф., 23 - 24 квіт. 2019 р. – Харків, 2019. - С.16-17.

2. Серков О.А., Князев В.В., Лазуренко Б.О., Яковенко І.В., Чурюмов Г.І., Токарєв В.В. Надширокосмугові технології в задачах забезпечення електромагнітної сумісності рухомих об'єктів / О.А. Серков, В.В. Князев, Б.О. Лазуренко, І.В. Яковенко, Г.І. Чурюмов, В.В. Токарєв // Проблеми електромагнітної сумісності перспективних бездротових мереж зв'язку (EMC-2019): збірник наукових робіт четвертої міжн. наук.-техн. конф., 24 жовт. 2019 р. - Харків, 2019. - С. 55-57.
3. Krivoulya G., Ilna I., Tokariev V., Shcherbak V. Mathematical Model for Finding Probability of Detecting Victims of Man-Made Disasters Using Distributed Computer System with Reconfigurable Structure and Programmable Logic / G. Krivoulya, V. Tokariev, I. Ilna, V. Shcherbak // IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology: (PIC S&T), 06-09 oct. 2020y. - Kharkiv, 2020. - P.573 - 576.

Науковий керівник: Токарєв Володимир Володимирович, кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки

*Осадчий О.О., студент, кафедра Електронних обчислювальних машин, Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка, кафедра Електронних обчислювальних машин, Харківський національний університет радіоелектроніки, м. Харків*

ВАЖЛИВІСТЬ ВИКОРИСТАННЯ ПРИВАТНИХ МЕРЕЖ ДЛЯ ІНТЕРНЕТУ РЕЧЕЙ

В останнє десятиліття все більшої популярності на ІТ-ринку набувають хмарні технології. Розробка цих технологій зумовлена зростанням у геометричній прогресії потоків і обсягів інформації та потреби у її доступності для користувачів у будь-який момент [1]. Збільшення кількості хмарних обчислень детермінує, у свою чергу, розвиток хмарних платформ і сервісів для створення додатків, а також протоколів роботи в платформі.

При цьому актуальним є питання вибору ефективної хмарної платформи для обміну даними великого обсягу в контексті інтернету речей [2]. Під Інтернетом речей (Internet of Things, IoT) вітчизняні та зарубіжні науковці розуміють концепцію обчислювальної мережі, що складається із взаємозв'язаних фізичних пристроїв, що мають вбудовані сенсори і програмне забезпечення для обміну даними з іншими пристроями.

Існує можливість з'єднати всю ІТ-інфраструктуру, що стосується пристроїв IoT, в одну мережу — і ця приватна мережа дозволяє забезпечити безпечний зв'язок між розгорнутими пристроями IoT та інфраструктурою, яка контролює чи отримує з них дані [3].

Завдяки досягненням у криптографії, обчислювальних технологіях та поширеності Інтернету можна шифрувати трафік даних та тунелювати його через Інтернет на сервер, розташований у приватній мережі. Захищений тунель

створює віртуальне посилення, яке поширює приватну мережу на загальнодоступну [4]. Цей вид мережі, який використовує загальнодоступні мережі для забезпечення підключення до приватної, називається віртуальною приватною мережею (VPN).

Використовуючи сервер доступу OpenVPN як рішення безпеки IoT, користувач може створити власну приватну мережу для IoT заради безпечного ведення та налагодження зв'язку з пристроями, а також для уникнення атак, що можуть загрожувати зміною даних або несанкціонованим стеженням через розумні пристрої. Втручання зловмисником у незахищену мережу Інтернету речей може загрожувати втратою цінної інформації.

VPN — це чудове рішення для захисту даних, що надсилаються та приймаються різними пристроями, які утворюють інтернет речей [5]. Застосувавши VPN у мережах Інтернету речей, є можливість зробити ці мережі набагато надійнішими та безпечнішими. Це можна зробити так, що лише авторизовані пристрої можуть стати частиною приватної мережі.

Література:

1. T. Vitalii, B. Anna, H. Kateryna and D. Hrebenuk, "Method of Building Dynamic Multi-Hop VPN Chains for Ensuring Security of Terminal Access Systems," 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T), 2020, pp. 613-618, doi: 10.1109/PICST51311.2020.9467953.
2. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
3. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Pp. 25-26.
4. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:<https://doi.org/10.26906/SUNZ.2021.1.090>.
5. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii Tkachov, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdet'ska // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020.

*Потьомкіна К.О., студентка, кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка, кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

ПЕРЕВАГИ VPN МЕРЕЖ ДЛЯ ОСОБИСТОГО ВИКОРИСТАННЯ

VPN (англ. Virtual Private Network — віртуальна приватна мережа) — це безпечне, зашифроване підключення між двома мережами або між окремим користувачем та мережею. Мережі VPN дозволяють користуватися Інтернетом, зберігаючи при цьому конфіденційну інформацію [1]. Тому, користуючись мережами VPN, абонент отримує ряд прерогатив.

По-перше, це безпека в мережі. Без відома користувача програма або веб-сайт можуть відстежувати активність в Інтернеті. Потім вони мають змогу аналізувати зібрані дані та використовувати їх, щоб намагатися націлити на користувача рекламу. Без VPN можна відчутти вплив “спливаючих” оголошень, які можуть переривати роботу веб-перегляду. При використанні VPN, це може завадити зловмисникам, програмному забезпеченню та веб-браузерам отримати доступ до з'єднання [2, 3]. Це забезпечує безпеку та анонімність інформації, яку користувач передає та отримує.

По-друге, це може допомогти зберегти особисту інформацію. Хакери мають змогу використовувати різні методи для перехоплення конференційної інформації, яку користувач заповнює на веб-сайтах. Використовуючи цю інформацію, вони можуть спробувати видати себе за інших людей, отримавши доступ до банківських рахунків, інформації про кредитну картку.

По-третє, це допоможе запобігти регулюванню даних. Регулювання даних відбувається, коли користувач витратив певний обсяг доступних даних, а потім інтернет-провайдер (ISP) вирішує уповільнити роботу сервісу. Якщо користувач має VPN, він може уникнути обмеження даних, особливо тому, що навіть інтернет-провайдер не може бачити, скільки даних абонент використовує [4]. Це може бути особливо корисно для співробітників, яким доводиться використовувати тарифні плани на приватних інтелектуальних пристроях при доступі до Інтернету в дорозі.

Таким чином, підсумовуючи всі переваги, стає очевидним, що використання VPN мереж значною мірою покращує якість користування Інтернетом [5].

Література:

1. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
2. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Pp. 25-26.

3. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:https://doi.org/10.26906/SUNZ.2021.1.090.
4. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii Tkachov, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdetska // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020.
5. V. Tkachov, M. Bondarenko, O. Ulyanov and O. Reznichenko, "Overlay Network Infrastructure for Remote Control of Radio Astronomy Observatory," 2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT), 2019, pp. 161-165, doi: 10.1109/ATIT49449.2019.9030494.

Потьомкіна К.О., студентка, кафедра Електронних обчислювальних машин, Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка, кафедра Електронних обчислювальних машин, Харківський національний університет радіоелектроніки, м. Харків

ПЕРЕВАГИ VPN МЕРЕЖ ДЛЯ ОСОБИСТОГО ВИКОРИСТАННЯ

VPN–з'єднання встановлює безпечне з'єднання між користувачем й Інтернетом. Через VPN весь трафік даних спрямовується через зашифрований віртуальний тунель. Це маскує особисту IP-адресу користувача, коли він користується Інтернетом, роблячи його місце розташування невидимим для всіх. Існує багато різних типів VPN, але є лише три основні:

- перший тип — це SSL VPN. Не завжди співробітники компанії мають доступ до корпоративного ноутбука, який вони використовують для роботи з дому. У таких випадках часто вдаються до використання особистих пристроїв (ПК, ноутбука, планшета, мобільного телефону). У цьому випадку компанії вдаються до використання SSL–VPN, яке зазвичай реалізується через відповідний апаратний блок. Необхідною умовою зазвичай є браузер з підтримкою HTML–5, який використовується для виклику сторінки входу в систему. Браузери з підтримкою HTML-5 доступні практично для будь-якої операційної системи. Доступ захищений ім'ям користувача і паролем.

- другий тип — міжсайтовий VPN. Цей тип, ще називають VPN типу "мережа до мережі" – це приватна мережа, призначена для приховування приватних інтрамереж і наданням користувачам безпечних мереж доступу до ресурсів один одного. VPN типу "мережа до мережі" корисний, якщо є кілька філій у компанії, кожне з яких має власну локальну мережу (LAN), підключену до WAN (глобальної мережі). VPN–з'єднання типу "мережа до мережі" також корисні, якщо є дві окремі інтрамережі, між якими можна відправляти файли без явного доступу користувачів з однієї локальної мережі до іншої.

- третій тип — VPN-клієнт – сервер. Підключення через VPN-клієнт можна уявити, як підключення домашнього комп'ютера до офісу за допомогою подовжувального кабелю. Співробітники можуть підключатися до корпоративної мережі з домашнього офісу через захищене з'єднання і працювати ніби з офісу. Це більш поширена форма VPN, яка особливо корисна для провайдерів незахищених загальнодоступних WLAN. Він запобігає доступу третіх осіб до мережевого з'єднання і його компрометацію, а також шифрує дані на всьому шляху до провайдера. Отже, підсумовуючи всі факти, можна зробити висновок, що таким чином, знаючи типи чинних VPN мереж, можна з легкістю обрати те, що потрібно користувачеві виходячи з ситуації.

Література:

1. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
2. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Pp. 25-26.
3. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:<https://doi.org/10.26906/SUNZ.2021.1.090>.
4. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii Tkachov, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdetka // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020.
5. V. Tkachov, M. Bondarenko, O. Ulyanov and O. Reznichenko, "Overlay Network Infrastructure for Remote Control of Radio Astronomy Observatory," 2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT), 2019, pp. 161-165, doi: 10.1109/ATIT49449.2019.9030494.

*Репалюк О.І., студент, кафедра інформатики та програмної інженерії,
Національний технічний університет України "КПІ ім. Ігоря Сікорського",
м. Київ*

ПРОГНОЗУВАННЯ РЕЗУЛЬТАТІВ СПОРТИВНИХ ЗМАГАНЬ НА ОСНОВІ СТАТИСТИЧНИХ МОДЕЛЕЙ

Оцінка ефективності команд та спортсменів, а також прогнозування їх успішності є важливою для багатьох сфер бізнесу: формування спонсорських контрактів, букмекерська діяльність, проведення та обслуговування спортивних

подій. Програмне забезпечення, що реалізує моделі аналізу статистичних даних може використовуватись у щоденній діяльності для побудови довгострокової стратегії відносин зі спортивними організаціями. Однак на сьогодні компанії, які пропонують прогнози щодо результатів матчу, зазвичай не розкривають дані щодо використовуваних алгоритмів і моделей. Актуальною є задача створення відкритого програмного забезпечення для перевірки ефективності статистичних моделей прогнозування результатів на основі даних попередніх ігор та поточних характеристик.

Метою даної роботи є підвищення ефективності взаємовідносин бізнесу зі спортивними організаціями за рахунок розробки програмного модулю аналізу результатів спортивних змагань, комунікація користувача з яким відбуватиметься через систему управління взаємовідносинами з клієнтами. Для досягнення поставленої мети необхідно провести аналіз існуючих підходів до прогнозування результатів спортивних змагань, обрати моделі для подальшого дослідження. На основі обраних алгоритмів пропонується створити мікросервісне рішення, яке дозволить користувачеві порівняти результати за кількома моделями.

Для прогнозування результатів спортивних змагань використовуються наступні алгоритми: лінійна регресія [1], логістична регресія, нейронні мережі, метод опорних векторів, наївний баєсів класифікатор, випадковий ліс [2].

В якості вхідних даних дослідження обрано підсумки зустрічей команд з трьох видів спорту: хокею, баскетболу, бейсболу за кожен сезон з 2010 року. Результатом роботи системи є величина, що визначає ймовірність перемоги однієї з команд. Для створення системи було обрано алгоритми, що базуються на методах лінійної та логістичної регресії.

Для побудови статистичних моделей було обрано наступні значимі характеристики: рахунок (кількість забитих та пропущених голів або кількість набраних очок); розташування змагання (вдома/в гостях); ефективність атаки/захисту; процент перемог вдома/в гостях; показник особистих зустрічей за декілька сезонів; параметри, характерні для окремого виду спорту (баскетбол: вільні кидки, відскоки, підбори. бейсбол: рейтинг пітчерів).



Рис. 1. Узагальнена схема програмного за стосунку

Отже, інтеграція системи прогнозування в систему управління взаємовідносинами з клієнтами дозволить спростити комунікації та підвищити ефективність роботи за рахунок надання прозорих прогнозів щодо результатів змагань.

Література:

1. Прогнозування за допомогою лінійної регресії – The Power Rank [Електронний ресурс]. – Режим доступу: <https://thepowerrank.com/2018/08/14/how-to-make-accurate-football-predictions-with-linear-regression/>
2. Corentin Herbinet. Predicting Football Results Using Machine Learning Techniques [Електронний ресурс]. – Режим доступу до ресурсу: <https://www.imperial.ac.uk/media/imperial-college/faculty-of-engineering/computing/public/1718-ug-projects/Corentin-Herbinet-Using-Machine-Learning-techniques-to-predict-the-outcome-of-professional-football-matches.pdf>

*Рибальченко О.Г., старший викладач, кафедра моделювання та програмного забезпечення, Криворізький національний університет, м. Кривий Ріг;
 Зайка Б.В., студент, кафедра моделювання та програмного забезпечення, Криворізький національний університет, м. Кривий Ріг*

НЕЙРОМЕРЕЖЕВЕ ВИЯВЛЕННЯ DDOS-АТАК

У час стрімкого розвитку інформаційних технологій компанії та підприємства активно зберігають і поширюють інформацію в мережі Інтернет. Це, безумовно, зменшує обсяг додаткових витрат на необхідні рекламні

послуги, заробітну плату співробітників тощо. Але з появою нових можливостей виникають нові загрози, які пов'язані з недостатнім захистом великих мережеских вузлів. В будь-який момент система, що працює бездоганно, може бути піддана DDoS-атаці (Distributed Denial of Service attack). Метою такої атаки є створення умов, при яких буде ускладнений або повністю обмежений доступ до системи [1].

Для ефективної протидії DDoS-атакам потрібно вирішити дві послідовні задачі. Першою задачею є розподіл мережевого трафіку на звичайний та аномальний. Другою задачею є інтелектуальний аналіз аномального трафіку, достовірне діагностування DDoS-атаки на ранніх стадіях. Це дозволить вчасно протидіяти атакам, а саме налаштувати правила міжмережеских екранів, увімкнути мережескі фільтри, задіяти резервні канали тощо.

Всі сучасні DDoS-атаки проводять з використанням ботнету, із застосуванням підміни IP-адрес або комбінованим методом. Їх можна розподілити на три великі групи [2]. Перша група – це flood-атаки з насиченням смуги пропускання, вони направлені на переповнення каналу зв'язку. Їх метою є створити потужний потік запитів, який займає всю виділену смугу трафіку, пакети користувачів не проходять і ресурс змушений відмовляти їм в обслуговуванні (UDP, ICMP та інші). Друга група – це атаки на рівні протоколів, вони використовують вразливості стека мережеских протоколів. При атаці через помилки протоколів TCP/IP можуть використовуватися SYN-пакети (запити на відкриття з'єднання), в результаті чого на комп'ютері, що атакується, швидко вичерпується кількість доступних сокетів і сервер припиняє відповідати (так званий SYN- flood). Третя група – це низькоінтенсивні і малопотужні атаки (low-rate DDoS). Відмова в обслуговуванні досягається приховано, невеликою кількістю трафіку і не вимагає виснаження смуги пропускання. Атакуючий відкриває безліч нескінченних з'єднань і при перевищенні деякого порога викликає в мережі «жертви» відмову в обслуговуванні. Використовуються протоколи транспортного (TCP) або прикладного (HTTP) рівнів моделі OSI. Такі атаки дуже важко виявити, оскільки самі по собі такі з'єднання не є «аномальною» поведінкою.

Виділяють наступні класи методів виявлення DDoS-атак: поведінкові методи, методи на основі знань, методи машинного навчання, методи штучного інтелекту [3]. До поведінкових методів віднесені наступні методи: вейвлет-аналіз, статистичний аналіз, аналіз ентропії, спектральний аналіз, фрактальний аналіз, кластерний аналіз. Ці методи засновані на використанні інформації про нормальну поведінку системи та її порівнянні з параметрами поточної поведінки. Випадок значних відхилень може розглядатися як свідчення наявності атаки. Представлена група методів орієнтована на побудову моделі штатного, або нормального, функціонування системи або користувача.

До методів на основі знань віднесені такі методи, які в контексті заданих фактів і правил зіставлення, що відображають ознаки заданих атак, виконують дії по виявленню атак на основі закладеного механізму пошуку. Своєю назвою ці методи зобов'язані тому, що такі системи працюють з базою знань, в якій

містяться дані щодо вже відомих атак. До методів на основі знань віднесені експертні системи, кінцеві автомати, мережі Петрі, сигнатурний метод тощо.

Методи машинного навчання і методи штучного інтелекту застосовують як при виявленні аномалій, так і при виявленні зловживань. Це пояснюється тим, що зазначені підходи в якості вихідних даних для навчання часто використовують шаблони як нормальної, так і аномальної поведінки в мережі. До методів машинного навчання віднесені наступні підходи до виявлення мережових атак: дерева рішень, Байєсівські мережі, MAP-сплайни, алгоритми кластеризації та алгоритми регресії. До методів штучного інтелекту віднесені штучна нейронна мережа (ШНМ), генетичні алгоритми, нечітка логіка, імунні системи, роеві алгоритми тощо.

Поставлена задача виявлення ступеню небезпеки для web-ресурсу є задачею класифікації – об'єкти описані зазначеним набором ознак і відома приналежність кожного об'єкту до певного класу. Оскільки у досліджуваній проблемі наявні лише два класи – “норма” і “атака”, що будуть закодовані відповідно як “0” і “1”, задача належить до типу бінарної класифікації.

На сьогоднішній день найпотужнішим засобом, здатним вирішити поставлену задачу, є технологія машинного навчання – набір методів побудови алгоритмів, що навчаються. До методів, здатних ефективно вирішувати завдання класифікації, належать композиційні методи: Bagging, Random Forest, Gradient Boosting тощо. Цей клас методів заснований на побудові ансамблю простих базових алгоритмів, наприклад, вирішальних дерев, кожне з яких робить внесок у роботу всієї моделі. Перевагою композиційних алгоритмів є можливість розпаралелювання процесу побудови моделі, а також доволі висока якість їх роботи. До недоліків належить велика кількість гіперпараметрів методу. Гіперпараметри – це такі параметри, що заздалегідь невідомі і не настроюються під час навчання моделі, але їх необхідно визначити перед запуском процесу навчання. Серед таких параметрів кількість базових алгоритмів у композиції, глибина дерев, ознаки і їх порогові значення, які визначаються в вершинах кожного дерева, параметр рандомізації вибірок, що потрапляють до навчання тощо. Наявність такої кількості ступенів свободи моделі уповільнює настройку параметрів композиції.

Ще одним класом методів машинного навчання є ШНМ. Модель ШНМ подібна до спрощеної біологічної моделі нейронної мережі (НМ), де нейрони зв'язані між собою синапсами, по яким інформація передається у вигляді електричних імпульсів. Останнім часом ШНМ набули великої популярності у вирішенні задач високої складності, алгоритми розв'язання яких часом не відомі або зовсім не існують. Така здатність ШНМ вирішувати складні задачі обумовлена перевагами біологічних НМ, оскільки модель обробки інформації в них однакова. ШНМ вчиться так само, як мозок живої істоти – на прецедентах. Процес пред'явлення ШНМ об'єктів із зазначенням належності кожного об'єкту до певного класу є навчанням за прецедентами, або навчанням з учителем (supervised learning). Маючи велику кількість прецедентів і відомих відповідей на цих прикладах, можна за кінцевий час навчити ШНМ з заданою точністю класифікувати об'єкти, які навіть не брали участь у процесі навчання, згідно

теореми збіжності перцептрона Ф. Розенблатта [4]. Крім того, ШНМ є універсальною моделлю, яка здатна апроксимувати будь-які поверхні, і, за теоремою про універсальне наближення [5], може бути представлена у вигляді суперпозиції функцій від однієї змінної – вектору ознак об'єкту.

На користь застосування ШНМ для вирішення поставленого завдання також свідчить ряд переваг цього класу методів: стійкість до зашумлених даних, адаптація до зміни вхідних даних, швидкодія тощо. Серед недоліків ШНМ є ймовірність збіжності методу навчання у локальному мінімумі в процесі оптимізації функціоналу помилки ШНМ при використанні градієнтних методів оптимізації, а за умови застосування стохастичних оптимізаційних методів, які завжди знаходять глобальний мінімум, результат роботи ШНМ не завжди передбачуваний.

Література:

1. Мурасов Р.К. Завчасне попередження про DDoS атаку на базі методів прогнозування / Мурасов Р.К., Мельник Я.В.. // Національний університет оборони України імені Івана Черняхівського. – 2016. – С. 59.
2. Классификация DDoS-атак: краткий обзор современных подходов [Электронный ресурс] – Режим доступа до ресурсу: <https://ddos-guard.net/ru/info/blog-detail/classification-of-ddos-attacks-a-short-overview-of-modern-approaches>
3. Тарасов Я. В. Метод обнаружения низкоинтенсивных DDoS-атак на основе гибридной нейронной сети / Я. В. Тарасов // Известия ЮФУ. Технические науки. – 2014. – С. 47-57.
4. Розенблатт Ф. Принципы нейродинамики: перцептроны и теория механизмов мозга = Principles of Neurodynamic: perceptrons and the theory of brain mechanisms / Фрэнк Розенблатт. – М.: Мир, 1965. – 480 с.
5. Ian Goodfellow. Deep learning (Adaptive computation and machine learning series). / Ian Goodfellow, Yoshua Bengio, Aaron Courville. – Cambridge, MA: MIT Press, 2017. – 775 с.

*Ротань Д.В., магістр, кафедра електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

МОДЕЛЬ ВЗАЄМОДІЇ D2D В БЕЗДРОТОВИХ МЕРЕЖАХ

Архітектура цифрових об'єктів (Digital Object Architecture – DOA) і пов'язана з нею система резолюцій Handle System були розроблені корпорацією національних дослідницьких ініціатив (CNRI) на початку 1990-х років, ґрунтуючись на роботах над цифровими бібліотеками для Управління перспективних дослідницьких проєктів Міністерства оборони США (DARPA). Розробка архітектури цифрових об'єктів стала спробою переходу від представлення даних в Інтернеті за допомогою наборів вузлів і транспорту до виявлення і доставці інформації у вигляді цифрових об'єктів. Мета створення

архітектури цифрових об'єктів – вирішення наступних проблем управління цифровою інформацією:

- забезпечення стандартного доступу до розрізної інформації (ідентифікація, пошук інформації і надання даних, забезпечення безпеки, типізація);
- взаємодія з різноманітними інформаційними системами;
- незалежність від конкретних базових технологій, які використовуються для розміщення та обслуговування інформації;
- взаємодія протягом тривалих періодів часу;
- активне управління системами, на яких поширюється інформація;
- забезпечення великого рівня масштабованості;
- розподілена архітектура;
- відкрита архітектура;
- стандартні протоколи і процедури взаємодії компонентів системи.

Архітектура цифрових об'єктів – архітектура розподіленої системи зберігання, визначення місця розташування і пошуку інформації в Internet рис.1.

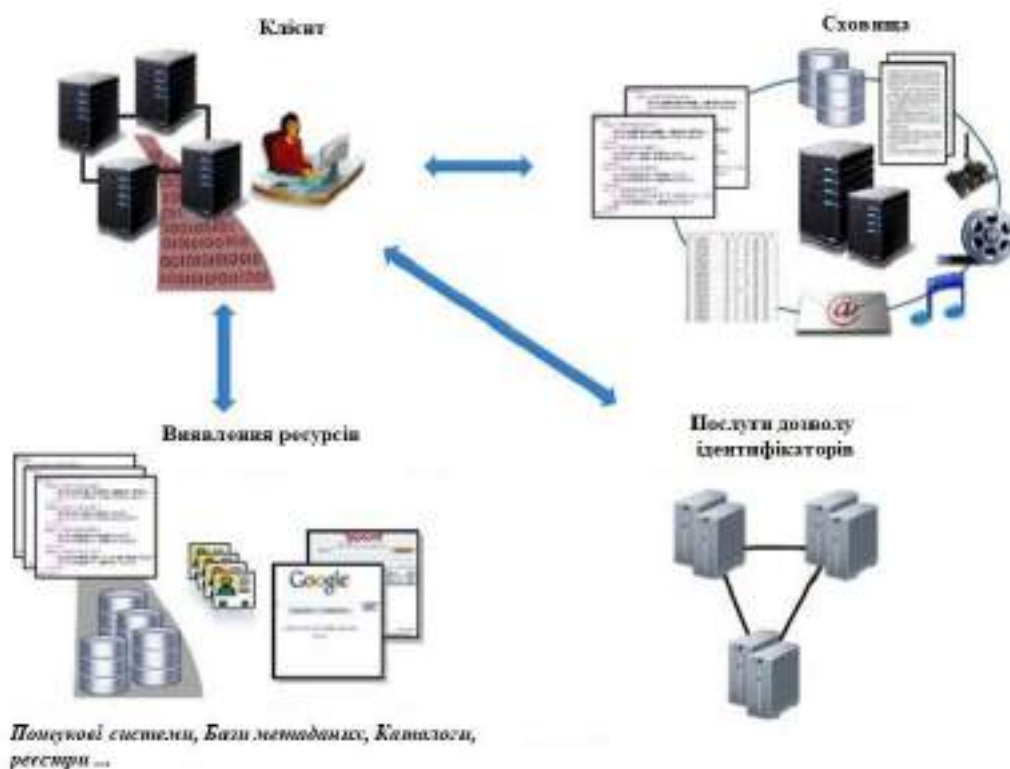


Рис.1. Фундаментальні компоненти архітектури цифрових об'єктів

Література:

1. Churyumov G., Tokariev V., Tkachov V. Problem of self-organization of s-bot group movement in unorganized physical environment / G. Churyumov, V. Tokariev, V. Tkachov // Комп'ютерні та інформаційні системи і технології: тези доповідей третьої міжн. наук.-техн. конф., 23 - 24 квіт. 2019 р. – Харків, 2019. - С.16-17.

2. Серков О.А., Князев В.В., Лазуренко Б.О., Яковенко І.В., Чурюмов Г.І., Токарєв В.В. Надширокосмугові технології в задачах забезпечення електромагнітної сумісності рухомих об'єктів / О.А. Серков, В.В. Князев, Б.О. Лазуренко, І.В. Яковенко, Г.І. Чурюмов, В.В. Токарєв // Проблеми електромагнітної сумісності перспективних бездротових мереж зв'язку (EMC-2019): збірник наукових робіт четвертої міжн. наук.-техн. конф., 24 жовт. 2019 р. - Харків, 2019. - С. 55-57.
3. Krivoulya G., Ilna I., Tokariev V., Shcherbak V. Mathematical Model for Finding Probability of Detecting Victims of Man-Made Disasters Using Distributed Computer System with Reconfigurable Structure and Programmable Logic / G. Krivoulya, V. Tokariev, I. Ilna, V. Shcherbak // IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology: (PIC S&T), 06-09 oct. 2020y. - Kharkiv, 2020. - P.573 - 576.

Науковий керівник: Науковий керівник: Токарєв Володимир Володимирович, к.т.н., доцент, Харківський національний університет радіоелектроніки

Савелов Є.Е., студент, кафедра автоматизованих систем управління, Національний університет «Львівська політехніка», м. Львів

РЕАЛІЗАЦІЯ ВЕБ-ОРІЄНТОВАНОГО ПРОГРАМНОГО РІШЕННЯ ДЛЯ ЗДІЙСНЕННЯ ОПИТУВАНЬ В ОНЛАЙН РЕЖИМІ

Веб-орієнтоване програмне рішення для здійснення опитувань в онлайн режимі складається з двох частин:

- 1) Веб-застосунку для створення опитувань та адміністрування;
- 2) Веб-сайту, який дозволяє проходити онлайн-опитування.

Веб-орієнтоване програмне рішення для здійснення опитувань в онлайн режимі працює за трирівневою структурою клієнт-сервер. Така архітектура ділить програму на три рівні: рівень клієнта, рівень сервера та рівень даних. Серверна та клієнтська частини застосунку здійснюють запити за допомогою технології WebSocket та HTTP. Формат передачі даних – JSON.

Клієнтський рівень – той, що запитує інформацію. У даному випадку це графічний інтерфейс користувача.

Рівень сервера – діє як інтерфейс між клієнтом і рівнем даних. Це допомагає в підтримці програмної системи, а також забезпечує безпеку [1].

Основним модулем у серверній частині застосунку є Voter, що складається з таких шарів: Core, DAL, Services, API.

Core – базовий шар модуля, що містить його інфраструктуру. Цей шар описує сутності, контракти, моделі, об'єкти передачі даних.

DAL – шар, що відповідає за забезпечення доступу до даних.

Services – шар, що містить логіку опрацювання даних у додатку.

API – шар, що є фасадом сервера та містить у собі необхідні налаштування.

Серверна частина системи містить у собі наступні модулі:

Core – базовий модуль сервер, що реалізує головну інфраструктуру серверної частини системи. У цьому модулі містяться всі необхідні загальні класи для роботи серверної програми.

DatatableServerProcessingLibrary – модуль, що містить базовий функціонал для обробки таблиць у системі.

GenericDataAccessHelper – модуль, що містить базові визначення контролерів та сервісів, що потім будуть наслідувані в інших модулях.

ObjectMapping – модуль, що відповідає за конвертування між різними типами даних у системі.

Membership – модуль, що реалізує функції управління користувачами та їх ролями у системі.

Notification – модуль, що відповідає за відправку повідомлень користувачам.

Survey – модуль, що відповідає за створення та управління опитуваннями.

UploadService – модуль, що відповідає за вивантаження файлів різного формату користувачів на сервер. Ці файли можуть потім бути використані для створення чи редагування опитувань.

На рис. 1 наведена діаграма компонентів серверної частини системи.

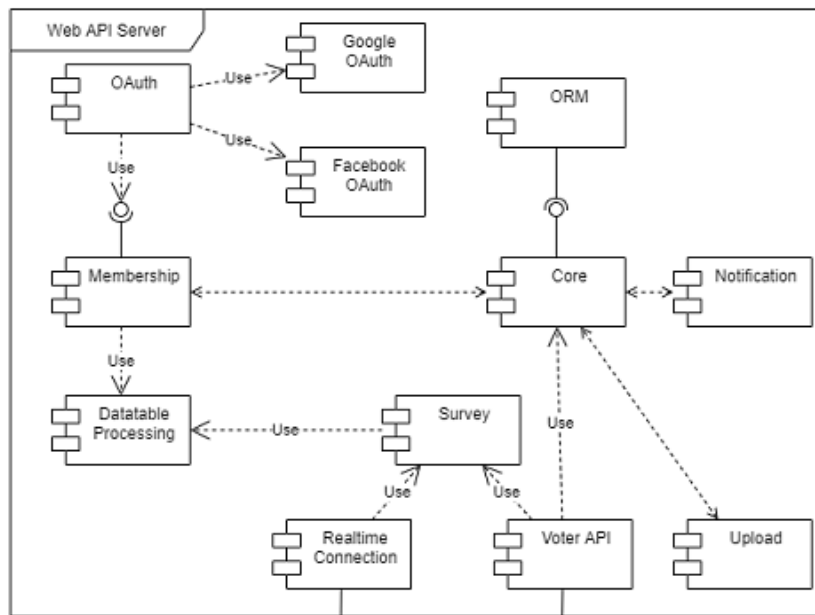


Рис. 1. Діаграма компонентів серверної частини системи

Клієнтська частина веб-застосунку реалізована з допомогою Angular Framework. Angular – це платформа для побудови односторінкових клієнтських додатків за допомогою HTML та TypeScript. Архітектура додатку, написаному на Angular Framework спирається на певні фундаментальні концепції. Основним будівельним блоком є декоратор NgModule. Додаток, написаний на Angular визначається набором модулів [2].

У клієнтській частині системи для проходження та створення онлайн-опитувань можна виділити наступні модулі: CommonSharedModule – загальний модуль, AppModule – модуль для авторизації та реєстрації користувачів системи, OrganizationModule – модуль для управління організаціями системи, SurveyModule – модуль, що надає графічний інтерфейс для перегляду, створення та налаштування опитувань у системі, ProfileModule – модуль, що використовується для налаштування даних про користувача.

На рис. 2 подано діаграму компонентів клієнтської частини системи для створення та проходження опитувань.

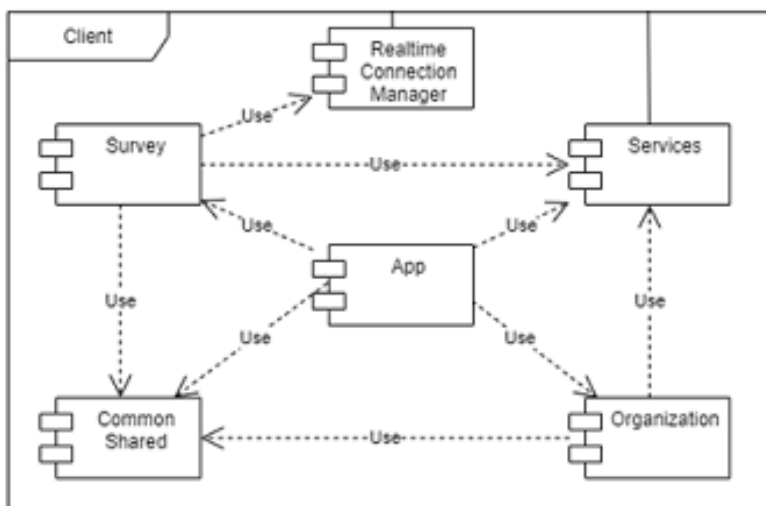


Рис. 2. Діаграма компонентів клієнтської частини системи

Література:

1. ASP.NET Core. Разработка приложений / Джеймс Чамберс, Дэвид Пэккетт, Саймон Тиммс. – Питер, 2018. – 484 с.
2. Angular для профессионалов / Фримен Адам. – Питер, 2018 – 800 с.

Сагайдачний О.М., студент, кафедра Електронних обчислювальних машин, Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка, кафедра Електронних обчислювальних машин, Харківський національний університет радіоелектроніки, м. Харків

ПРОБЛЕМИ БЕЗПЕКИ В БЕЗДРОВОВИХ СЕНСОРНИХ МЕРЕЖАХ

Безпека і конфіденційність – величезні проблеми у всіх типах бездротових мереж, які є привабливими мішенями для проникнень і інших атак. У мережах, створених для відстеження цілей та моніторингу інфраструктури, порушення коректного алгоритму функціонування мережі можуть мати дуже серйозні наслідки [1]. Сенсорні мережі часто використовуються в віддалених областях, без можливості постійного контролю за працездатністю мережі, і тим самим

вони стають легкою мішенню для фізичних атак, несанкціонованого доступу і пошкоджень.

Сенсорні вузли зазвичай дуже обмежені ресурсами і працюють в суворих умовах, що полегшує дискредитацію вузлів [2]. У порівнянні з традиційними атаками і механізмами безпеки, розробленими для Інтернету, бездротові сенсорні мережі мають безліч унікальних проблем, які потрібно розглянути, звертаючись до проблем безпеки:

- Обмеження ресурсу. Традиційні механізми безпеки не придатні для бездротових сенсорних мереж з обмеженими ресурсами [3]. Багато механізмів безпеки в обчислювальному відношенні дороги або вимагають зв'язку з іншими вузлами або віддаленими пристроями (наприклад, з метою авторизації), що призводить до енергетичних витрат. Маленькі сенсорні пристрої також обмежені в своїй доступною пам'яті. Традиційні алгоритми безпеки, що вимагають істотної кількості пам'яті і простору пам'яті, неможливі для таких сенсорів.

- Відсутність центрального управління. Часто неможливо мати центральну точку управління в сенсорних мережах, наприклад, через їх великого масштабу, обмежень ресурсу і мережевий динаміки (топологічних змін, поділу мережі). Тому рішення щодо забезпечення безпеки повинні бути децентралізовані і вузли повинні співпрацювати, щоб досягти безпеки [4].

- Віддалене розташування. Перша лінія захисту проти атак безпеки - це забезпечення контрольованого фізичного доступу до сенсорного вузла. Багато бездротових сенсорних мереж розміщуються у віддалених і важкодоступних місцях, розгорнутих в середовищах, відкритих для публічного доступу, або настільки великих, що буде неможливо постійно контролювати і захищати сенсорні вузли від фізичних атак.

З іншого боку, певні характеристики сенсорних мереж спрощують умови безпеки. Наприклад, самоврядкування і самовідродження функціонування бездротової сенсорної мережі можуть дозволити їй продовжувати роботу, навіть якщо сенсорний вузол або область сенсорної мережі були поставлені під загрозу [5].

Література:

1. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
2. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Pp. 25-26.
3. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:<https://doi.org/10.26906/SUNZ.2021.1.090>.

4. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii Tkachov, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdetska // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020.
5. Спосіб передачі цифрових даних мультикоптерною системою між сегментами розподіленої сенсорної мережі та базовою станцією : пат. на корисну модель 118921 Україна / Ткачов В. М., Токарев В. В. ; ХНУРЕ. – 2017

*Снайчук Я.Л., студент, кафедра автоматизованих систем управління,
Національний університет «Львівська політехніка», м. Львів*

РЕАЛІЗАЦІЯ ВЕБ-СИСТЕМИ ДЛЯ ЗАПИСУ ВІЗИТІВ У МЕДИЧНИЙ ЗАКЛАД

Веб-системи для запису візитів користуються популярністю серед бізнесу, оскільки дозволяють легко та ефективно розподіляти навантаження на персонал або устаткування. Такі системи є універсальним та мають широку сферу застосування, проте для наочності розглянемо систему для запису візитів у медичний заклад.

Зазвичай такі системи повинні містити наступні функції:

1. Авторизація та реєстрація користувачів в системі. Надання доступу до системи в залежності від прав користувача;
2. Перегляд та пошук у списку лікарів, пошук найближчої медичної установи;
3. Відображення доступних дат та годин для запису пацієнта на прийом до лікаря;
4. Перегляд історії записів та нагадування пацієнтам про прийом за допомогою доступних каналів сповіщень;
5. Інструменти для адміністрування системи.

З метою вивчення ринку було розглянуто наявні рішення, а саме Helsei, asker, pb4.

Веб-система побудована за клієнт-серверною архітектурою [1] використовує HTTPS-запити для захисту важливої інформації від спроб крадіжок. Усі контракти між клієнтом та сервером мають відповідати REST-підходу [2].

Клієнт відповідає за надання користувачу веб-застосунку зручного і адаптивного інтерфейсу. Клієнт системи буде легким, таким чином він міститиме лише невелику частину логіки системи. Він реагуватиме на дії користувача та надсилатиме запити на сервер.

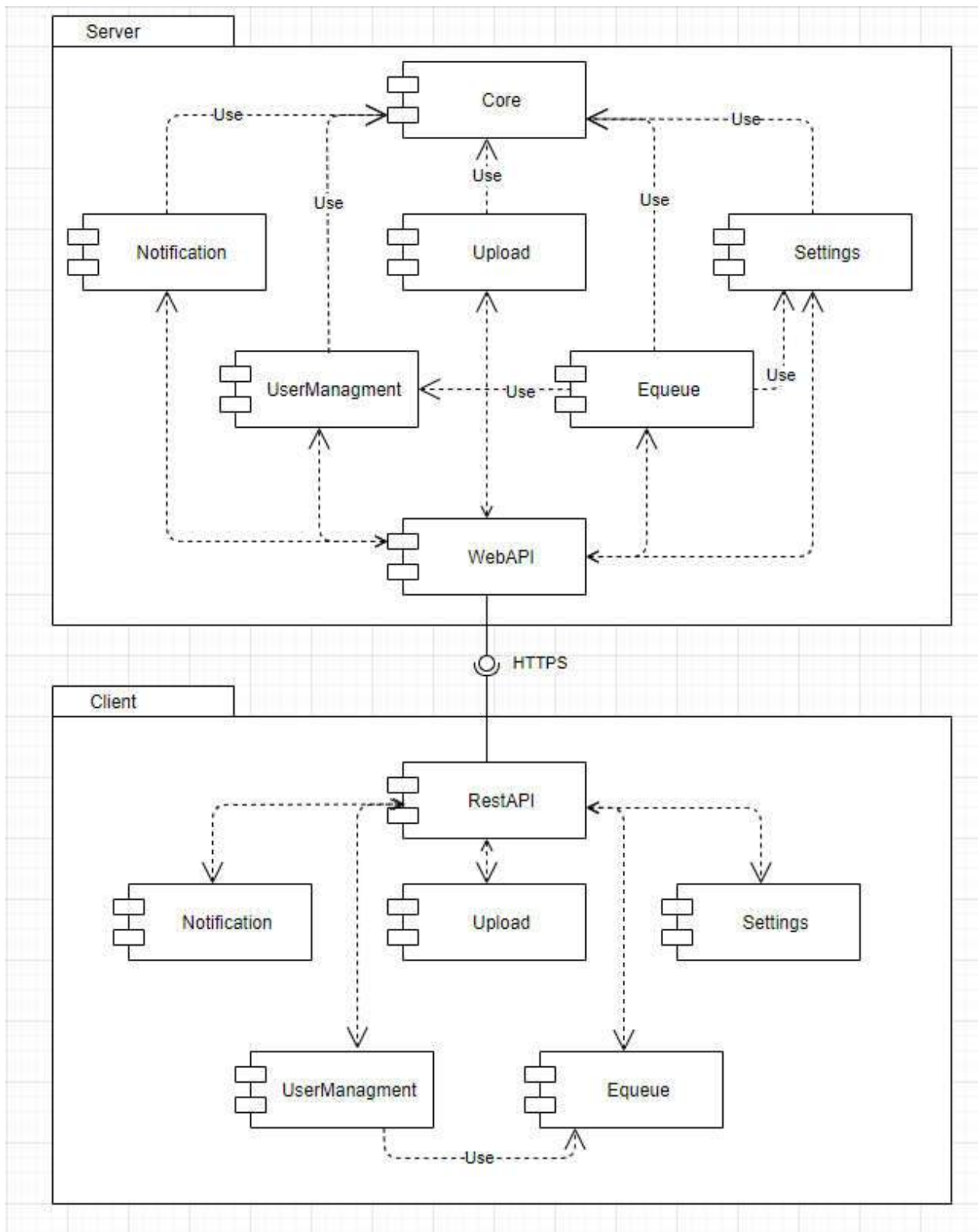


Рис. 1. Діаграма компонентів веб-системи для запису візитів

Серверна і клієнтська частини будуть поділені на модулі, які складатимуться з рівнів представлення і сервісів [3]. Основні модулі веб-системи для запису візитів у медичний заклад:

1. UserManagement – відповідає за авторизацію та автентифікацію користувачів;

2. Queue – модуль для роботи з розкладом роботи та візитами;
3. Settings – керує загальними налаштуваннями системи та не є прив'язаним до конкретної системи;
4. Notification – модуль, що відповідає відправлення електронних листів, повідомлень чи інших сповіщень;
5. Upload – відповідає за завантаження документів та фотографій у систему;
6. Core – містить базовий функціонал, що необхідний іншим модулям для роботи.

Контролери на сервері та сервіси клієнтської частини взаємодіють через REST API. Таким чином клієнтські модулі можуть спілкуватися з різними точками входу на сервері.

Література:

- 1 Клієнт-серверна архітектура – Wikipedia [Електронний ресурс]. – Режим доступу: uk.wikipedia.org/wiki/Клієнт-серверна_архітектура (2021).
- 2 REST API – Wikipedia [Електронний ресурс]. – Режим доступу: uk.wikipedia.org/wiki/REST (2021).
- 3 Фрімен А. Entity Framework Core 2 для ASP.NET Core MVC для професіоналов / А. Фрімен. – Москва, 2016. – 1328 с.

Соботник Е.Л., магістр, кафедра інженерії програмного забезпечення, Івано-Франківський національний технічний університет нафти і газу;
Юрчишин В.М., доктор технічних наук, професор, кафедра інженерії програмного забезпечення, Івано-Франківський національний технічний університет нафти і газу

ОСОБЛИВОСТІ БАГАТОПОТОКОВОЇ ОБРОБКИ ДАНИХ У ДОДАТКАХ, ПОБУДОВАНИХ НА БАЗІ NODE.JS

Багатопотокова обробка даних (англ. *Multi-data-stream processing*) — програмний та/або апаратний спосіб побудови архітектури обладнання, що дає змогу здійснювати одночасне оброблення двох або більше послідовностей даних під керуванням однієї або більше послідовностей команд.

Із проблемами багатопотокової обробки даних стикаються постійно через перевантаженість інформаційних систем файлами великих розмірів та невірний підхід до їх обробки, оскільки при цьому виникають збої у роботі програмного забезпечення. Саме проблему завантаженості програмного забезпечення у своїй праці «Node.js: Using JavaScript to Build High-Performance Network Programs» висвітлили Stefan Tilkov та Steve Vinoski [1]. Вони продемонстрували різні архітектурні підходи щодо організації опрацювання складної навантаженої бізнес-логіки для додатків різної складності.

Розглянемо загальний принцип обробки файлів великих розмірів засобами Node.js. Node.js спроектований для побудови масштабованих

мережевих додатків. Те що Node.js спроектований без багатопоточності, не означає, що його можливості не можна використовувати для кількох ядер. Node.js був створений як експеримент для асинхронної обробки даних. Ідея полягала в тому, що асинхронна обробка в одному потоці може забезпечити більшу продуктивність і масштабованість для типових веб-навантажень, ніж типова реалізація на основі потоків.

Платформа Node.js підтримує обробку файлів, проте зчитування файлів великих розмірів потребує багатопотокової обробки, оскільки така операція є синхронною, то і файли великих розмірів призведуть до завантаженості однопотокових процесів та ефекту «зависання» через брак оперативної пам'яті, виділеної середовищем виконання програмного забезпечення.

На рис. 1 зображена найпростіша операція зчитування інформації з файлу. Так, такий спосіб цілком підходить для файлів невеликих розмірів. Дослідження показали, що максимально можливий розмір файлу для обробки таким способом не повинен перевищувати 512 МБ для 32-бітних систем та до 1024 МБ для 64-бітних систем відповідно [2].

```
var fs = require('fs')

fs.readFile('very_large.csv', function(err, data) {
  // do something with data
})
```

Рисунок 1 – Загальний вигляд оператора зчитування файлу

Проте, при перевищенні даного обмеження інформаційна система потрапляє під ризик, а саме некоректну роботу та й можливе аварійне завершення роботи програмного забезпечення в цілому. Щоб вирішити проблему обробки файлів великих розмірів необхідно звернутися до API Node.js та дізнатися інші способи опрацювання файлів засобами даної платформи. Оскільки платформа Node.js розглядає файл як окремий потік, то, відповідно, надає можливість завантажити будь-який файл частинами (чанками), що можна спостерігати на рис. 2.

```
var readerStream = fs.createReadStream('very_large.csv') //create a readable stream
readerStream.setEncoding('UTF8') // set the encoding to be utf8.

// handle stream events -> data, end, and error
readerStream.on('data', function(chunk) {
  data += chunk
  //do something with data
})
```

Рисунок 2 – Загальний вигляд оператора зчитування файлу частинами.

В цілому, такий спосіб міг би і підійти, проте, на жаль, під час зчитування файлів необхідно виконувати певні асинхронні операції, наприклад калькуляція параметрів та подальше їх збереження у базу даних чи групування за певними

критеріями і розповсюдження інформації різними засобами зв'язку, зокрема через електронні скриньки. Оскільки, як раніше було зазначено, зчитування файлу є синхронним, то програмне забезпечення не буде «очікувати» виконання асинхронних операцій. Тож якщо частин файлу буде багато, то це призведе до швидкого заповнення стеку процесу і до унеможливлення подальшої роботи інформаційної системи. Такий спосіб чудово підходить тільки для синхронних операцій. Проте розробники платформи передбачили можливість виконання асинхронних операцій при зчитуванні файлів великих розмірів. Для цього необхідно використовувати додатковий інструмент `stream.pipe` (див. рис. 3).

```
let readable = fs.createReadStream('large.csv');

readable
  .pipe(streamA())
  .pipe(streamB())
  .pipe(streamC())
  .pipe(streamD())
  .pipe(streamE());
```

Рисунок 3 – Загальний вигляд оператора асинхронного зчитування файлу частинами.

Як бачимо, такий оператор дозволяє об'єднувати потрібні потоки та формувати складну бізнес-логіку, що залежить від попередніх результатів потоків, що виконувалися. Щоб головний потік міг правильно «спілкуватися» та керувати усіма іншими потоками (дочірніми потоками), необхідно щоб вони реалізовували спеціальну поведінку, а саме метод `_transform`, що приймає у собі як вхідні аргументи: частину файлу, його кодування, та функцію-callback `done`, яка повинна сповістити батьківський потік про завершення обробки інформації, а також у разі помилки повідомити про неї. Такий підхід дозволить накладати потоки один на одного, щоб вибудувати архітектуру ланцюга. Цей ланцюг допоможе розділити логіку складних ресурсозатратних операцій на дрібніші функціональні одиниці. Відомо, що при виконанні програмного забезпечення можуть виникати як помилки передбачені бізнес-логікою, так і аварійні непередбачувані помилки. Спосіб «сповіщення» головного потоку про виникнення помилки нами було розглянуто вище, проте варто згадати, що платформа Node.js дозволяє відловити будь-яку помилку, що виникає під час багатопотокового опрацювання інформації та виконати необхідні дії. Зазначимо, що Node.js не зупиняє усі дочірні потоки батьківського потоку при виникненні принаймні однієї помилки, саме тому інженерам програмного забезпечення надважливо є коректне завершення роботи усіх дочірніх потоків, щоб уникнути непередбачуваних витоків пам'яті. З версії Node.js 10.x була додана можливість автоматичного відлову помилок та закриття усіх дочірніх потоків за рахунок використання нового методу `pipeline` (див. рис. 4), що дозволяє аналогічно комбінувати потоки, але з можливістю відстеження виконання усіх потоків, а також зчитування можливих помилок.

```
const fs = require('fs');
const { pipeline } = require('stream');
const { Readable } = require('stream');
const { Readable } = require('stream');
const readable = fs.createReadStream('large.txt');

// This file contains the code to read a stream of data
// pipeline and get notified when the pipeline is fully done
pipeline(
  readable,
  new Readable(),
  (err) => {
    if (err) console.error('Pipeline failed.', err);
  }
);
```

Рисунок 4 – Загальний вигляд оператора pipeline.

Отже, шляхом вирішення проблеми опрацювання файлів великих розмірів з можливістю виконання асинхронних операцій є розбиття на потоки, які відповідають за свою частину файлу та передають інформацію далі по основному потоці, що було зображено на рис. 3 та рис. 4. Такий підхід дозволяє будувати складну бізнес-логіку, а також забезпечить подальшу масштабованість будь-якої інформаційної системи. Перспективою подальшого дослідження даної предметної області є її поширення серед сучасних додатків, які виконують обробку інформації файлів засобами Node.js задля покращення архітектури додатків, уникнення збоїв та пришвидшення роботи.

Список використаних джерел:

1. Tilkov S, Vinoski S. Node. js: Using JavaScript to build high-performance network programs. IEEE Internet Computing. 2010 Nov 1;14(6):80-3.
2. <https://askinglot.com/how-much-memory-can-node-js-use>

*Суцєнко І.В., магістр, кафедра електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

**СИСТЕМА ІДЕНТИФІКАЦІЇ НА БАЗІ АРХІТЕКТУРИ ЦИФРОВИХ
ОБ'ЄКТІВ**

В даний час в результаті швидкого розвитку інформаційних технологій, зростання обсягів різноманітної інформації в мережі зв'язку загального користування (МЗЗК), а також через повсюдне впровадження технологій інтернету речей з'явилася нагальна потреба впровадження механізмів однозначної ідентифікації пристроїв і додатків IoT, що дозволяють відслідковувати достовірність інформації в мережі і боротися з контрафактною ІКТ-продукцією. Для розробки такого сервісу спочатку необхідно вибрати найбільш оптимальну систему ідентифікації. Для ідентифікації можна використовувати безліч різних програмних і апаратних рішень, наприклад,

системи апаратної ідентифікації IPv6, зв'язку IPv4 + MAC, IMEI і ін.

Однак загальними недоліками цих систем є можливість програмної і апаратної зміни ідентифікатора мережевого інтерфейсу і прив'язка до апаратних ідентифікаторів, яка виключає можливість ідентифікації цифрового контенту, що відноситься до віртуальних сутностей інтернету речей і теж вимагає ідентифікації. Цих недоліків позбавлені альтернативні програмні рішення для ідентифікації, такі як DOA, URI, XRI, IRI і ін., які дозволяють ідентифікувати будь-який віртуальний або реальний об'єкт в мережі зв'язку загального користування, незалежно від наявності або відсутності у нього мережевого інтерфейсу. Ці системи так само, як і системи апаратної ідентифікації, використовують для аутентифікації фізичних і цифрових об'єктів різні сторонні технології. Вибір оптимальної системи визначається наступними вимогами до технологій ідентифікації, які враховують її використання в МЗЗК:

- системи ідентифікації повинні відповідати на множинні запити;
- для роботи з ідентифікаторами необхідно реалізувати різні рівні доступу, тобто систему авторизації користувачів;
- база, яка містить дані, повинна бути відокремлена від самого об'єкта ідентифікації;
- ідентифікатори не повинні містити динамічні елементи або метадані.

На рис.1 представлена загальна архітектура для системи резолюції Handle System.

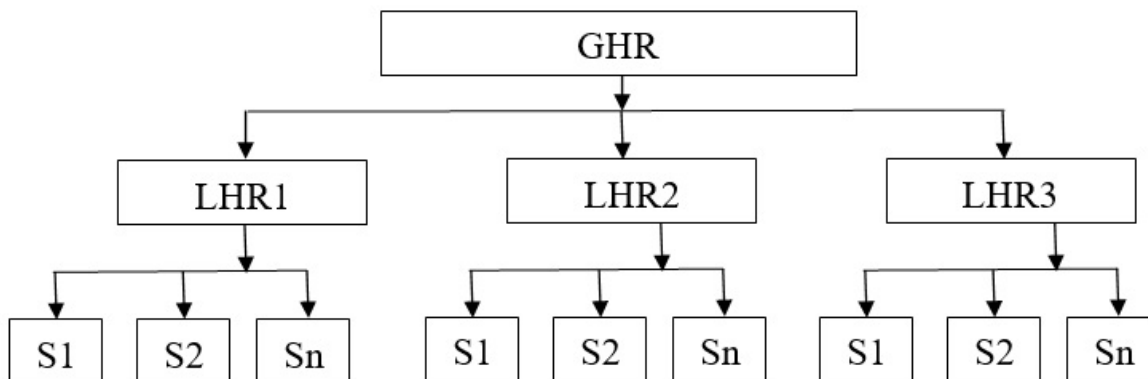


Рис.1. Загальна архітектура для системи резолюції Handle System

Література:

1. Churyumov G., Tokariev V., Tkachov V. Problem of self-organization of s-bot group movement in unorganized physical environment / G. Churyumov, V. Tokariev, V. Tkachov // Комп'ютерні та інформаційні системи і технології: тези доповідей третьої міжн. наук.-техн. конф., 23 - 24 квіт. 2019 р. – Харків, 2019. - С.16-17.
2. Серков О.А., Князев В.В., Лазуренко Б.О., Яковенко І.В., Чурюмов Г.І., Токарев В.В. Надширокосмугові технології в задачах забезпечення електромагнітної сумісності рухомих об'єктів / О.А. Серков, В.В. Князев, Б.О. Лазуренко, І.В. Яковенко, Г.І. Чурюмов, В.В. Токарев // Проблеми електромагнітної сумісності перспективних бездротових мереж зв'язку (ЕМС-

2019):збірник наукових робіт четвертої міжн. наук.-техн. конф., 24 жовт. 2019 р. - Харків, 2019. - С. 55-57.

3. Krivoulya G., Ilina I., Tokariev V., Shcherbak V. Mathematical Model for Finding Probability of Detecting Victims of Man-Made Disasters Using Distributed Computer System with Reconfigurable Structure and Programmable Logic / G. Krivoulya, V. Tokariev, I. Ilina, V. Shcherbak // IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology: (PIC S&T), 06-09 oct. 2020y. - Kharkiv, 2020. - P.573 - 576.

Науковий керівник: Токарев Володимир Володимирович, кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки

Танцинець В.В., студент кафедри програмного забезпечення, Національний університет «Львівська політехніка», м. Львів;
Журавчак Л.М., професор кафедри програмного забезпечення, Національний університет «Львівська політехніка», м. Львів;
Шинкаренко Г.А., професор кафедри інформаційних систем, Львівський національний університет імені Івана Франка, м. Львів

h-АДАПТИВНИЙ МЕТОД СКІНЧЕННИХ ЕЛЕМЕНТІВ З КУСКОВО-КУБІЧНИМИ АПРОКСИМАЦІЯМИ НА ТРИКУТНИКАХ

Дана робота присвячена розв'язанню варіаційної задачі дифузії-адвекції-реакції:

$$\begin{cases} \text{знайти } u \in V := \{v \in H^1(\Omega) : v = 0 \text{ на } \Gamma_u\} \text{ таку, що} \\ a(u, v) = \langle l, v \rangle \quad \forall v \in V, \end{cases} \quad (1)$$

де

$$\begin{cases} a(u, v) := \int_{\Omega} [(\mu \nabla u) \cdot \nabla v + v(\beta \cdot \nabla u + \sigma u)] dx + \int_{\Gamma_q} \alpha u v d\gamma \quad \forall u, v \in V, \\ \langle l, v \rangle := \int_{\Omega} f v dx + \int_{\Gamma_q} g v d\gamma \quad \forall v \in V, \end{cases} \quad (2)$$

$\Omega \subset R^2$ – обмежена зв'язна область з ліпшицевою межею

$$\Gamma = \Gamma_u \cup \Gamma_q, \Gamma_u \cap \Gamma_q = \emptyset,$$

$\mu = \mu(x) > 0$, $\beta = \{\beta_i(x)\}_{i=1}^2$, $\sigma = \sigma(x) \geq 0$, $f = f(x)$, $g = g(x) \geq 0$ і $\alpha = \alpha(x) \geq 0$ – задані достатньо регулярні функції, за деталями див. [1, 2, 4].

Для розв'язання задачі (1) застосовується h-адаптивна схема методу скінченних елементів (МСЕ) з кубічними апроксимаціями на трикутниках [3, 5].

Покриємо область Ω сіткою $\mathfrak{T}_h = \{K\}$ скінченних елементів K та перенесемо розв'язування задачі (1) у підпростір апроксимацій $V_h \subset V$, знайдемо наближений розв'язок задачі (1) як розв'язок дискретизованої задачі:

$$\begin{cases} \text{задано } h = \text{const} > 0 \text{ i } V_h \in V, \dim V_h = n < +\infty, \\ \text{знайти } u_h \in V_h : a(u_h, v) = \langle l, v \rangle \quad \forall v \in V_h. \end{cases} \quad (3)$$

Наближений розв'язок u_h можна подати у вигляді лінійної комбінації

$$u_h(x) = \sum_{j=1}^N q_j \varphi_j(x), \quad (4)$$

де $\varphi_j(x) \in P_3(K)$ – кубічні базисні функції МСЕ, а коефіцієнти q_1, \dots, q_N визначаються як розв'язок системи лінійних алгебричних рівнянь

$$\sum_{j=1}^N a(\varphi_j, \varphi_i) q_j = \langle l, \varphi_i \rangle, \quad i = 1, 2, \dots, N, \quad (5)$$

матриця якої є симетричною і має стрічкову структуру.

Щоб оцінити похибку $e := u - u_h$ знайденого наближення $u_h \in V_h$ ми будемо таку варіаційну задачу про лишок

$$\begin{cases} \text{задано } u_h \in V_h; \text{ знайти } e := u - u_h \in E, V = E \oplus V_h, \text{ таку, що} \\ a(e, v) = \langle l, v \rangle - a(u_h, v) \quad \forall v \in E. \end{cases} \quad (6)$$

Наближені розв'язки (апостеріорні оцінювачі похибок (АОП) апроксимацій МСЕ) задачі (6) шукаємо за схемою Гальоркіна в підпросторах $E_h \subset E, \dim E_h < +\infty$, які конструюються в такий спосіб [3]

$$e_h(x) = \sum_{K \in \mathfrak{Z}_h} L_1^K(x) L_2^K(x) L_3^K(x) [\lambda_1^K L_1^K(x) + \lambda_2^K L_2^K(x) + \lambda_3^K L_3^K(x)], \quad (7)$$

де $\{L_i^K(x)\}_{i=1}^3$ – барицентричні координати трикутника K . Знаходження значень коефіцієнтів $\{\lambda_i^K\}_{i=1}^3$ вимагає (згідно процедури Гальоркіна) формування і розв'язання системи із трьох лінійних алгебричних рівнянь на кожному скінченному елементі сітки.

Для обчислення апроксимацій МСЕ із наперед заданою точністю $TOL > 0$ застосовувався ітераційний алгоритм h -адаптування (локального покращення триангуляцій з використанням методу бісекції [4]). Тут спочатку обчислювався розподіл індикаторів АОП $\{\|e_h\|_K\}_{K \in \mathfrak{Z}_h}$, де

$$\|v\|_K^2 := \int_K [(\mu \nabla v) \cdot \nabla v + v(\beta \cdot \nabla v + \sigma v)] dx + \int_{\Gamma_q \cap \partial K} \alpha v^2 d\gamma \quad \forall K \in \mathfrak{Z}_h.$$

Пізніше визначалася множина \mathfrak{R}_h трикутників $K \in \mathfrak{Z}_h$, які не задовольняють умову наперед заданої допустимої похибки

$$\eta_K := \sqrt{N} \frac{\|e_h\|_K}{\|u_h + e_h\|_\Omega} 100\% \leq TOL, \quad (8)$$

де $\|v\|_\Omega = \sqrt{a(v, v)} = \sqrt{\sum_{K \in \mathfrak{Z}_h} \|v\|_K^2}$ і, нарешті, локальне покращення якості триангуляції здійснювалося поділом трикутників множини \mathfrak{R}_h методом бісекції [4].

Для розв'язування двовимірних задач дифузії-адвекції-реакції розроблений програмний комплекс в середовищі Microsoft Visual Studio з використанням мови програмування C#. В рамках цього комплексу двовимірні

задачі розв'язуються за допомогою лінійних, квадратичних та кубічних апроксимацій МСЕ на трикутниках.

Числові результати. Властивості побудованої схеми аналізувалися за результатами числових експериментів з модельною задачею [6]

$$\begin{cases} -\nabla(\nabla u) + 10.0u = f & \text{в } \Omega = (-1,1)^2, \\ u = g & \text{на } \Gamma = \partial\Omega, \end{cases}$$

з точним розв'язком у вигляді

$$u(x, y) = \frac{1.0}{(x-0.5)^2 + (y-0.5)^2 + 0.01} - \frac{1.0}{(x+0.5)^2 + (y+0.5)^2 + 0.01}.$$

Допустимий рівень похибки задавався $TOL = 1\%$. Результати рівномірного згущення наведено в табл. 1, а в табл. 2 такі ж результати для h -адаптивного згущення. Тут $E_i = Card(\mathfrak{T}_h)$ та $N_i = Nod(\mathfrak{T}_h)$ – кількість елементів та вузлів триангуляції \mathfrak{T}_h на i -му кроці обчислень апроксимації, $\varepsilon_h = \|e_h\|_{\Omega} \|u_h\|_{\Omega}^{-1} 100\%$ та $\varepsilon(u_h) = \|u - u_h\|_{\Omega} \|u\|_{\Omega}^{-1} 100\%$ – відносні похибки наближень, $p_h = 2 \ln(\|e_h\|_{\Omega} \|e_h\|_{\Omega}^{-1}) / \ln(E_{i+1} E_i^{-1})$, $p = 2 \ln(\|e\|_{\Omega} \|e\|_{\Omega}^{-1}) / \ln(E_{i+1} E_i^{-1})$ – порядки збіжності оцінювача та похибки відповідно, обчислені на двох послідовних кроках уточнення апроксимації.

Таблиця 1. АОП і порядки збіжності кубічних апроксимацій МСЕ на рівномірно згущуваних триангуляціях

i	N_i	E_i	$\varepsilon_h \%$	$\varepsilon(u_h), \%$	p_h	p	$Time, c$
1	181	36	13.882	61.863			0.040
2	685	144	5.629	23.351	1.39	1.34	0.064
3	2665	576	1.989	7.978	1.46	1.55	0.434
4	10513	2304	0.576	0.83	1.78	3.26	4.624

Таблиця 2. АОП і порядки збіжності h -адаптивних кубічних апроксимацій МСЕ

i	N_i	E_i	$\varepsilon_h \%$	$\varepsilon(u_h), \%$	p_h	p	$Time, c$
1	181	36	13.882	61.863			0.035
2	277	56	16.489	55.293	0.13	0.32	0.024
3	463	96	5.629	23.351	3.47	3.18	0.054
4	679	144	2.763	15.202	3.43	2.12	0.109
5	895	192	2.001	7.986	2.19	4.48	0.156
6	1111	240	1.253	2.904	4.17	9.07	0.156
7	1327	288	0.657	0.995	7.09	11.75	0.234

На рис. 1 наведена сітка на останньому кроці h -адаптивного алгоритму, що складається з 288 елементів, а на рис. 2 – графік наближеного розв'язку, отриманий на третьому кроці адаптування сітки.

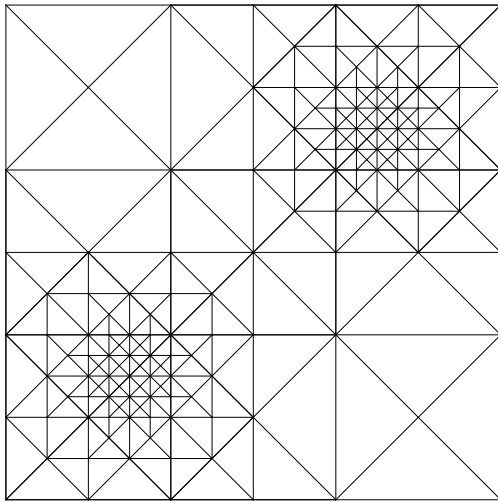


Рис. 1. \mathfrak{Z}_7 (Card $\mathfrak{Z}_7 = 288$)

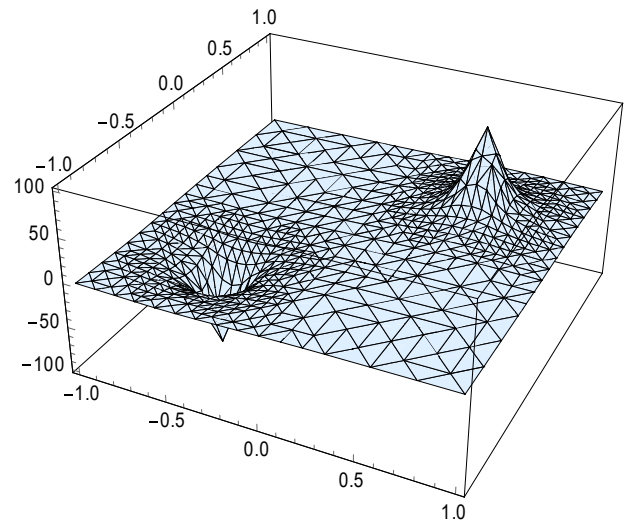


Рис. 2. Апроксимація u_h

Як видно з таблиць, за допомогою рівномірного згущення сітки можна отримати наближений розв'язок із необхідною точністю за чотири ітерації, тоді як h -адаптивна схема виконує сім ітерацій, але остаточна адаптивна сітка складається з 288 скінченних елементів, на відміну від 2304 скінченних елементів рівномірного згущення. Тобто використання h -адаптування дозволяє отримати розв'язок з набагато меншими обчислювальними витратами.

Література:

1. *Танчинець В.* h -адаптивний метод скінченних елементів з кусково-квадратичними апроксимаціями на трикутниках. Міжнародна студентська наукова конференція з питань прикладної математики та комп'ютерних наук МСНКПМК-2020: тези доповідей (Львів, 23-24 квітня, 2020 р.). Львів: ЛНУ ім. І. Франка. 2020. С. 81-85.
2. *Трушевський В.М., Шинкаренко Г.А., Щербина Н.М.* Метод скінченних елементів і штучні нейронні мережі. Теоретичні аспекти та застосування – Львів : ЛНУ імені Івана Франка, 2014 – 394с.
3. *Шинкаренко Г., Вовк О., Танчинець В.* Високоточні h -адаптивні методи скінченних елементів з кусково-поліноміальними апроксимаціями на трикутниках. XXVI Міжнародна наукова конференція «Сучасні проблеми прикладної математики та комп'ютерних наук» АРАМС-2021: тези доповідей (Львів, 27-28 вересня, 2021 р.). Львів: ЛНУ ім. І. Франка. 2021. С. 184-188.
4. *Ostapov O.Yu., Shynkarenko H.A., Vovk O.V.,* A posteriori error estimator and h -adaptive finite element method for diffusion-advection-reaction problems. Recent Advances in Computational Mechanics, London, Taylor & Francis Group. – 2014. – P. 329-337.
5. *Zienkiewicz O. C., Taylor R.L., Zhu J.Z.* The Finite Element Method: Its Basis and Fundamentals Sixth edition – Oxford, 2005 – p. 733.

6. Zhang W., Nie Y., Gu Y. Adaptive finite element analysis of elliptic problems based on bubble-type local mesh generation. J. Comput. App. Math. 280 (2015) – P. 42 – 58.

Науковий керівник: Журавчак Л.М., професор кафедри програмного забезпечення, Національний університет «Львівська політехніка», м. Львів

*Тарасов М.С., студент, кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка, кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

ОГЛЯД НОВИХ СТАНДАРТІВ, ЩО ВИКОРИСТОВУЮТЬСЯ В INTERNET OF THINGS

Інтернет речей (ІоТ), як і багато інших сучасних ідей, являє собою просту концепцію. Підключаючи об'єкти до мережі Інтернет, можна об'єднати можливості віртуального і реального світів.

Поєднання двох елементів, які зазвичай функціонують окремо, відкривають величезні можливості для більш ефективного використання ресурсів [1]. Промисловість використовує ІоТ для оптимізації процесів розробки і виробництва продукції [2, 3]. Люди ж використовують дані можливості для підвищення комфорту у своєму житті, наприклад, починаючи від оптимізації надання тепла, світла, безпеки, закінчуючи наданням їжі, пристосованої до їхніх потреб.

Наразі одними з популярних стандартів в сфері Інтернету речей і пов'язаними з ними технологіями є три нових стандарти, розроблені ISO, які допомагають більш вигідно реалізувати потенціал ІоТ:

- ISO / IEC 21823-2 містить структуру і вимоги до функціональної сумісності з транспортом, в цілях забезпечення можливості побудови систем ІоТ з обміном інформації, одноранговими сполуками і безшовним зв'язком як між різними системами ІоТ, так і між організаціями в системі інтернету речей.

- ISO / IEC TR 30164 містить загальні поняття, термінологію, характеристики, приклади використання і технології (включаючи управління даними, координацію, обробку, мережеві функції, гетерогенні обчислення, безпеку, апаратну / програмну оптимізацію) периферійні обчислення для додатків систем ІоТ. Прикордонні обчислення — це галузь технологій, які лежать в основі служб віддалених обчислень і зберігання, таких як «хмара».

- ISO / IEC TR 30166 застосовується у загальних системах і у промисловому секторі (ІоТ), з описом характеристик, технічних аспектів і функціональних, а також не функціональних елементів структури ІоТ і

переліку організацій по стандартизації, консорціумів і спільнот з відкритим вихідним кодом, що працюють над усіма аспектами IoT.

Основною значущою тенденцією Інтернету речей в останні роки є вибухова динаміка пристроїв, підключених і контрольованих інтернетом [4]. Інтернет–засоби створюють можливості для більш прямого фізичного контакту в комп'ютерні системи, що дозволяє підвищити ефективність, економічні вигоди і зниження навантаження на людину, саме для цього постійно з'являються нові стандарти, що допомагають підвищити ефективність та швидкість зв'язку розумних приладів з Інтернетом речей [5].

Література:

1. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Pp. 25-26.
2. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:<https://doi.org/10.26906/SUNZ.2021.1.090>.
3. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
4. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii Tkachov, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdetka // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020
5. Ruban I.V., Churyumov G.I., Tokariiev V.V., Tkachov V.M. Structural-functional reconfiguration of computer systems with reconstruct structure. Проблеми інформатики та моделювання: тези доповідей 19-ї міжн. наук.-техн. конф., м. Одеса, 11-16 вер. 2019р. Одеса, С.71 — 72.

*Філіп'єва М.В., студентка, кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків;
Гвоздецька К.П., студентка, кафедра Електронних обчислювальних машин,
Харківський національний університет радіоелектроніки, м. Харків*

ПОРІВНЯННЯ СИМЕТРИЧНОГО І АСИМЕТРИЧНОГО ШИФРУВАННЯ

Криптографічні системи розділені на дві основні галузі дослідження: симетрична і асиметрична криптографія. Симетричне шифрування часто

використовується як синонім симетричною криптографії, а асиметрична криптографія охоплює два основні варіанти використання, це асиметричне шифрування і цифрові підписи.

Алгоритми шифрування діляться на дві категорії, відомі як симетричне і асиметричне шифрування. Принципова відмінність між цими двома методами полягає в тому, що алгоритми симетричного шифрування використовують один ключ, в той час як асиметричні використовують два різних, але пов'язаних між собою ключа [1].

Симетричний алгоритм підходить для передачі великих обсягів шифрованих даних. Асиметричний алгоритм, за інших рівних, буде значно повільніше. Крім того, для організації обміну даними по асиметричному алгоритму або обом сторонам повинні бути відомі відкритий і закритий ключ, або таких пар повинно бути дві (по парі на кожену сторону) [2].

Асиметричне шифрування дозволяє зробити безпечне з'єднання без зусиль з боку користувача, а симетричний алгоритм передбачає, що користувачеві необхідно ще дізнатися пароль [3]. Проте, варто розуміти, що асиметричні алгоритми так само не забезпечують повної безпеки. Наприклад, вони схильні до атак " Man-in-the-middle". Суть останньої полягає в тому, що між вами і сервером встановлюється комп'ютер, який для вас відсилає свій відкритий ключ, а для передачі даних від вас використовує відкритий ключ сервера.

Симетричні алгоритми зазвичай будуються на основі деяких блоків з математичними функціями перетворення [4]. Тому модифікувати такі алгоритми легше. Асиметричні ж алгоритми зазвичай будуються на деяких математичних задачах, наприклад, RSA побудований на завданні зведення в ступінь і взяття по модулю. Тому їх практично неможливо або дуже складно модифікувати.

Як симетричне, так і асиметричне шифрування грає важливу роль в забезпеченні безпеки конфіденційної інформації та комунікації в сучасному цифровому світі [5]. Обидва шифру можуть бути корисні, адже у кожного з них є свої переваги і недоліки, тому вони застосовуються в різних випадках. Оскільки криптографія як наука продовжує розвиватися для захисту від новіших і більш серйозних загроз, симетричні і асиметричні криптографічні системи завжди будуть мати відношення до комп'ютерної безпеки.

Література:

1. T. Vitalii, B. Anna, H. Kateryna and D. Hrebenuk, "Method of Building Dynamic Multi-Hop VPN Chains for Ensuring Security of Terminal Access Systems," 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T), 2020, pp. 613-618, doi: 10.1109/PICST51311.2020.9467953.
2. Tkachov V. Principles of Constructing an Overlay Network Based on Cellular Communication Systems for Secure Control of Intelligent Mobile Objects / Vitalii

- Ткачов, Andriy Kovalenko, Mykhailo Hunko and Kateryna Hvozdetska // Информационные технологии и безопасность. Материалы XIX Международной научно-практической конференции ИТБ-2020. – К.: ООО «Инжиниринг», 2020
3. Метод забезпечення живучості високомобільної комп'ютерної мережі / В. М. Ткачов [та ін.] // Сучасні інформаційні системи = Advanced Information Systems. – 2021. – Т. 5, № 2. – С. 159-165.
 4. Коваленко А.А. Метод забезпечення живучості комп'ютерної мережі на основі VPN-тунелювання / А.А. Коваленко, Г.А. Кучук, В.М. Ткачов // Системи управління, навігації та зв'язку. Збірник наукових праць. – Полтава: ПНТУ, 2021. – Т. 1 (63). – С. 90-95. – doi:<https://doi.org/10.26906/SUNZ.2021.1.090>.
 5. Predicting Traffic Anomalies in Container Virtualization / N. Kuchuk, A. Kovalenko, V. Tkachov, D. Rosinskiy, H. Kuchuk // Fifth International Scientific and Technical Conference "Computer and Information Systems and Technologies", 2021. – Pp. 25-26.

Секція 2. Економічні науки

*Боднар О.В., к.е.н., доцент, кафедра обліку і аудиту,
ВП НУБіП України «Брежанський агротехнічний інститут»,
м. Брежани*

УПРАВЛІНСЬКИЙ АНАЛІЗ ТА ЙОГО РОЛЬ В ДІЯЛЬНОСТІ ПІДПРИЄМСТВА

Успіх та розвиток діяльності будь-якого підприємства визначається обгрунтованістю управлінських рішень. Але часто рішення, що приймаються керівництвом, не мають належного економічного обгрунтування. Ефективність роботи окремих сегментів діяльності на сільськогосподарських підприємствах аналізується не завжди. У зв'язку із цим актуальними є становлення та розвиток управлінського аналізу.

Методика здійснення управлінського аналізу для діяльності сільськогосподарських підприємств досліджувалися у працях таких вітчизняних та зарубіжних вчених, як І. Ансофф, М.А. Вахрушина, Є. В. Мних, Н.А. Нікіфорова, Б. Райан, І. Д. Фаріон, М. Г. Чумаченко та інші.

Управлінський аналіз покликаний перетворити економічну та неекономічну інформацію в придатну для прийняття рішення. Логічна обробка, вивчення, узагальнення фактів, їх систематизація, висновки, пропозиції, пошук резервів – все це завдання управлінського аналізу, який покликаний забезпечити обгрунтованість управлінського рішення і підвищити його ефективність.

На думку Г.В. Савицької, управлінський аналіз проводять всі служби підприємства з метою отримання інформації, необхідної для планування, контролю та прийняття оптимальних управлінських рішень, розроблення стратегії і тактики з питань фінансової політики, маркетингової діяльності, вдосконалення техніки, технології і організації виробництва. Він носить оперативний характер, його результати – комерційна таємниця [4, с. 35].

Більшість вчених трактують цей вид аналізу як аналіз, результати якого використовуються керівництвом всередині підприємства для прийняття управлінських рішень.

Дослідивши різні підходи до сутності поняття «управлінський» аналіз, можна виділити такі ознаки управлінського аналізу:

- орієнтація на внутрішнє споживання;
- використання релевантної неpubлічної інформації;
- перспективна спрямованість результатів аналізу;
- основа для прийняття управлінських рішень;
- конфіденційний характер;
- оперативність;
- галузева орієнтація.

Вважаємо, що ефективність проведення управлінського аналізу багато в чому залежить від того, наскільки управлінський облік адаптований до реальних потреб підприємства.

Отже, управлінський аналіз як функція керуючої системи включає оцінку внутрішніх і зовнішніх факторів ситуації, загальних тенденцій розвитку економічних процесів, можливих резервів підвищення ефективності виробництва; передбачає оцінку ступеня напруженості і виконання плану по всіх видах показників, вивчення ходу оперативного виконання плану, збурюючих причин, шляхів їх усунення.

Управлінський аналіз, спираючись на дані обліку, становить базу обґрунтованого планування та завершує виконання плану і йде в ході його оперативного здійснення.

Література:

1. Мних Є. В. Економічний аналіз : підручник. Київ : Знання, 2011. 630 с.
2. Кіндрацька Г.І., Білик М.С., Загородній Г. Економічний аналіз : навч. посібник. Київ: Знання, 2008. 487с.
3. Гайдаєнко О.М., Шевчук Н.С. Управлінський аналіз: навч. посібник. Одеса, 2015. 151 с.
4. Савицька Г.В. Економічний аналіз діяльності підприємства : навч. посібник. Київ : Знання, 2007. 668 с.

Левчук О.В., к.е.н., доцент, провідний науковий співробітник науково-дослідного відділу воєнно-економічних проблем, Центр воєнно-стратегічних досліджень Національного університету оборони України імені Івана Черняхівського, м. Київ

Онофрійчук О.А., ад'юнкт науково-дослідного відділу воєнно-економічних проблем, Центр воєнно-стратегічних досліджень Національного університету оборони України імені Івана Черняхівського, м. Київ

ВИЗНАЧЕННЯ МЕХАНІЗМУ РОЗПОДІЛУ ФІНАНСОВИХ ОБОРОННИХ РЕСУРСІВ ДЛЯ ЕФЕКТИВНОГО РОЗВИТКУ СПРОМОЖНОСТЕЙ СИЛ ОБОРОНИ УКРАЇНИ

Утримання державою високоякісної професійної армії, спроможної забезпечити воєнну безпеку держави в сучасних умовах глобальних політичних, економічних, енергоресурсних та екологічних змін існування людства неможливо без всебічного обґрунтування її призначення, цілей та способів застосування, якісним забезпеченням відповідними ресурсами, їх розподілом, використанням і відновленням для досягнення та підтримання необхідних спроможностей.

Якісне забезпечення та розподіл оборонних ресурсів в арміях провідних країн світу базується на сучасному науковому підході – методології програмно-

цільового планування, яке запроваджується в Україні зі змінним успіхом з 1996 року. Провідні країни світу та країни блоку НАТО протягом останніх десятиліть використовують саме цю методологію, адаптувавши її до умов національних економік та чинного законодавства.

Цим шляхом йде і Україна, яка поступово формує та впроваджує свою воєнно-економічну політику, метою якої є всебічне задоволення обґрунтованих і визначених з точки зору оборонної достатності потреб Збройних Сил України та інших суб'єктів сил оборони у фінансових і матеріальних ресурсах. Разом із тим, в питаннях впровадження методу оборонного планування, заснованого на спроможностях (ОПОС) в практику сил оборони, існують певні труднощі. Справа в тім, що процедури даного методу планування є доволі трудомісткими і потребують значної кількості підготовлених фахівців. З іншого боку, існує певна неврегульованість питань в нормативно-правовій базі країни. Це стосується визначення нормативним порядком чіткого алгоритму реалізації методу оборонного планування, заснованого на спроможностях і орієнтованого на загрози, відповідальності певних посадових осіб (організаційних структур) за відпрацювання певних етапів (фаз) оборонного планування, питання підпорядкованості між складовими сил оборони.

Крім того, існують певні труднощі у “копіюванні” досвіду оборонного планування в провідних країнах світу. Національні особливості досвіду військового будівництва в Україні, відсутність жорстко регламентованих правил оборонного планування, заснованого на спроможностях в країнах-членах НАТО призводить, з одного боку, до певної свободи дій, а, з іншого боку – ставить перед необхідністю самостійного вироблення свого національного методу оборонного планування, заснованого на спроможностях і орієнтованого на загрози.

Одним із важливих питань впровадження методу ОПОС в практику сил оборони – є розроблення автоматизованої системи підтримки прийняття рішень (СППР) процесу оборонного планування в силах оборони.

Так, в стратегічному оборонному бюлетені України, введеному в дію Указом Президента України від 6 червня 2016 року вказується, що проведена в рамках комплексного огляду сектору безпеки і оборони оцінка стану воєнної безпеки держави виявила низку проблем функціонування сил оборони в умовах існуючих та потенційних загроз, серед яких визначені наступні:

- недосконалість процедур оборонного планування, їх недостатня узгодженість з бюджетним процесом, недосконалість механізмів програмного управління оборонними ресурсами;

- невідповідність потужностей виробництва потребам оборонного замовлення, критичне фізичне і моральне зношення основних виробничих фондів;

- недостатні оперативні (бойові, спеціальні) спроможності сил оборони;

- відсутність ефективної об'єднаної системи логістики, яка здатна підтримувати роботу всіх складових сил оборони;

- критично низький рівень оперативних запасів матеріально-технічних засобів;

відсутність автоматизованої системи управління у сфері матеріально-технічного забезпечення.

Для удосконалення процедури оборонного планування розвитку сил оборони України на основі спроможностей з наукової точки зору необхідно вирішити три глобальних завдання: визначитись з теоретико-методологічним підґрунтям вирішуваного завдання; привести у відповідність нормативно-правову базу держави і розробити автоматизовану СППР, яка сприятиме впровадженню нового для України методу оборонного планування з урахуванням стандартної послідовності виконання обов'язкових процедур.

Таким чином, вирішення завдання удосконалення процедури оборонного планування розвитку сил оборони України на основі спроможностей вбачається у розробленні автоматизованої системи підтримки прийняття рішення процесу оборонного планування в силах оборони України.

Автоматизація процесу оборонного планування, заснованого на спроможностях, дозволить вирішити певне коло завдань, а саме:

знизити певний рівень вимог до осіб, які займаються оборонним плануванням стосовно вирішення оптимізаційних завдань;

спростити процес виконання обов'язкових процедур для відпрацювання оборонного планування з використанням методу оборонного планування, заснованого на спроможностях.

Це дозволить фахівцям з оборонного планування зосередити свою увагу на суті оборонного планування та менше звертати увагу на рутинні моменти здійснення процедури оборонного планування. Зазначене вирішить проблему складності виконання процедури, позбавить від великої кількості помилок, які пов'язані з технічними моментами, зробить роботу фахівців з оборонного планування більш комфортною. Подальше розповсюдження СППР дозволить стандартизувати процедуру оборонного планування в силах оборони України та розширити коло фахівців з оборонного планування.

Все зазначене сприятиме вирішенню завдання впровадження оборонного планування, заснованого на спроможностях та орієнтованого на загрози, що призведе до підвищення рівня ефективності розвитку сил оборони України та безпосереднього підвищення рівня відповідності сил оборони України своєму призначенню.

Мямлін В.В., д.т.н., с.н.с., професор, кафедра «Вагони та вагонне господарство», Дніпровський національний університет залізничного транспорту імені академіка В. Лазаряна, м. Дніпро

ДО ПИТАННЯ ПРО ВРЕГУЛЮВАННЯ РІВНОВАГИ МІЖ ЗАГАЛЬНОЮ ВАРТІСТЮ ТОВАРІВ І КІЛЬКІСТЮ ГРОШЕЙ, ПРИЗНАЧЕНИХ ДЛЯ ЇХ СПОЖИВАННЯ

В економіці існують дуже гострі проблеми, про які не прийнято говорити, але від правильного рішення яких цілком залежить ефективність функціонування всієї макроекономічної системи держави. Одним з таких

питань є питання співвідношення загальної вартості товарів і кількістю грошей, які можуть потенційно використовуватися для придбання цих же товарів. Автор вже неодноразово звертався до цього питання [1-11], але навколо нього до сих пір існує інформаційний вакуум, який свідчить про те, що економічна громадськість поки ще недостатньо глибоко усвідомила всю важливість цього питання для розвитку економіки держави.

Макроекономічна система, як і будь-яка інша реальна система, складається з безлічі елементів, які тісно взаємопов'язані між собою. Коли протиріч між елементами немає, системи функціонують нормально, без збоїв. Коли з'являються протиріччя, системи перестають виконувати свою функцію або виконують її дуже неефективно. В економіці така ситуація називається кризою. Механізм функціонування макроекономічної системи може ґрунтуватися на різних принципах, і результати функціонування при різних принципах можуть дуже сильно відрізнятись один від одного. Чи можна зробити так, щоб криз не було взагалі? Можно! Треба визначити між якими елементами системи виникають суперечності та усунути їх.

В основу даного дослідження покладено такі логічні припущення: 1. Вартість товарів (продукти і послуги) пов'язана тільки з кількістю та якістю витраченої людської праці, а її чисельне значення визначається ринком; 2. Все, що створено Природою самостійно, без участі людської праці, вартості не має, хоча може мати певну цінність; 3. Всі гроші в країні повністю належать державі, яка здійснює строгий контроль за ними. Люди і підприємства можуть тільки короткочасно користуватися грошима для можливості товарообміну.

Пояснимо це на прикладах. Сам газ, який знаходиться в надрах землі, не є товаром і ніякої вартості не має, тому що людина не використовувала свою працю для його створення. Але вартість має праця з розвідки газових родовищ, видобутку газу, транспортування, зберігання, будівництва, а також обслуговування і ремонту трубопроводів та обладнання. Тому газ, який надходить споживачеві, теж має вартість, так як в нього вкладено людську працю. Вартість цього газу повинна дорівнювати сумарній вартості вкладеної праці у весь технологічний ланцюжок від видобутку газу до його поставки споживачеві. Це є науково обґрунтованим рішенням. Якщо ціна газу перевищує вартість праці або нижче її вартості – це вже не науково обґрунтована економіка, а соціально-економічна політика.

Що стосується грошей, то вони повинні бути в безготівковій формі, повністю належати державі та контролюватися державою, а то виходить, що держава, яка здійснює емісію грошей, сама весь час в них потребує, про що свідчить постійний дефіцит бюджету. Незважаючи на те, що вже з'явилися нові технічні можливості, загальна фінансово-економічна схема держави продовжує функціонувати по старому.

Свого часу гроші були винайдені для полегшення обміну товарами, і вони дійсно стали дуже зручним посередником в цих угодах. Спочатку Товар 1 обмінювався на гроші, а потім гроші обмінювалися на Товар 2 ($T_1 \rightarrow G \rightarrow T_2$). Деякий час гроші мали внутрішню цінність, так як виготовлялися з дорогоцінних металів, які теж були товаром. Згодом гроші втратили внутрішню

цінність, але за звичкою продовжували мати цінність зовнішню. Крім того, зараз виготовлення грошей практично нічого не коштує, але купівельну спроможність вони мають і тому продовжують залишатися вартісним замінником товару. У цьому теж не було б нічого поганого, якби якісь не зовсім чесні люди не вирішили виробляти гроші приватно і в величезній кількості. Логіка їх проста: навіщо виробляти товар – це дуже складно і трудомістко, краще виробляти самі гроші та за них купувати вже готовий товар. Відбулося розмежування між товарами і грошима. Вся справа в тому, що гроші пішли в самотійне плавання і вже давно перестали грати свою головну роль, для якої були спочатку призначені – бути фінансовим еквівалентом товару. Відбувається якийсь парадокс. При їх вже колосальній кількості в світі та триваючому зростанні грошової маси, їх постійно не вистачає. Насправді ж кількість грошей в обігу має бути незмінною. А додаткова потреба, що виникає в них, повинна вирішуватися шляхом збільшення швидкості їх обігу, тобто інтенсивним способом, а не збільшенням грошової маси в обігу (екстенсивний шлях).

Крім того, розвиток національної економіки не повинен залежати від екзогенних факторів, від того, дадуть їй гроші міжнародні фінансові структури чи не дадуть? Національна економіка, тим більше незалежної держави, повинна залежати тільки від своїх внутрішніх рішень. Якщо це правильні рішення, то економіка буде ефективно функціонувати та інтенсивно розвиватися, якщо рішення неправильні, то економіка буде схильна до всіх негативних явищ, властивих ліберальній економіці.

Справжня, правильна економіка – це виробництво товарів і розподіл їх між членами суспільства за допомогою грошового еквівалента вартості товарів. Якщо ж грошового еквівалента буде недостатньо, то товари усі викуплені не будуть, якщо його буде більше, то ціни на товари будуть зростати.

Між підприємствами і державою мають існувати партнерські відносини. Держава повинна не просто збирати оброк з підприємств у вигляді маси різних податків, а максимально сприяти роботі підприємств, в тому числі й щодо забезпечення безпроцентними позиками. А підприємства в подяку повинні справно віддавати державі її «частку».

Припустимо, що попит на товар є завжди, так як товар виробляється тільки такий, який суспільство потребує. Так як товар продається за гроші, то у споживачів на руках має бути кількість грошей, відповідна вартості товару. Для розумного і ефективного функціонування макроекономіки, кількість грошей в обігу має бути строго фіксованою, а їх додаткова потреба повинна регулюватися швидкістю обігу грошової маси.

При існуючих принципах функціонування економіки, грошей весь час не вистачає. Тому в них і постійно виникає потреба. Як вчить нас сучасна економічна теорія, в ціні товару закладена заробітна плата, яка становить лише частину ціни товару, а товар треба продавати дорожче, щоб отримати «прибуток». З огляду на той факт, що товар призначений для тих же самих людей, які виготовили цей товар, виникає резонне питання: як люди, отримуючи у вигляді зарплати тільки частину вартості товарів, можуть купувати за неї весь товар, адже частина не дорівнює цілому. В цьому і полягає

основне протиріччя сучасної економіки, а саме протиріччя між вартістю товарів і кількістю грошей в вигляді зарплати за свою працю, що входять в цю вартість. Тому-то і відбувається час від часу підвищення зарплат і пенсій за рахунок додаткових емісій, так як купівельна спроможність населення сильно знижується. Теж саме можна сказати і про підприємства, яким також весь час не вистачає обігових коштів. Але збільшення грошової маси викликає інфляцію, що також негативно позначається на купівельній спроможності населення. Категорія «прибуток» для того і була впроваджена в економічну теорію, щоб виправдати роботу друкарського верстата і дозволити фінансовим олігархам здійснювати неконтрольовану емісію грошей в своїх особистих інтересах. Насправді ж економіка без емісії буде розвиватися набагато краще.

Зовсім іншу картину можна побачити в разі рівноважної моделі, коли вартість товару дорівнює вартості праці.

Валовий дохід підприємства (ВДП) може бути розкладений на такі складові

$$\text{ВДП} = \text{ОП} + \text{ЗП} + \text{ЧД},$$

де ОП – оплата постачальникам (сировина, комплектуючі, енергоносії, обладнання, повернення позик тощо);

ЗП – зарплата безпосередньо працівників підприємства, у тому числі і власників;

ЧД – «частка держави» (один єдиний податок).

А оплата постачальникам в свою чергу також поділяється на оплату працівників підприємств-постачальників, «частку держави», виплачену постачальниками і оплату іншим підприємствам-постачальникам. І так далі.

Таким чином ВВП держави буде дорівнювати

$$\text{ВВП} = \sum \text{ВДП} \text{ або } \text{ВВП} = \sum \text{ЗП} + \sum \text{ЧД},$$

де $\sum \text{ЗП}$ – сумарна заробітна плата працівників всіх підприємств за рік;

$\sum \text{ЧД}$ – сумарна «частка держави», сплачена всіма підприємствами за рік.

З усього сказаного випливає висновок, що економіка держави могла б ефективно функціонувати без криз, без інфляції, з відсутністю безробіття, якби ВВП відповідав би сумі всіх зарплат і соціальних допомог, отриманих протягом цього ж періоду, як працівниками підприємств всіх форм власності, а також бюджетної сфери і громадянами, які отримують всі види соціальної допомоги.

Пропонована економічна Модель ліквідує вказаний вище дисбаланс і дозволить національній економіці розвиватися прискореними темпами, крім того держава стає самодостатньою і незалежною від іноземних кредитів.

Тому, якщо прибрати помилкову категорію «прибуток», що порушує дисбаланс між вказаними елементами і працює на світових фінансових олігархів, то відпаде і потреба в емісії. А «економіко-фінансова» система почне нормально функціонувати і розвиватися швидкими темпами. У правильній, науково обґрунтованій економіці, ціни повинні весь час знижуватися за рахунок досягнень науково-технічного прогресу, а не підвищуватися, як це відбувається в даний час.

Імітаційне моделювання зазначених принципів на комп'ютері повністю підтвердило правильність пропонованих рішень для ефективного

функціонування національної економіки.

Нові науково обґрунтовані принципи функціонування механізму макроекономічної системи, пропонуються урядам різних країн для якнайшвидшого усунення негативних явищ національних економік та різкого підвищення їх ефективності.

Література:

1. Мямлин В. В. Существующая прибыльно-финансовая модель хозяйствования - основная причина кризиса мировой экономической системы и краха финансовой системы. *Вісник ДНУЗТ. Вип. 25.* Дніпропетровськ : Вид-во ДНУЗТ, 2008. С. 241-247.
2. Мямлин В. В. Теория бесприбыльной альтернативной экономики как основа нового экономического мировоззрения. *Вісник ДНУЗТ. Вип. 26.* Дніпропетровськ : Вид-во ДНУЗТ, 2009. С. 222-230.
3. Мямлин В. В. К вопросу о категории прибыли. *Вісник ДНУЗТ. Вип. 29.* Дніпропетровськ: Вид-во ДНУЗТ, 2010. С. 268-279.
4. Мямлин В. В., Мямлин С. В. Основные научные принципы создания высокоэффективного экономико-финансового «механизма» в государстве. *Независимое исследование с позиций IT-технологий: монография.* Киев-Днепр : Монолит, 2019. С. 40-58.
5. Мямлін В., Мямлін С. Як Україні запустити потужний економічний «двигун»? *Світ.* 2018. № 3-4. С. 3.
6. Мямлин В. В., Мямлин С. В. Сбалансированность товарной и денежной систем без дополнительных эмиссий денег как основное условие интенсивного развития экономики государства. *Світ економічної науки. Випуск 9 : зб. тез міжн. наук.-практ. інтернет-конф. (м. Тернопіль, 27 листопада 2018 р.).* Тернопіль, 2018. С. 135-140.
7. Мямлин В. В. К вопросу о ложных категориях в экономической науке. *Science and education: problems, prospects and innovations. Proceedings of the 7th International scientific and practical conference.* CPN Publishing Group. Kyoto, Japan. 2021. Pp. 738-747.
8. Мямлін В. В. Що потрібно змінити в економічній теорії, щоб вона стала потужним творчим інструментарієм? *Проблеми та перспективи розвитку залізничного транспорту: матер. 81 міжн. наук.-практ. конф. (м. Дніпро, 22.04-23.04.2021 р.).* Дніпро, 2021. С. 274-276.
9. Мямлін В. В. Фінансова рівновага виробництва і споживання як основа нормального функціонування економіки. *Світ наукових досліджень. Вип. 1 : матер. міжн. наук.-практ. інтернет-конф. (м. Тернопіль, 22.06.2021 р.).* Тернопіль, 2021. С. 14-18.
10. Мямлін В. В. «Соціально-економічна» технологія як інструмент створення високоефективного господарського механізму в державі. *Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення (вип. 56) : зб. тез допов. міжн. наук. інтернет-конф. (м. Тернопіль, 10 березня 2021 р.).* Тернопіль, 2021. С. 62-66.

11. Мямлін В. В. Аксиоматичний підхід до теорії високоефективної національної економіки. *П'ятдесят четверті економічно-правові дискусії* : матер. міжн. наук.-практ. інтернет-конф. (м. Львів, 24 лютого 2021 р.). Львів, 2021. С. 25-28.

*Сливінська О.Б., к.е.н., доцент, кафедра обліку і аудиту,
ВП Національного університету біоресурсів і природокористування України
«Бережанський агротехнічний інститут», м. Бережани*

РОЛЬ БІЗНЕС-ДІАГНОСТИКИ В УПРАВЛІННІ ПІДПРИЄМСТВОМ

Існування будь-якого підприємства пов'язано з розробкою, прийняттям і реалізацією управлінських рішень. Від того, які саме управлінські рішення розробляються і реалізуються, залежить поточна та перспективна конкурентоспроможність підприємства, ефективність його діяльності. Від забезпечення системи управління інформацією про стан та перспективи діяльності, існуючі й потенційні проблеми, альтернативні сценарії розвитку залежить ефективність роботи суб'єкта господарювання.

У сучасних умовах господарювання керівництво підприємств має враховувати динамічність зовнішнього оточення, підвищений рівень ризику виконання поточної діяльності, що обумовлений як внутрішніми, так і зовнішніми факторами. Розробка, ухвалення і реалізація рішень тактичного й стратегічного характеру здійснюється на основі аналізу поточного стану підприємства та перспектив його розвитку. Такий аналіз проводиться шляхом застосування методів бізнес-діагностики з метою виявлення та вирішення проблем підприємства в процесі досягнення поставлених цілей.

Діагностика – це комплексний аналіз поточного стану і перспектив розвитку підприємства з метою розв'язання проблем і використання переваг, що виникають і створюються в процесі його діяльності. Діагностика повинна інформаційно забезпечувати прийняття рішень, на які істотно впливають фактичні або прогнозні дані про фінансовий стан підприємства. Завдання діагностики фінансового стану повинні бути підпорядковані завданням управління підприємством у трьох сферах діяльності: операційній (виробничій), інвестиційній та фінансовій, що поєднанні рухом фінансових ресурсів.

Мета економічної діагностики – комплексна оцінка фінансового стану, ділової активності підприємства для пошуку шляхів поліпшення його платоспроможності, виявлення змін у фінансовому стані в просторі і часі, визначення і прогнозування основних факторів впливу на фінансовий стан підприємства. Діагностика підприємства – це метод аналізу стану внутрішнього середовища організації, встановлення проблем та вузьких місць, які спричиняють відхилення та деформації об'єкта від норми чи цілі, виявлення потенційно сильних та слабких сторін, що, впливають на формування та реалізацію ефективною стратегії організації.

Важливість здійснення діагностики як методу дослідження визначається її необхідністю у розробці та реалізації стратегії розвитку підприємства та

реалізації ефективного менеджменту. Фінансова діагностика надає максимально достовірні та переконливі результати аналізу всіх сфер діяльності підприємства. Результати діагностики та запропоновані рекомендації допоможуть керівництву у розробці гнучкої стратегії ефективного функціонування та подальшого динамічного розвитку підприємства.

Отже, діагностика створює інформаційну систему підтримки прийняття управлінських рішень, яка ґрунтується на комплексному та систематичному дослідженні усіх сторін фінансово-господарської діяльності підприємства, а також визначає способи впливу на фінансові параметри його роботи.

Література:

1. Кривов'язюк, І. В. Економічна діагностика: навч. посіб. К.: Центр учбової літератури, 2013. 456 с.
2. Суздальцев, О. М. Антикризове управління як один із напрямів підвищення ефективності діяльності підприємства. *Вісник Донецького університету економіки та права*. 2012. № 1. С. 88-92.
3. Чернишов, В. Фінансова діагностика як система забезпечення фінансової стійкості підприємства. *Вісник Харківського національного економічного університету*. 2012 р. № 4. С. 76-80.

Секція 3. Технічні науки

Bozhko K.M., PhD, Assistant professor, Department of Information and Measuring Technologies, National Technical University of Ukraine

“Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv;

Morozova I.V., graduate student, Department of Information and Measuring Technologies, National Technical University of Ukraine

“Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv;

Kirilenko E.A., student, Department of Information and Measuring Technologies, National Technical University of Ukraine

“Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv

AIR QUALITY MONITOR DM72T IN MEASUREMENTS OF PM_{2,5} AND PM₁₀ DUST CONCENTRATION

Air quality monitoring in Kyiv is by stationary meteorological stations. Measurement results is an interactive map in real time [1]. Dust concentration of PM_{2,5} and PM₁₀ is the main parameter for air quality.

Air quality monitor DM72T provides measurement of particular matter (PM) in the range 0-999 mg/m³. The resolution of the device is 0,001 mg/m³. Measuring principle is the Laser Scattering (LS).

PM_{2,5} air quality rating range:

- Excellent: 0-35 mg/m³;
- Good: 36-75 mg/m³;
- Light pollution: 76-115 mg/m³;
- Moderate pollution: 116-150 mg/m³;
- Heavy pollution: 151-250 mg/m³;
- Serious pollution: more than 250 mg/m³.

There are also similar ratings for PM₁₀.

Results of measuring for PM_{2,5} in Kyiv (October 12, 2021):

- Pechersk-1: 39,2 mg/m³;
- Pechersk-2: 12,8 mg/m³;
- Shulyavka-1: 21,2 mg/m³;
- Shulyavka-2: 45,7 mg/m³;
- Vynogradar: 18,8 mg/m³;
- Rusanovsky Sady: 35,4 mg/m³.

Results of measuring for PM₁₀:

- Pechersk-1: 48,9 mg/m³;
- Pechersk-2: 25,3 mg/m³;
- Shulyavka-1: 55,2 mg/m³;
- Shulyavka-2: 30,7 mg/m³;
- Vynogradar: 33,9 mg/m³;
- Rusanovsky Sady: 58,5 mg/m³.

Device DM72T (Fig.1) is a multifunctional air quality monitor that detects Particular Matter (PM_{2,5}/1.0/10).



Figure 1. – Device DM72T

The device in the conditions for Kyiv.

Література:

1. <https://www.saveecobot.com/ru/maps/kyiv>

*Липенков І.В., старший викладач,
кафедра інженерних дисциплін
Дунайського інституту Національного університету
«Одеська морська академія»*

АНАЛІЗ РОБОТИ СИСТЕМИ ВЕНТИЛЯЦІЇ ВАНТАЖНИХ ТРЮМІВ

Технічне кондиціонування забезпечує підтримку заданих умов (кондицій) атмосфери (з відносною вологістю менше 50%) наприклад в трюмах суховантажних суден, що сприяє збереженню вантажів і значному зменшенню корозії внутрішніх поверхонь трюмів і танків.

Всі вантажі, що перевозяться на суховантажних судах, можна підрозділити на дві основні категорії: гігроскопічні вантажі, що містять вологу, кількість якої залежить від умов довкілля і властивостей самого вантажу (пшениця, мука, бавовна, кава, тютюн, прянощі і т.і.), і негігроскопічні вантажі, що не містять вологу (метал і вироби з нього, машини, устаткування, інструмент і т. п.).

Вантажі володіють великою теплоакumuляційною здатністю; це наводить до випадання вологи з повітря при переходах судна з районів тропіків в північні широти і, навпаки, з холодних районів в теплі. Так, якщо судно переходить з північних районів в південні, то температура вантажу підвищується повільно, тобто він довгий час залишається холодним; з теплого вологого повітря трюму (у південних широтах) випадає волога на вантаж. Якщо судно переходить з районів тропіків в північні широти, то, завдяки теплоакumuляційної здатності вантажу, повітря трюму тривалий час залишається теплим і вологим, а температура внутрішніх частин корпусу судна знижується швидко (холодна забортна вода). Тому з повітря на ці поверхні випадатиме волога. Повітря в трюмах насичується парами води при випарі вологи з вантажів (наприклад, незрілих хлібних злаків, рису, соєвих бобів, бавовни, целюлози та ін.), а також унаслідок попадання снігу або дощу під час вантаження.

Випадання вологи на вантаж і металеві поверхні корпусу судна приводить до псування вантажу й посиленої корозії внутрішніх поверхонь трюмів. Так, по даним великої англійської компанії Атлантик Конвеєр, збитки від корозії внутрішніх поверхонь трюмів на двохстах судах через необхідність їхнього капітального ремонту вже після 7 ... 10 років експлуатації, а іноді й раніше, становлять щорічно близько 2 млн. фунтів стерлінгів.

Як показали проведені в багатьох країнах дослідження, найбільш ефективним засобом боротьби із псуванням вантажів і корозією металевих частин корпусу судна за зазначеними вище причинами є штучне регулювання точки роси повітря в трюмах і видалення з атмосфери трюмів вологи, тобто осушення повітря трюмів. Дослідження в Англії й Швеції показали, що при зниженні відносної вологості повітряного середовища в танках з 80 до 40...50% втрата у вазі стали від корозії зменшується більш ніж в 10... 12 разів, а при зниженні більш ніж на 50% корозія практично припиняється.

Основною частиною систем технічного кондиціонування повітря суховантажних і нафтоналивних судів служать повітряно-осушні установки. Крім того, в систему для суховантажного судна входять:

1. припливні і витяжні трюмні вентилятори, продуктивність яких, як правило, вибирають з розрахунку забезпечення 3-4-кратного обміну повітря в годину на порожній трюм;
2. повітроводи, забезпечуючи подачу і розподіл осушеного повітря в трюмах;
3. прилади і засоби автоматичного управління і контролю за установкою.

У систему технічного кондиціонування танкерів замість трюмних вентиляторів входить високо напірний нагнітач і трубопроводи осушеного

повітря з неповоротними клапанами, що забезпечують підтримку в танках надлишкового тиску повітряного середовища від 2000 до 8500 Па.

Завдяки застосуванню систем осушення повітря значно поліпшуються умови повітряного середовища в трюмах під час перевезення сипучих вантажів. Може підтримуватися мікроклімат, необхідний відповідно до санітарно-гігієнічних і технологічних вимог. Разом з тим потрібно уявляти собі можливості СОП, виходячи з конкретних умов. Технологія й СОП між собою нерозривно зв'язані. Організація осушення повітря в трюмах балкерів диктується в основному особливостями вантажу, що перевозиться, і умовами навколишнього середовища. Повністю вирішити проблему мікроклімату й ступеня осушення повітря в трюмі лише за рахунок удосконалювання СОП звичайно неможливо. Вона, як правило, може бути вирішена на основі здійснення комплексу заходів технічного й технологічного характеру. До числа останніх відносяться: герметизація технологічного обладнання з метою ліквідації або, у крайньому випадку, істотного зменшення шкідливих виділень; ізоляція поверхонь із високою температурою; застосування дистанційного керування й автоматики, що виключають безпосередній контакт працюючих з матеріалами; заміна відкритого транспорту матеріалів закритим тощо.

Практично всяка зміна технології (заміна устаткування, його модернізація, зміна продуктивності) повинне спричиняти відповідні зміни в СОП (застосування нового обладнання, установка двигунів іншої потужності, зміна схеми повітрярозподілення і т.п.).

СОП вимагають певних капітальних й експлуатаційних витрат. Найбільш дорогими, природно, є ефективні системи – з автоматичним регулюванням параметрів, що забезпечують глибоке осушення більших обсягів повітря. Але ці витрати в порівняно короткий строк (від декількох місяців до декількох років) окупаються, і досягається відчутний економічний ефект за рахунок зниження витрат на штрафні санкції за втрати вантажу або погіршення його якості.

При трансокеанських морських перевезеннях у результаті введення в дію СОП також може бути отриманий значний економічний ефект, про це свідчить наявний досвід, описаний у літературі. Під час перевезення зернових культур вищого сорту може бути досягнуте зниження витрат вантажу на 9... 14%, а під час перевезення тютюнових і бавовняних культур до 18%.

Перелік літератури:

1. Хордас Г. С. Технические системы кондиционирования воздуха и инертных газов на судах.- Л.: Судостроение, 1972.- 323 с.
2. Баркалов Б.В., Карпис Е.Е. Кондиционирование воздуха в промышленных, общественных и жилых зданиях. - М.: Стройиздат, 1982. - 213 с.
3. Богословский В.Н., Кокорин О.Я., Петров Л.В. Кондиционирование воздуха и холодоснабжение. - М.: Стройиздат, 1985.- 416 с.
4. Кокорин О.Я. Энергосберегающие технологии функционирования систем вентиляции, кондиционирования воздуха (системы ВОК).- М.: Проспект, 1999. - 208 с.

5. Глотов Ю.Г. и др. Безопасность жизнедеятельности человека на морских судах. Справочник. – М.: Транспорт, 1988.-220с.
6. Конопелько Г.И. и др. Охрана жизни на море. – М.: Транспорт, 1990-386с.
7. Козлитин А.М., Яковлев Б.Н. Чрезвычайные ситуации техногенного характера. Прогнозирование и оценка. Детерминированные методы количественной оценки опасностей техносферы: Учеб. пособие /Под ред. А.И. Попова. Саратов: Саратов. гос. техн. ун-т, 2000. 124с.
8. Меньшиков В.В., Швыряев А.А. Опасные химические объекты и техногенный риск: Учебное пособие. - М.: Изд-во Химия, фак. Моск. ун-та, 2003. - 254 с.
9. РД 31.11.81.37-82 Правила морской перевозки химических грузов наливом.

*Мартинюк Р.Т., к.т.н., доцент,
кафедра газонафтопроводів та газонафтоховищ,
Івано-Франківський національний технічний університет нафти і газу*

ОРГАНІЗАЦІЯ ВИГОТОВЛЕННЯ ТЕХНІЧНОЇ ДОКУМЕНТАЦІЇ ПРИ ВИКОНАННЯ ОКРЕМИХ ВИДІВ БУДІВЕЛЬНИХ РОБІТ

Типові технологічні карти виконання окремих видів будівельних, монтажних і спеціальних робіт, що використовують при прокладанні трубопроводів, розробляють з метою забезпечення рішень по організації і технології проведення будівельного процесу. Всі ці рішення сприяють підвищенню продуктивності праці, покращенню якості і зниженню собівартості робіт.

Технологічні карти, які розробляються з прив'язкою до конкретних умов будівництва, входять до складу проекту виконання робіт. Ця така прив'язка полягає [1]:

- в уточненні об'ємів робіт та засобів механізації;
- потреби в трудових і матеріально-технічних ресурсах;
- в обліку генерального плану і транспортної схеми будівництва трубопроводу.

Технологічні карти розробляють на основі передового досвіду управління організації будівництва трубопроводів, сучасного рівня техніки та технології. В технологічних картах передбачають:

- застосування технологічних будівельних і монтажних операцій, що забезпечують високий рівень продуктивності праці і якості виконання робіт;
- синхронна по об'ємах і в часі поставка труб, ізоляційних і інших матеріалів, конструкцій, виробів, деталей та пристосувань;
- виконання робіт з врахуванням технологічно допустимих зближень лінійних потоків;
- впровадження комплексної механізації з максимальним використанням машин та механізмів та широке застосування засобів малої механізації;

- застосування вузлів регулюючої та відключаючої арматури трубопроводу, що виготовлені в заводських або базових умовах;
- дотримання правил охорони праці і техніки безпеки;
- виконання заходів щодо охорони навколишнього середовища.

Матеріалами для розробки технологічних карт є: робочі креслення; інструкції по розробці проекту виконання робіт; єдині норми і розцінки на будівельні і монтажні роботи; відомчі і місцеві норми. Для будівництва лінійної частини трубопроводів типові технологічні карти на окремі види робіт часто розробляють в декількох варіантах (наприклад, для трубопроводів різного діаметру, для різних природно-кліматичних умов) [2].

Типову технологічну карту складають з розділів:

- 1 Область застосування;
- 2 Організація і технологія будівельного процесу;
- 3 Техніко-економічні показники;
- 4 Матеріально-технічні ресурси.

Для спорудження лінійної частини трубопроводу необхідно до типової технологічної карти ввести додаткові розділи:

- 5 Організація і методи праці;
- 6 Контроль якості виконання робіт;
- 7 Техніка безпеки при виконанні робіт.

При розробці типових технологічних необхідно надати дані:

- про спеціалізацію бригад, що споруджують переходи;
- про організації і виконавців робіт, їх виробничі підрозділи та границі ділянок роботи цих організацій;
- про директивний графік виконання робіт (виконують в масштабі).

Література:

- 1 Баталин Ю.П. Организация строительства магистральных трубопроводов / Баталин Ю.П., Березин В.Л., Телегин Л.Г. и др. – М. : Недра, 1980. – 344 с.
- 2 Організація спорудження та ремонту газонафтопроводів : конспект лекцій / Р. Т. Мартинюк. – Івано-Франківськ: ІФНТУНГ, 2019. – 70 с.

*Семенова А.М., кафедра управлінських технологій,
Університет економіки та права «КРОК», м. Київ*

ОРГАНІЗАЦІЯ ПРОВЕДЕННЯ ОЦІНКИ ВІДПОВІДНОСТІ ЗАСОБІВ ВИМІРЮВАЛЬНОЇ ТЕХНІКИ В ДП "УКРМЕТРТЕСТСТАНДАРТ"

Створення та функціонування єдиного світового ринку є неможливим без вільного транскордонного переміщення товарів і послуг. Виходу української продукції на європейські та міжнародні ринки, а також доступові вітчизняних споживачів до якісних закордонних продуктів заважає наявність у міжнародній торгівлі технічних бар'єрів. Механізми їх ліквідації у сфері торгівлі та надання послуг базуються на взаємному визнанні результатів оцінки відповідності в

результаті технічної гармонізації. Остання вирішальною мірою залежить від наявності в країні сучасної національної інфраструктури якості, що відповідає загально визнаним міжнародним нормам і правилам [1]. Наразі Україна — єдина країна серед країн СНД, яка здійснила перехід від застарілої системи державних випробувань засобів вимірювальної техніки (далі – ЗВТ) до оцінки відповідності вимогам технічних регламентів [2]. У 2016 році набули чинності нові редакції відповідних законодавчих актів України [3–5]. Відповідно до чинного законодавства суб'єкти господарювання, які реалізують засоби вимірювальної техніки, перш ніж виставити продукцію на ринок повинні перевірити її на наявність належного маркування, передбаченого технічними регламентами, та мати всю необхідну супровідну технічну документацію, зокрема й декларацію про відповідність. Саме тому об'єктом дослідження став процес підготовки процедур проведення оцінки відповідності ЗВТ в ДП «УКРМЕТРТЕСТСТАНДАРТ», що є однією з головних умов його виходу на ринок із конкурентоспроможними послугами. Предметом дослідження є теоретико-методичні та науково-практичні аспекти організації підготовки підприємства до проведення оцінки відповідності ЗВТ. Обрані об'єкт і предмет визначили мету дослідження: аналіз чинного порядку проведення оцінки відповідності, виявлення недосконалостей в її організації та розробка рекомендацій щодо удосконалення організаційних і технічних заходів під час підготовки підприємства до проведення ОВ ЗВТ. Відповідно до поставленої мети були визначені й вирішені такі завдання дослідження: проаналізовано методологічні принципи в галузі сертифікації ЗВТ в Україні; проаналізовано й узагальнено стан наявного порядку проведення оцінки відповідності ЗВТ; розроблено пропозиції щодо удосконалення організаційних, технічних і економічних заходів для проведення ОВ ЗВТ в ДП «УКРМЕТРТЕСТСТАНДАРТ». Теоретичною і методологічною базою дослідження слугували праці вітчизняних і зарубіжних вчених, практиків і фахівців, які останніми роками оприлюднили праці в області реформування системи технічного регулювання стандартизації, метрології і метрологічної діяльності. Зокрема, проблеми формування та переходу до нової системи висвітлювалися в роботах: О.С. Букреєва, О.М. Величко, О. Гіленко, Л.М. Віткіна, Ю.В. Кузьменко, І.В. Рибалко, С.В. Ціпоренко, С.В. Чередниченко, Марек Свобода, Вільям Мазур та інших. Інформаційною та правовою базою дослідження стали законодавчі акти Верховної Ради України, урядові рішення й постанови. На різних етапах дослідження використовувались відповідні методи та методологічні підходи: аналіз наукових праць вітчизняних фахівців, законодавчо-нормативних документів і матеріалів, дотичних до проблеми оцінки відповідності, дав змогу з'ясувати рівень наукового забезпечення процесу проведення ОВ ЗВТ; класифікація, систематизація даних уможливили структурне узагальнення теоретико-методологічних основ сертифікації; порівняльний аналіз і синтез, прогнозування дали змогу з'ясувати конкурентні переваги сертифікованої продукції; вивчення зарубіжного досвіду, побудова сценарію зробили можливим виявлення світових тенденцій розвитку системи сертифікації для запровадження їх в Україні. Отже, на першому етапі

дослідження було вивчено й проаналізовано стан теоретичного забезпечення в сфері сертифікації і стандартизації. Розроблена концепція дослідження, визначено об'єкт, предмет, сформульовано мету та завдання дослідження, опрацьовано методика організації дослідження, визначено програму проведення дослідницької роботи. Здійснено порівняльний аналіз теоретико-методологічних основ національної системи сертифікації України з чинною практикою та визначено її недоліки. На другому етапі дослідження здійснено аналіз організації проведення ОВ ЗВТ і лічильників води в ДП «УКРМЕТРТЕСТСТАНДАРТ» та визначено її недоліки. На третьому етапі дослідження узагальнено результати дослідження, сформульовано та обґрунтовано пропозиції та практичні рекомендації щодо адаптації порядку проведення ОВ до європейських стандартів якості. Доведено, що за рахунок покращення організації проведення оцінки відповідності ЗВТ та підвищення якості послуг (усіх основних і допоміжних технологічних та управлінських процесів) можна підвищити результативність функціонування, конкурентоспроможність і прибутковість ДП «УКРМЕТРТЕСТСТАНДАРТ». Підсумовуючи наведене вище, можна зазначити, що розроблення практичних рекомендацій з удосконалення процедур системи ОВ ЗВТ є ще одним кроком із гармонізації національної системи метрології і метрологічної діяльності, що сприятиме забезпеченню необхідного рівня захищеності кожного громадянина та суспільства у цілому від небезпечної продукції і наслідків недостовірних вимірювань, підвищенню конкурентоспроможності вітчизняних ЗВТ та усуненню технічних бар'єрів у торгівлі.

Література:

1. Кузьменко Ю. Оцінка відповідності засобів вимірювальної техніки — реалізація в Україні європейських підходів (принципів) / Кузьменко Ю., Черепков С., Дуля В. // Науково-виробничий журнал «Метрологія та прилади». — 2018. — № 1(69). — С. 11–16.
2. Віткін Л. Аналіз системи технічного регулювання, стандартизації, метрології в Україні та заходи щодо її удосконалення на 2018 рік / Віткін Л. // Науково-виробничий журнал «Метрологія та прилади». — 2018. — № 1(69). — С. 3–7.
3. Закон України «Про метрологію та метрологічну діяльність» від 05.06.2014 № 1314-VII. [Електронний ресурс]. Режим доступу: <https://zakon.rada.gov.ua/laws/show/1314-18#Text>
4. Технічний регламент засобів вимірювальної техніки, затверджений постановою Кабінету Міністрів України від 24.02.2016 № 163. [Електронний ресурс]. Режим доступу: <https://zakon.rada.gov.ua/laws/show/163-2016-%D0%BF#Text>
5. Технічний регламент законодавчо регульованих засобів вимірювальної техніки, затверджений постановою Кабінету Міністрів України від 13.01.2016 № 94. [Електронний ресурс]. Режим доступу: <https://zakon.rada.gov.ua/laws/show/94-2016-%D0%BF#Text>

*Теслюк Г.В., к.т.н.,
доцент, кафедра тракторів і сільськогосподарських машин, Дніпровський
державний аграрно-економічний університет, м. Дніпро;*

*Харченко В.О., магістрант,
кафедра тракторів і сільськогосподарських машин, Дніпровський державний
аграрно-економічний університет, м. Дніпро*

ОГЛЯД КОНСТРУКЦІЙ РОТАЦІЙНИХ БОРИН ДЛЯ ПОВЕРХНЕВОГО ОБРОБІТКУ ҐРУНТУ

Ротаційні борони призначені для та до післявсходового боронування посівів польових культур (зернових, пропашних, технічних), для поверхневого рихлення і аерації ґрунту, нищення нитєвидних сходів бур'яну.

Ротаційні борони відносять до безпривідних ґрунтообробних машин, які мають робочі органи різного конструктивного виконання: дискові ротори з пелюстковими, гольчастими, зубовими і ножевидними рихлячими елементами [1, 2,]

Найбільш поширеними робочими органами ротаційної борони є гольчасті колеса. Які в залежності від виробника мають різні кінцівки і діаметр. Конструктивно, колеса попарно об'єднані в секції і мають єдиний пружний стояк (рис.1.1).



Рисунок 1.1 – Секція ротаційної борони:

1- голчасте колесо; 2 – стояк; 3 – пружний елемент; 4 – ступиця.

Конструкція секції дозволяє її використовувати в складі агрегатів. Як суцільного, так і міжрядного обробітку.

Технологічний процес роботи ротаційної борони виглядає наступним чином. В процесі руху машини, голчасті колеса починають обертатись і заглиблюються у ґрунт на 3 – 5 см, в результаті чого руйнується ґрунтова кірка

і відбувається насичення поверхневого шару повітрям, в результаті азот з повітря потрапляє в ґрунт. Таким чином застосування ротаційної борони сприяє зменшенню потрібної кількості азотних добрив.

Оглядом конструкцій виявлені наступні варіанти виконання голчастих коліс: цільнолиті (рис.1.2.), штамповані (рис.1.3), Складні, або такі що мають декілька складових одиниць (рис.1.4).



Рисунок 1.2 – Варіант цільнолитого виконання колеса ротаційної борони (важка борона)



Рисунок 1.3 – Варіант штампованого виконання колеса ротаційної борони

Функціонально легкі ротаційні борони не занурюють у ґрунт на глибину більше за 2-3 см і тому вони повинні бути легкими. На рис.1.4. представлений варіант виконання диска такої борони. Диск має 5 складових одиниць, що безумовно є негативним фактором. Але голки можна легко міняти і в разі потреби збільшити чи зменшити їх довжину і пружність. Окрім того покращується показник ремонтпригодності.



Рисунок 1.4 – Колесо з декількох складових одиниць. (легка борона):
1 – голка; 2 – диск; 3 – елементи кріплення.

Ротаційні борони використовують як самостійний агрегат (рис.1.2), так і у складі комбінованого агрегату (рис.1.5). Представлена конструкція включає практично всі відомі ротаційні робочі органи, тому машина досить важка, але ротаційні робочі органи мають від’ємний кут входження у ґрунт і тому потребують примусового заглиблення. Це є однією з причин того, що рами таких машин роблять важкими. Підвищеної ваги також роблять і допоміжні робочі органи.



Рисунок 1.5 – Секція ротаційної борони в складі комбінованого ґрунтообробного агрегату:
1 – рама; 2 – колтер; 3 – секція ротаційної борони; 4 – секція турбодискова; 5 – долотоподібний глибокорозпушувач.

Таким чином, аналіз конструкцій показує, що знаряддя не зважаючи на окремі відмінності в конструктивних рішеннях в основному схожі і функціонально подібні.

Для ефективної роботи такої голки необхідно забезпечити підпірне різання, тобто для неї потрібен необроблений попередньо поверхневий шар. Аналітично обґрунтована глибина занурення до 3 см в грантових умовах Дніпропетровської області умовах

Література:

1. Скрипник В.І. Розробка, виробництво, конструктивні особливості нової сільськогосподарської техніки: навчальний посібник для здобувачів професійної освіти / В. І. Скрипник – К.: Літера ЛТД, 2019. – 256 с.
2. Войтюк Д. Г. Сільськогосподарські та меліоративні машини: підручник / Д. Г. Войтюк, В. О. Дубровін, Т. Д. Іщенко [та ін.]; за ред. Д. Г. Войтюка. – К.: Вища освіта, 2004. – 544 .

Зміст

Секція 1. Інформаційні системи і технології

Vash Y., Rol M. USING OF THE UTILITY SYSTEMS WITH BEHAVIOR TREES.....	3
Vash Y., Rol M. REVISITING ENVIRONMENT QUERY SYSTEM.....	6
Бичковський В.О., Реутська Ю.Ю. ФОРМАЛІЗАЦІЯ ЗАДАЧІ ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ ПРОЦЕСУ ПОШУКУ ОБ'ЄКТІВ.....	7
Бровіна О.І., Іосіпов Р.К. ВИКОРИСТАННЯ PyCairo У РОБОТІ З ВЕКТОРНИМИ ЗОБРАЖЕННЯМИ.....	9
Бухарова Л.Д., Гвоздецька К.П. ОГЛЯД НАЙПОПУЛЯРНІШИХ МЕТОДІВ ШИФРУВАННЯ.....	12
Бухарова Л.Д., Гвоздецька К.П. РОЗГЛЯД VPN ТА АНОНІМАЙЗЕРІВ В ЯКОСТІ ЗАСОБІВ БЕЗПЕЧНОГО ВИКОРИСТАННЯ НЕЗАХИЩЕНИХ МЕРЕЖ.....	13
Бухарова Л.Д., Гвоздецька К.П. КОНЦЕПЦІЯ ВІРТУАЛЬНОЇ ПРИВАТНОЇ МЕРЕЖІ В ПОРІВНЯННІ З МЕРЕЖАМИ КОНВЕРТОВОГО РАДІОРЕЛЕЙНОГО ЗВ'ЯЗКУ.....	15
Бухарова Л.Д., Гвоздецька К.П. ОСНОВИ ТУНЕЛЮВАННЯ, ЯК РЕАЛІЗАЦІЯ ТЕХНОЛОГІЇ ВІРТУАЛЬНИХ ПРИВАТНИХ МЕРЕЖ.....	16
Глюза А.П. МОСК-ОБ'ЄКТИ В ЮНІТ-ТЕСТУВАННІ.....	18
Глюза А.П. МУТАЦІЙНЕ ТЕСТУВАННЯ.....	19
Гончаренко Д.В. СЕРВІСИ AWS.....	21
Гончаренко Д.В. TYPES OF CLOUD SERVICES.....	22

Думчиков С.А., Лукічов В.В. ВИЯВЛЕННЯ ФІШИНГОВИХ ІНФОРМАЦІЙНИХ РЕСУРСІВ.....	24
Жакун Г.А., Лукічов В.В. СИСТЕМА ВИЯВЛЕННЯ АТАК З ВИКОРИСТАННЯМ HONEYROT.....	27
Кметь О.І. МУРАШИНИЙ АЛГОРИТМ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧ МАРШРУТИЗАЦІЇ ТРАНСПОРТУ.....	29
Колпак М.В. ПОВНОТЕКСТОВИЙ ПОШУК У ВІДСКАНОВАНИХ ДОКУМЕНТАХ В СИСТЕМАХ ЕЛЕКТРОННОГО ДОКУМЕНТООБІГУ.....	30
Красовська Є.В., Красовський О.Д. СУЧАСНІ ТЕНДЕНЦІЇ ЦИФРОВОЇ ТРАНСФОРМАЦІЇ БАЗ ДАНИХ.....	32
Крят Д.С. МОДЕЛЬ МОБІЛЬНОЇ СЕНСОРНОЇ МЕРЕЖІ ДЛЯ VANET.....	35
Лапшов Д.К. МОДЕЛЬ ДОСТАВКИ ДАНИХ З WEARABLE THINGS НА БАЗІ DRONE.....	37
Любіва А.В. СИСТЕМА ІДЕНТИФІКАЦІЇ ОБ'ЄКТІВ В INTERNET.....	39
Матвійчук Є.Є. РЕАЛІЗАЦІЯ ВЕБ-ОРІЄНТОВАНОГО ПРОГРАМНОГО РІШЕННЯ ДЛЯ НАВЧАННЯ СТУДЕНТІВ ПРОГРАМУВАННЮ.....	40
Нефедченко О.О. МОДЕЛЬ ІНФОРМАЦІЙНОЇ ВЗАЄМОДІЇ В МЕРЕЖАХ ІоТ.....	42
Осадчий О.О., Гвоздецька К.П. ВАЖЛИВІСТЬ ВИКОРИСТАННЯ ПРИВАТНИХ МЕРЕЖ ДЛЯ ІНТЕРНЕТУ РЕЧЕЙ.....	44
Потьомкіна К.О., Гвоздецька К.П. ПЕРЕВАГИ VPN МЕРЕЖ ДЛЯ ОСОБИСТОГО ВИКОРИСТАННЯ.....	46
Потьомкіна К.О., Гвоздецька К.П. ПЕРЕВАГИ VPN МЕРЕЖ ДЛЯ ОСОБИСТОГО ВИКОРИСТАННЯ.....	47

Репалюк О.І. ПРОГНОЗУВАННЯ РЕЗУЛЬТАТІВ СПОРТИВНИХ ЗМАГАНЬ НА ОСНОВІ СТАТИСТИЧНИХ МОДЕЛЕЙ.....	48
Рибальченко О.Г., Заїка Б.В. НЕЙРОМЕРЕЖЕВЕ ВИЯВЛЕННЯ DDOS-АТАК.....	50
Ротань Д.В. МОДЕЛЬ ВЗАЄМОДІЇ D2D В БЕЗДРотовИХ МЕРЕЖАХ.....	53
Савелов Є.Є. РЕАЛІЗАЦІЯ ВЕБ-ОРІЄНТОВАНОГО ПРОГРАМНОГО РІШЕННЯ ДЛЯ ЗДІЙСНЕННЯ ОПИТУВАНЬ В ОНЛАЙН РЕЖИМІ.....	55
Сагайдачний О.М., Гвоздецька К.П. ПРОБЛЕМИ БЕЗПЕКИ В БЕЗДРотовИХ СЕНСОРНИХ МЕРЕЖАХ.....	57
Снайчук Я.Л. РЕАЛІЗАЦІЯ ВЕБ-СИСТЕМИ ДЛЯ ЗАПИСУ ВІЗИТІВ У МЕДИЧНИЙ ЗАКЛАД.....	59
Соботник Е.Л., Юрчишин В.М. ОСОБЛИВОСТІ БАГАТОПОТОКОВОЇ ОБРОБКИ ДАНИХ У ДОДАТКАХ, ПОБУДОВАНИХ НА БАЗІ NODE.JS.....	61
Сущенко І.В. СИСТЕМА ІДЕНТИФІКАЦІЇ НА БАЗІ АРХІТЕКТУРИ ЦИФРОВИХ ОБ'ЄКТІВ.....	64
Танчинець В.В., Журавчак Л.М., Шинкаренко Г.А. h-АДАПТИВНИЙ МЕТОД СКІНЧЕННИХ ЕЛЕМЕНТІВ З КУСКОВО- КУБІЧНИМИ АПРОКСИМАЦІЯМИ НА ТРИКУТНИКАХ.....	66
Тарасов М.С., Гвоздецька К.П. ОГЛЯД НОВИХ СТАНДАРТИВ, ЩО ВИКОРИСТОВУЮТЬСЯ В INTERNET OF THINGS.....	70
Філіп'єва М.В., Гвоздецька К.П. ПОРІВНЯННЯ СИМЕТРИЧНОГО І АСИМЕТРИЧНОГО ШИФРУВАННЯ.....	71

Секція 2. Економічні науки

- Боднар О.В.**
УПРАВЛІНСЬКИЙ АНАЛІЗ ТА ЙОГО РОЛЬ В ДІЯЛЬНОСТІ
ПІДПРИЄМСТВА.....74
- Левчук О.В., Онофрійчук О.А.**
ВИЗНАЧЕННЯ МЕХАНІЗМУ РОЗПОДІЛУ ФІНАНСОВИХ ОБОРОННИХ
РЕСУРСІВ ДЛЯ ЕФЕКТИВНОГО РОЗВИТКУ СПРОМОЖНОСТЕЙ СИЛ
ОБОРОНИ УКРАЇНИ.....75
- Мямлін В.В.**
ДО ПИТАННЯ ПРО ВРЕГУЛЮВАННЯ РІВНОВАГИ МІЖ ЗАГАЛЬНОЮ
ВАРТІСТЮ ТОВАРІВ І КІЛЬКІСТЮ ГРОШЕЙ, ПРИЗНАЧЕНИХ ДЛЯ ЇХ
СПОЖИВАННЯ.....77
- Сливінська О.Б.**
РОЛЬ БІЗНЕС-ДІАГНОСТИКИ В УПРАВЛІННІ ПІДПРИЄМСТВОМ.....82

Секція 3. Технічні науки

- Bozhko K.M., Morozova I.V., Kirilenko E.A.**
AIR QUALITY MONITOR DM72T IN MEASUREMENTS OF PM_{2,5} AND
PM₁₀ DUST CONCENTRATION.....84
- Липенков І.В.**
АНАЛІЗ РОБОТИ СИСТЕМИ ВЕНТИЛЯЦІЇ ВАНТАЖНИХ ТРІУМІВ.....85
- Мартинюк Р.Т.**
ОРГАНІЗАЦІЯ ВИГОТОВЛЕННЯ ТЕХНІЧНОЇ ДОКУМЕНТАЦІЇ ПРИ
ВИКОНАННЯ ОКРЕМИХ ВИДІВ БУДІВЕЛЬНИХ РОБІТ.....88
- Семенова А.М.**
ОРГАНІЗАЦІЯ ПРОВЕДЕННЯ ОЦІНКИ ВІДПОВІДНОСТІ ЗАСОБІВ
ВИМІРЮВАЛЬНОЇ ТЕХНІКИ В ДП "УКРМЕТРТЕСТСТАНДАРТ".....89
- Теслюк Г.В., Харченко В.О.**
ОГЛЯД КОНСТРУКЦІЙ РОТАЦІЙНИХ БОРІН ДЛЯ ПОВЕРХНЕВОГО
ОБРОБІТКУ ҐРУНТУ.....92

Підписано до друку 29.10.2021
Формат 60x84/16. Папір офсетний. Друк на дублікаторі.
Умов.-друк. арк. 4,5. Обл.-вид. Арк 4,95.
Тираж 80 прим.

Віддруковано ФО-П Шпак В.Б.
Свідоцтво про державну реєстрацію № 073743
СПП № 465644
Тел. 097 299 38 99
E-mail: tooums@ukr.net