МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ
УНІВЕРСИТЕТ ІМЕНІ СЕМЕНА КУЗНЕЦЯ

КАФЕДРА КІБЕРБЕЗПЕКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

МАТЕРІАЛИ

XV-ої МІЖНАРОДНОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ
«FREE AND OPEN SOURCE SOFTWARE»

Дякуємо за підтримку

13-14 лютого 2024 р.
м. Харків

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

КАФЕДРА КІБЕРБЕЗПЕКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

МАТЕРІАЛИ

XV-ої МІЖНАРОДНОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ

«FREE AND OPEN SOURCE SOFTWARE»

13-14 лютого 2024 р.

ХАРКІВ 2024

УДК 004
БК 32.973.202

Матеріали XV-ої Міжнародної науково-практичної конференції «Free and Open Source Software», Харків, 13-14 лютого 2024 р. – Харків: Харківський національний економічний університет імені Семена Кузнеця, 2024. – 148 с.

Представлено матеріали пленарних та секційних засідань XV-ої Міжнародної науково-практичної конференції «Free and Open Source Software». Обговорено основні проблеми, науково-технічні досягнення, впровадження і досвід використання сучасних технологій в області безкоштовних програмних продуктів, а також з відкритим вихідним кодом. Висвітлено основні питання безкоштовного прикладного, серверного програмного забезпечення та прикладного програмного забезпечення з відкритим вихідним кодом, безкоштовних сервісів, в тому числі в контексті кібербезпеки, ліцензування та правові аспекти використання безкоштовного програмного забезпечення. Для фахівців науково-дослідних, комерційних організацій, аспірантів та студентів.

Відповідальний за випуск:
Старкова О.В.

Роботи надруковані з авторських оригіналів, що надані оргкомітету, за авторської редакції.

Електронний варіант матеріалів конференції доступний на сайті конференції:

https://foss.kn-it.info/

ЗМІСТ

СЕКЦІЯ 1

СЕКЦІЯ 2

СЕКЦІЯ 3

Секція 1

USE FREE SOFTWARE ON MOBILE PHONES TO DIAGNOSE ANDROID MOBA GAME LATENCY ISSUES UNDER 5G NETWORKS

Gao Shuzhi
Supervisor: Natalya Dolgova
E-mail: 535655854@qq.com
Kharkiv, Simon Kuznets Kharkiv National University of Economics

With the continuous updating of smartphone products and the upgrade of hardware configurations, mobile 4G networks are developing rapidly, and 5G applications are becoming more abundant and diverse. According to Internet industry statistics, China's game industry revenue will exceed 430 billion in 2023, with mobile games accounting for 65%. The main factors that affect game perception can be subdivided into terminals, wireless networks, transmission, core networks, and servers. Each sub-item needs targeted performance improvements to ultimately improve game players' perception [1].

The game client and server implement delay detection, link keepalive player synchronization, and interaction through three data streams respectively [2].

1) The UE initiates a UDP message to the server for link delay evaluation. The message contains the Start field, the period is 5s, and the payload is 60 Bytes; the server response message contains the Stop field, and the payload is 58 Bytes.
2) TCP heartbeat detection flow: UE initiates TCP packets to the server with a cycle of 3s and keeps the link active.
3) Player synchronization interaction flow: Realize status synchronization and information interaction between players through UDP flow, pursuing low latency

Through research on the MOBA game delay business mechanism, we decompose the game process and simulate some simple requirements, so that some non-professionals can also diagnose simple problems.

- Wireless networks: We can use some free signal detection software to evaluate the quality of wireless signals, such as cellular-z Pro, Network Signal Guru, and other free detection software.
- Transmission: We can use some packet capture software to obtain the game interaction server. According to different types of game interaction (UDP/TCP), some commonly used software, Packet Capture, Reqable and other software
- Server latency detection software: Let’s use some simple network analysis software, such as PingTools, to evaluate the latency from the server to the mobile application.
- Mobile phone evaluation software: Simultaneously evaluate some mobile phone performance, heating simulation, and other computing testing software to evaluate the performance of the game.

While modern trends and tools make mobile end-to-end game latency optimization more feasible and convenient, some technical understanding and practical experience are still required. For more complex problems and in-depth optimization, the involvement of specialized personnel may still be necessary. Therefore, continuing to enhance the knowledge and skills of individuals is crucial to better address the challenges of mobile end-to-end game latency optimization.

References

[1] Razie Roostaei; Zahra Dabiri; Zeinab Movahedi: A game-theoretic joint optimal pricing and resource allocation for Mobile Edge Computing in NOMA-based 5G networks and beyond// Computer Networks, ISSN: 1389-1286, Vol: 198, Page: 108352 // AccessMode: https://www.sciencedirect.com/science/article/abs/pii/S138912862100342X

[2] Junchao Yang; Ali Kashif Bashir; Zhiwei Guo; Keping Yu; Mohsen Guizani: Intelligent cache and buffer optimization for mobile VR adaptive transmission in 5G edge computing networks//Digital Communications and Networks, ISSN: 2352-8648 // AccessMode: https://www.sciencedirect.com/science/article/pii/S2352864823001190

FREE TOOLS FOR DEVOPS

Zhuravka A.V., Akulynichev A.A., Ivanov A.
E-mail: andrii.zhuravka@nure.ua
Kharkiv, Kharkiv National University of Radioelectronics

DevOps, which combines development and operations, is a cultural approach aimed at improving collaboration and automating processes between development and IT operations teams. This enables organizations to deliver high-quality software faster and more efficiently. In this context, free DevOps tools play a crucial role.

There are numerous free and open-source tools available in the DevOps ecosystem that help teams enhance collaboration, automate processes, and improve overall software development and IT operations efficiency. Some of them include Git for version control, Jenkins for continuous integration/continuous delivery (CI/CD), Ansible for configuration management, Docker for containerization, Kubernetes for container orchestration, Prometheus for monitoring, ELK Stack (Elasticsearch, Logstash, Kibana) for log management, Terraform for infrastructure as code, and Slack for collaboration and communication.

Free DevOps tools offer several advantages, including cost reduction, flexibility, availability, and a high degree of control and customization. They also contribute to improving code quality, accelerating the development process, and increasing team productivity.

Like any other methodology or approach, DevOps faces several challenges that can hinder its successful implementation. These challenges may arise for various reasons, such as resistance to change, unclear goals, lack of skills, tool and infrastructure limitations, and organizational silos.

The growth, innovation, and adoption of DevOps are expected to increase in the future. We will witness teams mastering new technologies to meet business priorities and revenue goals. In this context, free DevOps tools will continue to play a significant role.

Git: provides powerful version control capabilities, allowing developers to work in parallel by creating branches for developing specific features and then safely merging these changes; ensures reliability and security by preserving the complete history of code changes, enabling tracking and fixing of errors; supports distributed development, simplifying the scaling of engineering teams.

Jenkins: automates the process of building, testing, and deploying applications, speeding up development and enhancing software delivery reliability; offers an extensive ecosystem of plugins that extend its functionality, making it highly customizable and adaptable to various project requirements; supports continuous integration and continuous delivery (CI/CD), allowing quick and efficient responses to code changes.

Ansible: enables automation of tasks related to configuration management and application deployment, simplifying infrastructure management; uses a user-friendly YAML configuration definition language, making it easy to write and understand automation scripts; supports a wide range of platforms and applications, making it a versatile tool for automation.

Docker: allows the creation and launch of applications in isolated containers, ensuring consistency of the environment throughout development and deployment stages; facilitates the deployment process by enabling developers to package applications along with their dependencies into standardized units; supports a microservices architecture, allowing the development and scaling of services independently.

Kubernetes: provides automated deployment, scaling, and management of containerized applications; supports service discovery and load balancing, ensuring high availability and efficient resource utilization; enables automatic recovery of applications and nodes, enhancing system reliability.

Prometheus: provides powerful capabilities for monitoring systems and alerting about errors; collects and stores metrics as time series, allowing tracking changes in the system state over time; supports a flexible query language (PromQL) for extracting and processing real-time metrics.

Terraform: allows automation of infrastructure creation, modification, and management using code; supports a wide range of cloud providers, enabling infrastructure management in various cloud environments; uses a declarative configuration definition language, making it easy to write and understand automation scripts.

КРИТЕРІЇ ВИБОРУ DLP СИСТЕМИ БЕЗПЕКИ КОРПОРАТИВНОЇ МЕРЕЖІ

Бойко В.І.
Керівник: Долгова Н.Г.
E-mail: wrepleys91@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Система запобігання витоку інформації (DLP - Data Loss Prevention) є спеціалізованим програмним забезпеченням, створеним для захисту компанії від незаконного витоку конфіденційної інформації.

DLP-система включає в себе низку методів та інструментів для виявлення та запобігання розголошенню даних, які можуть бути використані зловмисниками або особами, які недбало розголошують важливу інформацію, використовуючи свої або вкрадені облікові дані.

Залежно від місця розташування у мережі компанії, DLP-системи можуть бути:

1) хостові
2) мережеві
3) змішані

Існують як платні, так і безкоштовні системи DLP (Data Loss Prevention). Вибір між ними залежить від конкретних потреб компанії, обсягу і важливості даних, бюджету, доступності технічної підтримки та інших факторів.

Можна виділити наступні переваги платних систем DLP:

1. Використання розширеного функціоналу: платні системи DLP часто мають більший набір функцій і можливостей.
2. Своєчасна професійна підтримка: користувачі отримують доступ до професійної технічної підтримки у разі виникнення проблем.
3. Гарантована більш висока надійність і безпека: платні системи часто пропонують більш високий рівень захисту даних і безпеки.

Ці системи можуть бути розташовані на клієнтських пристроях (хостах), на мережевому обладнанні (зазвичай на серверах) або мати розподілені програмні інструменти, які функціонують як на серверах, так і на клієнтських пристроях. ПЗ для запобігання витоку даних завжди враховує розмір та архітектуру мережі, структуру та обсяг даних. Це визначає набір технологій та сценаріїв, які реалізовуються в цих системах.

Розпізнавання витоків даних може бути ефективним для типових документів (наприклад, номерів телефонів і адрес), але також і для великих обсягів інформації.

Щодо реагування на факти витоку інформації, DLP-системи можна поділити на активні та пасивні за сценаріями. Активні системи вживають заходів для блокування передачі даних згідно з програмним алгоритмом. У випадку пасивних систем функції контролю передачі даних делегуються службі безпеки або до спеціаліста з інформаційної безпеки. Це може впливати на продуктивність та ефективність DLP-систем у корпораціях різних розмірів.

Звісно, жодна DLP-система не може гарантувати абсолютний захист від наслідків діяльності працівників. Проте такі системи дозволяють суттєво зменшити ризики та наслідки людських помилок, а також гарантують дотримання правил щодо захисту конфіденційної інформації.

Література

[1] Застосування DLP-систем [Электроний ресурс]. – Режим доступа до ресурсу: https://techexpert.ua

[2] Системи запобігання витоку даних [Электроний ресурс]. – Режим доступа до ресурсу: http://allta.com.ua

АНАЛІЗ РИЗИКІВ ТА РОЗРОБКА ЗАХОДІВ БЕЗПЕКИ В МЕРЕЖАХ 5G

Воронцов І.О.
Керівник: Мерлак О.В.
E-mail: vorontsovilia69@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Мережі 5G знаменують собою революцію в телекомунікаціях, пропонуючи безпрецедентні можливості для розвитку та інновацій. Ця нова технологія обіцяє значно більшу швидкість, кращу продуктивність, ширшу conectivitu та меншу затримку, що може призвести до революційних змін у багатьох галузях, таких як:

– промисловість: автоматизація та роботизація завдяки 5G стануть доступнішими та ефективнішими, що призведе до значного зростання продуктивності;

– транспорт: 5G може зробити транспорт більш безпечним, економним та екологічно чистим, завдяки розвитку автономних транспортних засобів та систем «розумних міст»;

– медицина: 5G може революціонізувати телемедицину, дозволяючи лікарям діагностувати та лікувати пацієнтів на відстані, а також розширити доступ до якісної медичної допомоги;

– освіта: 5G може зробити освіту більш доступною та інтерактивною, дозволяючи учням навчатися в будь-якому місці та в будь-який час;

– розваги: 5G може революціонізувати індустрію розваг, пропонуючи нові й захоплюючі способи споживати контент.[1]

Однак, окрім цих вражаючих можливостей, 5G також несе в собі нові ризики та проблеми безпеки, які потребують ретельного аналізу та рішучих заходів.

Безпека мереж 5G є надзвичайно важливою, адже вони будуть використовуватися для критичної інфраструктури, таких як енергетичні мережі, транспортні системи та медичні заклади.

В представленій роботі досліджуються ризики та проблеми безпеки, пов'язані з мережами 5G, та пропонуються заходи для їхнього пом'якшення.

Більш ретельного аналізу потребують наступні ризики:

– атаки на доступ: 5G використовує нові протоколи та технології, які можуть бути вразливими до атак;

– атаки на конфіденційність: 5G може збирати та обробляти велику кількість даних, що може призвести до проблем з конфіденційністю;

– атаки на цілісність: 5G використовується для критичної інфраструктури, тому атаки на цілісність мережі можуть мати серйозні наслідки;

– атаки на доступність: 5G має забезпечувати безперебійну роботу, тому атаки на доступність мережі можуть мати значний вплив.

Для запобігання деструктивному впливу ризиків, пропонуються наступні заходи безпеки: аутентифікація та авторизація: сильні методи аутентифікації та авторизації; шифрування даних; сегментація мережі: розбиття мережі на сегменти; моніторинг та аналітика: постійний моніторинг мережі; регулярне оновлення програмного забезпечення.[2]

Безпека мереж 5G є надзвичайно важливою, адже від неї залежить не лише безперебійна робота критичної інфраструктури, але й добробут та безпека людей. Це може бути досягнуто завдяки вживанню комплексного та багаторівневого підходу до кібербезпеки, який включає в себе: сильну аутентифікацію та авторизацію; шифрування даних; сегментацію мережі; моніторинг та аналітику.

Регулярне оновлення програмного забезпечення можна мінімізувати ризики та забезпечити безпечне та надійне використання мереж 5G. Важливо зазначити, що кібербезпека – це не одноразова дія, а постійний процес, який потребує постійного вдосконалення та адаптації до нових викликів.[1]

Література

[1] Network Security Solutions | NordLayer. Network Access & Security Solutions | NordLayer. URL: https://nordlayer.com/network-security/?gclid=CjwKCAiAt5euBhB9EiwAdkXWO813uNqpvR9iYXvt-S4CBAOrVfQXPsifeoctx4e57ZLklNRmsOXGfxoCc8QQAvD_BwE.

[2] Кейси – smartpower. smartpower. URL: https://www.smartpower.com.ua/kejsy/

ДОСЛІДЖЕННЯ ЗАГРОЗ ТА РОЗРОБКА МЕТОДІВ ЗАХИСТУ ДОМАШНІХ МЕРЕЖ З ФОКУСОМ НА БЕЗПЕКУ WI-FI З'ЄДНАНЬ

Грезнєва М.В

Керівник : Лимаренко В.В.

E-mail: masha.grezneva.03@gmail.com

Харків, Харківський національний економічний університет імені Семена Кузнеця

В інформаційному суспільстві, яке переплетено з цифрових технологій, розуміння та вивчення загроз, пов'язаних з використанням мережевих технологій в домашніх умовах, стає актуальним завданням. Однією з ключових областей цього дослідження є безпека домашніх мереж, з особливим акцентом на захист від потенційних загроз, пов'язаних із бездротовими з'єднаннями Wi-Fi.

Сьогодні бездротовий доступ до Інтернету став невід'ємною частиною нашого повсякденного життя, проте ця технологія несе ризики, які важливо розуміти. Велика кількість домашніх мереж стають вразливими перед різноманітними загрозами, такими як несанкціонований доступ, крадіжка інформації, віруси та інші форми кіберзлочинності.

Існує кілька ефективних інструментів та методів захисту домашніх мереж з акцентом на безпеку Wi-Fi з'єднань:

1. Шифрування Wi-Fi: WPA3 (Wi-Fi Protected Access 3): Стандарт безпеки Wi-Fi, який надає більш потужні та безпечні методи шифрування для захисту бездротових з'єднань.
2. Зміна паролю: Регулярна зміна паролю для Wi-Fi мережі допомагає уникнути несанкціонованого доступу. Міцний та унікальний пароль грає ключову роль у захисті мережі.
3. Firewall (Брандмауер): Встановлення брандмауера на роутер або на самі пристрої в мережі може допомогти блокувати небажані підключення та захищати від потенційних загроз.

Об'єктом дослідження є потенційні загрози у домашніх мережах та розробка ефективних методів захисту, з основним фокусом на забезпеченні безпеки бездротових з'єднань Wi-Fi. Основна мета дослідження - розкрити сутність ризиків та вивчити можливі наслідки вразливостей домашніх мереж, а також запропонувати конкретні рішення для підвищення рівня безпеки.

В рамках дослідження планується провести аналіз сучасних тенденцій в кібербезпеці, вивчити вразливості, що виявлені в домашніх мережах, і розглянути існуючі або потенційні методи захисту для мінімізації ризиків. Значущим елементом буде також розгляд соціальних та етичних аспектів безпеки мереж у домашньому оточенні.

Література

[1] Вікіпедія "Wi-Fi" [Електронний ресурс]. – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Wi-Fi

[2] Вікіпедія "WPA (Wi-Fi Protected Access)" [Електронний ресурс]. – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Wi-Fi_Protected_Access

[3] Вікіпедія "Firewall (Брандмауер): " [Електронний ресурс]. – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Мережевий_екран

[4] Вікіпедія "Cybersecurity" [Електронний ресурс]. – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Cybersecurity

АНАЛІЗ ЗАСОБІВ ВИЯВЛЕННЯ ТА ПОПЕРЕДЖЕННЯ DDOS АТАК У КІБЕРПРОСТОРІ

Дудкін В.М.
Керівник: Шаповалова О.О.
E-mail: vovadedkov0@gmail.com, olena.shapovalova@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

В умовах війни, зокрема, в кіберпросторі, захист інформації від несанкціонованого доступу та відмови в обслуговуванні є вкрай важливим, а аналіз засобів попередження та виявлення DDoS атак не втрачає своєї актуальності вже багато років і є важливою частиною інформаційної безпеки. До ознак сьогоденності в цієї сфері можна віднести наступні:

– зростання останнім часом кількості та складності DDoS атак, які стають більш вишуканими та різноманітними, зокрема атаки на різних рівнях протоколів, аплікаційні атаки та атаки, спрямовані на використання різних типів трафіку;

– зростання важливості онлайн-присутності, що пов’язано із вагомим збільшенням кількості бізнес-операцій та взаємодій в онлайн-середовищі. Великі компанії, сервіси, урядові та фінансові установи приділяють багато уваги заходам кіберзахисту;

– використання DDoS атак для відволікання уваги на механізми відмови в обслуговуванні, щоб приховати інші, більш складні атаки або спроби проникнення;

– необхідність постійного вдосконалення заходів кібербезпеки, при чому аналіз і підвищення ефективності існуючих засобів попередження та виявлення DDoS атак є ключовим елементом стратегії кіберзахисту;

– зростання ризиків для бізнесу та інфраструктури за рахунок DDoS атак, які можуть погіршити репутацію компанії, призвести до значних фінансових втрат та відмови в обслуговуванні, що може бути особливо критичним для бізнес-процесів та інфраструктури.

Отже, проведення дослідження в сфері аналізу та вдосконалення засобів попередження та виявлення DDoS атак має велике значення в контексті забезпечення стійкості та безпеки сучасних інформаційних систем та мереж. В межах досліждення планується оцінити існуючі засоби попередження та виявлення DDoS атак у кіберпросторі з метою визначення їхньої ефективності з подальшим вдосконаленням заходів кібербезпеки. Робота передбачає аналіз технічних аспектів виявлення та протидії DDoS атакам, зокрема з використанням мережевих інструментів, систем виявлення вторгрень (intrusions) та інших технологій. На основі отриманих результатів буде розроблено рекомендації щодо вдосконалення заходів кіберзахисту та вибору оптимальних інструментів для конкретного середовища.

В ході роботи планується розглянути такі ключові аспекти як:

– аналіз існуючих методів виявлення DDoS атак, зоерема дослідити різноманітні техніки виявлення DDoS атак, включаючи сигнатурний аналіз, аналіз аномалій та глибинне навчання;

– ефективність інструментів попередження та виявлення, зокрема провести експериментальне тестування існуючих засобів кіберзахисту для оцінки їхньої ефективності в умовах сучасних DDoS атак;

– розробка рекомендацій та вдосконалення систем безпеки, в ході чого на основі отриманих результатів запропонувати конкретні заходи щодо вдосконалення систем безпеки та вибору оптимальних інструментів для запобігання та виявлення DDoS атак;

– врахування специфіки конкретних середовищ, в ході чого визначити особливості та вимоги конкретного середовища (мережі, веб-сайту, додатку) для адаптації рекомендацій та інструментів до конкретних потреб користувача.

Безкоштовні інструменти, які планується використати в роботі:

– Wireshark [1] є інструментом аналізу мережевого трафіку, який дозволяє перехоплювати та аналізувати пакети даних в реальному часі (рис.1). Використання Wireshark є корисним для вивчення пакетів, що входять у мережу та виходять з неї, для виявлення незвичайного трафіку, який може бути зв'язаний з DDoS атакою;

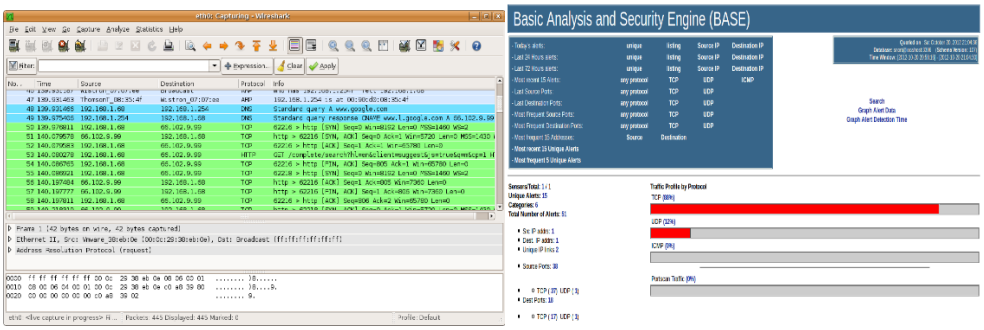


Рисунок 1. Вікна роботи Wireshark та Snort

– Snort [2] є системою виявлення вторгнень (IDS) та може слугувати для виявлення аномального та зловмисного трафіку в реальному часі (рис.1). Налаштувати Snort можливо для моніторингу мережі та виявлення атак, що можуть бути пов'язані з DDoS;

– OpenVAS (Open Vulnerability Assessment System) [3] є сканером вразливостей, який дозволяє ідентифікувати потенційно слабкі місця в системі безпеки (рис. 2). Також OpenVAS застосовується для сканування мережі та серверів на предмет вразливостей, які можуть бути використані в DDoS атаках;

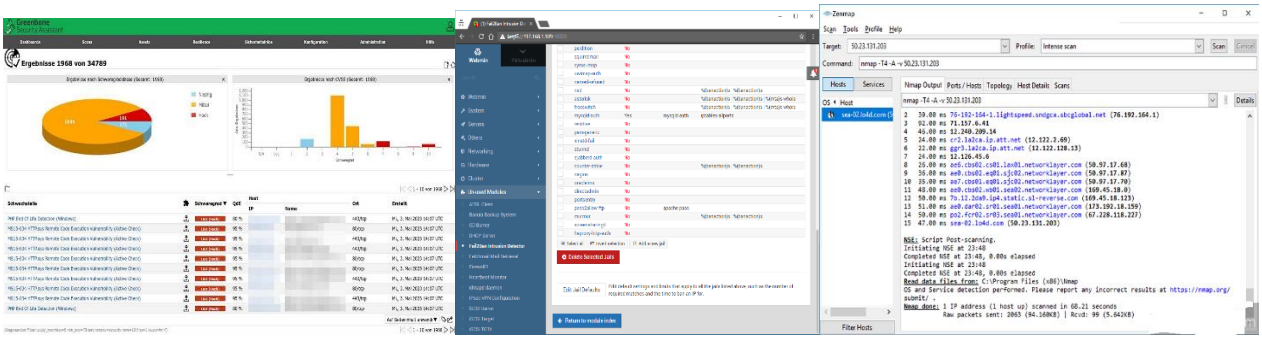


Рисунок 2. Вікна роботи OpenVAS, Fail2Ban та Nmap

– Fail2Ban є програмним засобом запобігання атакам на сервери шляхом автоматичного блокування IP-адрес, які здійснюють надто багато невдалих спроб авторизації (рис. 2). Встановлення Fail2Ban сприятиме виявленню та блокуванню IP-адрес, які можуть бути пов'язаними з DDoS атаками;

– Nmap [4] є інструментом сканування мережі, який може виявляти активні хости та служби в мережі (рис. 2). Використання Nmap сприятиме дослідженню мережі та виявленню надлишкових обсягів трафіку, що може вказувати на DDoS атаку.

Литература

[1] Wireshark, сайт [Електронний ресурс]. – Режим доступу до ресурсу: https://www.wireshark.org/

[2] Snort, сайт [Електронний ресурс]. – Режим доступу до ресурсу: https://www.snort.org/

[3] OpenVas сайт [Електронний ресурс]. – Режим доступу до ресурсу: https://www.openvas.org/

[4] Nmap сайт [Електронний ресурс]. – Режим доступу до ресурсу: https://nmap.org/

ПРОГРАМНІ ЗАСОБИ ДЛЯ АВТОМАТИЗОВАНОЇ ПЕРЕВІРКИ ЦІЛІСНОСТІ ФАЙЛІВ

Єфімов М.Ю.
Керівник: Венгріна О.С.
E-mail: mishaefimov2002@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

У цифрову епоху, що стрімко розвивається, зростаючий обсяг цифрових даних підкреслює першорядну важливість забезпечення цілісності та безпеки файлів. Оскільки організації орієнтуються в складних цифрових екосистемах, потреба в автоматизованих системах, які можуть ефективно і надійно перевіряти цілісність файлів, стала критичною вимогою. У даному досліджені розглядається сучасна актуальність автоматизованої перевірки цілісності файлів, зокрема за допомогою хеш-функцій і цифрових підписів, у вирішенні проблем кібербезпеки та підтримці цілісності цифрових активів.

Крім того, сувора нормативна база, що регулює захист даних і приватності, вимагає надійних механізмів перевірки цілісності файлів. Дотримання таких нормативних актів, як GDPR, HIPAA та інших, зобов'язує організації впроваджувати ефективні заходи для забезпечення цілісності та автентичності конфіденційної інформації. Невиконання цих вимог не лише наражає організації на юридичні наслідки, але й підриває довіру до їхньої здатності відповідально поводитися з конфіденційними даними.

В умовах динамічної зміни загроз кібербезпеки, різні безкоштовні програмні рішення вимальовуються як цінний інструмент для автоматизованої перевірки цілісності файлів. Ці інструменти використовують передові технології, в тому числі хеш-функції та цифрові підписи, щоб зміцнити безпеку організацій, не накладаючи на них фінансового тягаря.

Детально розглянемо програмне забезпечення, таке як HashCheck, md5deep, GnuPG та OpenSSL, їхню роль у забезпеченні безпеки цифрових активів та методи, які вони використовують для перевірки цілісності файлів:

HashCheck - легкий інструмент для Windows, який легко інтегрується в операційну систему, пропонуючи надійну підтримку різноманітних алгоритмів хешування. Його інтуїтивно зрозумілий інтерфейс спрощує розрахунок контрольної суми, надаючи користувачам ефективний засіб перевірки цілісності файлів. Незважаючи на свою дружню природу, HashCheck забезпечує високий рівень безпеки, що робить його ідеальним вибором для користувачів Windows, які шукають баланс між простотою і функціональністю [1].

md5deep - кросплатформенна утиліта командного рядка, md5deep відмінно справляється з глибоким хешуванням і перевіркою цілісності файлів. Її перевага полягає у полегшенні рекурсивного хешування, що дозволяє користувачам перевіряти цілісність цілих структур каталогів. Вміння утиліти генерувати хеш-значення у різних форматах підвищує її адаптивність до різних потреб перевірки. Хоча md5deep може бути складнішим у вивченні через інтерфейс командного рядка, його потужність і гнучкість роблять його кращим вибором для користувачів, які звикли до роботи з командним рядком. [2].

Крім функцій хешування, такі інструменти, як GnuPG (GNU Privacy Guard) і OpenSSL, виділяються своєю майстерністю в області цифрових підписів і криптографії з відкритим ключем:

GnuPG, будучи реалізацією OpenPGP з відкритим вихідним кодом, забезпечує безпечну перевірку цілісності файлів і пропонує надійну систему управління ключами. Підтримка цифрових підписів і криптографії з відкритим ключем робить його комплексним рішенням для тих, кому важлива надійна основа безпеки. Хоча GnuPG може мати крутішу криву навчання для початківців, його широкі можливості роблять його незамінним інструментом для користувачів, які піклуються про безпеку.

OpenSSL, відомий своєю універсальністю, підтримує SSL, TLS і цифрові підписи, надаючи комплексні криптографічні функції для безпечного спілкування та перевірки цілісності файлів. Його широке використання в різних додатках свідчить про його надійність

та ефективність. Інтерфейс командного рядка OpenSSL може відлякувати деяких користувачів, але велика документація та підтримка спільноти роблять його придатним для тих, хто шукає потужне рішення, що легко налаштовується [3].

Аналіз цих інструментів за ознаками наведено в табл.1

Таблиця 1. Порівняння характеристик інструменів хешування

Інструмент	HashCheck	md5deep	GnuPG	OpenSSL
Інтерфейс	Графічний	Командный рядок	Командний рядок	Командний рядок
Функциональність	Хешування	Хешування	Хешування, цифровые подписи, криптография	Хешування, цифрові підписи, криптографія
Платформа	Windows	Кросплатформна	Кросплатформна	Кросплатформна
Складність	Простий	Складний	Складний	Складний
Підтримка алгоритмов хешування	MD5, SHA1, SHA256	MD5, SHA1, SHA256	MD5, SHA1, SHA256	MD5, SHA1, SHA256
Гнучкість	Низька	Висока	Висока	Висока
Додаткові функции	-	Рекурсивне хешування, порівняння файлів	Цифрові підписи, криптографія з відкритим ключем	Широкий спектр криптографічних функцій
Рекомендації	Для початківців	Для користувачів з досвідом	Для забезпечення максимальної безпеки	Для універсальності
Додаткові ресурси	-	https://md5deep.sourceforge.io/	https://www.gnupg.org/	https://www.openssl.org/

Кожен інструмент має свої унікальні сильні сторони, задовольняючи різні уподобання та вимоги користувачів. HashCheck і QuickHash GUI надають перевагу зручному інтерфейсу, що робить їх доступними для широкої аудиторії. md5deep, з його можливостями командного рядка, підходить для користувачів, які шукають розширені функції та автоматизацію. GnuPG і OpenSSL, з їх акцентом на цифрових підписах і криптографії, приваблюють тих, хто надає перевагу надійним основам безпеки. Зрештою, вибір між цими інструментами залежить від конкретних потреб, технічних навичок та уподобань користувачів.

Отже, автоматизована перевірка цілісності файлів за допомогою таких інструментів, як HashCheck, md5deep, GnuPG та OpenSSL, має вирішальне значення для протидії зростаючим загрозам кібербезпеки. Ці інструменти пропонують різноманітні рішення, що відповідають різним уподобанням користувачів і потребам організацій, посилюючи безпеку і надійність цифрових активів.

Литература

[1] code.kliu.org [Електронний ресурс]. – Режим доступу: https://code.kliu.org/hashcheck/

[2] md5deep and hashdeep – Latest version 4.4 [Електронний ресурс]. – Режим доступу: https://md5deep.sourceforge.net/

[3] OpenSSL vs GPG: Which Is Right for You [Електронний ресурс]. – Режим доступу: https://techcolleague.com/openssl-vs-gpg/

СТАНДАРТИ БЕЗПЕКИ ДЛЯ ПРИСТРОЇВ IOT

Захарова О.О.
Керівник: Долгова Н.Г.
E-mail: olga-zakharova@ukr.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Інтернет речей (IoT) - це концепція, що описується як мережа фізичних об'єктів (речей), у які вбудовані датчики, програмне забезпечення та інші технологічні інструменти для збору та обміну даними з іншими пристроями та системами через Інтернет. Ці пристрої можуть бути різних типів - від звичайних побутових речей до складних промислових інструментів.

В наслідок великої різноманітності сфер застосування та, у зв'язку з цим, різних вимог щодо протоколів безпеки, на сьогоднішній день не існує одного єдиного всесвітнього стандарту безпеки, який був би загальноприйнятим для всіх пристроїв Інтернету речей (IoT). Натомість, існують різні стандарти, рекомендації та рамки, що розробляються певними міжнародними організаціями, агентствами з безпеки та приватними групами.

Розглянемо одні з найбільш відомих існуючих стандартів безпеки:

1. Серія ISO/IEC 27000 - це набір міжнародних стандартів інформаційної безпеки, який був розроблений та опублікований спільно Міжнародною організацією зі стандартизації (ISO) та Міжнародним електротехнічним комітетом (IEC). Ця серія надає загальні рекомендації щодо управління інформаційною безпекою (управління інформаційними ризиками через засоби контролю ІБ) у контексті загальної системи управління безпекою (СУІБ) [1]. Ці стандарти використовуються великими підприємствами, оскільки їх підтримка коштовна для малого бізнесу. Однак перевага цієї серії полягає в тому, що вона визнана в нормативних актах та різних відповідних структурах.

2. ISA/IEC 62443 – це міжнародний стандарт, розроблений Міжнародною Комісією Електротехнічних Стандартів (IEC), спрямований на забезпечення кібербезпеки в промислових автоматизованих системах (Industrial Automation and Control Systems, IACS). Головна мета цього стандарту полягає в забезпеченні виробництва безпечних продуктів на основі принципів «за замовчуванням» та «при конструкції». Це означає впровадження засобів контролю безпеки на етапах проектування та впровадження систем керування для ефективного зменшення поточних і потенційних вразливостей безпеки [2].

3. ETSI EN 303 645 - це стандарт, розроблений Європейським інститутом стандартів телекомунікацій (ETSI). Він містить основні вказівки для організацій, які займаються розробкою та виробництвом споживчого Інтернету речей щодо того, як реалізувати ці положення [3].

4. NIST Cybersecurity Framework – це набір рекомендацій та керівництва, розроблений для забезпечення кібербезпеки в критичних інфраструктурних секторах, таких як енергетика, транспорт, фінанси і телекомунікації. Основна мета цього фреймворку полягає в створенні спільної мови для розуміння, управління та вираження ризиків кібербезпеки, які виникають у сфері критичної інфраструктури, як для внутрішніх, так і для зовнішніх зацікавлених сторін [4].

5. ENISA IoT Security Baseline – це документ, розроблений Європейським агентством з мереж і інформаційної безпеки (ENISA), який містить рекомендації та принципи безпеки для пристроїв IoT. Його ціль полягає в створенні базового рівня безпеки для різних типів пристроїв IoT. Основними аспектами є ідентифікація загроз і вразливостей, захист від атак, керування доступом, апаратні та програмні засоби безпеки, а також конфіденційність і цілісність даних [5].

Можна зробити висновок, що у Інтернеті речей (IoT) безпека відіграє вирішальну роль, оскільки пристрої IoT стають все більш поширеними в різних галузях, включаючи промисловість, транспорт, медицину та споживчий сектор. Враховуючи різноманітність застосування та вимог щодо безпеки для пристроїв IoT виникає необхідність використовувати різні стандарти безпеки.

Література

[1] IT Governance [Електронний ресурс]. – Режим доступу до ресурсу: https://www.itgovernance.co.uk/iso27000-family

[2] International Society of Automation [Електронний ресурс]. – Режим доступу до ресурсу: https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards

[3] European Telecommunications Standards Institute [Електронний ресурс]. – Режим доступу до ресурсу: https://www.etsi.org/technologies/consumer-iot-security?jjj=1707347303189

[4] NIST Cybersecurity for IoT Program [Електронний ресурс]. – Режим доступу до ресурсу: https://www.nist.gov/itl/applied-cybersecurity/nist-cybersecurity-iot-program

[5] Baseline Security Recommendations for IoT [Електронний ресурс]. – Режим доступу до ресурсу: https://www.enisa.europa.eu/publications/baseline-security-recommendations-for-iot

ІНСТРУМЕНТАЛЬНІ ЗАСОБИ ПОШУКУ ВРАЗЛИВОСТЕЙ ВЕБ-ДОДАТКІВ

Кондренко Я.Г.
Керівник:Долгова Н.Г.
E-mail: kond2992@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Ефективний захист персональних даних – одна з проблем сучасного світу. Викликом є те, що сучасні технології прогресують дуже швидко, а разом із цим є нагальна потреба в постійному вдосконаленні захисту інформації. Постачальники веб-додатків мусять постійно працювати над вдосконаленням свого ПЗ, виявляти слабкі місця задля забезпечення цілісності персональних даних користувачів. Веб-додатки стали невід'ємною частиною суспільного життя, тому їх використання майже у багатьох сферах змушують створювати надійніший захист задля попередження кіберзагроз. Для забезпечення стабільної роботи веб-додатку та захисту, цілісності, конфіденційності даних користувачів, додаток необхідно обов'язково перевірити на різні вразливості. Перевірка на вразливості може попередити наступні небажані наслідки, такі як: викрадення конфіденційних даних, внесення змін в існуючу інформацію, перехоплення сесій, застосування шкідливого коду. Отже, вчасне виявлення вразливостей є головною засадою коректного функціонування веб-додатку.

Слід звернути увагу, на причини виникнення вразливостей, ними можуть бути:

– використання слабких паролів користувачів;
– помилки програмного забезпечення;
– наявність шпигунського ПЗ;
– недостатнє тестування на вразливості;
– недостатнє розмежування доступу до інформації;
– складність системи. що збільшує ризик дефектів/

Виконаємо огляд декількох популярних інструментальних засобів, які допоможуть в пошуку вразливостей веб-додатків, одними з них є:

Burp Suite. Програма, що розроблена компанією PortSwigger [3], використовується для тестування веб-додатка на проникнення. Має багато різних функцій, а отже забезпечить виявлення та аналіз вразливостей. представлений як в платній версії так і в безкоштовній. Містить в собі такі інструменти як: Burp Proxy (виступає посередником між клієнтом та сервером, та може перехоплювати запити), Burp Spider(сканує сторінки на вразливості), Burp Intruder(використовується для проведення атак на додаток), Burp Scanner(виконує автоматичне сканування вразливостей), Burp Repeater(відтворює запити та аналізує відповіді сервера). Кожен з наведених інструментів відіграє важливу роль в попередженні та виявленні вразливостей, що дає змогу забезпечити якісний захист. [4]

Zed Attack Proxy. Один з поширених, простих та безкоштовних інструментів-сканерів веб-додатків на вразливості, реалізований на мові програмування Java,з відкритим вихідним кодом. Розроблений спільнотою Open Web Application Security Project (OWASP), що створює вільно доступні методології, інструменти та технології в сфері веб-застосунків. Має різні функціональні можливості, наприклад:сканування вразливостей, ручне тестування, перехоплення трафіку, та інші, призначений як для розробників так і для тестувальників. ZAP має інсталятори для Windows, Linux та macOS. [1]

Nessus. Комерційний інструмент, автоматичного пошуку вад у захисті системи. Розроблений компанією Tenable Network Security [2]. Nessus виявляє та аналізує вразливості в комп'ютерних мережах, системах або веб-додатках. Проводить перевірку сервісів маючи велику базу даних вразливостей, що має щотижневе оновлення. Зручний інтерфейс, гнучкі налаштуванням сканування та створення звітів про виявлення недоліків, робить Nessus оптимальним інструментом для захисту. Нижче наведемо найпоширеніші типи вразливостей, які можуть бути виявлені даним сервісом:

– помилки в конфігурації;

– наявність слабких паролів;

– присутність вразливих версій сервісів.

Одною з особливостей Nessus є функція, що класифікує вразливості за ступенем серйозності, щоб допомогти користувачам усунути найбільші загрози в терміновому порядку. Також перевагою сервісу є низький рівень хибних результатів, що дає можливість одразу звернути увагу на реальну загрозу [5].

Вище перераховані сервіси є одними з сучасних, потужних програм, що допомагають з виявленням проблем у веб-додатках .Кожен з них має свої недоліки та переваги, та вибір між ними залежить від потреб веб-додатка, його характеристики, бюджету та сфери застосування.Інструментальні засоби для виявлення вразливостей веб-додатків відіграють важливу роль у забезпеченні захисту та надійності систем. Безпека один з надважливих елементів при створенні та запуску веб-додатків. Досліджені нами програми, такі як: Burp Suite, ZAP і Nessus, можуть допомогти розробникам у підвищенні якості додатку. Регулярний аналіз з використанням вище перерахованих інструментів, може забезпечити своєчасне виявлених вразливостей. Таким чином, програми з виявлення вразливостей веб-додатків є важливим кроком під час розробки веб-додатку.

Література:

[1] Zed Attack Proxy - [Електронний ресурс] - URL: https://www.zaproxy.org/

[2] Nessus - [Електронний ресурс] -URL: https://www.tenable.com/products/nessus/nessus-expert

[3] Burp Suite - [Електронний ресурс]-URL: https://portswigger.net/burp/enterprise#section1

[4] Burp Suite - [Електронний ресурс]-URL: https://www.vaadata.com/blog/introduction-to-burp-suite-the-tool-dedicated-to-web-application-security/

[5] What is Nessus? - [Електронний ресурс] -URL: https://www.techtarget.com/searchnetworking/definition/Nessus

ЯК АТАКИ ТИПУ SQL-ІН'ЄКЦІЇ ВПЛИВАЮТЬ НА БЕЗПЕКУ ВЕБ-САЙТІВ МЕРЕЖІ

Зозуляк О.О.
Керівник: Лимаренко В.В.
E-mail: sashacawa2266@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

В сучасному цифровому середовищі веб-сайти відіграють ключову роль, діючи як платформи для розповсюдження інформації, комунікації та проведення онлайн-транзакцій. Проте, завдяки їх широкому використанню, вони стають привабливими цілями для зловмисних дій, з SQL-ін'єкціями серед основних загроз.

Власникам та адміністраторам веб-сайтів надзвичайно важливо бути в курсі останніх загроз безпеки та систематично оновлювати свої системи та додатки для усунення вразливостей. Велике значення має досвід користувачів щодо безпечних практик в Інтернеті, таких як створення надійних паролів і обережність при обміні особистою інформацією, що сприяє загальним практикам кібербезпеки.

Що таке SQL-ін'єкція? SQL-ін'єкція це одна з найпоширеніших тактик, які використовують хакери та зловмисники. Під час процесу прохожить посилання в систему шкідливого код. Цей код здатний зламати вашу базу даних та достати певні данні. Хакер поміщає шкідливий код у SQL-запити, які передаються через вхідні дані веб-сайту.

Поширений сценарій, у якому це відбувається, може мати наступний вигляд: відвідувачу сайту необхідно надати інформацію, наприклад логін або ім'я користувача. Замість імені користувача відвідувач надає вам SQL-запит. Цей SQL-запит потрапляє на ваш сайт, де без вашого відома працює з вашою базою даних.

Хакери використовують подібні атаки, тому що їх відносно легко здійснити. Власникам і операторам зламаних сайтів також буває складно зрозуміти, що вони зазнали атаки. Повільне реагування на атаку може завдати незліченної шкоди постраждалій системі.

Ефективний захист ваших клієнтів від потенційних витоків даних прямо впливає на рівень довіри, який вони мають до вашого бізнесу. Клієнти будуть більш схильні вірити вам, якщо вони переконані у безпеці своєї інформації. Проте, забезпечення захисту вашого веб-сайту від потенційних загроз також має прямий вплив на сам бізнес.

Можно привести приклад: якщо ваш веб-сайт стане жертвою компрометації, це може автоматично призвести до припинення діяльності вашого бізнесу або втрати частини клієнтів, що в результаті призводить до погіршення довіри до вас. Тому критично важливо надавати належну увагу актуальності та безпеці вашого веб-сайту. Він не лише є візитівкою вашого бізнесу, але також містить найбільш конфіденційну та цінну інформацію, яку слід належним чином захищати.

Насамкінець слід зазначити, захист від SQL-ін'єкцій вимагає комплексного підходу, включаючи безпечне кодування, регулярні аудити безпеки та застосування захисних засобів, таких як веб-захист (WAF). При активному розвитку кіберзагроз наявність проактивного підходу та стеження за безпекою залишаються критичними для забезпечення цілісності та безпеки онлайн-платформ.

Література

[1] SQL Injection [Электроний ресурс]. – Режим доступа до ресурса: https://owasp.org/wwwcommunity/attacks/SQL_Injection#:~:text=SQL%20Injection%20has%20become%20a,attempted%20attack%20of%20this%20kind.

[2] SQL Injection Attack: How it Works, Examples and Prevention [Электроний ресурс]. – Режим доступа до ресурсу: https://brightsec.com/blog/sql-injection-attack/

РОЗРОБКА КОНЦЕПЦІЇ ЗАХИСТУ ПІДПРИЄМСТВА НА ОСНОВІ АНАЛІЗУ НОВІТНІХ ТЕНДЕНЦІЙ У КІБЕРЗЛОЧИННОСТІ

Карнаушенко А.О.
Керівник: Старкова О.В.
E-mail: andrykarnaushenko@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Інформаційний злочин – незаконні дії спрямовані на розкрадання або руйнування інформації в інформаційних системах і мережах, які виходять з корисливих або хуліганських спонукань[1]. До основних видів кіберзлочинності можна віднести такі: розповсюдження шкідливого програмного забезпечення, крадіжка номерів кредитних карт і банківських рахунків, злом паролів, порушення авторських прав[1].

Сучасний світ стає все більш залежним від цифрових технологій. Це робить кібербезпеку одним із найважливіших пріоритетів для будь-якого підприємства. Кіберзлочинність постійно еволюціонує, тому важливо знати про новітні тенденції та розробляти відповідні стратегії захисту.

До інформаційних злочинів можна віднести злочини, скоєні за статтями Кримінального кодексу України, що входять до Розділу 16 «Злочини у сфері використання електронно-обчислювальних машин (комп'ютерів), систем та комп'ютерних мереж і мереж електрозв'язку»[2].

Однією з ключових складових ефективної концепції захисту є глибокий аналіз новітніх тенденцій у кіберзлочинності. Злочинці постійно вдосконалюють свої методи, використовуючи нові технології та тактики. Розуміння цих тенденцій дозволяє підприємствам адаптувати свої заходи захисту, прогнозуючи можливі напрямки атак і вчасно реагуючи на них.

Проведення глибокого аналізу повинно містити вивчення сучасних методів атак, використання широкого спектра загроз, таких як розкрадання даних, атаки з використанням шкідливих програм та інші. Також важливо враховувати тенденції у сфері кіберзлочинності, такі як розширення атак на Інтернет речей, використання штучного інтелекту для зломів та еволюція методів, які дозволяють злочинцям отримати грошову винагороду за повернення вкраденої інформації.

На основі отриманих даних, концепція захисту підприємства повинна передбачати інтеграцію передових технологій та розробку гнучких стратегій, здатних адаптуватися до нових загроз. Ефективний захист містить застосування сучасних систем виявлення інцидентів, моніторингу мережі та прогнозування вразливостей. Важливо також ставити акцент на підвищенні кваліфікації персоналу та регулярних тренінгах з питань кібербезпеки.

Враховуючи викладене, важливо прийняти зважене рішення щодо напрямків і пріоритетів захисту ключових інформаційних та автоматизованих систем управління в організаціях. При цьому треба враховувати тенденції багатовекторності та швидкозмінності схем атак, орієнтованих на невідомі вразливості або відсутність ефективних систем захисту від них на фоні обмеженого фінансування в сферах інформаційної технологій.

Література

[1] Вікіпедія. Інформаційні злочини [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Інформаційні_злочини

[2] Д. В. Дубов Розділ 4. Забезпечення національних інтересів України в глобальному та національному кіберпросторах. Кіберпростір як новий вимір геополітичного суперництва Монографія. Київ: Національний інститут стратегічних досліджень, 2014. [Електронний ресурс] – Режим доступу до ресурсу: https://www.niss.gov.ua/sites/default/files/2015-02/Dubov_mon-89e8e.pdf

АНАЛІЗ ОСНОВНИХ МЕТОДІВ СОЦІАЛЬНОЇ ІНЖЕНЕРІЇ ТА РОЗРОБКА РЕКОМЕНДАЦІЙ ДЛЯ ПІДВИЩЕННЯ ОБІЗНАНОСТІ СПІВРОБІТНИКІВ В ОФІСНОМУ СЕРЕДОВИЩІ

Кравець С.О.
Керівник: Міхєєв І.А.
E-mail: steven.dnepr@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Соціальна інженерія в останні роки стала однією з найбільш поширених та небезпечних загроз інформаційній безпеці в офісних середовищах. Ця методика маніпулювання людьми для отримання конфіденційної інформації може призвести до серйозних наслідків для організацій. Аналіз основних методів соціальної інженерії, таких як фішинг, перехоплення ідентифікаційних даних та імперсоніфікація, вказує на їхню високу ефективність та потенційну небезпеку для безпеки інформації.

У межах дослідження розробляються рекомендації для підвищення обізнаності співробітників в офісному середовищі щодо методів соціальної інженерії. Ці рекомендації включають в себе проведення регулярних навчань та тренінгів з інформаційної безпеки, надання інструкцій щодо розпізнавання підозрілих ситуацій та використання безпечних практик у повсякденній роботі.

Актуальність теми дослідження полягає в зростаючій загрозі соціальної інженерії для організацій та їх інформаційної безпеки. Соціальна інженерія - це маніпулювання людьми з метою отримання конфіденційної інформації або надання доступу до захищених систем. Відповідно, обізнаність співробітників з цими методами стає критичною для запобігання інцидентам безпеки в офісному середовищі.

У межах дослідження проводиться аналіз основних методів соціальної інженерії, таких як фішинг, перехоплення ідентифікаційної інформації, імперсоніфікація тощо. Дослідження також включає вивчення реальних випадків соціальної інженерії та їх наслідків для організацій [3].

Аналіз ефективності різних методів усвідомлення та навчання співробітників виявляє, які підходи є найбільш успішними для підвищення їх обізнаності та практичного реагування на соціальну інженерію. Це може включати навчання на основі симуляційних вправ, проведення обов'язкових курсів з безпеки, а також надання інструкцій та рекомендацій.

Загальне значення цього дослідження полягає в підвищенні рівня обізнаності та здатності співробітників ефективно реагувати на соціальну інженерію в офісному середовищі. Це допомагає запобігти витоку конфіденційної інформації, фінансовим втратам та іншим наслідкам невдалих атак [2].

Важливий компонент дослідження - це розробка рекомендацій для підвищення обізнаності співробітників щодо можливих загроз соціальної інженерії. Ці рекомендації мають на меті навчити персонал розпізнавати потенційні атаки та уникати їх, а також надати інструменти для ефективної реакції в разі виявлення підозрілих ситуацій.

Крім того, в рамках дослідження враховується контекст офісного середовища, що передбачає специфічні вимоги та сценарії використання інформації. Рекомендації повинні бути підготовлені з урахуванням цих особливостей, щоб бути максимально ефективними та придатними для впровадження в конкретному офісному середовищі [1].

Література

[1] Мітник К. Мистецтво обману / К. Мітник. – John Wiley & Sons, 2002. – 304 с.

[2] Шудрова К. Соціальна інженерія в інформаційній безпеці / К. Шудрова // Директор з безпеки. - 2012. - №10. - с. 13-17

[3] Інформаційна та кібербезпека: соціотехнічний аспект: підручник / [В. Л. Бурячок, В. Б. Толубко, В. О. Хорошко, С. В. Толюпа]; за заг. ред. д-ра техн. наук, професора В. Б. Толубка.— К.: ДУТ, 2015.— 288 с.

МЕТОДИ ТА ЗАСОБИ ПІДВИЩЕННЯ ОБІЗНАНОСТІ ПЕРСОНАЛУ З КІБЕРБЕЗПЕКИ НА ПРОМИСЛОВОМУ ПІДПРИЄМСТВІ

Красільніков М.В.
Керівник: Старкова О.В.
E-mail: maksud11112002@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Кібербезпека – це критичне питання для сучасних промислових підприємств. Кібератаки можуть призвести до значних фінансових втрат, зупинки виробництва, шкоди для репутації та інших негативних наслідків. Тому важливо, щоб персонал підприємства був обізнаний з питань кібербезпеки та мав уявлення про захист від кіберзагроз.

Традиційно, підвищення обізнаності про кібербезпеку фокусувалося на односторонньому інформуванні персоналу, наприклад, через лекції, семінари або інформаційні кампанії. Цей підхід не завжди ефективний, адже він не враховує людський фактор та не дає людям мотивації для впровадження правил кібербезпеки. Новий підхід до кібербезпеки спирається на концепції моделі поділеної відповідальності, поширеної у хмарних технологіях. У цій моделі кібербезпека розглядається як корпоративна стратегічна мета, а не просто як відповідальність служби безпеки.

Ключові елементи нового підходу є:

1) оцінка безпеки – кожен співробітник має «оціночний лист безпеки», включений до його щорічних показників оцінки ефективності;

2) спрощення – інструменти та процеси безпеки повинні бути зручними та простими у використанні;

3) співпраця – служба безпеки співпрацює з іншими відділами (наприклад, HR, комунікації та маркетингу) для кращого залучення персоналу;

4) пояснення – співробітникам пояснюють, чому кібербезпека важлива та як вони можуть стати частиною рішення;

5) доступність – керівники служби безпеки регулярно спілкуються з персоналом, щоб підкріплювати інформацію, відповідати на питання та збирати відгуки;

6) мотивація – програма кібербезпеки будується з урахуванням того, що мотивує співробітників [1].

Перехід до моделі поділеної відповідальності – це складний процес, але він необхідний для створення ефективної культури кібербезпеки на промисловому підприємстві.

Окрім вищезгаданих елементів, важливо також використовувати різні канали комунікації, щоб донести інформацію про кібербезпеку до персоналу. Це можуть бути лекції, семінари, онлайн-курси, електронні листи, брошури, плакати, відео та інші. Важливо також заохочувати співробітників повідомляти про підозрілі активності. Це може допомогти службі безпеки вчасно виявити та запобігти кібератакам.

Підвищення обізнаності персоналу з кібербезпеки – це постійний процес, який потребує постійної уваги та зусиль. Використання нового підходу, який спирається на модель поділеної відповідальності, може допомогти промисловим підприємствам значно підвищити рівень кібербезпеки та захистити себе від кібератак.

Література

[1] Chanel for IT [Електронний ресурс] – Режим доступу до ресурсу: https://channel4it.com/publications/znannya-pro-kberbezpeku-shst-porad-yak-dopomozhut-vashim-spvrobtnikam-stati-kberumnimi.html Знання про кібербезпеку: шість порад, які допоможуть вашим співробітникам стати «кіберумними».

AXENSE NETTOOLS – УПРАВЛІННЯ ТА КОНТРОЛЬ МЕРЕЖІ

Крощенко М.В., Леуненко О.В.
E-mail: kroshch3228@gmail.com, Oleksii.Leunenko@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

У світі, де технології є невід'ємною частиною кожного бізнесу, керування мережею стає критично важливим завданням для всіх IT-фахівців. Оскільки компанії все більше покладаються на мережеву інфраструктуру, дуже важливо мати низку надійних рішень для моніторингу, керування та діагностики мережі, яке допоможе підтримувати високу ефективність мережі та оперативно вирішувати будь-які проблеми.

Моніторинг мережі є важливим практичним завданням. Загалом, всі інструменти для аналізу та діагностики комп'ютерних мереж можна розділити на кілька великих класів:

– інтегровані діагностичні засоби – ці інструменти вбудовані в мережеве обладнання та надають інформацію про його стан і ефективність;

– фізичні діагностичні засоби - спрямовані на виявлення фізичних проблем, таких як збої в кабелях чи обладнанні;

– аналізатори протоколів - дозволяють аналізувати та моніторити дані протоколів, що передаються по мережі;

– монітор мережі - спеціалізовані інструменти для спостереження за мережевим трафіком та роботою пристроїв;

– програмні діагностичні засоби - забезпечують різноманітні функції для тестування та діагностики мережі через програмний інтерфейс;

– засоби керування системою - допомагають в управлінні конфігурацією та ресурсами мереж.

Одним з найкращих безкоштовних комплексних програмних засобів, що поєднують в собі декілька класів, є Axsense NetTools – це потужне та комплексне рішення, яке пропонує широкий спектр інструментів та функцій для моніторингу мережі та діагностики, яке допоможе ефективно керувати мережевою інфраструктурою. Програма забезпечує комплексне уявлення про мережеву інфраструктуру, включаючи продуктивність пристрою та якість мережі. Завдяки широкому набору функцій і зручному дизайну Axsense NetTools є ідеальним інструментом для оптимізації продуктивності мережі.

Однією з найважливіших переваг Axsense NetTools є його система моніторингу та сповіщень хоста. Ця функція дозволяє відстежувати працездатність і стан мережевих пристроїв і отримувати сповіщення в режимі реального часу, коли виникають проблеми. За допомогою цієї функції можливо швидко виявити та вирішити проблеми з мережею до того, як вони спричинять значний застій або втрату даних.

Система моніторингу та сповіщень хоста в Axsense NetTools легко налаштовується. Є можливість встановити порогові значення для різних мережевих пристроїв і отримувати сповіщення, коли ці порогові значення перевищено. Ця функція особливо корисна для IT-фахівців, яким потрібно швидко виявляти та вирішувати проблеми у великому мережевому середовищі.

Іншою важливою особливістю Axsense NetTools є можливість сканування мережі. Ця функція забезпечує комплексне уявлення про мережеву інфраструктуру, включаючи продуктивність пристрою та якість мережі. За допомогою цієї інформації можливо оцінити ефективність мережі та визначити потенційні проблеми, які можуть вплинути на роботу мережі.

Функцію мережевого сканування в Axsense NetTools можна налаштувати. Є можливість виконати швидке сканування мережі або виконати детальне сканування, яке надасть повне уявлення про мережеву інфраструктуру. Ця функція особливо корисна для IT-фахівців, яким необхідно завчасно виявляти та вирішувати проблеми мережі.

Axsense NetTools також пропонує низку стандартних IP-інструментів, які дозволяють вимірювати пропускну здатність, відстежувати пакети та виконувати інші важливі мережеві

завдання, не викликаючи перевантаження мережі. Крім того, браузер Simple Network Management Protocol (SNMP) дозволяє легко керувати мережевими пристроями та контролювати їх навіть новачкам.

Стандартні інструменти IP в Axsense NetTools використовуються для виконання широкого спектру мережевих завдань, включаючи ping, traceroute і DNS-пошук. Ця функція особливо корисна для IT-фахівців, яким потрібно швидко й ефективно виконувати мережеві завдання.

Крім того, Axsense NetTools надає інформацію про вхідні та вихідні з'єднання, що дозволяє швидко ідентифікувати мережеві вузли та їхні служби. Зручний графічний інтерфейс інструменту та параметри експорту даних у формати тексту, Excel і HTML полегшують аналіз та інтерпретацію даних, що робить його потужним інструментом для адміністрування системи та підтримки мережевої інфраструктури.

Програма дає можливість налаштувати інтерфейс для відображення необхідної інформації, полегшуючи навігацію та використання. Ця функція є досить корисною для користувачів, яким потрібен швидкий і ефективний доступ до інформації про мережу.

Оскільки виробник програмного забезпечення не надав мінімальні системні вимоги, для запуску програми, після проведення самостійного дослідження, важливо зазначити, що Axsense NetTools обмежений підтримкою тільки операційної системи Windows, версією від Windows XP до Windows 11 і вимагає наявності лише 300 МБ вільного місця на диску. Це може вплинути на доступність програми для користувачів, які використовують інші операційні системи.

Axsense NetTools є цінним ресурсом як для особистого, так і для корпоративного мережевого середовища. Його функції та функціональність роблять його чудовим вибором для користувачів та IT-фахівців, які хочуть підтримувати високу ефективність мережі та оперативно вирішувати будь-які проблеми. Завдяки широкому набору функцій і зручному дизайну Axsense NetTools є ідеальним інструментом для оптимізації продуктивності мережі.

Підсумовуючи, Axsense NetTools – це потужне та комплексне рішення для моніторингу, керування та діагностики мережевих проблем. Його система моніторингу та сповіщення хоста, функція сканування мережі, стандартні інструменти IP і браузер SNMP роблять його цінним інструментом як для особистого, так і для корпоративного мережевого середовища. Його зручний графічний інтерфейс і параметри експорту даних у різні формати тексту роблять його потужним інструментом для адміністрування системи та підтримки мережевої інфраструктури. Якщо ви шукаєте інструмент, який допоможе вам підтримувати високу ефективність мережі та оперативно вирішувати будь-які проблеми, Axsense NetTools – чудовий вибір.

Література

[1] Axense. NetTools - Free and functional network diagnostic tool [Електронний ресурс]. – Режим доступу: https://axence.net/en/axence-nettools

[2] Uptodown. Axence NetTools Download [Електронний ресурс]. – Режим доступу: https://axence-nettools.en.uptodown.com/windows

[3] Діагностика локальних обчислювальних мереж [Електронний ресурс]. – Режим доступу: https://vuzlit.com/963604/instrumenti_diagnostiki

[4] Cnet. Axence NetTools Download [Електронний ресурс]. – Режим доступу: https://download.cnet.com/axence-nettools/3000-2085_4-10395430.html

ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ВИЯВЛЕННЯ ТА ЗАПОБІГАННЯ КІБЕРАТАКАМ

Кузьмінов Б.Р.
Керівник: Шаповалова О.О.
E-mail: bogdan122119@gmail.com, olena.shapovalova@hneu.net
Харків, Харківський національний економічний університет імені Семен Кузнеця

Сучасний світ переживає стійке нарощування кількості кіберзагроз та інцидентів інформаційної безпеки, що ставить під загрозу як приватну, так і державну сфери. У зв'язку з цим виникає необхідність впровадження надійних та ефективних заходів забезпечення безпеки, одним з ключових інструментів якої є штучний інтелект [1].

Штучний інтелект є потужним інструментом для виявлення, аналізу та реагування на кіберзагрози. Здатність інтерпретувати стан середовища та розпізнавати аномалії дозволяє системам штучного інтелекту ефективно виявляти вразливості та попереджати можливі інциденти. Технології штучного інтелекту, серед яких машинне навчання, аналітика поведінки користувачів та алгоритми автоматичної реакції, допомагають створити системи моніторингу, які можуть надійно захищати інформацію в реальному часі.

Однією з ключових сфер застосування штучного інтелекту є захист додатків та мереж. Зокрема, системи протидії шахрайства (антифрод) та захисту прикладних застосунків використовують штучний інтелект для виявлення аномальних дій та попередження введення в оману та вчинення протиправних дій. Крім того, інтелектуальні системи підтримки прийняття рішень та управління ідентифікацією допомагають у реагуванні на загрози та забезпеченні безпеки даних.

В контексті інформаційної безпеки штучний інтелект – програмне забезпечення, яке здатне інтерпретувати стан середовища, розпізнати певні події та самостійно прийняти необхідні заходи. Серед платформ, які використовують технології штучного інтелекту для протидії кіберзагрозам, можна виділити наступні.

EDR (Endpoint Detection and Response) – платформи виявлення атак на робочих станціях, серверах, будь-яких комп'ютерних пристроях (кінцевих точках) та оперативне реагування на них (рис.1). За допомогою технологій штучного інтелекту продукти даної категорії можуть виявляти невідомі шкідливі програми, автоматично класифікувати загрози та самостійно реагувати на них, передаючи дані в центр управління. Штучний інтелект приймає рішення на основі загальної бази знань, накопиченої шляхом збору даних з множини пристроїв.

NDR (Network Detection and Response) – пристрої та аналітичні платформи, які виявляють атаки на мережевому рівні та дозволяють оперативним чином на них реагувати. Використовуючи накопичену статистику та базу знань про загрози, продукти даного типу за допомогою технологій штучного інтелекту виявляють загрози в мережевому трафіку та можуть автоматично реагувати на них відповідним чином, змінюючи конфігурацію мережевих пристроїв та шлюзів. Частина продуктів даного типу спеціалізується на захисті хмарних провайдерів та їхньої інфраструктури.

UEBA (User and Entity Behavior Analytics) – системи поведінкового аналізу користувачів та інформаційних сутностей. Вони виявляють випадки незвичної поведінки та використовують їх для детектування внутрішніх та зовнішніх загроз. Основним сценарієм застосування технологій штучного інтелекту в продуктах типу UEBA є автоматичне виявлення аномалій в поведінкових моделях (відхилення від норми або невідповідність шаблону) для користувачів та різних інформаційних систем. Виявлені аномалії за допомогою штучного інтелекту класифікують як різноманітні загрози та ризики для держави.

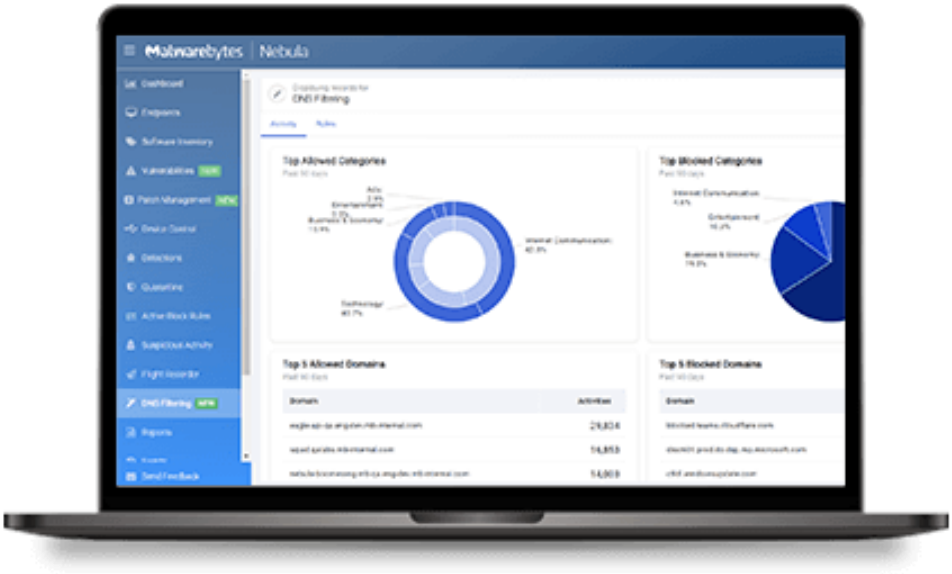


Рисунок 1. Вікно роботи EDR

TIP (Threat Intelligence Platform) – платформи попереднього детектування загроз та реагування на них, що діють, спираючись на значний обсяг накопичених різноманітних даних (Data Lake) та індикаторів компрометації (IoC). Застосування штучного інтелекту дозволяє підвищити ефективність виявлення невідомих загроз на ранніх етапах; сценарій дещо схожий на роботу SIEM-систем, але націлений на зовнішні джерела даних та зовнішні загрози.

SIEM (Security Information and Event Management) – рішення, які здійснюють моніторинг інформаційних систем в режимі реального часу та аналізують події безпеки, що поступають від мережевих пристроїв, засобів захисту інформації, IT-сервісів, інфраструктури систем та додатків, та допомагають виявити інциденти в інформаційній безпеці. В системах такого класу накопичується велика кількість даних з різних джерел, а застосування технологій штучного інтелекту дає можливість евристичними методами виявити аномалії та скоротити хибні спрацьовування під час зміни моделей даних [2].

Однак разом зі зростанням застосування штучного інтелекту в сфері інформаційної безпеки виникають нові виклики, зокрема, пов'язані з етичними питаннями та зі збереженням приватності. Забезпечення безпеки конфіденційної інформації та захист прав користувачів стає важливим завданням для подальшого розвитку і вдосконалення систем безпеки на основі штучного інтелекту.

Отже, штучний інтелект відіграє ключову роль у забезпеченні інформаційної безпеки, допомагаючи виявляти, аналізувати та реагувати на кіберзагрози в реальному часі. Його впровадження сприяє створенню надійних систем захисту, але водночас вимагає уваги до етичних аспектів та забезпечення приватності.

Література

[1] Роль штучного інтелекту в кібербезпеці: передбачення та запобігання атакам [Електроний ресурс] - Режим доступу до ресурсу: https://eba.com.ua/rol-shtuchnogo- intelektu-v-kiberbezpetsi-peredbachennya-ta- zapobigannya-atakam/

[2] Використання штучного інтелекту в інформаційній безпеці України [Електроний ресурс] - Режим доступу до ресурсу: http://www.dy.nayka.com.ua/pdf/1_2022/4.pdf

ВАЖЛИВІСТЬ БІОМЕТРИЧНОГО ЗАХИСТУ ПРИВАТНОЇ ІНФОРМАЦІЇ

Кулик О. В.
Керівник: Долгова Н. Г.
E-mail: s.v.kulyk55@gmail.com
Харків, Харківський національний економічний університет ім. Семена Кузнеця

Біометричний захист є важливою складовою забезпечення приватності інформації, оскільки він базується на унікальних фізичних характеристиках кожної особи, таких як відбитки пальців, розпізнавання обличчя, голосу тощо.

Що ж таке біометрія, біометричні дані, біометричні параметри?

Біометричні дані представляють собою інформацію про конкретну особу, отриману шляхом фіксації її характеристик, які відрізняються від аналогічних параметрів інших осіб та мають стійку стабільність. Ці параметри можуть включати відцифрований підпис, обличчя чи відбитки пальців.

Біометричні параметри - це фізичні характеристики або особистісні риси, що вимірюються і використовуються для ідентифікації особи або перевірки наданої інформації про її ідентифікацію.

Біометрія - це сукупність автоматизованих методів та засобів ідентифікації особи, що базуються на її фізичних або поведінкових характеристиках [1].

Нижче я наведу декілька причин, чому біометричний захист важливий для приватної інформації:

– Унікальність: Біометричні дані є унікальними для кожної особи і важко підробити або скопіювати. Це робить їх ефективними для ідентифікації особи та захисту її особистої інформації.

– Непередаваність: Особливості біометричних даних, такі як відбитки пальців або розпізнавання обличчя, зазвичай не можуть бути передані або використані кимось іншим без належної авторизації.

– Безпека: Використання біометричних даних може зменшити ризик використання паролів або інших ідентифікаторів, які можуть бути вкрадені або відновлені зловмисниками.

– Зручність: Біометричний захист може забезпечити швидкий та зручний доступ до інформації без необхідності запам'ятовування складних паролів чи кодів.

– Контроль доступу: З використанням біометричних технологій можна здійснювати більш точний контроль доступу до об'єктів, систем або приміщень.

Проте важливо також пам'ятати, що біометричні дані також можуть стати об'єктом зловживання, якщо вони потраплять у ненадійні руки. Тому необхідно вживати заходів безпеки для захисту самого процесу збереження та обробки біометричних даних, а також забезпечити їх конфіденційність та недоступність для несанкціонованого доступу.

Отже, системи біометричного захисту інформації не можна вважати абсолютно надійними через можливість виникнення проблем з різних причин, включаючи фізичні фактори. Однак прогрес в розробці сенсорних технологій, алгоритмів обробки даних та машинного навчання може сприяти подоланню цих проблем і покращенню точності та надійності систем біометричного захисту. Таким чином, такі системи будуть постійно вдосконалюватися, оскільки, незважаючи на обмеження і можливі помилки, наприклад, хибні спрацювання, вони залишаються ефективними засобами підвищення безпеки.

Література

[1] "Біометричні дані: збір і захист у Європі, США та Україні" [Електронний ресурс] – Access mode: https://yur-gazeta.com/publications/practice/inshe/biometrichni-dani-zbir-i-zahist-u-evropi-ssha-ta-ukrayini.html

АНАЛІЗ ІСНУЮЧИХ АНТИВІРУСНИХ РІШЕНЬ ДЛЯ ВИЯВЛЕННЯ ТА ЛІКВІДАЦІЇ ЗАГРОЗ У КІБЕРПРОСТОРІ

Насибулін Є.С.
Керівник: Старкова О.В.
E-mail: eugenehelpme@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

За даними звіту [1] протягом 2023 році в Україні було зафіксовано та оброблено 1105 кіберінцидентів, що на 62,5% більше, ніж у 2022 році. Підсистемами виявлення вразливостей і реагування на кіберінциденти та кібератаки було детектовано 1 516 861 підозрілих файлів, серед яких у категорії «02 Шкідливий програмний код (Malicious Code)» переважають SmokeLoader, Agent Tesla, Snake Keylogger, Remcos, Formbook [2]. Враховуючи стрімке зростання кількості кібератак, актуальною та невідкладною потребою стає забезпечення захисту комп'ютерних систем. Це пов'язано з тим, що кібератаки можуть завдати значної шкоди як приватним особам, так і організаціям.

Значна частина користувачів нехтує питаннями кібербезпеки пристроїв, що спричиняє виникнення суттєвих ризиків для безпеки. Поширена думка, що за відсутності завантаження неперевіреного програмного забезпечення з Інтернету потреба в антивірусному програмному забезпеченні (ПЗ) відсутня. Однак, це твердження є хибним. Існує широкий спектр вразливостей, наприклад, «вразливості нульового дня» [3], які дають можливість зловмисникам завдати шкоди пристрою, навіть без завантаження неперевіреного ПЗ. Аналіз статистичних даних [4] свідчить, що 15% громадян Сполучених Штатів Америки не використовують антивірусні програми. Наведені дані свідчать про низький рівень обізнаності значної частини користувачів з питань кібербезпеки, що робить їх вразливими до кібератак. Вплив вірусного програмного забезпечення може варіюватися від сповільнення роботи комп'ютерних систем до викрадення та втрати даних, а в окремих випадках – до незворотного пошкодження та руйнування комп'ютерних систем. Враховуючи значні ризики, пов'язані з вірусними атаками, стає очевидною нагальна потреба у забезпеченні захисту комп'ютерних систем за допомогою антивірусного ПЗ.

В суспільстві відбувається дискусія щодо питання, яке антивірусне рішення краще обрати – платне чи безкоштовне. Аналіз даних [4] свідчить, що 61,2% громадян Сполучених Штатів Америки використовують безкоштовне антивірусне ПЗ, натомість 32,4% користувачів надають перевагу платному антивірусному ПЗ. Але залишається питання щодо кращого вибору антивірусного рішення – платне чи безкоштовне. Результати дослідження свідчать про те, що рівень зараження шкідливими вірусами серед користувачів платного антивірусного програмного забезпечення на 2% нижчий, ніж у користувачів безкоштовного ПЗ. Наведені дані свідчать про несуттєві розбіжності в рівні захисту, що надається платним та безкоштовним антивірусним ПЗ. Також необхідно пам’ятати, що деякі безкоштовні програми можуть бути неефективними або навіть шкідливими для комп’ютерної системи користувача.

Враховуючи вищезазначене, актуалізується питання вибору оптимального безкоштовного антивірусного програмного забезпечення для здійснення належного захисту. В Україні офіційно сертифіковано ряд антивірусних ПЗ [5], декілька з яких пропонують безкоштовний план користування. Найбільш розширений функціонал серед таких антивірусних засобів пропонують такі ПЗ, як Avast Free Antivirus, McAfee Internet Security, Bitdefender Total Security. До ключових функцій, що реалізовані в даних ПЗ, відносяться:

– антивірусний захист: виявлення та нейтралізація шкідливого програмного забезпечення;

– антишпигунське забезпечення: захист від шпигунських програм, що можуть красти особисті дані;

– захист від фішингу: запобігання переходу на шахрайські веб-сайти, що імітують легальні ресурси;

– antirootkit: виявлення та видалення руткітів [6], які дозволяють зловмисникам отримати несанкціонований доступ до системи;

– HIPS (Host-based Intrusion Prevention System): система запобігання вторгненням на хост-машині, що блокує підозрілу активність;

– захист у реальному часі: постійний моніторинг системи для запобігання кібератакам.

– персональний фаєрвол: контроль мережевого трафіку та блокування несанкціонованого доступу.

– антиспам: блокує небажану електронну пошту;

– захист від мережевих атак: блокує атаки, спрямовані на експлуатацію вразливостей мережевого протоколу;

– менеджер паролів: система управління паролями, що допомагає користувачу створювати та зберігати складні та безпечні паролі для різних сайтів та програм.

Важливо також акцентувати увагу на антивірусному програмному забезпеченні Defender Antivirus, що інтегровано в операційну систему Windows, який у грудні 2022 року у рейтингу антивірусних програм від незалежної організації «AV-Test» [7] отримав 16 з 17 можливих балів, що свідчить про те, що це антивірусне ПЗ знаходиться на рівні з іншими антивірусами сторонніх розробників. Також необхідно зазначити, що у 2022 році Федеральна комісія із зв'язку США додала до переліку «неприйнятного ризику» [8] щодо використання антивірусного програмного забезпечення російського походження Kaspersky Antivirus.

Підсумовуючи вищесказане, можна з упевненістю стверджувати, що на сучасному етапі розвитку інформаційних технологій наявність антивірусного захисту для комп'ютерних систем є невід'ємною складовою їх безпечної експлуатації. На ринку представлено широкий спектр надійних платних та безкоштовних антивірусних програмних засобів, що постійно оновлюються та вдосконалюються. Це дає можливість користувачам підібрати оптимальний варіант для захисту своїх пристроїв від кібератак.

Література

[1] Статистичний звіт за результатами роботи Системи виявлення вразливостей і реагування на кіберінциденти та кібератаки в 2023 році [Електронний ресурс]. – Режим доступу: https://scpc.gov.ua/uk/articles/334.

[2] Державна служба спеціального зв'язку та захисту інформації України. Перелік категорій кіберінцидентів [Електронний ресурс]. – Режим доступу: https://scpc.gov.ua/uk/articles/334https://www.cip.gov.ua/ua/news/perelik-kategorii-kiberincidentiv.

[3] Вікіпедія. Вразливість нульового дня [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/Вразливість_нульового_дня.

[4] 2023 Antivirus Market Annual Report [Електронний ресурс]. – Режим доступу: www.security.org/antivirus/antivirus-consumer-report-annual/.

[5] Analysis and research of the characteristics of standardized in Ukraine antivirus software [Електронний ресурс]. – Режим доступу: https://www.researchgate.net/publication/334198724_analysis_and_research_of_the_characteristics_of_standardized_in_ukraine_antivirus_software.

[6] Wikipedia. Rootkit [Електронний ресурс]. – Режим доступу: https://en.wikipedia.org/wiki/Rootkit

[7] AV-TEST Product Review and Certification Report – Nov-Dec/2022 [Електронний ресурс]. – Режим доступу: https://www.av-test.org/en/antivirus/home-windows/windows-10/december-2022/microsoft-defender-4.18-221615/.

[8] Public safety and homeland security bureau announces additions to the list of equipment and services covered by section 2 of the secure networks act [Електронний ресурс]. – Режим доступу: https://docs.fcc.gov/public/attachments/DA-22-320A1.pdf.

АНАЛІЗ ЗАГРОЗ ТА МЕТОДІВ ЗАХИСТУ МОБІЛЬНИХ ПРИСТРОЇВ В КІБЕРПРОСТОРІ

Поповиченко Д.С.
Керівник: Шаповалова О.О.
E-mail: dimapopovichenko3@gmail.com, olena.shapovalova@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Аналіз потенційних загроз для мобільних пристроїв та розробка ефективних методів захисту від несанкціонованого втручання в їхню діяльність є вкрай актуальним напрямом досліджень в сучасному цифровому суспільстві. Зі зростанням популярності мобільних пристроїв серед всіх верств населення та застосуванням їх майже у всіх сферах життя водночас зростає чутливість користувачів засобів мобільного зв'язку до небезпек з боку різноманітних загроз у кіберпросторі, зокрема вірусів, шкідливих програм, атак на конфіденційні дані тощо, на все те, що може серйозно нашкодити та підірвати довіру до безпеки інформаційних каналів.

Одним із важливих кроків на шляху до забезпечення захисту мобільного зв'язку є аналіз сучасних загроз для мобільних пристроїв, враховуючи їхню різноманітність та ступінь складності. Визначення та вивчення різних типів вірусів, методів атак та виявлення потенційних вразливостей мережі та використовуваних девайсів є ключовим етапом для розробки ефективних методів їхнього захисту.

Іншим важливим моментом для безпечної комунікації в мобільній мережі є розробка методів захисту, спрямованих на зменшення вразливостей та мінімізацію ризиків для користувачів мобільних пристроїв. Використання опенсорсних інструментів у цьому контексті відкриває можливості для створення ефективних рішень, які поєднують у собі високий рівень безпеки та гнучкість у налаштуваннях. У межах дослідження загроз для мобільних пристроїв та розробки ефективних методів захисту планується провести детальний аналіз різноманітних методів та засобів, спрямованих на запобігання та протидію кіберзагрозам [1].

Однією з ключових задач дослідження є аналіз сучасних антивірусних програм та методів виявлення шкідливих програм для мобільних пристроїв. Особлива увага при цьому приділяється ефективності антивірусних движків та їхній здатності реагувати на нові види загроз. Не менш важливим завданням дослідження є вивчення методів біометричної аутентифікації та їх відповідності умовам мобільного використання. Також проводиться огляд систем керування доступом з метою визначення їхньої придатності для захисту від несанкціонованого доступу.

Вивчення сучасних алгоритмів шифрування для забезпечення конфіденційності та цілісності інформації на мобільних пристроях та оцінка їхньої якості допоможе обрати найкращий варіант у кожному конкретному випадку. Системи захисту даних від втрати та крадіжки, зокрема шифрування файлів та комунікацій дозволить підвищити ступінь безпеки зв'язку та позбавити користувачів від реалізації загроз кіберзлочинців.

Аналіз ефективності фаєрволів для мобільних пристроїв та їх роль у захисті від зовнішніх атак, а також вивчення систем виявлення вторгнень та їхньої здатності вчасно реагувати на потенційні загрози є іншими важливими аспектами дослідження [2].

Всі методи дослідження спрямовані на отримання кращого розуміння сучасних підходів до забезпечення безпеки мобільних пристроїв та розробку практичних рекомендацій у сфері кібербезпеки. Загальне значення роботи полягає у створенні стійких, ефективних та надійних методів захисту мобільних пристроїв, що сприятиме забезпеченню конфіденційності та цілісності інформації, яка обробляється на цих пристроях.

Таким чином, основний матеріал дослідження орієнтується на вивчення і аналіз сучасних опенсорсних інструментів, призначених для захисту мобільних пристроїв. Наразі існує множина відкритих (opensource) інструментів, призначених для захисту мобільних пристроїв.

Так, Haven є відкритим застосунком, який за бажанням власника перетворює його смартфон на персональний домашній монітор, який за допомогою датчиків, камери та інших функцій мобільного пристрою може стежити за оточенням. Застосунок використовує вбудовані сенсори смартфона, такі як акселерометр і датчик світла, для виявлення будь-якого руху в області навколишнього простору. Haven може реагувати на сторонні звуки, наприклад, виявляти звуки проникнення або інші непередбачені події. На разі виявлення будь-якої підозрілої активності, Haven сповіщає про це за допомогою повідомлень або записів аудіо та фото з використанням камери пристрою. До того ж застосунок використовує шифрування для захисту записаних даних та гарантує їх конфіденційність. Дані, що збираються та оброблюються застосунком Haven можна керувати локально з пристрою, що також сприяє більш високому рівню конфіденційності [3, 4].

WireGuard - це відкритий VPN-тунель, який забезпечує приватність та безпеку з'єднання для мобільних пристроїв. Відомий своєю швидкістю та простотою використання; намагається уникати зайвих шарів абстракції, що сприяє зменшенню накладних витрат та підвищенню продуктивності. WireGuard використовує сучасні криптографічні примітиви та протоколи для безпечної передачі даних. Створений з урахуванням безпеки та уникнення пасток, що часто спостерігаються у старіших протоколах. Застосунок WireGuard призначений для використання в різних сценаріях, зокрема віддалений доступ, мережеві тунелі та інші випадки. Може легко інтегруватися в різноманітні середовища та продовжує активно розвиватися [5].

Signal є месенджер з відкритим вхідним входом, який забезпечує шифрування end-to-end для повідомлень, дзвінків та відеодзвінків. Відомий рівнем захисту своїх користувачів, безпечний та надійний, його вихідний код доступний для перевірки. За рівнем безпеки перевищує Viber та WhatsApp.

Використовуючи відкриті технології, такі інструменти можуть забезпечити ефективний рівень безпеки та враховувати високий ступінь гнучкості для користувачів. Проводиться докладний огляд функціоналу та можливостей опенсорсних антивірусів, файрволів, систем виявлення вторгнень та інших інструментів, спрямованих на захист мобільних пристроїв [1].

В результаті дослідження формулюються рекомендації з ефективного використання опенсорсних інструментів для захисту мобільних пристроїв, враховуючи специфіку їхнього застосування та потреби користувачів. Висновки роботи підкреслюють значущість опенсорсних рішень у забезпеченні безпеки інформації та пропонують перспективи подальших досліджень у цьому напрямку.

Література

[1] Войтович О.П. Засіб моніторингу операційної системи Android //Науковий журналі «Вісник Хмельницького національного університету».- Київ, 2017.- С.35-46.

[2] Куперштейн Л.М. Базно-орієнтований підхід до захисту даних в операційній системі Android// Науковий журналі «Вісник Хмельницького національного університету».- Київ, 2018.- С.51-62.

[3] Офіційний веб-сайт Guardian Project [Електронний ресурс]. – Режим доступу до ресурсу: https://guardianproject.info/apps/org.havenapp.main/

[4] Офіційний веб-сайт Freedom of the Press Foundation [Електронний ресурс]. – Режим доступу до ресурсу: https://freedom.press/

[5] Офіційний веб-сайт Wireguard [Електронний ресурс]. – Режим доступу до ресурсу: https://www.wireguard.com

ОЦІНКА РИЗИКІВ КІБЕРБЕЗПЕКИ ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ

Пуценко Д.С.
Керівник: Лимаренко В.В.
E-mail: danil.putzenko.0@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Мобільні пристрої стали невід'ємною частиною нашого життя. Ми використовуємо їх для роботи, спілкування, розваг та зберігання особистої інформації. Зростання залежності від мобільних пристроїв робить їх привабливою мішенню для кіберзлочинців [1].

Ризики кібербезпеки:

– втрата або крадіжка: мобільні пристрої легко втратити або вкрасти. Це може призвести до крадіжки особистої інформації, паролів та фінансових даних;

– шкідливі програми: мобільні пристрої можуть бути заражені шкідливими програмами, які можуть вкрасти дані, пошкодити програмне забезпечення або відстежувати ваше місцезнаходження;

– фішинг: кіберзлочинці можуть використовувати фішингові атаки, щоб обманом змусити вас розкрити особисті дані або завантажити шкідливі програми;

– несанкціонований доступ: мобільні пристрої можуть бути вразливими до несанкціонованого доступу, якщо не використовувати надійні паролі та методи шифрування.

Оцінка ризиків. Щоб оцінити ризики кібербезпеки для мобільних пристроїв, важливо врахувати наступні фактори:

– тип пристрою: Деякі мобільні пристрої більш вразливі до кібератак, ніж інші.

– операційна система: Деякі операційні системи більш безпечні, ніж інші.

– використання пристрою: Як ви використовуєте свій мобільний пристрій?

– зберігання даних: Які дані ви зберігаєте на своєму мобільному пристрої?

Методи зниження ризиків:

– використовуйте надійні паролі та методи шифрування.

– встановлюйте оновлення програмного забезпечення.

– завантажуйте програми лише з надійних джерел.

– будьте обережні з посиланнями та вкладеннями в електронних листах і текстових повідомленнях.

– використовуйте антивірусне програмне забезпечення.

– регулярно створюйте резервні копії даних.

Література:

[1] Apache Log4j Vulnerability Guidance [Електронний ресурс] – Режим доступу до ресурсу: https://www.cisa.gov/uscert/apache-log4j-vulnerability-guidance

ВИКОРИСТАННЯ ВІДКРИТОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ЕФЕКТИВНОСТІ СМАРТ-КОНТРАКТІВ У ПРОЦЕСАХ БЕЗПЕКИ

Пчолка В.Е.
Керівник: Венгріна О.С.
E-mail: s.pchelka03@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

У сучасному цифровому світі, де безпека даних та автоматизація бізнес-процесів набувають все більшої важливості, смарт-контракти виступають як ключовий інструмент для досягнення цих цілей. Смарт-контракти, що є самовиконуючимися контрактами з вбудованими умовами виконання, забезпечують високий рівень надійності та прозорості у цифрових транзакціях. Застосування відкритого програмного забезпечення у розробці смарт-контрактів відкриває нові можливості для підвищення їх ефективності та безпеки. Відкрите

програмне забезпечення надає спільноті розробників гнучкість у вдосконаленні та адаптації смарт-контрактів до специфічних потреб користувачів, а також сприяє підвищенню їх безпеки через постійні перевірки та оновлення спільнотою. Цей підхід може відіграти важливу роль у розвитку цифрової економіки, забезпечуючи більш ефективні та безпечні процеси для підприємств та індивідуальних користувачів.

В даному дослідженні розглянуто відкрите програмне забезпечення, яке може включати різні платформи та інструменти та які використовуються для розробки, тестування й реалізації смарт-контрактів, а саме:

– Ethereum – найвідоміша платформа для створення смарт-контрактів, вона використовує мову програмування Solidity для написання контрактів.

– Hyperledger Fabric – проект від Linux Foundation, що забезпечує інфраструктуру для розробки рішень на базі блокчейну, включаючи смарт-контракти.

– Truffle Suite – набір інструментів для розробки Ethereum, що дозволяє легко тестувати та розгортати смарт-контракти.

– OpenZeppelin – бібліотека безпечних смарт-контрактів, яка надає стандартизовані, перевірені контракти для використання у розробці.

– Ganache – частина Truffle Suite, що надає локальне тестове середовище блокчейну для розробки та тестування смарт-контрактів.

– Remix IDE – браузерне середовище для написання, тестування та розгортання смарт-контрактів на Ethereum.

Ці інструменти та платформи є відкритими, що означає, що вони доступні для використання, модифікації та розповсюдження спільнотою. Використання відкритого програмного забезпечення у контексті смарт-контрактів дозволяє розробникам використовувати готові рішення та адаптувати їх під конкретні потреби, сприяючи інноваціям та підвищуючи загальну безпеку системи.

Одним з ключових аспектів розробки смарт-контрактів є вибір відповідного програмного забезпечення, яке забезпечує не тільки необхідні функціональні можливості, але й відповідає стандартам безпеки та ефективності. Нижче наведено таблицю (табл.1), яка демонструє перелік популярних платформ та інструментів для розробки смарт-контрактів, їх основні переваги та потенційні недоліки, що дозволяє глибше зрозуміти їхню придатність для конкретних задач.

Таблиця 1. – Переваги та недоліки платформ та інструментів для розробки смарт-контракті

Програмне забезпечення	Переваги	Недоліки
Ethereum	Велика спільнота, універсальність, підтримка Solidity	Складність масштабування, проблеми з ефективністю
Програмне забезпечення	Переваги	Недоліки
Hyperledger Fabric	Гнучкість, конфіденційність, підтримка різних мов програмування	Вимагає певних технічних знань, менш популярний
Truffle Suite	Інтегровані інструменти для розробки, тестування та розгортання	Може бути складним для новачків
OpenZeppelin	Стандартизовані безпечні контракти, зменшення ризиків	Обмежена гнучкість, залежить від сторонніх контрактів
Ganache	Легке локальне тестування, швидка налаштування	Тільки для Ethereum, не підходить для продакшну
Remix IDE	Просте веб-середовище для написання та тестування контрактів	Обмежені функціональні можливості порівняно з повноцінними IDE

Підсумовуючи, роль відкритого програмного забезпечення у розвитку та впровадженні смарт-контрактів є надзвичайно важливою. Використання таких платформ та інструментів, як Ethereum, Hyperledger Fabric, Truffle Suite, OpenZeppelin, Ganache, та Remix IDE, дозволяє розробникам створювати більш безпечні, ефективні та адаптивні смарт-контракти. Це сприяє не лише технічному прогресу в галузі блокчейн технологій, але й забезпечує більш міцну основу для розвитку цифрової економіки. Таким чином, відкрите програмне забезпечення відіграє ключову роль у підвищенні загальної ефективності та безпеки смарт-контрактів, що є невід'ємною частиною сучасного цифрового ландшафту.

Литература

[1] Ethereum. Ethereum Foundation. [Електронний ресурс] – Режим доступу до ресурсу: https://www.ethereum.org/

[2] Hyperledger Fabric. The Linux Foundation. [Електронний ресурс] – Режим доступу до ресурсу: https://www.hyperledger.org/use/fabric

[3] Truffle Suite. Trufflesuite. [Електронний ресурс] – Режим доступу до ресурсу: https://www.trufflesuite.com/

[4] OpenZeppelin. OpenZeppelin. [Електронний ресурс] – Режим доступу до ресурсу: https://openzeppelin.com/

[5] Ganache. Trufflesuite. [Електронний ресурс] – Режим доступу до ресурсу: https://www.trufflesuite.com/ganache

[6] Remix - Ethereum IDE. Remix. [Електронний ресурс] – Режим доступу до ресурсу: https://remix.ethereum.org/

АНАЛІЗ ТА ПОКРАЩЕННЯ ЗАХОДІВ КІБЕРБЕЗПЕКИ В МЕДИЧНИХ ІНФОРМАЦІЙНИХ СИСТЕМАХ

Рева В.О
Керівник: Старкова О.В.
E-mail: slavik.reva327@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Медичні інформаційні системи (МІС) є важливими елементами сучасної медичної практики, які сприяють збереженню, обробці та обміну медичною інформацією для поліпшення якості надання медичних послуг та збереження здоров’я пацієнтів. МІС містять величезний обсяг конфіденційних даних про пацієнтів, включаючи їхню медичну історію, діагнози, результати обстежень та інші особисті дані. Необхідність захисту медичних даних від несанкціонованого доступу, крадіжок та руйнування стає надзвичайно важливою, оскільки порушення конфіденційності може призвести до серйозних наслідків для пацієнтів, включаючи фінансові втрати, репутаційну шкоду та навіть загрози для їхнього здоров’я та життя [1].

Проте зростаюча комп’ютеризація та цифровізація в медичній сфері також призводять до збільшення загроз кібербезпеці. Зростаюча кількість кібератак та загроз кібербезпеці, спрямованих на медичні установи та інформаційні системи, створює серйозні загрози для конфіденційності, цілісності та доступності медичних даних. Атаки з метою отримання доступу до медичних даних, втручання в роботу систем або викрадення конфіденційної інформації стають дедалі більшими загрозами. Тому аналіз та покращення заходів кібербезпеки в медичних інформаційних системах є невідкладним завданням, спрямованим на забезпечення безпеки та захисту медичних даних у цьому чутливому секторі [1].

Для забезпечення кібербезпеки в МІС необхідно враховувати різні рівні захисту, такі як фізичний, логічний, адміністративний та правовий. Фізичний рівень захисту включає в себе захист від несанкціонованого фізичного доступу до медичних пристроїв, серверів, мережевого обладнання та інших компонентів МІС. Логічний рівень захисту включає в себе

захист від несанкціонованого логічного доступу до медичних даних, програмного забезпечення, операційних систем та інших ресурсів МІС. Адміністративний рівень захисту включає в себе захист від несанкціонованого адміністративного доступу до налаштувань, конфігурацій, політик та інших параметрів МІС. Правовий рівень захисту включає в себе захист від несанкціонованого правового доступу до медичних даних, що може бути здійснений за допомогою судових рішень, договорів, законів та інших нормативних актів [2].

Кожен з цих рівнів захисту вимагає відповідних заходів кібербезпеки, які можуть бути технічними, організаційними та правовими. Технічні заходи можуть включати удосконалення аутентифікації та авторизації користувачів, впровадження механізмів шифрування для захисту даних в покоївках та під час передачі, вдосконалення систем виявлення та відновлення після інцидентів безпеки. Організаційні заходи включають в себе навчання персоналу щодо правил безпеки, регулярні огляди та аудити безпеки, розробку та впровадження політик безпеки і ризиків. Правові заходи включають в себе дотримання відповідних законів та регулятивних вимог, таких як GDPR у Європейському Союзі або HIPAA в Сполучених Штатах [2].

Одним із сучасних технологічних рішень, які можуть покращити кібербезпеку в медичних інформаційних системах, є використання блокчейну. Блокчейн – це розподілена база даних, яка зберігає інформацію в ланцюжку блоків, які захищені криптографічно. Блокчейн може забезпечити високий рівень прозорості, незмінності та децентралізації медичних даних, що може запобігти їхньому підробленню, втраті або зловживанню [3].

Блокчейн також може полегшити обмін медичною інформацією між різними сторонами, такими як пацієнти, лікарі, страхові компанії та інші, з дотриманням правил конфіденційності та згоди. Блокчейн може бути використаний для створення електронних медичних записів, електронних рецептів, електронних сертифікатів, електронних контрактів та інших документів, пов’язаних з медичною сферою [3].

Іншим сучасним підходом, який може покращити кібербезпеку в медичних інформаційних системах, є використання штучного інтелекту (ШІ). ШІ – це галузь науки, яка займається створенням інтелектуальних систем, які можуть виконувати завдання, які зазвичай вимагають людського розуму. ШІ може бути використаний для підвищення ефективності та якості медичних послуг, таких як діагностика, прогнозування, лікування, моніторинг, підтримка прийняття рішень та інші. ШІ також може бути використаний для підвищення рівня кібербезпеки в медичних інформаційних системах, наприклад, для виявлення та запобігання кібератак, для аналізу та класифікації медичних даних, для адаптації та оптимізації систем безпеки, для автоматизації та розподілу ресурсів безпеки та інші [4].

Однією з переваг ШІ є його здатність навчатися з даних та адаптуватися до змінних умов. ШІ може використовувати різні методи машинного навчання, такі як нейронні мережі, глибоке навчання, підсилювальне навчання, кластеризація, класифікація, регресія та інші, для аналізу великих обсягів медичних даних та виявлення закономірностей, аномалій, тенденцій та прогнозів . ШІ може також використовувати різні методи оптимізації, такі як генетичні алгоритми, штучні бджолині сім’ї, штучні імунні системи, штучні мурашині колонії та інші, для підбору оптимальних параметрів систем безпеки та розподілу ресурсів безпеки . ШІ може також використовувати різні методи експертних систем, такі як нечітка логіка, нейро-нечітка логіка, байєсовські мережі, системи, засновані на правилах, системи, засновані на знаннях та інші, для підтримки прийняття рішень та рекомендацій щодо заходів безпеки [4].

Загальне значення цього дослідження полягає у підвищенні рівня кібербезпеки в медичних інформаційних системах, що сприятиме захисту конфіденційності та цілісності медичних даних та запобігатиме можливим кібератакам. Узагальнюючи, аналіз та покращення заходів кібербезпеки в медичних інформаційних системах є важливим завданням, щоб забезпечити конфіденційність, цілісність та доступність медичної інформації

та запобігти можливим кібератакам, які можуть мати серйозні наслідки для пацієнтів та медичних установ.

Література.

[1] Аналіз проблеми забезпечення кібербезпеки медичних комп'ютерних систем / О. Г. Трофименко, Я. В. Дубовой, Н. І. Логінова, Ю. В. Прокоп, О. В. Задерейко // Захист інформації. – Київ : Національний авіаційний університет. – 2021. – Т. 23. – № 1. – С. 30-39. DOI: 10.18372/2410-7840.23.15153.

[2] Стрєлкіна А. А., Узун Д. Д. Забезпечення кібербезпеки медичних систем: виклики і рішення в контексті Інтернету речей // Радіоелектронні і комп’ютерні системи, 2017, № 1 (81) – С. 44-50. DOI: https://doi.org/10.32620/reks.2017.1.05.

[3] Блокчейн у медицині [Електроний ресурс]. – Режим доступу до ресурсу: https://blog.whitebit.com/uk/blockchain-in-medicine/#heading-0.

[4] Омельченко С. О. Використання штучного інтелекту в медицині / С. О. Омельченко // Радіоелектроніка та молодь в XXI столітті : матеріали 26-го Міжнародного молодіжного форуму, 20-22 квітня 2022 р. – Харків : ХНУРЕ, 2022. – Т. 5. – С. 36–37. URI https://openarchive.nure.ua/handle/document/23626.

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ РОЗВІДКИ В СФЕРІ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

Резвін А.А.
Керівник:Мерлак О.В.
E-mail:andreyrezvin3@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Дослідження, що розглядається, є вкрай актуальним у сучасному цифровому світі, де загрози кібербезпеки постійно зростають. З розвитком технологій злочинці стають більш винахідливими, тож це вимагає від організацій удосконалення їхніх методів розвідки для запобігання кібератак та збереження конфіденційної інформації.

Актуальність дослідження ефективності розвідки в цій сфері обумовлена наступними чинниками [1]:

1. Зростання кіберзагроз: інформаційна безпека стає все важливішою для підприємств та організацій у зв’язку з зростанням кіберзагроз і кібератак.
2. Необхідність ефективної оборони: підприємства та установи потребують ефективних засобів розвідки для вчасного виявлення потенційних загроз і вчасної реакції на них.
3. Фінансові втрати: кібератаки можуть призвести до значних фінансових втрат, порушення діяльності підприємства та пошкодження репутації.
4. Регулятивні вимоги: багато секторів економіки підпорядковані строгим регулятивним вимогам стосовно захисту конфіденційної інформації, що вимагає відповідного рівня розвідки та захисту.
5. Швидкість змін: швидкість змін у технологіях та тактиках кіберзлочинців вимагає постійного оновлення методів розвідки для ефективного протидії новим загрозам.

Серед сучасних методів та інструментів розвідки, які використовуються для забезпечення безпеки інформаційних систем, найбільш ефективними є наступні [2, 3]:

1. Аналіз загроз та вразливостей: методи аналізу загроз та вразливостей дозволяють ідентифікувати потенційні загрози для інформаційної системи та визначати їхні слабкі місця.
2. Виявлення інцидентів безпеки: інструменти для виявлення інцидентів безпеки допомагають вчасно виявляти атаки або несправності в системі.
3. Моніторинг мережі: програмні та апаратні засоби моніторингу мережі дозволяють відстежувати активність в мережі та виявляти підозрілі дії.

4. Системи аналітики поведінки: інструменти аналітики поведінки допомагають виявляти аномальні патерни поведінки користувачів або системних процесів, що можуть свідчити про атаку або порушення безпеки.

5. Системи інтелектуальної аналітики: використання штучного інтелекту та машинного навчання для аналізу великих обсягів даних дозволяє виявляти нові загрози та прогнозувати їхній вплив на безпеку інформаційної системи.

6. Інструменти кіберзахисту: використання спеціалізованих інструментів для захисту інформаційних систем від різних видів атак, таких як файрволи, антивіруси, системи виявлення вторгнень та інші.

Актуальність дослідження ефективності розвідки в сфері інформаційної безпеки включає також розгляд правових аспектів збору та використання інформації. Важливість цього аспекту пов’язана з наступними причинами [4, 5]:

1. Забезпечення конфіденційності та приватності: вимоги до збору та використання інформації повинні бути відповідні законодавству з питань конфіденційності та приватності, щоб забезпечити захист прав користувачів.

2. Законність доказів: інформація, зібрана за допомогою розвідки, може використовуватися в правових процедурах, тому важливо, щоб збір цієї інформації був проведений в рамках відповідних правових норм.

3. Відповідність регуляторним вимогам: багато секторів економіки підпорядковані строгим регулятивним вимогам стосовно збереження та використання конфіденційної інформації. Дотримання цих вимог є обов’язковим для уникнення штрафів та інших правових наслідків.

4. Захист від порушень законодавства: недодержання правових норм під час збору та використання інформації може призвести до правових проблем для організацій, включаючи скарги користувачів, штрафи та інші правові наслідки:

Під час проведення дослідження пропонується задіяти наступні методи.

1. Колекція інформації: дослідження включатиме аналіз вже існуючих даних, а також активний збір нової інформації про потенційні загрози та вразливості.

2. Аналіз: оцінка зібраної інформації для виявлення трендів, вразливостей та можливих сценаріїв атак.

3. Моделювання: використання математичних та статистичних моделей для передбачення можливих наслідків інцидентів та оцінки ризиків.

4. Експерименти: проведення симуляцій та тестування заходів безпеки для оцінки їхньої ефективності в реальних умовах.

Література

[1] Journal of cybersecurity-Oxford Academic [Електронний ресурс]. – Режим доступу до ресурсу: https://academic.oup.com/cybersecurity

[2] COSE|Computer& Security|Journal [Електронний ресурс]. – Режим доступу до ресурсу: https://www.sciencedirect.com/journal/computers-and-security

[3] IEEE Xplore [Електронний ресурс]. – Режим доступу до ресурсу: https://ieeexplore.ieee.org/Xplore/home.jsp

[4] ACM Digital Library [Електронний ресурс]. – Режим доступу до ресурсу: https://dl.acm.org/subject/is

[5] ScienceDirect.com[Електронний ресурс]. – Режим доступу до ресурсу: https://www.sciencedirect.com/

ЗАСТОСУВАННЯ ВІДКРИТИХ IDS/IPS ТА АНАЛІЗАТОРІВ МЕРЕЖ ДЛЯ ЗАБЕЗПЕЧЕННЯ КІБЕРБЕЗПЕКИ

Сивуха А.Л.
Керівник: Венгріна О.С.
E-mail: anastasiasivuha7@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

У сучасному цифровому світі, де кіберзагрози стають все більш витонченими та небезпечними, забезпечення кібербезпеки є надзвичайно важливим завданням. Кібератаки можуть завдати серйозної шкоди організаціям та індивідуальним користувачам, а тому необхідно мати ефективні засоби виявлення та запобігання таким загрозам. У цьому контексті використання відкритих IDS/IPS (систем виявлення та запобігання вторгнень) та аналізаторів мережевого трафіку стає актуальною стратегією для забезпечення кібербезпеки.

Використання відкритих IDS/IPS, таких як Snort, Suricata, та аналізаторів мережевого трафіку, як Wireshark, Bro/Zeek, та системи виявлення вторгнень на рівні хоста, як OSSEC, відкриває можливості для глибокого аналізу мережевої активності та виявлення потенційних загроз. Ці інструменти є безкоштовними та відкритими для спільноти, що робить їх доступними для багатьох організацій та користувачів.

У рамках розробки та впровадження стратегій кібербезпеки, особливе значення набуває використання спеціалізованих інструментів для аналізу та моніторингу мережевих процесів. Ці інструменти надають можливість для глибокого розуміння мережевої активності, виявлення аномалій та протидії потенційним кіберзагрозам. Оскільки відкриті джерела відіграють ключову роль у розвитку та адаптації технологій кібербезпеки, низка безкоштовних інструментів, доступних на ринку, стають незамінними помічниками для фахівців у цій галузі. Серед таких інструментів можна виділити наступні:

1. Snort - відкритий та безкоштовний IDS/IPS, який може слугувати основою для аналізу поведінки трафіку в мережі.
2. Wireshark – безкоштовний аналізатор мережевого трафіку, що може бути використаний для моніторингу та аналізу пакетів, які проходять через мережеві вузли.
3. Bro/Zeek – потужна система моніторингу мережі, яка забезпечує детальний аналіз мережевих з'єднань.
4. Suricata – також відкритий IDS/IPS, який працює в багатопоточному режимі і підтримує сучасні функції аналізу мережевого трафіку.
5. OSSEC – система виявлення вторгнень, яка працює на рівні хоста і забезпечує глибокий аналіз системних логів та активності.

Враховуючи постійне зростання складності та витонченості кіберзагроз, відкриті IDS/IPS системи та мережеві аналізатори відіграють ключову роль у захисті інформаційних систем. Інструменти, такі як Snort, Wireshark, Bro/Zeek, Suricata, та OSSEC, не тільки надають гнучкість та адаптивність для виявлення та реагування на потенційні загрози, але й сприяють розвитку спільноти експертів з кібербезпеки через обмін знаннями та досвідом. Ці інструменти дозволяють ефективно аналізувати мережеву активність, виявляти аномалії, а також розробляти та тестувати нові методи захисту, що є критично важливим для адаптації до постійно змінюваного цифрового ландшафту. Таким чином, використання відкритих IDS/IPS систем та мережевих аналізаторів є фундаментальним для створення міцної та ефективної стратегії кібербезпеки, здатної витримувати сучасні та майбутні виклики в цій галузі.

СТВОРЕННЯ ЗАХИЩЕНОГО МЕСЕНДЖЕРА З END-TO-END ШИФРУВАННЯМ ДЛЯ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ ТА КОНФІДЕНЦІЙНОСТІ ОСОБИСТОЇ ПЕРЕПИСКИ

Сирбу А.В.
Керівник: Леуненко О.В.
E-mail: ttest202012@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

У сучасному світі, коли ми все більше спілкуємося через месенджери, безпека та конфіденційність нашої особистої переписки стають дедалі важливішими. Чи знаєте ви, хто може читати ваші повідомлення у популярних месенджерах, таких як Viber та WhatsApp? Згідно зі заявами голови WhatsApp, ніхто не може читати ваші особисті повідомлення завдяки наскрізному шифруванню (end-to-end encryption) [1]. Однак, MediaSapiens провів аналіз того, як ці месенджери захищають нашу приватність. Засновник Viber Телмон Марко заявив, що компанія не має можливості прослуховувати розмови, а повідомлення зберігаються протягом двох тижнів або поки вони не відкриті одержувачем. Близько 80% повідомлень видаляються швидше, ніж за секунду [1]. Таким чином, Viber забезпечує високий рівень захисту і конфіденційності.

Однак, у світі, де війна та конфлікти стають все більш поширеними, безпека спілкування стає особливо важливою. Деякі месенджери, такі як Telegram та Viber, надають додаткові можливості для захисту особистої переписки. Наприклад, Telegram пропонує можливість створення секретних чатів з автоматичним видаленням повідомлень та можливістю встановлення паролю для доступу до облікового запису [2]. Це дозволяє зберегти конфіденційність і захистити ваші повідомлення від несанкціонованого доступу.

У світі, де наша особиста інформація є цінним активом, важливо обирати месенджери, які забезпечують найвищий рівень захисту та конфіденційності. Один з таких месенджерів - Telegram. Особливістю Telegram є його end-to-end шифрування, що означає, що ваші повідомлення залишаються приватними та недоступними для сторонніх осіб. Ще одним безпечним месенджером є Signal. Цей месенджер також пропонує end-to-end шифрування та забезпечує конфіденційність вашої комунікації [3]. Signal був розроблений американським дослідником з кібербезпеки та використовує технологію, яка робить ваші повідомлення незрозумілими для будь-кого, окрім вас та отримувача [3]. Крім того, ці месенджери надають додаткові функції, такі як можливість встановлення паролю для доступу до додатку та автоматичне видалення повідомлень, що дозволяє зберегти вашу приватність [4].

Отже, вибір захищеного месенджера з end-to-end шифруванням є важливим для забезпечення безпеки та конфіденційності вашої особистої переписки. Ви можете обрати месенджер, який надає додаткові функції для захисту вашої приватності, такі як автоматичне видалення повідомлень та можливість встановлення паролю для доступу до додатку. Не забувайте, що ваша особиста інформація є цінним активом, і ви маєте право на її захист [2][4].

Література

[1] Хто може читати повідомлення у Viber та WhatsApp? [Електронний ресурс]. – Режим доступу: https://ms.detector.media/kiberbezpeka/post/16325/2016-03-28-khto-mozhe-chytaty-povidomlennya-u-viber-ta-whatsapp/

[2] DOU. Спілкування під час війни: Telegram, Viber чи щось краще [Електронний ресурс]. – Режим доступу: https://dou.ua/forums/topic/37416/

[3] Українські експерти назвали топ безпечних месенджерів - Mind.ua [Електронний ресурс]. – Режим доступу: https://mind.ua/news/20239355-onovleno-ukrayinski-eksperti-nazvali-top-bezpechnih-mesendzheriv

[4] "Google" та "WhatsApp": міжнародна сторона втручання у приватне спілкування [Електронний ресурс]. – Режим доступу: https://yur-gazeta.com/publications/practice/inshe/google-ta-whatsapp-mizhnarodna-storona-vtruchannya-u-privatne-spilkuvannya-.html

АНАЛІЗ РИЗИКІВ І НАСЛІДКІВ МЕРЕЖЕВИХ АТАК ВЕБ-САЙТІВ ДЛЯ БІЗНЕСУ

Стасюк К.В.
Керівник: Старкова О.В.
E-mail: StasuikKyrylo@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Управління вразливостями - ключовий елемент захисту інформації: за даними Державного центру кіберзахисту, у 2022 році було зареєстровано у 2,8 раза більше кіберінцидентів, ніж у 2021 році. Підходи до управління вразливостями різні, але мають спільну мету - знизити ризик втрати або пошкодження інформації. Управління вразливостями є невід'ємною частиною ефективної стратегії кібербезпеки для будь-якої компанії на шляху цифрової трансформації. Це особливо важливо для таких галузей, як фінанси і банківська справа, сільське господарство, важка промисловість, обробна промисловість, енергетика і комунальні послуги, роздрібна торгівля, транспорт і державне управління.

Підприємства, схильні до ризику кібератак, можуть постраждати від втрати конфіденційної інформації, фінансових втрат і шкоди репутації. Крім того, існує ризик переривання діяльності та значних збитків унаслідок атак на інфраструктуру. Наприклад, у банківському секторі загрози включають атаки з метою отримання конфіденційної інформації про клієнтів, а в залізничному секторі - атаки на автоматизовані системи управління поїздами. Щоб запобігти таким наслідкам, організаціям необхідно впровадити комплексну програму управління вразливостями і підтримувати її в актуальному стані.

Для цього необхідно зробити низку кроків:

– розробити політику безпеки і план реагування на інциденти; провести інвентаризацію інформаційних активів, оцінити пов'язані з ними ризики та відстежувати зміни;

– виявити й оцінити потенційні загрози, визначити вразливості та зрозуміти, які вразливості наразі використовуються зловмисниками. Важливо також зазначити, що 2022 року було виявлено 25 080 вразливостей, що на 18,78 % більше, ніж 2021 року, а цьогоріч було виявлено понад 11 000 нових вразливостей, до того ж щодня з'являються нові методи й засоби експлуатації відомих вразливостей;

– впровадження ефективних процесів виправлення вразливостей і оцінки результатів. Це включає в себе ефективне визначення пріоритетів вразливостей, щоб зосередити ресурси команди на тих вразливостях, які становлять найбільший ризик; впровадження та підтримання в актуальному стані заходів інформаційної безпеки; постійне вдосконалення програми управління вразливостями шляхом відстеження змін в інфраструктурі та безперервного розвитку загроз;

– для аналізу та виявлення вразливостей використовують різні інструменти: Мережеві сканери - це програмні інструменти, які виявляють слабкі місця в системі та мережі на основі сканування портів та аналізу реакції системи;

– сканери веб-додатків використовуються для перевірки веб-сайтів і додатків.

Ці інструменти можуть виявляти вразливості за списком OWASP TOP 10; Сканери, що виявляють уразливості в ресурсах у хмарі або контейнерах; Сканери для виявлення вразливостей на пристроях за межами організації, наприклад на віддалених робочих станціях. Для аналізу виявлених вразливостей використовується Mitre Att&ck.

Література.

[1] Netwave [Електронний ресурс]. – Режим доступу до ресурсу: https://netwave.ua/cybersecurity-in-business/.

[2] Attack.Mitre [Електронний ресурс]. – Режим доступу до ресурсу: https://attack.mitre.org/.

ЗАСТОСУВАННЯ МЕТОДІВ ГЕЙМІФІКАЦІЇ ДЛЯ ПІДВИЩЕННЯ РІВНЯ ОБІЗНАНОСТІ СПІВРОБІТНИКІВ ПІДПРИЄМСТВА З ПИТАНЬ КІБЕРБЕЗПЕКИ

Толстик О.А.
Керівник: Старкова О.В.
E-mail: lesatolstik@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Питання забезпечення кібербезпеки є важливими для компаній будь-якого розміру. З часом об'єм даних збільшується, разом із тим збільшується кількість користувачів, що віддалено працюють і спілкуються. Кіберзлочинці розробляють складні методи, щоб отримувати доступ до ресурсів, викрадати дані, саботувати роботу компаній або вимагати гроші. Щороку кількість атак збільшується, а зловмисники розробляють нові методи для уникнення виявлення [1, 2]. Ці атаки можуть мати руйнівний вплив на бізнес, спричиняючи фінансові втрати, втрати репутації та навіть юридичну відповідальність.

Поінформованість співробітників про кібербезпеку є критично важливим компонентом стійкості компаній. Співробітники відіграють ключову роль у захисті конфіденційної інформації та захисті від кіберзагроз. Однак для багатьох підприємств залишається складним завданням формування міцного розуміння практик кібербезпеки серед співробітників.

Гейміфікація – це процес введення ігрових елементів в діяльність організації на різних рівнях – від окремого фахівця, відділів, проєктних команд – до підприємств, установ, громадських організацій в цілому [3].

В результаті застосування елементів і принципів ігрового дизайну в неігрових контекстах, гейміфікація є потужним інструментом у сфері навчання кібербезпеці. Інтегруючи такі елементи, як конкуренція, винагороди та інтерактивні завдання, у навчальні програми з кібербезпеки, компанії можуть ефективно залучати працівників і надавати їм можливість розпізнавати потенційні загрози та реагувати на них [3].

Є декілька ключових аспектів впровадження гейміфікації у навчальний процес для підвищення рівня обізнаності працівників з питань кібербезпеки: створити цікаві, захоплюючі навчальні сценарії, які викликають інтерес у працівників; створити симуляції кібератак, вправи з виявлення вразливостей та інші практичні завдання; використовувати системи відстеження прогресу; сприяти створенню спільноти працівників, які навчаються кібербезпеці; адаптувати методи до різних рівнів навчальної підготовки та стилів навчання працівників.

Наведений перелік не є вичерпним, адже теорія та практика застосування гейміфікації при навчання постійно розвивається та удосконалюється.

Література.

[1] Звіт про роботу у 2022 р. Оперативний центр реагування на кіберінциденти, Державний центр кіберзахисту, Державна служба спеціального зв'язку та захисту інформації України [Електронний ресурс]. – Режим доступу до ресурсу: https://scpc.gov.ua/api/docs/4eeb6a10-b7aa-4396-8b04-e0e4b7fca1ba/4eeb6a10-b7aa-4396-8b04-e0e4b7fca1ba.pdf.

[2] Звіт про роботу у 2023 р. Оперативний центр реагування на кіберінциденти, Державний центр кіберзахисту, Державна служба спеціального зв'язку та захисту інформації України [Електронний ресурс]. – Режим доступу до ресурсу: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwiooaS0w5uEAxU1S_EDHWpxC8wQFnoECCgQAQ&url=https%3A%2F%2Fcip.gov.ua%2Fservices%2Fcm%2Fapi%2Fattachment%2Fdownload%3Fid%3D53656&usg=AOvVaw0sLZQzggIloHvo9_7fCpOW&opi=89978449.

[3] Навчаємося граючи. Що таке гейміфікація [Електронний ресурс]. – Режим доступу до ресурсу: https://buki.com.ua/news/scho-take-geimifikatsiia/.

ЗАСТОСУВАННЯ ТЕХНОЛОГІЙ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ВИЯВЛЕННЯ ТА УПРАВЛІННЯ АНТРОПОГЕННИМИ ВИКЛИКАМИ

Точилкін М.І.
Керівник: Шаповалова О.О.
E-mail: Mykyta.Tochylkin@hneu.net, olena.shapovalova@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Штучний інтелект (ШІ) стає все більш важливим і ефективним інструментом для вирішення проблем в сфері інформаційної безпеки через свої можливості в аналізі даних, виявленні вразливостей та в управлінні ризиками. Актуальність теми обумовлена не тільки швидким розвитком інформаційних технологій, але і зростанням різноманітних технологічних загроз, які мають великий вплив на інформаційну безпеку. Сучасний світ високо цінить обмін інформацією, але це також робить системи та мережі більш вразливими перед різними видами загроз, зокрема тими, які виникають через людську діяльність. Таким чином, використання технологій штучного інтелекту стає критично важливим для ефективного виявлення та управління антропогенними викликами в інформаційній безпеці.

Аналіз функцій та можливостей штучного інтелекту в контексті виявлення та аналізу антропогенних загроз розкриває потужний потенціал цих технологій. Від семантичного аналізу текстів до комплексних моделей машинного навчання, ШІ може швидко і точно розпізнавати незвичайні патерни та аномалії у великих обсягах даних, що допомагає оперативно реагувати на потенційні загрози та запобігати їх поширенню. ШІ є рушійною силою будь-якої ефективної стратегії аналізу даних. Це потужний, ефективний і доступний спосіб обробки даних.

Для інтелектуальної обробки даних доцільно використовувати наступні інструменти:

– Julius AI, інтелектуальний інструмент аналізу даних, який інтерпретує, аналізує та візуалізує складні дані інтуїтивно зрозумілим і зручним способом. До його переваг можна віднести його здатність робити аналіз даних доступним і зрозумілим навіть для тих, хто не є спеціалістами в обробці даних або статистичних дослідженнях.

– Microsoft Power BI, один з найкращих інструментів ШІ для аналізу даних, корисна платформа бізнес-аналітики, яка дозволяє користувачам сортувати свої дані та візуалізувати їх для аналізу. До того ж, платформа забезпечує користувачів зручними інструментами для імпортування даних практично з будь-якого джерела з подальшою обробкою та формуванням запитів, звітів, дашбордів та інформаційних панелей (рис.1).

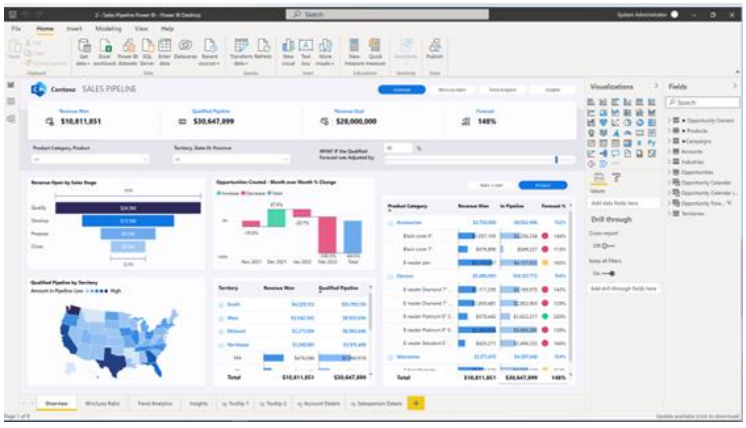


Рисунок 1. Результати роботи Microsoft Power BI

У контексті інформаційної безпеки важливо використовувати ефективні методи виявлення антропогенних викликів. Це може включати в себе не тільки технічні засоби, але й вивчення взаємодії користувачів з інформаційними системами та виявлення аномальних патернів в їх поведінці. Для цього, наприклад, можуть стати у пригоді такі інструменти як:

– IBM Security QRadar Suite, платформа для виявлення, розслідування та реагування на загрози, яка використовує розширений ШІ та автоматизацію для прискорення та оптимізації роботи аналітиків безпеки [1] (рис.2).

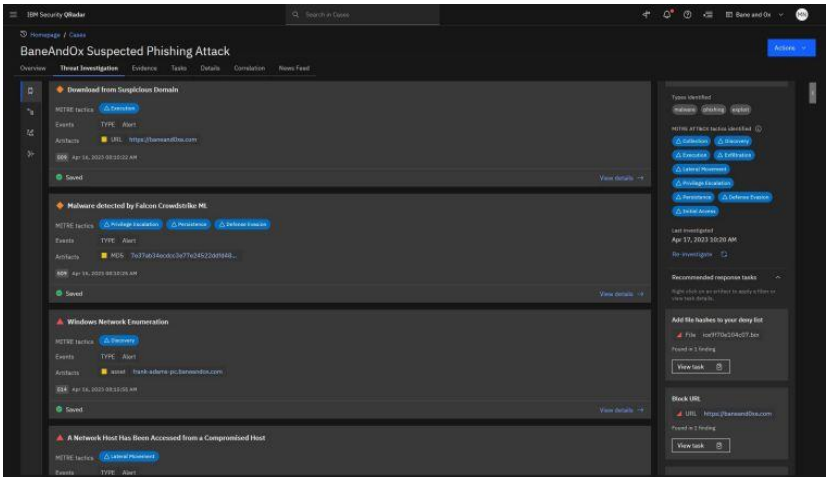


Figure 2. Вікно IBM Security QRadar Suite

– Fortinet, платформа, що пропонує рішення для мережевої безпеки з використанням ШІ для запобігання нульових денних загроз, що використовують наперед невідомі вразливості [2].

– Tessian, платформа для захисту від загроз, пов’язаних з електронною поштою, яке використовує ШІ для аналізу поведінки користувачів та контексту комунікації, щоб запобігти фішингу, шахрайству та витоку даних.

Наскільки важливим є управління антропогенними викликами з застосуванням технології ШІ, настільки ж важливим є розгляд можливості побудови систем, що сприяють взаємодії штучного інтелекту з людьми. Визначення етичних та правових аспектів використання ШІ в інформаційній безпеці стає ключовим завданням. Опис конкретних випадків впровадження технологій ШІ для виявлення та управління антропогенними викликами не лише розкриває практичні аспекти, але і дозволяє взяти до уваги виклики та нюанси, з якими можуть зіткнутися організації під час впровадження.

Можна констатувати, що використання технологій ШІ в інформаційній безпеці визначає нові стандарти та можливості для захисту від антропогенних загроз. Це стає не тільки питанням технічної безпеки, але й етики, правових аспектів та взаємодії з користувачами. Перспективи подальших досліджень та розвитку сфери застосування ШІ для управління антропогенними викликами в інформаційній безпеці відкривають широкий простір для інновацій та вдосконалення систем безпеки.

Література

[1] Бурлака В. І., Пометун О. А. Кібербезпека: проблеми та шляхи вирішення в Україні. Київ: Національний університет оборони України імені Івана Черняховського. 2019 – 256с.

[2] Григоренко Ю. М., Пилипчук В. І. Інтелектуальний аналіз даних в системах інформаційної безпеки. Київ: Національний технічний університет України "КПІ". 2019 – 354с.

[3] Мельник М. І. "Штучний інтелект: навчальний посібник." Київ: Видавництво "Каравела". 2018 – 453с.

[4] Goodfellow I., Bengio Y., & Courville A. Deep Learning. MIT Press. 2016 – 154p.

[5] Nilsson N. J. Artificial Intelligence: A New Synthesis. Morgan Kaufmann. 2014-654p.

[6] Russell S., & Norvig P. Artificial Intelligence: A Modern Approach. Pearson. 2018 – 384p.

ZILLYA! АНТИВІРУС – ЗАХИСТ КОМП'ЮТЕРА ВІД ВІРУСІВ, ТРОЯНІВ ТА ІНШИХ ШКІДЛИВИХ ПРОГРАМ

Чайка А.В.
Керівник: Сажко Г.І.
E-mail: wowalinachaika@gmail.com
Харків, Українська інженерно-педагогічна академія

Актуальність: в сучасному цифровому світі, де комп'ютери та інші електронні пристрої стали невід'ємною частиною повсякденного життя, захист від кіберзагроз стає критично важливим. Інтернет-простір насичений різноманітними загрозами, такими як віруси, троянські програми, шпигунське програмне забезпечення, рекламні атаки та інші форми шкідливих програм. Напади кіберзлочинців стають все більш складними та хитрими, існують нові техніки та методи атак, які можуть завдати значної шкоди як приватним користувачам, так і бізнесам. Крім того, збільшується кількість кіберзагроз, спрямованих на крадіжку особистої та фінансової інформації, а також на розповсюдження шкідливого програмного забезпечення для вимагання викупу.

У зв'язку з цим, існує критична потреба у надійних антивірусних програмах, які здатні ефективно виявляти та блокувати шкідливі загрози, а також забезпечувати захист в режимі реального часу. Програми, такі як Zillya! Антивірус, стають невід'ємною складовою частиною цієї захисної інфраструктури, сприяючи безпеці та захисту користувачів у цифровому середовищі.

Постановка проблеми: з огляду на постійно зростаючу кількість шкідливих програм та кіберзагроз у цифровому просторі, безпека комп'ютерних систем стає надзвичайно актуальним питанням для приватних користувачів та бізнесів. Потенційні загрози включають в себе віруси, троянські програми, шпигунське програмне забезпечення, рекламні атаки та інші форми шкідливих програм, які можуть спричинити втрату даних, порушення конфіденційності або навіть фінансові втрати. У зв'язку з цим, існує загальна необхідність у надійних засобах захисту, які здатні вчасно виявляти та нейтралізувати ці загрози, забезпечуючи безпеку інформації та безперебійну роботу комп'ютерних систем.

Мета даного дослідження полягає в оцінці ефективності та функціональних можливостей антивірусного програмного забезпечення Zillya! Антивірус. Основні аспекти, що розглядаються, включають:

– ефективність виявлення загроз: дослідження ефективності програми у виявленні різноманітних шкідливих програм, включаючи віруси, троянські програми, шпигунське програмне забезпечення та інші форми загроз;

– реакція в реальному часі: оцінка швидкості та ефективності програми у виявленні та блокуванні загроз у режимі реального часу, щоб запобігти поширенню шкідливого програмного забезпечення;

– ресурсозбереження: аналіз впливу програми на ресурси комп'ютера, такі як процесор та пам'ять, з метою забезпечення оптимальної продуктивності комп'ютерної системи під час роботи антивірусного програмного забезпечення;

– можливості налаштування та додаткові функції: вивчення можливостей налаштування програми та наявності додаткових функцій, які можуть підвищити рівень захисту та забезпечити більш гнучкий контроль користувача.

Результат дослідження: Zillya! Антивірус - це програмне забезпечення для захисту комп'ютерів від шкідливих програм, таких як віруси, троянські програми, шпигунське програмне забезпечення тощо. Він розроблений компанією Zillya! і пропонує різні функції для виявлення та видалення загроз для безпеки комп'ютера.

Перша версія з'явилася у квітні 2009 року. В жовтні 2013 р. був анонсований і звісно відбувся 13 листопада 2013 р. вихід Zillya! Антивірус для Бізнесу - який призначений для захисту мереж. 30 травня 2014 р. відбувся реліз, технічна підтримка завершена з 1 лютого 2017 р., а у 2021 року завершився випуск продукту — Zillya! Антивірус Безкоштовний. 15

березня 2016 р. відбувся реліз Zillya! Mobile Antivirus, який призначений для мобільних Android-пристроїв. 25 березня 2019 р. вийшла оновлена версія Zillya! Internet Security for Android 2.0. 12 лютого 2020 р. стало відомо що Zillya!Total Security 3.0 отримав Gold сертифікацію OPSWAT [1].

Антивірусні продукти Zillya! базуються на власних розробках компанії, які увібрали в себе кращі світові досягнення в сфері інформаційного захисту та дозволяють говорити про інноваційність наших технологій.

Використовуючи нестандартні підходи до виявлення та запобігання потраплянню шкідливих програм на комп'ютер користувача, спеціалісти компанії завжди намагаються бути на крок попереду від зловмисників.

Прикладом "гри на випередження" слугує вбудований в усі продукті Zillya! евристичний аналізатор - інтелектуальний модуль захисту. На основі "коду" програмного забезпечення комп'ютера користувача він може визначити вірусне походження програми та знешкодити її, навіть, якщо така програма не була раніше включена до списку шкідливих.

Завдяки власним технологіям компанії стала сертифікованим партнером великого американського інтегратора систем інформаційної безпеки - OPSWAT. Вдалося налагодити співпрацю і ведення обмін зразками вірусів з безліччю антивірусних компаній з усього світу. Наші продукти отримали сертифікат компанії Intel, що підтверджує оптимальність роботи на багатоядерних системах [2].

Серед основних можливостей Zillya! Антивірус можна виділити:

1. Сканування системи: програма проводить періодичне сканування файлів і папок на комп'ютері для виявлення потенційно небезпечних програм.
2. Виявлення і вилучення загроз: після сканування Zillya! Антивірус може виявляти і вилучати віруси, троянські програми, шпигунське програмне забезпечення та інші загрози.
3. Захист в реальному часі: програма може працювати в фоновому режимі, аналізуючи активні процеси та файли, що виконуються на комп'ютері, для виявлення можливих загроз.
4. Оновлення бази даних: Zillya! Антивірус регулярно оновлює свою базу даних вірусних сигнатур, щоб виявляти нові шкідливі програми.
5. Інші додаткові функції: деякі версії програми можуть включати додаткові функції, такі як брандмауер, захист від фішингу, захист електронної пошти тощо.
6. Легкий в ресурсах: деякі антивірусні програми можуть бути важкими для ресурсів системи, що може сповільнювати роботу комп'ютера. Zillya! Антивірус пропонує легкий в ресурсах режим, який дозволяє забезпечити захист без значного впливу на продуктивність комп'ютера.
7. Захист в реальному часі від онлайн-загроз: окрім захисту від загроз, що знаходяться в системі, Zillya! Антивірус також може виявляти та блокувати онлайн-загрози, такі як шкідливі посилання або веб-скрипти, які можуть завдати шкоди під час веб-серфінгу.
8. Захист від розповсюдження через USB-пристрої: деякі версії Zillya! Антивірус мають можливість автоматично перевіряти USB-пристрої, які підключаються до комп'ютера, на наявність загроз і автоматично обробляти їх для запобігання поширенню вірусів через зовнішні пристрої.
9. Режим глибокого сканування: крім стандартного швидкого сканування, Zillya! Антивірус може мати можливість проводити глибоке сканування, яке аналізує всі файли на комп'ютері більш детально для виявлення навіть найменших загроз.
10. Захист від вразливостей: деякі версії програми можуть включати інструменти для виявлення та виправлення вразливостей у системі, таких як застарілі програми або незахищені налаштування, щоб запобігти можливим атакам [3].

Zillya! Антивірус - це один із варіантів антивірусного програмного забезпечення, який може бути корисним для забезпечення безпеки комп'ютера. Проте перед використанням будь-якого антивірусного програмного забезпечення рекомендується ретельно ознайомитися з його можливостями та оглядами, а також забезпечити його регулярне оновлення для найкращої ефективності.

Також цей продукт було використано для аналізу його можливостей в реальних умовах, а саме:

1. Ефективність виявлення загроз: для оцінки ефективності Zillya! Антивірус у виявленні загроз було проведено серію тестів, які включали сканування системи за допомогою відомих вірусних сигнатур та аналіз поведінки програм. Результати показали, що програма здатна ефективно виявляти шкідливе програмне забезпечення різних типів, включаючи навіть ті, які використовують нові алгоритми або техніки маскування.

2. Реакція в реальному часі: для оцінки реакції в реальному часі були використані сценарії симуляції атак, під час яких намагалися виконати небезпечні дії на комп'ютері, такі як завантаження шкідливих файлів або спроби виконання небажаних дій. Zillya! Антивірус виявив здатність ефективно реагувати на такі загрози, блокуючи їх та надаючи повідомлення користувачеві про виявлені загрози.

3. Ресурсозбереження: для оцінки впливу на ресурси комп'ютера були проведені тести, під час яких вимірювалася використана програмою пам'ять та обсяг обчислювальних ресурсів. Результати показали, що Zillya! Антивірус має незначний вплив на продуктивність системи, забезпечуючи при цьому високий рівень захисту.

4. Можливості налаштування та додаткові функції: аналізувалися можливості програми з налаштування та додаткові функції, такі як контроль батьківства, фільтрація спаму, вогнемір тощо. Оцінка базувалася на доступності налаштувань, їхній зручності використання та ефективності у запобіганні різноманітним загрозам.

Загальний результат дослідження підтвердив високу ефективність Zillya! Антивірус як надійного інструменту захисту комп'ютерних систем від кіберзагроз. Його можливості забезпечують не лише виявлення та блокування загроз, але й дозволяють користувачам налаштовувати захист з урахуванням їхніх індивідуальних потреб та вимог.

Отже, Zillya! Антивірус є програмним забезпеченням, спрямованим на захист комп'ютерів від різноманітних загроз, таких як віруси, троянські програми та інші шкідливі програми. Він пропонує різноманітні функції, включаючи сканування системи, виявлення та вилучення загроз, захист в реальному часі, захист від онлайн-загроз і захист від розповсюдження через USB-пристрої. Zillya! Антивірус також може мати легкий в ресурсах режим, глибоке сканування та захист від вразливостей. Загалом, він пропонує комплексний підхід до безпеки комп'ютера, що може бути корисним для користувачів у попередженні загроз і забезпеченні безпеки під час використання Інтернету та роботи з файлами.

Рекомендації та висновки: підсумовуючи вище викладене, автори дійшли висновку, що Zillya! Антивірус є ефективним і надійним інструментом для захисту комп'ютерних систем від широкого спектру кіберзагроз. Його висока ефективність у виявленні та блокуванні шкідливих програм, а також здатність швидко реагувати на загрози в режимі реального часу, робить його привабливим вибором для користувачів будь-якого рівня.

Висновки також підтверджують, що Zillya! Антивірус має незначний вплив на ресурси комп'ютера, що робить його оптимальним рішенням для тих, хто шукає збалансований захист без великого споживання системних ресурсів.

Рекомендуємо користувачам регулярно оновлювати антивірусне програмне забезпечення, а також регулярно проводити сканування системи для максимальної ефективності захисту. Крім того, варто враховувати індивідуальні потреби та вимоги та використовувати доступні налаштування програми для оптимізації захисту.

Література:

[1] Zillya! — Вікіпедія. [Електронний ресурс]. – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Zillya!

[2] Zillya! Антивірус — український антивірус. [Електронний ресурс]. – Режим доступу до ресурсу: https://zillya.ua

[3] Огляд Zillya! Антивірус Безкоштовний – YouTube. [Електронний ресурс]. – Режим доступу до ресурсу: https://youtu.be/KW12EXfiUWI?si=BGIHoNjaSBSyS7v5

АНАЛІЗ РИЗИКІВ ТА РОЗВИТОК ПРЕВЕНТИВНИХ СТРАТЕГІЙ В КІБЕРБЕЗПЕЦІ НА ОСНОВІ ВПЛИВУ СОЦІАЛЬНО-ІНЖЕНЕРНИХ АТАК НА ОБ'ЄКТИ БЕЗПЕКИ

Чусов Е.Є.
Керівник: Старкова О.В.
E-mail: edikchusov20035@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Кібербезпека в сучасному світі стає все більш важливою проблемою. Зростання кількості та складності кібератак, а також їх значний вплив на критичну інфраструктуру та особисті дані роблять цю проблему надзвичайно актуальною.

Соціально-інженерні атаки – це тип кібератаки, який використовує психологічні методи для маніпулювання людьми з метою отримання доступу до інформації, систем або ресурсів. Ці атаки можуть бути спрямовані на будь-кого, незалежно від його технічної кваліфікації.

Існує багато різних типів соціально-інженерних атак, але деякі з найпоширеніших включають: фішинг – шахраї розсилають електронні листи або текстові повідомлення, які здаються легітимними, щоб змусити людей розкрити особисті дані або перейти за посиланням, яке заражає їх комп'ютер шкідливим програмним забезпеченням; квітинг – шахраї вдають із себе співробітників служби підтримки або інших авторитетних осіб, щоб отримати доступ до інформації або систем; підміна особистості – шахраї видають себе за інших людей, щоб отримати доступ до інформації або ресурсів [1].

Соціально-інженерні атаки можуть мати значний вплив на об'єкти безпеки. Ці атаки можуть призвести до:

1. крадіжки даних – особисті дані, фінансова інформація, комерційна таємниця та інші конфіденційні дані можуть бути вкрадені;
2. втрата доступу – шахраї можуть отримати доступ до систем або ресурсів, що може призвести до порушення роботи або крадіжки даних;
3. фінансові втрати – шахраї можуть викрасти гроші або інші цінні активи;
4. пошкодження репутації – соціально-інженерні атаки можуть завдати шкоди репутації організації [1].

Організації повинні впровадити політики та процедури, які допоможуть захистити від соціально-інженерних атак. Ці політики та процедури повинні включати: політику паролів; контроль доступу; навчання співробітників [1].

Існують технології, які можуть допомогти захистити від соціально-інженерних атак. Ці технології включають: фільтри спаму; антивірусне програмне забезпечення; системи виявлення вторгнень (IDS) [1].

Крім того, важливо пам'ятати, що соціально-інженерні атаки можуть бути спрямовані не лише на співробітників, але й на клієнтів, партнерів та інших зацікавлених сторін. Тому важливо, щоб всі, хто має доступ до інформації або систем організації, знали про ризики соціально-інженерних атак і вживали заходів для їхнього захисту.

Захист від соціально-інженерних атак – це постійний процес. Важливо бути пильними, знати про ризики та вживати відповідні заходи для їхньої мінімізації.

Література

[1] Borsukovskyi, Y. V., Borsukovska, V. Y. (2018). Прикладні аспекти захисту інформації в умовах обмеженого фінансування. Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка», 1(1), С. 26–34. [Електронний ресурс]. – Режим доступу до ресурсу: https://csecurity.kubg.edu.ua/index.php/journal/article/view/13.

ВИКОРИСТАННЯ УТИЛІТИ VENTOY ДЛЯ СТВОРЕННЯ ФЛЕШ-ДИСКУ З МОЖЛИВІСТЮ МУЛЬТИЗАВАНТАЖЕННЯ ОПЕРАЦІЙНИХ СИСТЕМ

Шапо В.Ф.[1], Воловщиков В.Ю.[2]
E-mail: [1]vladlen.shapo@gmail.com, [2]valvol98@gmail.com
[1]Одеса, Інститут Військово-Морських Сил
[2]Харків, Національний технічний університет "Харківський політехнічний інститут"

В сучасному світі, наповненому різноманітними комп'ютерами, в величезній кількості випадків дуже велике значення має їх працездатність. Якщо не працюють персональні комп'ютери користувачів (про сервери в цій роботі взагалі не йдеться) в офісній локальній мережі, це як мінімум може збільшити час на копіювання, друкування, редагування документів, як максимум може привести до фінансових втрат при веденні бізнесу. Якщо не працюють комп'ютери касирів в залізничних або авіакасах, це призводить до втрат часу та, можливо, грошей на придбання квитків пасажирами. Якщо не працюють комп'ютери операторів на паливній заправці, стає неможливим заправлення паливом. Якщо не працюють комп'ютери касирів в супермаркетах, стає неможливим придбання різноманітних товарів, і т. ін. Такі ситуації призводять до лише втрат часу та грошей, і на можливість виникнення аварій з величезними фінансовими втратами та/або людськими жертвами це безпосередньо не впливає.

Але в ситуаціях, коли комп'ютери працюють на виробництві або в військовій сфері, їх непрацездатність може призвести до різного роду аварій, катастроф, величезних матеріальних та людських втрат. При цьому абстрактний системний адміністратор, що має відновлювати працездатність комп'ютерів, може бути у відрядженні, у відпустці, хворіти, бути пораненим чи загинути. Дані ж з комп'ютерів треба у багатьох ситуаціях отримувати якнайшвидше, бо від цього може залежати багато людських життів та збереження матеріальних цінностей.

Взагалі кожна хвилина простоювання робочих комп'ютерів може мати суттєве значення, а результати їх непрацездатності можуть бути невиправними.

Одним з підходів, що дозволяє отримати доступ до даних, що розміщені на комп'ютері, який вийшов з ладу, є використання створеного та підготовленого належним чином заздалегідь флеш-диску, що дозволить завантажити деяку операційну систему, запустити антивірусне програмне забезпечення, спробувати знайти необхідні дані на жорсткому диску, якщо він не вийшов з ладу, та скопіювати їх на відповідний носій.

Безкоштовна утиліта Ventoy [1] дозволяє створювати накопичувачі з можливістю мультизавантаження. Остання версія 1.0.97 вийшла 24 січня 2024 р. Існують версії для Windows, Linux, та варіант у вигляді LiveCD. Її основні можливості представлені нижче, і ними перелік можливостей не обмежений.

1. Можливість встановлення на USB/Local Disk/SSD/NVMe/SD карту.
2. Безпосереднє завантаження з файлів ISO/WIM/IMG/VHD(x)/EFI, нема необхідності в розпакуванні.
3. Є можливість розмічати флеш-пристрій (цільовий носій) під MBR чи GPT.
4. Оновлення без втрати даних на носії.
5. Підтримка Secure Boot.
6. Має встановлювальними під Linux (CLI / Web / Qt & GTK GUI) та Windows (GUI).
7. Може завантажувати ISO Windows, Windows PE, Linux, *BSD, Android X86 та інші. Список протестованих версій операційних систем насчитывает більше ніж 1100 позицій [2] на дату виходу останньої версії.
8. Може працювати з архітектурами: x86_64, x86, IA32 UEFI, ARM64 UEFI, MIPS64 UEFI.
9. Підтримує завантаження x86 OpenWRT.

10. Може завантажувати floppy, VHD, VDI та RAW іміджі.

11. Має підтримку Persistence для LiveCD Fedora, Ubuntu, Arch, Mint та інших.

12. Може працювати з файлами для unattended installation різних дистрибутивів Linux та Windows.

13. Може працювати в Memdisk Mode (режим, коли образ повністю завантажується в пам'ять та запускається вже звідти. Це дозволяє, наприклад, завантажити KolibriOS та деякі диски для відновлення на основі DOS).

14. Може переключатися WIMBOOT mode при наявності проблем с запуском образів Windows в нормальному режимі.

15. Має розвинуту систему плагінів, яку можна конфігурувати головним чином за допомогою json-файла конфігурації.

16. Підтримує ISO-файли більші за розміром, ніж 4 Гбайти.

17. Підтримує USB-диски з захистом від запису.

18. Дозволяє працювати з USB у звичайному режимі.

19. Під час оновлення версії дані зберігаються неушкодженими.

20. Нема необхідності в оновленні Ventoy, коли випускається новий дистрибутив будь-якої операційної системи.

Деякі популярні дистрибутиви операційних систем сімейства Linux, що підтримуються Ventoy, та відповідні файли представлені в табл. 1. Вказані у табл. 1 дистрибутиви Linux можуть працювати як з BIOS, так і з UEFI. Меню вибору операційних систем при завантаженні Ventoy представлено на рис. 1. Можливий вигляд розділів флеш-накопичувача представлений на рис. 2.

Таблиця 1 – Дистрибутиви операційних систем сімейства Linux та відповідні файли

Дистрибутив	Файл ISO
Centos8	CentOS-8.3.2011-x86_64-dvd1.iso
Centos7	CentOS-7-x86_64-Minimal-2009.iso
Deepin	deepin-desktop-community-1010-amd64.iso
Fedora	Fedora-Workstation-Live-x86_64-33-1.2.iso
Ubuntu Desktop	ubuntu-20.04-desktop-amd64.iso
Ubuntu Server	ubuntu-20.04.1-live-server-amd64.iso
Linux Mint	linuxmint-20.2-cinnamon-64bit.iso
MX Linux	MX-19.2_x64.iso
Kali Linux	kali-linux-2021.1-installer-amd64.iso
Manjaro	manjaro-xfce-20.0.1-200511-linux56.iso
Archman	Archman_KDE_20200209.iso
Mageia	Mageia-7.1-x86_64.iso
openSUSE	openSUSE-Leap-15.2-DVD-x86_64.iso

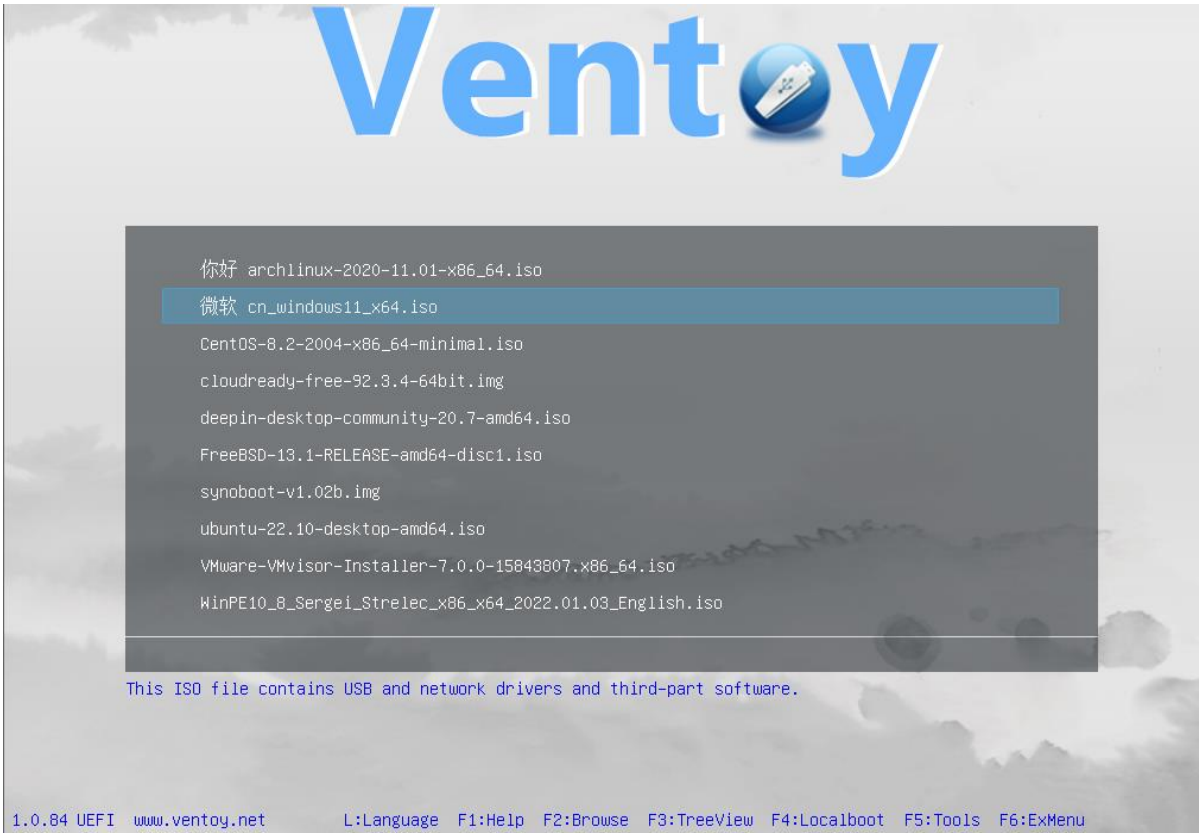


Рис. 1. Меню вибору операційних систем при завантаженні Ventoy

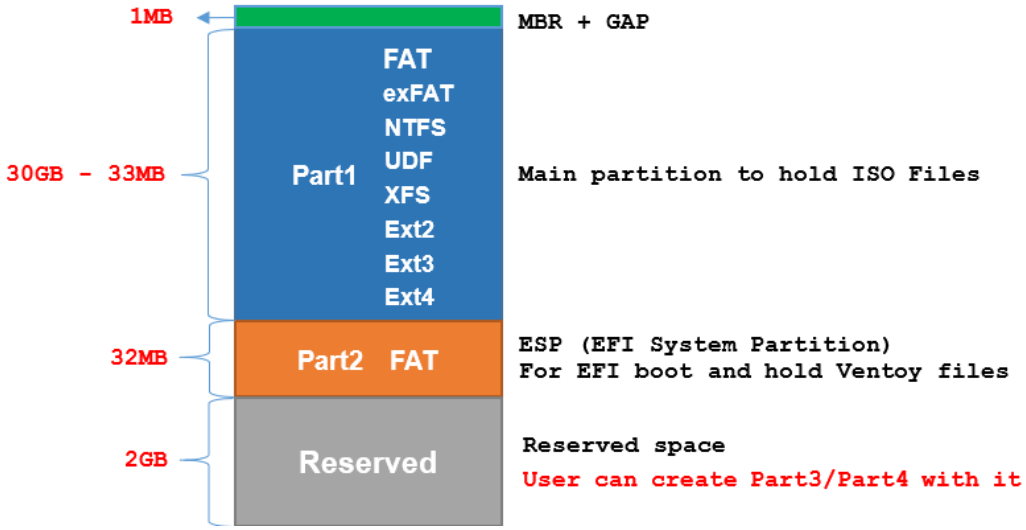


Рис. 2. Можливий вигляд розділів флеш-накопичувача при роботі з Ventoy

Оглянуті можливості Ventoy дозволяють зробити висновок, що її використання може суттєво покращити можливості доступу до даних на виведених з ладу комп'ютерах, в той же час Ventoy дозволяє засвоїти багато комп'ютерних термінів та технологій.

Література

[1] A New Bootable USB Solution [Электронний ресурс]. – Режим доступу до ресурсу: https://www.ventoy.net/en/index.html

[2] Ventoy tested OS series [Электронний ресурс]. – Режим доступу до ресурсу: https://www.ventoy.net/en/isolist.html

АУДИТ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ ПІДПРИЄМСТВА

Яковенко Д.В.
Керівник: Мерлак О.В.
E-mail: qdenis328@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

В сучасному світі, охопленому стрімкими технологічними змінами та постійною загрозою кібератак, питання забезпечення інформаційної безпеки стає дедалі більш актуальним для підприємств усіх галузей. Відомості, які вони обробляють і зберігають, стають сутою валютою в сучасному бізнесі, тому необхідність захисту цих активів стає вирішальною для забезпечення стабільності та успішності підприємства.

Загрози кібербезпеки постійно зростають, і це ставить перед керівництвом підприємств виклик: забезпечити ефективний захист інформації в умовах постійних технологічних змін та вдосконалення методів атак. Інциденти порушення безпеки, такі як витік конфіденційної інформації, крадіжки даних або вимагання виплат за доступ до цієї інформації, можуть суттєво пошкодити репутацію підприємства, призвести до фінансових втрат та негативно вплинути на його діяльність.

Для ефективного забезпечення інформаційної безпеки підприємство повинно здійснювати аудит, щоб оцінити свої поточні заходи захисту, ідентифікувати потенційні вразливості та ризики та розробити стратегії для їхнього усунення [1]. Основні засоби аудиту включають перевірку систем безпеки, аналіз прав доступу до інформації, оцінку вразливостей мережі та перевірку відповідності стандартам безпеки.

Під час проведення аудиту інформаційної безпеки підприємства можна виділити наступні основні етапи.

1. Визначення цілей дослідження та об'єктів, на яких буде проводитися аудит інформаційної безпеки

2. Опис поточних загроз кібербезпеці та необхідність аудиту інформаційної безпеки для запобігання інцидентам та мінімізації ризиків.

3. Опис методів та інструментів, які використовуються при проведенні аудиту інформаційної безпеки, таких як технічне тестування, аналіз політик безпеки, інтерв'ю з персоналом тощо [2].

4. Розбиття процесу аудиту на конкретні етапи, включаючи планування, збір даних, аналіз, виявлення вразливостей, формулювання рекомендацій та підготовку звіту.

5. Процес оцінки результатів аудиту інформаційної безпеки та розробка стратегій для вдосконалення заходів захисту.

6. Визначення практичних кроків, які підприємство може здійснити на основі рекомендацій аудиту для покращення своєї інформаційної безпеки.

7. Аналіз можливих перешкод та викликів, які можуть виникнути під час проведення аудиту інформаційної безпеки та способи їх подолання. Практичне застосування результатів [3].

8. Підсумкові висновки щодо результатів дослідження і рекомендації для підприємства щодо подальших кроків у сфері забезпечення інформаційної безпеки.

Проведення аудиту інформаційної безпеки є критично важливим етапом для забезпечення стійкості та надійності функціонування підприємства в умовах сучасного бізнес-середовища, насиченого ризиками кібератак та інших загроз. Через виявлення потенційних вразливостей та розробку рекомендацій щодо їх усунення аудит інформаційної безпеки допомагає підприємствам зберегти конфіденційність, цілісність та доступність їх інформаційних активів. Наслідком цього є зниження ризиків фінансових втрат, порушення репутації та негативного впливу на довіру клієнтів. Інформація, отримана в результаті аудиту, дозволяє підприємствам приймати обгрунтовані рішення щодо подальших інвестицій у заходи захисту інформації та ефективного управління ризиками. Таким чином, аудит

інформаційної безпеки є необхідним елементом стратегії управління ризиками та забезпечення успішності підприємства в сучасному бізнес-світі.

Література

[1] Мартиненко, Н. (2016). Методика проведення аудиту інформаційної безпеки підприємства. Електронний ресурс "Науковий журнал Наукові праці Харківського національного університету внутрішніх справ", (2), 89-93.

[2] Коваль, О. В., & Чекушкіна, В. М. (2018). Аудит інформаційної безпеки як засіб підвищення конкурентоспроможності підприємства. Науковий вісник Херсонського державного університету. Серія: Економічні науки, (31), 128-132.

[4] Чернишова, І. (2019). Аудит інформаційної безпеки в умовах сучасних загроз. Електронний ресурс "Бібліотека кафедри обліку та аудиту", Донецький національний університет.

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ОПТИМІЗАЦІЇ ЛАЗЕРНОГО ЛІКУВАННЯ ДІАБЕТИЧНОЇ РЕТИНОПАТІЇ

Яськова Є.Г.[1], Чугай А.М.[2]
E-mail: yelizavetayaskova@gmail.com
[1]Харків, Харківський національний університет імені В.Н. Каразіна
[2]Харків, Харківський національний економічний університет імені Семена Кузнеця

Процедура лазерної фотокоагуляції сітківки є важливим методом лікування різних захворювань сітківки, включаючи діабетичну ретинопатію [1]. Для генерації тепла та створення опіку, який перетворюється на рубець в цільовій області сітківки, використовують лазер. Рубець допомагає заблокувати ненормальні кровоносні судини, які можуть розвиватися у разі певних захворювань сітківки, зокрема діабетичної ретинопатії.

PAttern SCAnning Laser (PASCAL, 2006) – це перший лазерна система для офтальмології з автоматичним позиціонуванням лазерних імпульсів на основі високошвидкісних дзеркал та набору шаблонів. Вона призначена для автоматичного позиціонування лазерних імпульсів на основі високошвидкісних дзеркал та набору шаблонів. Це досягається за допомогою спеціалізованого програмного забезпечення, яке контролює рух лазерного променя та його інтенсивність. Програмне забезпечення також дає змогу лікарю вибрати відповідний шаблон для лазерного випромінювання, що забезпечує точність та ефективність лікування [2].

Основна мета сучасних систем – поліпшення якості ретинальної коагуляції під час лікування діабетичної ретинопатії за рахунок обчислення більш ефективного плану лазерної коагуляції. Програмне забезпечення має реалізувати більш рівномірний розподіл енергії лазера пігментнім епітелієм порівняно з моноімпульсними та шаблонними техніками лазерної коагуляції. У статті [3] представлені алгоритми для розрахунку карти коагулятів як задачі щільного пакування кругів у випадковій області.

В [4] досліджено фреймворки для класифікації непроліферативної діабетичної ретинопатії, ексудатів, кровотечів та мікроаневризм, які ґрунтуються на машинному навчанні та глибоких нейронних мережах. Описано використання передових методів штучного інтелекту для створення ефективної та точної системи, яка може допомогти медичним фахівцям автоматично діагностувати діабетичну ретинопатію на ранніх стадіях без спеціальних клінічних ресурсів.

Таким чином, сучасне програмне забезпечення дає змогу автоматизувати процес лазерної фотокоагуляції, забезпечуючи більш точне та ефективне лікування. Воно допомагає медичним фахівцям краще розуміти хворобу та розробляти більш ефективні стратегії лікування.

Література

[1] Everett, L.A., Paulus, Y.M. Laser Therapy in the Treatment of Diabetic Retinopathy and Diabetic Macular Edema. Curr Diab Rep 21, 35 (2021). AccessMode: https://doi.org/10.1007/s11892-021-01403-6.

[2] Nemcansky, J., Stepanov, A., Nemcanska, et al. Single session of pattern scanning laser versus multiple sessions of conventional laser for panretinal photocoagulation in diabetic retinopathy: Efficacy, safety and painfulness. PLOS ONE 14(7): e0219282 (2019). AccessMode: https://doi.org/10.1371/journal.pone.0219282.

[3] Ilyasova, N., Kirsh, D., Paringer, R. et al. Coagulate map formation algorithms for laser eye treatment. In: The 3rd International Conference on Frontiers of Signal Processing (ICFSP), Paris, France, 2017. P. 120-124. AccessMode: https://doi.org/10.1109/ICFSP.2017.8097154.

[4] Agarwal, S., Bhat, A. A survey on recent developments in diabetic retinopathy detection through integration of deep learning. Multimed Tools Appl 82, 17321–17351 (2023). AccessMode: https://doi.org/10.1007/s11042-022-13837-5.

Секція 2

PYTHON CUPY AND CUDF LIBRARIES FOR GPU-ACCELERATED DATA PROCESSING

Latanska L.O., Fadieiev P.V.
E-mail: llatanskaya@gmail.com, pavelf200205@gmail.com
Mykolaiv, Admiral Makarov National University of Shipbuilding

When dealing with large amounts of data, it is possible to speed up the processing by using graphics processing units (GPUs) instead of CPUs. However, writing code to run on a GPU is not an easy task and requires in-depth knowledge of how to write efficient parallel algorithms and requires proficiency in GPU programming frameworks such as CUDA, OpenCL or ROCm. In many cases, this task can be accomplished through the use of specialised libraries that can simplify the writing of GPU-accelerated software. Among such libraries there are the open-source Python libraries CuPy and cuDF.

CuPy is a high performance data processing library for Python that has an interface that is highly compatible with the widely used libraries NumPy and SciPy. In most cases CuPy could be used as a drop-in replacement for these libraries with little code change [1]. CuPy primarily supports Nvidia GPUs, but also supports AMD's ROCm platform experimentally with some limitations on functionality. The speedup offered by CuPy over NumPy depends on the type of operation and the size of the data. For most operations, the larger the data size, the greater the advantage over NumPy.

An additional feature of CuPy is it's support for custom GPU kernels written by the user. This enables the extension of the library's functionality with additional operations for specific use cases without the need to write a complete application in C++ with CUDA.

For processing and analyzing the data represented in tabular form, data scientists and researchers often use Pandas library for Python [2]. Pandas is popular for it's rich feature set and ease of use. Pandas is usually fast enough for small datasets, but it may take a considerable amount of time to process larger datasets with the size of several gigabytes or more. However, it is possible to speed up the processing by using the cuDF library.

cuDF is a library for GPU-accelerated tabular data processing that provides an API similar to Pandas [3]. Recent versions of cuDF allow code written using the Pandas library to be accelerated by almost 150 times with zero code changes [4]. It is possible to enable cuDF acceleration for existing code in a Jupyter Notebook environment by simply using the %load_ext cudf.pandas cell magic at the beginning of the cell that needs to be accelerated. cuDF can also be used to speed up the entire Python modules and libraries. The big advantage of cuDF is that it can also speed up code that uses libraries that Pandas uses internally. In terms of hardware support, cuDF only supports Nvidia GPUs.

In summary, the CuPy and cuDF libraries are useful tools that unlock the power of GPU computing in a familiar Python environment, without the need to use C++ with hard-to-learn GPU programming frameworks such as CUDA. Furthermore, the API compatibility of CuPy with NumPy and SciPy, and of cuDF with Pandas, allows developers to create versions of their data processing software for both CPU and GPU using a single codebase.

References

[1] CuPy: NumPy & SciPy for GPU, URL: https://cupy.dev/

[2] Pandas - Python Data Analysis Library, AccessMode: https://pandas.pydata.org/

[3] cuDF - GPU DataFrame Library, AccessMode: https://github.com/rapidsai/cudf

[4] RAPIDS cuDF Accelerates pandas Nearly 150x with Zero Code Changes. AccessMode: https://developer.nvidia.com/blog/rapids-cudf-accelerates-pandas-nearly-150x-with-zero-code-changes/

THE USE OF TERRAFORM FOR THE MANAGEMENT OF HETEROGENEOUS CLOUD SYSTEMS IN THE PROCESSING OF RESOURCE-INTENSIVE DATA

Leunenko O.V.
E-mail: Oleksii.Leunenko@gmail.com
Kharkiv, Simon Kuznets Kharkiv National University of Economics

In the modern world, data is becoming an increasingly valuable resource. It is used in industries ranging from science to business, medicine to social media. However, processing this data can be challenging, especially when it comes to resource-intensive data that requires significant computing resources.

At the same time, cloud computing is becoming increasingly popular, providing organizations with the flexibility, scalability, and accessibility they need to process large amounts of data. However, managing these cloud resources can be challenging, especially when using heterogeneous cloud systems or multi-clouds.

Terraform is open-source software that automates the creation, modification, and destruction of cloud service infrastructure. It can be used to manage a wide range of cloud services, making it an ideal tool for managing heterogeneous cloud systems.

Terraform can be used to manage heterogeneous cloud systems for processing resource-intensive data. Terraform is particularly relevant for provisioning the resources needed to process data-intensive data, as it can simplify the management of multi-cloud environments.

Terraform is open-source software developed by HashiCorp for Infrastructure as Code (IaC) management. This means you can use code to create, modify, and destroy infrastructure across multiple cloud services and version infrastructure efficiently and securely.

There are some of the most important key aspects of Terraform:

– Declarative syntax: Terraform uses declarative syntax, which means you describe the desired state of the infrastructure rather than a sequence of actions to achieve that state. This simplifies the code and makes it easier to understand and read.

– Low-level components. Terraform covers both low-level components (such as compute instances, storage, and networking) and high-level components (such as DNS records and SaaS functions) [1, 2].

– Multi-vendor support: Terraform supports a wide range of cloud providers, including AWS, Google Cloud, Azure, and many others. This makes it possible to use a single tool to manage all your cloud resources from a variety of vendors.

– Modular structure: Terraform allows you to create modules, which in turn allows you to reuse code and simplify infrastructure management.

– Security and visibility: With Terraform, you can review the change plan before applying it to the infrastructure, ensuring security and transparency.

Using Terraform to manage cloud resources can greatly simplify deployment and scaling, especially when dealing with heterogeneous cloud systems.

Terraform opens new possibilities for managing heterogeneous cloud systems. With its flexibility and support for multiple cloud providers, Terraform can be a bridge between different cloud environments.

Terraform can be used to manage heterogeneous cloud systems in the following ways:

– Unified interface: Terraform provides a unified interface for managing resources across multiple clouds, allowing you to use the same syntax and commands to manage resources across Amazon Web Services, Google Cloud, Microsoft Azure, and other cloud services.

– Modularity: Modules allow you to reuse code to create the same resources across different clouds. This not only simplifies infrastructure management but also ensures consistency across different cloud environments.

– Automation: Terraform allows you to automate the processes of creating, modifying, and destroying resources, which can be especially useful when managing heterogeneous cloud systems where manual resource management can be difficult and time-consuming.

– Planning and security: Terraform allows you to review the change plan before it is applied, providing visibility and security. This is especially important when managing heterogeneous cloud systems, where the wrong changes can have big consequences.

Resource-intensive data, such as large data sets or complex computational tasks, require significant computing resources. Terraform can help automate the process of provisioning and scaling these resources across heterogeneous cloud systems.

For processing resource-intensive data, Terraform can be used:

– Automated deployment: With Terraform, you can automate the process of provisioning the resources needed to process resource-intensive data. For example, you can use Terraform to automatically create the virtual machines or containers you need to process your data.

– Automated scaling: Terraform can also be used to automate scaling processes. For example, you can use Terraform to automatically scale up or scale down the number of virtual machines you have based on your data processing needs.

– Disaster Recovery: Terraform can be used to automate disaster recovery processes, which is important to ensure continuity of processing for resource-intensive data.

– Configuration Management: With Terraform, you can centrally manage the configuration of resources used to process resource-intensive data. This can simplify management and ensure configuration consistency.

In the article "Terraform: Building Multi-Cloud Infrastructure for AWS and GCP" [3], the author discusses how to build an infrastructure on both cloud platforms (AWS and GCP) using Terraform. This can serve as a good example for your case. You can also check out the GitHub repository that demonstrates how to create Kubernetes clusters on multiple cloud platforms using Terraform [4].

Creating a VPN between AWS, GCP, and Azure using Terraform can ensure secure interaction between different clouds [5].

Terraform is a powerful tool for creating, modifying, and managing infrastructure in the Clouds. It uses the principle of "infrastructure as code", which allows you to use code to describe and create the desired infrastructure configuration.

However, when it comes to big data processing, you need to select the appropriate services to store, process, and analyze data in each cloud. This may include:

– Data stores: These are places where you store your data. They can be structured (such as relational databases) or unstructured (such as object storage).

– Hadoop clusters: Hadoop is an open-source framework for processing large amounts of data. It includes the Hadoop Distributed File System (HDFS) for the storage of data and MapReduce for the processing of data.

– Databases: You can use different types of databases, including relational, NoSQL, in-memory, and others, depending on your needs.

– Other tools: Depending on your business case, you may need other tools, such as streaming data processing systems (e.g., Apache Kafka), machine learning tools (e.g., TensorFlow), and data visualization tools (e.g., Tableau or Microsoft PowerBI).

It's important to remember that Terraform is just a tool for building and managing infrastructure. Choosing the right services for data storage, processing, and analysis is an important part of your big data strategy. Terraform can automate this process, but you still need to decide which services you need and how to configure them.

Using Terraform to process resource-intensive data in heterogeneous cloud systems can greatly simplify infrastructure management and provide greater flexibility and efficiency.

Managing heterogeneous cloud systems while processing data-intensive data is a complex task that requires effective tools. Terraform has proven to be an essential tool for this task with its flexibility, support for multiple cloud providers, and automation capabilities.

Using Terraform to manage heterogeneous cloud systems allows organizations to efficiently utilize cloud resources and ensure high performance when processing resource-intensive data. It also promotes the security, visibility, and consistency that are critical to successful cloud infrastructure management.

The future of Terraform and heterogeneous (multi-cloud) technologies in the context of processing resource-intensive data is promising. As technology evolves and data volumes increase, the need for effective cloud resource management tools will only grow. Terraform will undoubtedly play a key role in this process.

References

[1] What is Terraform | Terraform | HashiCorp Developer? [Electronic resource]. – Access mode to a resource: https://developer.hashicorp.com/terraform/intro.

[2] Terraform: what is it and what is it for? [Electronic resource]. – Access mode to a resource: https://blog.ithillel.ua/articles/terraform-shho-ce-i-dlya-cogo-vin-potriben.

[3] Terraform: Building Multi-Cloud Infrastructure for AWS and GCP: A Comparative Analysis | by Amit Maheshwari | Engineered @ Publicis Sapient | Medium? [Electronic resource]. – Access mode to a resource: https://medium.com/engineered-publicis-sapient/building-aws-and-gcp-cloud-infrastructure-with-terraform-a-comparative-analysis-16380cfd9dfd.

[4] GitHub - hajowieland/terraform-kubernetes-multi-cloud: Terraform to create Kubernetes clusters on multiple public cloud platforms (Aliyun, AWS, Azure, DO, GCP, OCI) ? [Electronic resource]. – Access mode to a resource: https://github.com/hajowieland/terraform-kubernetes-multi-cloud.

[5] Tutorial: Creating a Multi-Cloud VPN with Terraform between AWS, GCP, and Azure - Silectis? [Electronic resource]. – Access mode to a resource: https://www.silect.is/blog/multi-cloud-vpn-terraform.

COMBINATION OF .NET TECHNOLOGY AND ANGULAR FRAMEWORK TO DEVELOP APPLICATION FOR TESTING SQL LANGUAGE KNOWLEDGE

Naumenko V.[1], Shelest V.[1], Yakovleva O.[1,2]

E-mail: vadym.naumenko@nure.ua, volodymyr.shelest@nure.ua, olena.yakovleva@vsemba.sk

[1] Kharkiv, Kharkiv National University of Radio Electronics

[2] Bratislava, Bratislava University of Economics and Management

The paper is devoted to the development of an application for testing knowledge of the SQL language using the .NET technology [1] and the Angular framework [2]. .NET is developed and maintained by Microsoft, Angular is developed and maintained by Google. .NET and Angular are classified as Open Source Software and are distributed under the MIT license. This is one of the most flexible and less restrictive licenses in the open source world, allowing you to use, copy, modify, and distribute the software with virtually no restrictions. This gives developers considerable freedom to use, modify and distribute the software.

In the ever-changing landscape of web development, the combination of robust backend frameworks with dynamic front-end technologies has become the foundation for building powerful and scalable applications [3]. Among the dynamic duos in the field, the combination of .NET and Angular stands out as a compelling choice for developers seeking a seamless, efficient, and feature-rich application development experience. From streamlined development workflows to improved user experience, .NET and Angular bring a host of benefits. However, as with any technology, challenges and trade-offs are inevitable. By understanding the pros and cons of these frameworks, developers can make informed decisions, ensuring that the approach they choose is a perfect fit for the specific requirements and goals of their projects.

Let's look at their pros and cons in combination. First, let's introduce the .NET technology for creating server-side applications, which has the following advantages:

1. .NET supports several languages, such as C#, F#, and VB.NET, allowing developers to choose the language that best suits their preferences and project requirements.

2. The .NET ecosystem offers a complete set of tools, libraries, and frameworks that simplify development, increasing efficiency and consistency.

3. With the introduction of .NET Core, developers can create cross-platform applications, providing the flexibility to deploy on different operating systems.

4. .NET is known for its scalability and performance, making it an excellent choice for creating enterprise-level applications that require high efficiency and responsiveness.

.NET is a leader in its category, but it also has some drawbacks that are worth paying attention to:

1. The problem with memory leakage due to allocation in the closures for arrow functions.

2. Unfortunately, since .NET is managed by Microsoft, any changes or restrictions that the company may impose will inevitably affect projects made on this technology. This means that developers will have less control.

Secondly, let's consider the Angular frontend framework, the advantages of which include:

1. Two-way data binding in Angular simplifies the synchronization of data between the model and the view, reducing the amount of boilerplate code and increasing developer productivity.

2. The modular architecture of Angular allows you to create reusable components, which makes it easy to maintain and scale applications.

3. Angular's built-in dependency injection makes it easy to develop loosely coupled components, which leads to better testability and maintainability.

4. Angular has a large and active community that provides extensive support, tutorials, and documentation for developers, making it easy to troubleshoot development issues and learn.

Despite the many advantages and popularity of this framework, Angular has its drawbacks, which do not make it a permanent choice for front-end application development:

1. The wide range of features and complex concepts of Angular can be difficult for beginners, which requires a significant investment of time to understand its full potential.

2. The extensive features of Angular can lead to an increase in package size, which can potentially affect the initial loading time of applications, especially on slow networks.

3. Angular has a fast release cycle with frequent updates, which can pose challenges for developers and organizations in terms of keeping up with the latest changes.

So, both .NET and Angular offer powerful tools and capabilities, they can effectively complement each other in the development of modern web applications, where Angular is used to create the client side, .NET - for the server side.

Thus, .NET and Angular were chosen to create a client-server application for testing SQL knowledge.

There are many different database management systems that use the SQL language (but each of them has its own differences in some details): MS SQL Server, MySQL, PostgreSQL, and others. You can test and improve your practical knowledge of SQL using many services, such as CodeWars and LeetCode, but it is difficult, and often impossible, to build a learning process and establish communication with the student based on them.

Therefore, it was decided to develop an application for monitoring SQL knowledge for the educational process, namely for use in the study of relational database disciplines. The main difference between this application and other testing systems is that the application will allow the teacher to organize knowledge testing on the necessary topics, according to a given schedule, and most importantly, the system will allow students not to choose the correct answers from the list, but to write SQL scripts directly, and the system will evaluate the correctness of SQL scripts in automatic mode.

The system should support 3 types of users:

– teacher;

– administrator;

– student.

The server side of our application is written in .NET using microservice architecture. Microservices communicate with each other using the RPC protocol. The architecture diagram of the software application is shown in figure 1.

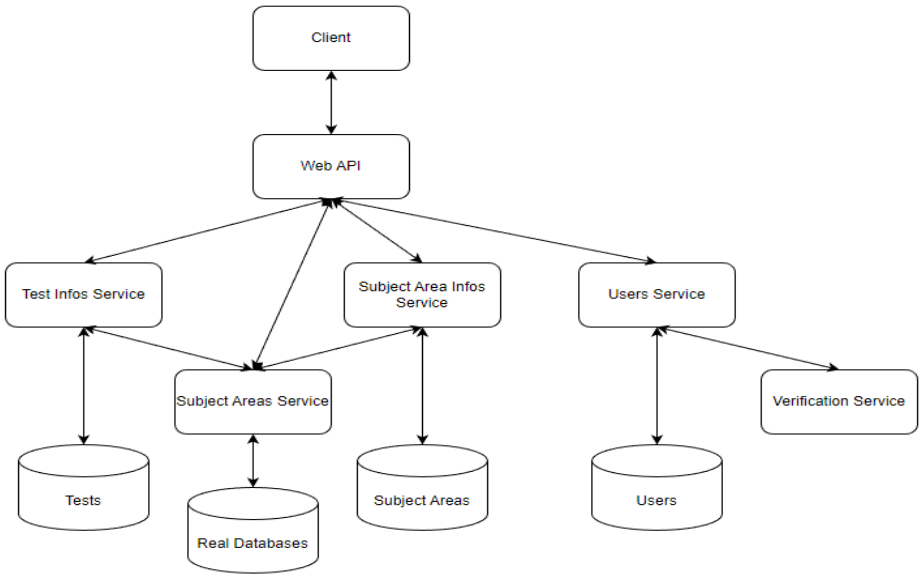
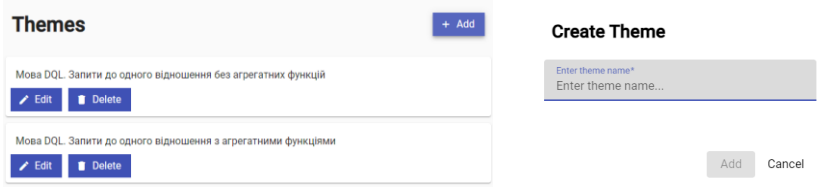


Figure 1 - Application architecture diagram

The Web API acts as an intermediary between the client and the server, processing all requests from the client service. In addition, this service is responsible for load balancing between other services. Test Infos Service processes requests for creating, editing, and deleting tests and questions, as well as answering students' questions. Subject Area Infos Service is responsible for processing requests for manipulating data in existing databases. It stores information about all available databases, tables, and attributes of these tables. Users Service manipulates user data and provides support for authorization, registration, and verification of student identity. Subject Areas Service is responsible for creating and deleting databases, as well as manipulating existing databases. It also executes students' SQL scripts and returns the result in JSON format.

The front-end part of the application is written using the Angular framework with the Material library, which has ready-made UI components that can be reused throughout the project. As for the design, all users have the same main panel with authorization and registration buttons and a navigation panel (only the content of this panel differs). Based on the requirements of the application, some types of users have some functional limitations. All of these restrictions affect what the user will see in front of them.

A user with the Administrator or Teacher role has 7 menu items on the navigation bar: Tests, Themes, Questions, Subjects, Users, Groups, and Schedule Tests. For example, the Themes menu displays the existing themes in the system (fig. 2(a)). The user can create, edit, or delete topics. In the create/edit menu (fig. 2(b)), the user is provided with only a field for the name of the topic. Based on the created data, a test will be created that students can take and receive an assessment and comments on their mistakes, which allows them to understand not only how much they know about a particular topic, but also what points they have problems with.



(a) (b)

Figure 2 - Themes menu: (a) view of the Themes menu; (b) view of the modal menu for creating/editing a theme

Thus, the developed application shows the feasibility of combining .NET technology and the Angular framework. This application is recommended to be used in the educational process when studying disciplines related to relational databases. It will be especially useful for universities, specialized online courses where students are grouped and have a schedule.

The work is funded by the EU NextGenerationEU through the Recovery and Resilience Plan for Slovakia under the project No. 09I03-03-V01-00115.

References:

[1] Skeet, J., & Lippert, E. (2019). C# in depth. Shelter Island, NY: Manning Publications.

[2] Seshadri, S. (2018). Angular: UP and running learning angular, step by step. Beijing, Boston, Farnham, Sebastopol, Tokyo: O'Reilly.

[3] Yakovleva, O., Kovač, M., Ardasov, V. & Yeremenko, I. (2023). Study on adding functionality to the Zoom online conference system for monitoring the participant activities. Public Administration and Regional Development, 19(1), pp. 158–184.

NEURAL NETWORK FRAMEWORKS FOR ARCHITECTURAL DRAWINGS

Toots R., Shapovalova O.
E-mail: toots.robert@hneu.net, olena.shapovalova@hneu.net
Kharkiv, Simon Kuznets Kharkiv National University of Economics

Working with architectural drawings through neural networks involves tasks like image recognition, pattern detection, and even generation of new designs based on learned patterns. Several frameworks and libraries are designed to facilitate these processes, leveraging deep learning and computer vision technologies. Here's a list of prominent neural network frameworks and tools that can be effectively used for working with architectural drawings.

TensorFlow [1] is an open-source library developed by the Google Brain team (fig. 1). It's widely used for numerical computation and large-scale machine learning. TensorFlow's flexible architecture allows for the deployment of computation across various platforms (CPUs, GPUs, TPUs), facilitating the development of models directly on architectural drawings for tasks like segmentation, detection, and classification.

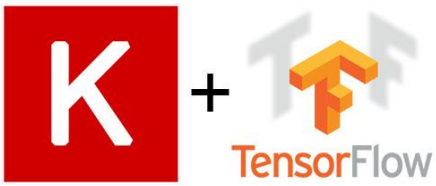


Figure 1. Keras and TensorFlow

PyTorch [2] is an open-source machine learning library based on the Torch library, and it's used for applications such as computer vision and natural language processing (fig. 2). Its dynamic computational graph enables a flexible and intuitive approach to building neural networks, making it suitable for processing architectural drawings, where the adaptability to changing design parameters is crucial.

Figure 2. PyTorch

Keras [3] is an open-source neural-network library written in Python (fig. 1). It's designed to enable fast experimentation with deep neural networks. Keras runs on top of TensorFlow, Theano, or Microsoft Cognitive Toolkit (CNTK), providing a high-level, user-friendly interface for building and training models. It's particularly useful for prototyping and experimenting with neural networks for architectural drawings.

OpenCV (Open-Source Computer Vision Library) [4], although not a neural network framework per se, OpenCV is a library of programming functions mainly aimed at real-time computer vision (fig. 3). It's used extensively for image and video analysis, including face recognition, motion tracking, and object detection. For architectural drawings, OpenCV can preprocess images for neural network models, perform feature extraction, and even handle tasks like automated drawing and layout recognition.

Figure 3. OpenCV and FastAI

FastAI [5] is built on top of PyTorch and is designed to make deep learning more accessible by providing higher-level components for building and training neural networks (fig. 3). FastAI simplifies the process of applying modern best practices to your models and data, such as architectural drawings, making it easier to achieve high-quality results without needing to be an expert in machine learning.

Detectron2 [6] developed by Facebook AI Research (FAIR), Detectron2 is an open-source platform for object detection and segmentation (fig. 4). It is built using PyTorch and is designed to be modular and scalable. Detectron2 is particularly useful for tasks in architectural drawings that involve detecting specific features or objects within complex layouts.

Figure 4. Detectron2 and AutoCAD

AutoCAD's AI and Machine Learning Features [7]. While AutoCAD itself is not a neural network framework, it's worth mentioning because of its integration of AI and machine learning features for architectural design (fig. 4). These features can automate tasks, suggest design modifications, and optimize workflows. Although these artificial intelligence enhancements are proprietary, they are increasingly becoming part of mainstream architectural drafting and design software.

References:

[1] Official TensorFlow Blog. [Electronic resource]. – Access mode to a resource: https://blog.tensorflow.org/

[2] PyTorch Official Tutorials. , Deep Learning Torch (Book by Eli Stevens, Luca Antiga, and Thomas Viehmann). [Electronic resource]. – Access mode to a resource: https://pytorch.org/tutorials/

[3] Deep Learning with Python (Book by François Chollet, the creator of Keras), Keras Official Documentation. [Electronic resource]. – Access mode to a resource: https://keras.io/

[4] Learning OpenCV 4 Computer Vision with Python 3 (Book by Joseph Howse, Joe Minichino), OpenCV Official Tutorials. [Electronic resource]. – Access mode to a resource: https://docs.opencv.org/4.x/d9/df8/tutorial_root.html

[5] Practical Deep Learning for Coders (Online course by FastAI). [Electronic resource]. – Access mode to a resource: https://course.fast.ai/

[6] Detectron2 Official Documentation. [Electronic resource]. – Access mode to a resource: https://detectron2.readthedocs.io/en/latest/

[7] Autodesk Universit. [Electronic resource]. – Access mode to a resource: www.autodesk.com/autodesk-university/search?fields.year=2023&fields.topic=Machine+Learning

AN OVERVIEW OF THE POPULAR FREE RESOURCES FOR TASK PLANNING AND SYSTEM MANAGEMENT

Yenhalychev S.O., Semenov S.G., Leunenko O.V.
E-mail: serhii.yenhalychev@hneu.net, serhii.semenov@hneu.net, oleksii.leunenko@hneu.net
Kharkiv, Simon Kuznets Kharkiv National University of Economics

Introduction. Currently, the enhancement of computational facility efficiency continues to be a pressing concern. This challenge gains significance due to the heightened need for communication and computing services across a wide range of computer systems, encompassing both industrial and general-use formats. Frequently, the approach to addressing this issue involves integrating computer and communication resources of varied scales and functions into a singular computing system.

Developers employ a range of software tools for the practical deployment of such systems. Such software are: Grid Engine, Apache YARN (Yet Another Resource Negotiator) and HTCondor.

The objective of this report is to conduct an analysis and comparative examination of commonly used task planners and systems management resources.

Grid Engine. The Sun Grid Engine (SGE) software has emerged as a pivotal tool in the realm of task planning and systems management resources, particularly in grid computing environments. Its design aligns with the evolving requirements of distributed computing, offering an integrated approach to manage complex computational tasks and resources, available free for a trial period.

The Sun Grid Engine is a distributed resource management (DRM) software that effectively manages and schedules tasks across a cluster of computers. Developed by Sun Microsystems, SGE enables users to harness the collective power of multiple computers within a network, making it a cornerstone in computational grid environments [1]. It facilitates the pooling of resources, allowing for the efficient execution of complex and resource-intensive tasks.

Advantages of Sun Grid Engine

1. Resource Optimization: SGE excels in optimizing the use of available computational resources, reducing idle times and improving overall efficiency.
2. Dynamic Resource Allocation: It supports the dynamic addition of computing resources, adapting to changing workload demands.
3. Advanced Job Scheduling: SGE's advanced job scheduling capabilities ensure that resources are allocated effectively to meet various computational needs.
4. Fault Management: The software includes mechanisms for managing faults and ensuring consistent performance even in complex grid environments.

Disadvantages of Sun Grid Engine

1. Complexity in Configuration and Management: Setting up and managing SGE can be complex, requiring specialized knowledge.
2. Limited Flexibility in Certain Environments: In some scenarios, SGE may offer limited flexibility compared to other grid management tools.
3. Dependence on Network Stability: SGE's performance is heavily reliant on network stability and connectivity.

The Sun Grid Engine represents a significant advancement in the field of distributed computing, offering robust solutions for task planning and systems management. While it offers numerous advantages in terms of resource optimization and job scheduling, its complexity and network dependence pose challenges. Nonetheless, as computational demands continue to grow, tools like SGE will remain vital in harnessing the power of distributed computing resources.

Apache YARN. Apache YARN (Yet Another Resource Negotiator) is a key component of the Apache Hadoop(is a collection of open-source software utilities) ecosystem, playing a crucial role in resource management and task scheduling. As a cornerstone of modern data processing, YARN facilitates efficient utilization of computing resources, offering a scalable and robust platform for handling big data applications.

Apache YARN is a resource management and job scheduling technology that orchestrates the processing of large datasets across a distributed environment. It separates the task of resource management from job scheduling and execution, allowing for more flexible data processing workflows beyond the traditional MapReduce model. YARN consists of a central Resource Manager, Node Managers on each cluster node, and Application Masters for each application [2].

Advantages of Apache YARN

1. Scalability and Flexibility: YARN efficiently scales to thousands of nodes and can handle diverse workloads, adapting to the changing resource requirements of different applications.

2. Improved Resource Utilization: By decoupling job scheduling from resource management, YARN optimizes the use of cluster resources, leading to better overall system performance [3].

3. Support for Multiple Data Processing Frameworks: YARN supports various data processing frameworks like MapReduce, Apache Spark, and Apache Tez, enhancing its versatility.

Disadvantages of Apache YARN

1. Complex Configuration and Management: Setting up and tuning YARN can be complex, requiring in-depth knowledge of the Hadoop ecosystem and its components.

2. Dependency on Network Stability: Like many distributed systems, YARN’s performance is dependent on the underlying network infrastructure’s reliability and stability.

3. Learning Curve: New users may find it challenging to navigate YARN's extensive features and capabilities.

Apache YARN has significantly transformed the landscape of big data processing, offering a dynamic and efficient way to manage computational resources in distributed environments. While it brings numerous advantages in terms of scalability, flexibility, and resource utilization, the complexities associated with its configuration and management can pose challenges. As big data continues to evolve, YARN’s role as a critical component in the Hadoop ecosystem is likely to grow, underscoring the need for continued development and optimization of this powerful tool.

HTCondor. HTCondor is a high-throughput computing (HTC) software framework designed for distributed job scheduling, making it an essential tool in scientific research environments. It is particularly well-suited for managing large volumes of batch jobs and computational tasks.

Developed by the University of Wisconsin-Madison, HTCondor is designed for the efficient utilization of distributed computing resources. It enables users to submit jobs from a local machine to other machines that are part of the HTCondor pool, thereby leveraging idle computing power across a network. Its key feature is the ability to dynamically allocate resources based on the job requirements and current load, optimizing overall computational throughput.

Advantages of HTCondor

1. Efficient Resource Utilization: HTCondor excels in maximizing the use of available computational resources, especially in environments with a large number of idle or underutilized computers [4].

2. Scalability: It is highly scalable, capable of managing large clusters and diverse job requirements efficiently.

3. Flexibility: HTCondor supports a variety of job types and can be configured to meet specific scheduling policies and requirements.

Disadvantages of HTCondor

1. Complex Setup and Management: Setting up and managing HTCondor can be complex, requiring technical expertise, particularly in large-scale deployments.

2. Limited Support for HPC Workloads: While excellent for HTC tasks, HTCondor may not be optimal for high-performance computing (HPC) workloads that require rapid inter-process communication.

3. Dependency on Network Infrastructure: Effective performance of HTCondor relies heavily on the underlying network stability and configuration.

HTCondor is a powerful and versatile tool in the field of scientific computing, offering efficient and scalable management of distributed computational tasks. Its ability to harness idle computing resources makes it particularly valuable for research institutions with large computational demands. Despite its complexity and some limitations in HPC contexts, HTCondor remains a popular choice for managing high-throughput computing tasks in various scientific fields.

Conclusions. In a review of prevalent task scheduling systems, the following conclusions were drawn:

1. Existing task planning systems generally depend on knowing the resource needs of data processing tasks, transferring the burden of optimization to the user.

2. Planning approaches that don't require knowledge of resource demands suffer from a critical limitation: they fail to consider the combined requirements of task groups, leading to suboptimal resource utilization.

3. A majority of these methods and tools do not account for uncertainties in input data or the diverse nature of real-world processes and services.

4. Consequently, there is a pressing need to develop innovative task scheduling methods for Distributed Systems Design Studios (DSDS) under conditions of incomplete information about resource requirements. These methods aim to enhance the efficiency of applied data processing tasks.

HTCondor is developed by the HTCondor team at the University of Wisconsin–Madison and is freely available for use. HTCondor follows an open-source philosophy and is licensed under the Apache License 2.0.[5]

References

[1] Sun Grid Engine: towards creating a compute power grid,W. Gentzsch Sun MicroSystems Inc.orporate, Palo Alto, CA, USA Published in: Proceedings First IEEE/ACM International Symposium on Cluster Computing and the Grid, 15-18 May 2001, [Electronic resource]. – Access mode to a resource: https://ieeexplore.ieee.org/document/923173/

[2] Apache Hadoop YARN: yet another resource negotiator, Vinod Kumar Vavilapalli, A. C. Murthy, C. Douglas, S. Agarwal, M. Konar, R. Evans, Thomas Graves, J. Lowe, Hitesh Shah, S. Seth, Bikas Saha, C. Curino, Owen O'Malley, S. Radia, B. Reed, Eric Baldeschwieler, SOCC '13: Proceedings of the 4th annual Symposium on Cloud Computing, October 2013, Article No.: 5, P. 1–16. [Electronic resource]. – Access mode to a resource: https://doi.org/10.1145/2523616.2523633

[3] Admission control in YARN clusters based on dynamic resource reservation, Yi Yao, Jason H. Lin, Jiayin Wang, N. Mi, B. Sheng, Published in: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), Date of Conference: 11-15 May 2015, [Electronic resource]. – Access mode to a resource: https://ieeexplore.ieee.org/document/7140389

[4] Lark: An effective approach for software-defined networking in high throughput computing clusters, Zhe Zhang, B. Bockelman, D. Carder, T. Tannenbaum, Future Generation Computer Systems, Volume 72, July 2017, P. 105-117. [Electronic resource]. – Access mode to a resource: https://doi.org/10.1016/j.future.2016.03.010

[5] HTCondor - License Information. [Electronic resource]. – Access mode to a resource: research.cs.wisc.edu.

ТЕХНОЛОГІЇ РОЗПОДІЛЕНИХ СХОВИЩ ДАНИХ

Алексієв В.О.
E-mail: vlax@hneu.edu.ua
Харків, Харківський національний економічний університет імені Семена Кузнеця

Зараз світ інформаційних технологій стрімко змінюється та надає бізнесу все більше уніфікованих й надійних рішень для побудови засобів реєстрації, обробки, аналізу і зберігання даних. В залежності від визначених завдань IT-підрозділ може обрати модель обчислень на замовлення: публічна хмара, мультихмара, приватне або гібридне рішення. Такий вибір моделі споживання ресурсів повинен враховувати особливості забезпечення безпеки даних.

У своєї більшості, основою для побудови надійних засобів обробки даних є розгортання сховища даних (рис. 1), як базової платформи щодо забезпечення усіх виконуваних дій з даними. Традиційно у якості такого сховища даних розглядають рішення SAN (Storage Area Network) або NAS (Network Attached Storage) [1].

Побудову сховища даних можна розглянути на рівні приватної хмари. З погляду на пропозиції на ринку серверних рішень слід виділити два сталих підходи для організації приватного центра обробки даних (ЦОД) – застосування найнадійніших новітніх серверних рішень, поруч із концепцією – застосування серверних платформ, що вже уходять з ринку, однак, мають різноманіття доступних взаємозамінних блоків й компонентів для оновлення у разі виходу їх з ладу. Звичайно це впливає на ціну побудови ЦОД та його обслуговування. Поруч із цим можна виділити «мікрорівень» – сам сервер, що буде виконувати роль вузла для збереження даних. Для окремого вузла важливим є баланс надійності збереження даних та швидкості їх запису, які доцільно забезпечити за допомогою RAID-масиву.

Можна провести аналогію відповідно до побудови ЦОД, так й на рівні сервера є вибір у формуванні апаратних чи програмних RAID [2]. Звичайно, програмні рішення є гнучкими та вирішують проблему відновлення RAID-масиву, наприклад, на відміну застосування апаратного RAID який може вийти з ладу в наслідок поламки RAID-контролеру. Також, оскільки програмний RAID підтримується майже всіма сучасними операційними системами, можливе його застосування й на рівні публічної хмари для підвищення швидкодії дискових операцій (завдяки розпаралелювання операцій запису) тощо.

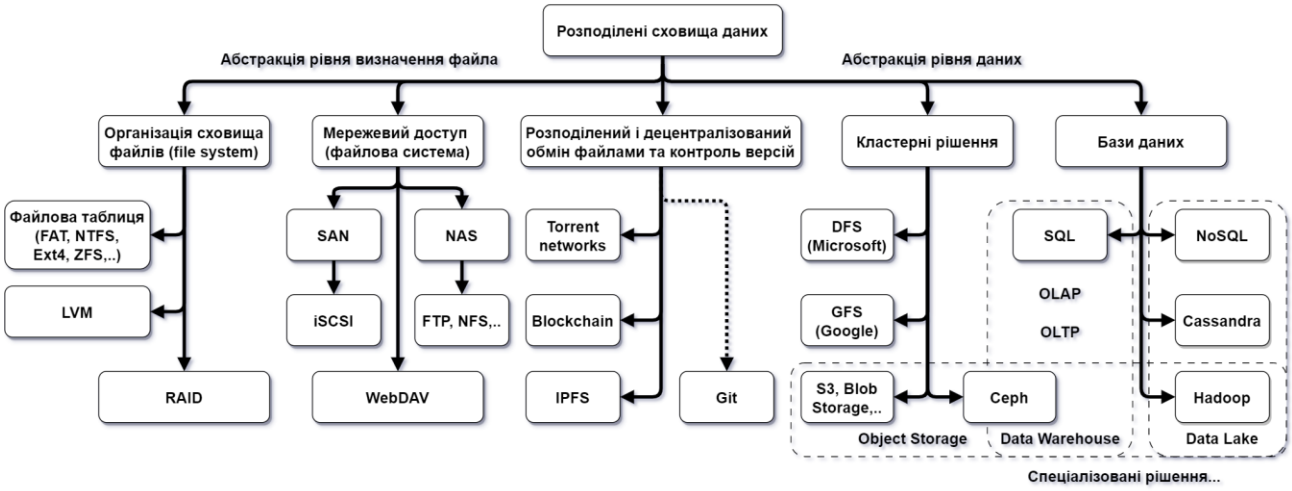


Рисунок 1 – Технології побудови сховищ даних

Абстракція побудови сховища даних рівня файлової системи дозволяє ефективно застосовувати дискові підсистеми на рівні серверу. Також, на цьому рівні слід виділити крім RAID ще застосування файлової системи OpenZFS [3].

Для побудови розподіленого сховища даних SAN/NAS на базі вільного програмного забезпечення можна відзначити застосування системи TrueNAS CORE або аналогів [4]. Також TrueNAS реалізує підтримку ZFS. Таким чином, TrueNAS CORE цікава як для розгортання на окремому сервері, так й у хмарному середовищі – для швидкого отримання сховища даних на базі потрібної технології мережевого доступу та зі зручною веб-панеллю управління.

На відміну від традиційних систем SAN/NAS слід розглянути реалізацію доступу до файлів за допомогою протоколу WebDAV та рішень Nextcloud (https://nextcloud.com/), ownCloud (https://owncloud.com/). Ці технології надають кінцевому користувачу абстракцію у вигляді представлення своєрідного віддаленого дискового накопичувача.

Поруч з цим, розподілені системи контролю версій можна вважати певною реалізацією розподіленого сховища даних, наприклад, у разі застосування методики DevOps та доставки коду застосунка на сервер за допомогою git (https://git-scm.com/) та ін.

Розподілені сховища даних це не тільки централізовані системи, існує чимало прикладів реалізації децентралізованих систем, серед яких слід виділити технології Blockchain [5].

В даний час багато завдань з обробки даних основані на BigData, тому побудова та застосування сучасних систем збереження даних потребує все більших обсягів сховищ інформації. Безумовно у цьому секторі, який можна вже назвати – «абстракція рівня даних», а одним із найцікавіших рішень на цьому рівні є система CEPH [6].

Ceph (https://ceph.io/) є відкритим рішенням для побудови кластеру збереження даних у масштабі петабайт. Ця технологія фактично є універсальним рішенням яке підтримує файлову систему CephFS, що сумісна з POSIX. Системи Linux можуть монтувати файлові системи CephFS нативно, через клієнт на основі FUSE або через шлюз NFSv4.

На базі Ceph можна побудувати блокове сховище RBD (RADOS Block Device). Це надає можливості у середовищі віртуалізації або приватної чи загальнодоступній хмарі отримати віртуальні диски, як об’єкти в кластері, розподіляючи дані та робоче навантаження між усіма доступними вузлами для надзвичайної масштабованості та продуктивності. Фактично це є аналогом SAN, однак, у цьому разі на низькому рівні застосується не концепція RAID, а програмна реалізація розподіленої системи кластеру для збереження даних.

Також кластер Ceph дозволяє отримати доступ до даних у форматі об’єктів, сформував, так зване, об’єктне сховище. У разі такого застосування кластеру можна отримати доступ до даних за API (Application Programming Interface), який є сумісний з Amazon S3 або OpenStack Swift API. Це відкриває достатньо можливостей для побудови гібридних хмар та прозорої міграції між приватною та публічною хмарою. Фактично Ceph на сьогодні підтримується, як основа для розподіленого сховища даних, так й побудови приватної хмари на базі рішень Proxmox VE (https://www.proxmox.com/en/proxmox-virtual-environment/overview) або OpenStack (https://www.openstack.org/), що робить це рішення достатньо універсальним.

Поруч з рішенням завдань побудови розподілених сховищ даних з абстракцією для користувача системи, у якості традиційного представлення файлової системи або об’єктного сховища, слід виділити клас завдань, які потребують застосування рішень керування базами даних. Такі системи надають можливості ефективного пошуку та представлення даних. Наприклад, завдання побудови традиційних сховищ даних (Data Warehouse) будуються на засобах що дозволяють на найвищому рівні представлення даних виконати завдання, наприклад, побудови OLAP (On-Line Analytical Processing) моделі. Звичайно це є рішеннями на базі реляційних баз даних. Однак, сховище даних – це все ж таки більше про потоки даних, онлайнова обробка транзакцій, тобто задіяння процесів OLTP (Online Transaction Processing). Такі рішення притаманні стандартизованим системам оперативного аналізу та прийняття рішень [7, 8].

Сучасні реалі конкурентного ринку вимагають від підприємств та організацій обробки вже не стандартизованих потоків даних. Такими даними стають звичайні документи або дані від систем IoT (Internet of things) тощо. Для збереження відповідних до предметної діяльності даних підходять як нереляційні бази даних так й об'єктні сховища. Побудова озер даних (Data Lakes) продиктована реаліями сучасності та швидкістю отримання первинних даних. В свою чергу, озера даних можуть містити й структуровані дані. Фактично абстракція сприйняття рівня даних вже є притаманної для хмарних рішень.

Виконаний аналіз можливості створення розподілених сховищ даних дозволяє визначити, що якоїсь загальної технології побудови відповідних платформ не існує. Для кожного типу завдань є свої рішення. Як верхній ступень уніфікації процесів та накопичення даних зараз стає технологія побудови озера даних. У свою чергу, якщо подивитися на рівень абстракції рівня «файлової системи», то можна побачити, що на сьогодні це є вже другорядним завданням. У рішеннях, що вбудовуються (Embedded systems), інтернету речей та ін. є стійка тенденція до застосування технологій граничних (периферійних) обчислень (Edge Computing) [9]. Для виконання таких завдань обчислень, як змога ближче до постачальників даних, не є критичним сприйняття даних у вигляді файлів. Тобто для цих застосувань можна задіяти API, наприклад, сумісний з Amazon S3, що дозволить застосовувати програмні рішення хмари, однак, вже на «границі». Тому у цьому сегменті дуже перспективним є рішення компанії Canonical – MicroCeph [10]. За допомогою Snap-репозиторію (автономні та прості в установці пакети Linux програм) можна розгорнути кластер Ceph на одному чи трьох вузлах у декілька хвилин. Це скриває межу між побудовою великого кластеру та мікро-рішенням для завдань граничних обчислень.

Таким чином, на сьогодні технологія Ceph стає своєрідним уніфікованим рішенням для побудови розподілених сховищ даних від рівня систем інтернету речей та граничних обчислень до побудови кластерів для забезпечення завдань озер даних. При цьому не слід забувати про інші рішення для побудови розподілених сховищ даних, із врахуванням поглядів на завдання, які потребують абстракції файлової системи або представлення даних.

Література

[1] What's the Diff: NAS vs. SAN [Електронний ресурс]. – Режим доступу: https://www.backblaze.com/blog/whats-the-diff-nas-vs-san/

[2] Software RAID or hardware RAID: what's better in 2024? [Електронний ресурс]. – Режим доступу: https://xinnor.io/blog/software-raid-or-hardware-raid-whats-better-in-2023/

[3] ZFS for Dummies [Электронный ресурс]. – Режим доступа к ресурсу: https://blog.victormendonca.com/2020/11/03/zfs-for-dummies/

[4] Best Alternatives To TrueNAS [Электронный ресурс]. – Режим доступа к ресурсу: https://youprogrammer.com/truenas/

[5] Кравченко П. Блокчейн і децентралізовані системи: навч. посібник для студ. закладів вищ. освіти: в 3 частинах. Ч. 2 / П. Кравченко, Б. Скрябін, О. Курбатов, О. Дубініна. – Харків, 2019. – 412 с.

[6] Ceph storage on VMware [Электронный ресурс]. – Режим доступа к ресурсу: https://ubuntu.com/blog/ceph-storage-on-vmware

[7] What's the Difference Between a Data Warehouse, Data Lake, and Data Mart? [Электронный ресурс]. – Режим доступа к ресурсу: https://aws.amazon.com/compare/the-difference-between-a-data-warehouse-data-lake-and-data-mart/

[8] Озеро даних vs Сховище даних [Электронный ресурс]. – Режим доступа к ресурсу: https://cloudfresh.com/ua/cloud-blog/ozero-danih-vs-shovishhe-danih/

[9] Що необхідно для сталого розвитку граничних обчислень? [Электронный ресурс]. – Режим доступа к ресурсу: https://ko.com.ua/shho_neobhidno_dlya_stalogo_rozvitku_granichnih_obchislen_145214

[10] Cloud storage at the edge with MicroCeph [Електронний ресурс]. – Режим доступу: https://ubuntu.com/blog/cloud-storage-at-the-edge-with-microceph

СТВОРЕННЯ ПРОГРАМНОГО КОНСУЛЬТАНТА НА ОСНОВІ БЕЗКОШТОВНИХ ІНСТРУМЕНТІВ ДЛЯ РОЗРОБКИ

Бейник В.А., Льовкін В.М.
E-mail: vliovkin@gmail.com
Запоріжжя, Національний університет «Запорізька політехніка»

Створення програмного консультанта, який здатен відповідати на запитання користувача щодо переліку завдань або іншої подібної інформації, може мати позитивний вплив на ефективність та зручність роботи з даними в різних сферах практичного застосування. Такі програмні рішення дозволяють користувачам отримувати швидкий доступ до необхідної інформації, вводячи запитання або команди замість того, щоб вручну виконувати запити до бази даних. Це полегшує і прискорює взаємодію з базою даних. Замість наявності графічного інтерфейсу консультант може оброблювати природну мову користувача, що робить взаємодію більш природною та доступною. У підсумку реалізація такого програмного рішення може значно полегшити взаємодію з базою даних, забезпечуючи ефективний та зручний доступ до інформації про завдання, що розглянуто в даній роботі, для користувачів на різних рівнях технічної підготовки.

Для розробки програми-консультанта було використано мову програмування Python, пакет обробки природної мови NLTK [1], а в якості системи керування базами даних використано MySQL, тобто всі обрані та використані в підсумку рішення є виключно безкоштовними.

Під час реалізації програмного рішення було створено інтерфейс взаємодії з користувачем, який передбачає постановку запитань природною українською мовою, при цьому програма аналізує поставлені запитання за допомогою засобів бібліотеки NLTK, яке є основним використовуваним зовнішнім інструментом.

Natural Language Toolkit (NLTK) – це бібліотека для обробки природної мови, яка надає можливості простої і зручної інтеграції в програми мовою програмування Python [1]. NLTK має широкий набір функцій, спрямованих на роботу з текстом. Бібліотека є потужним інструментом для дослідження та вирішення завдань у сфері обробки природної мови. Вона широко використовується в науці про дані, мовному аналізі, машинному навчанні та інших областях, де необхідно аналізувати та розуміти текст.

Алгоритм використання бібліотеки NLTK базується на використанні токенізації (розділенні на слова) введеного користувачем запитання або запитань для подальшої їх фільтрації від стоп-слів (слова, які відкидаються з запитання, бо вони не несуть значущості для пошуку). Далі виділені слова використовуються у генеруванні SQL-запитів до таблиць з необхідними даними (або усіма завданнями в розробленій програмі). Програма відшукує у записах слова, які схожі на ті, що задані у питанні, та якщо є збіг, то система виводить дане завдання користувачеві.

Оскільки для роботи консультанта використано українську мову, то бібліотека за замовчуванням не здатна опрацювати всі характерні ситуації, може виникати неправильна обробка слів та їх фільтрація, зокрема відсутність стоп-слів українською мовою, тому для подолання цієї проблеми було створено і додано текстовий файл зі стоп-словами до файлів бібліотеки, зокрема для коректного їх видалення при фільтрації слів запитання.

У підсумку програма є простою у використанні, проте може потребувати налаштування стосовно підключення до бази даних, адже при цьому мають бути правильно встановлені конфігураційні параметри підключення до бази даних MySQL та встановлено бібліотеку mysql.connector для роботи з базою даних.

Література

[1] NLTK: Natural Language Toolkit. [Electronic resource]. – Access mode: https://www.nltk.org/

ВИКОРИСТАННЯ GOOGLE COLABORATORY ДЛЯ ПОБУДОВИ МОДЕЛЕЙ ШТУЧНОГО ІНТЕЛЕКТУ

Блиндарук А.О.
Керівник: Шаповалова О.О.
E-mail: blindaruk@yahoo.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Google Colaboratory, часто називаний Colab, є революційним інструментом, розробленим Google для сприяння науковим дослідженням та розробці у сфері штучного інтелекту (ШІ) та машинного навчання (МН) [1]. Він надає дослідникам високопродуктивне обчислювальне середовище, доступне з будь-якого веб-браузера, що значно спрощує процес розробки та тестування моделей ШІ.

В свою чергу Google Colaboratory - це безкоштовний хмарний сервіс, який базується на Jupyter Notebooks і дозволяє користувачам писати та виконувати код Python у веб-браузері. Однією з ключових особливостей Colab є можливість легко ділитися кодом, аналізами та звітами з колегами, що робить його ідеальним інструментом для співпраці у сфері досліджень ШІ.

Google Colaboratory оснащений рядом функцій, що спрямовані на підтримку розробки моделей ШІ [1]. Воно надає доступ до потужних обчислювальних ресурсів, включаючи графічні процесори (GPU) та тензорні процесорні одиниці (TPU), які є критично важливими для тренування складних моделей МН [2].

Colab також інтегрований з Google Drive та іншими сервісами Google, що полегшує зберігання та обмін даними.

GPUs відомі своєю здатністю обробляти паралельні завдання, що робить їх ідеальними для тренування моделей машинного навчання, які вимагають великої кількості математичних обчислень. Google Colaboratory надає доступ до наступних типів GPU:

NVIDIA Tesla K80: Один з найбільш розповсюджених варіантів у Colab, підходить для різноманітних завдань машинного навчання [3].

NVIDIA Tesla T4: Цей GPU пропонує вищу продуктивність порівняно з K80 та підтримку технології Ray Tracing для високоякісних візуалізацій.

NVIDIA Tesla P100: Цей високопродуктивний GPU ідеально підходить для складніших завдань машинного навчання і глибокого навчання.

Використання GPU в Colab може значно прискорити процес тренування моделей, порівняно з використанням традиційних центральних процесорів (CPU) [3].

TPUs є спеціалізованими чіпами, розробленими Google для оптимізації роботи з тензорами, які є основою багатьох алгоритмів машинного навчання. Використання TPU в Colab може надати наступні переваги:

Висока продуктивність: TPU можуть обробляти мільйони операцій з тензорами за секунду, що робить їх надзвичайно ефективними для тренування моделей глибокого навчання.

Ефективність витрат: Завдяки високій продуктивності та ефективності, TPU можуть значно зменшити час та вартість тренування моделей.

Масштабованість: TPU підтримують масштабованість, дозволяючи ефективно розподіляти обчислення для тренування великих та складних моделей.

Google Colaboratory є незамінним інструментом для наукових дослідників в галузі штучного інтелекту завдяки його доступності, гнучкості та масштабованості. Він дозволяє дослідникам швидко експериментувати з різними моделями та параметрами, не турбуючись про обмеження локальних обчислювальних ресурсів.

Безкоштовний доступ до GPU та TPU робить Colab особливо привабливим для стартапів та індивідуальних дослідників, що не мають значних обчислювальних ресурсів.

Багато провідних технологічних компаній та стартапів активно використовують Google Colaboratory у своїй роботі над проєктами ШІ. Це включає розробку прототипів, тестування

нових алгоритмів, аналіз даних та навіть проведення навчальних курсів та воркшопів. Використання Colab сприяє зменшенню витрат на обчислювальні ресурси та полегшує співпрацю в команді.

Google Colaboratory представляє собою потужний інструмент, який може значно прискорити дослідження та розробку в області штучного інтелекту. Його легкість у використанні, співпраця в реальному часі, доступ до високопродуктивних обчислювальних ресурсів роблять його ідеальним вибором для науковців у будь-якій області. Для максимальної ефективності рекомендується активно використовувати можливості спільної роботи та інтеграції з іншими сервісами Google.

Google Colaboratory також надає різноманітні можливості для інтеграції з базами даних, сторонніми сервісами та системами управління версіями, такими як GitHub [4] . Ці можливості роблять Colab особливо потужним інструментом для розробників і дослідників у галузі штучного інтелекту, дозволяючи їм легко зберігати, обробляти та аналізувати великі обсяги даних, а також співпрацювати над проектами.

Colab дозволяє підключатися до різних типів баз даних, включаючи реляційні бази даних (наприклад, MySQL, PostgreSQL) та NoSQL бази даних (наприклад, MongoDB). Використовуючи бібліотеки Python, такі як sqlalchemy для реляційних баз даних або pymongo для MongoDB, користувачі можуть легко виконувати запити до баз даних прямо зі своїх ноутбуків Colab. Це дозволяє ефективно виконувати завантаження, обробку та аналіз даних у рамках їх дослідницьких проєктів [5].

Google Colaboratory надає вбудовану підтримку для GitHub, дозволяючи користувачам легко імпортувати ноутбуки з GitHub, а також зберігати зміни в ноутбуках прямо до репозиторіїв на GitHub. Ця інтеграція спрощує співпрацю над проектами, дозволяючи командам ефективно управляти версіями своїх ноутбуків і спільно працювати над кодом. Для імпорту ноутбука з GitHub достатньо лише вказати посилання на нього в Colab, а для збереження змін — використовувати інтерфейс GitHub у самому Colab.

Науковцям також слід бути уважними до обмежень безкоштовної версії, таких як час обмеження сесії, і розглянути можливість переходу на платну версію для доступу до додаткових ресурсів та функцій.

Google Colaboratory є цінним ресурсом, який відкриває нові можливості для досліджень та інновацій у сфері штучного інтелекту, забезпечуючи легкий доступ до потужних обчислювальних ресурсів та інструментів співпраці.

Литература

[1] Офіційна документація Google Colaboratory. [Electronic resource]. – Access mode: https://colab.research.google.com/notebooks/intro.ipynb

[2] TensorFlow документація по використанню TPU. [Electronic resource]. – Access mode: https://www.tensorflow.org/guide/tpu

[3] NVIDIA документація по GPU для глибокого навчання. [Electronic resource]. – Access mode: https://developer.nvidia.com/deep-learning

[4] GitHub та Colaboratory інтеграція. [Electronic resource]. – Access mode: https://help.github.com/en/github/managing-files-in-a-repository/working-with-jupyter-notebook-files-on-github

[5] Efficiently Scaling up Deep Learning Computations to Tens of Thousands of Cores. [Electronic resource]. – Access mode: https://ai.googleblog.com/2019/04/efficiently-scaling-up-deep-learning.html

DJANGO – ВИСОКОРІВНЕВИЙ ВЕБ – ФРЕЙМВОРК ДЛЯ PYTHON

Буренко Я.Д.
Керівник: Міхєєв І.А.
E-mail: yaroslav2003003@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Django - це відкритий високорівневий веб-фреймворк для мови програмування Python, який використовується для швидкої та ефективної розробки веб-додатків. Фреймворк надає готовий стартовий пакет для створення веб-додатків, де вже налаштовані багато базових елементів, що дозволяє розробникам зосередитися на бізнес-логіці своїх додатків, а не на налаштуванні низькорівневих складових.

Django був розроблений і вперше опублікований у 2005 році. Розробка Django почалася у Lawrence Journal-World, американській газеті, де виникла необхідність в швидкому та ефективному створенні веб-додатків для новинного веб-сайту. Група розробників, включаючи Адаріана Холовати, Саймона Вілліса, Джарада Реймонда та Голтон Барлоу, працювала над створенням фреймворку.

Django має численні особливості, які роблять його популярним:

1. ORM (Object-Relational Mapping): Django надає вбудований ORM, що дозволяє розробникам взаємодіяти з базами даних за допомогою об'єктно-орієнтованого підходу, спрощуючи роботу з базами даних.
2. MVC та MTV архітектура: Django використовує архітектуру MTV (Model-Template-View), що відповідає концепції Model-View-Controller (MVC), розділяючи логіку додатку на моделі (бізнес-логіка), шаблони (відображення) та представлення (логіка перегляду).
3. Адміністративний інтерфейс: Django надає вбудований адміністративний інтерфейс, який автоматично генерується для адміністрування додатків, що спрощує керування даними та їх моніторинг.
4. Шаблонізація та вигляди: Django використовує шаблони та вигляди для відокремлення логіки відображення від бізнес-логіки, дозволяючи легше управляти інтерфейсом.
5. Розширюваність через додатки: Django підтримує використання додатків, що дозволяє вам легко використовувати та розширювати функціональність вашого проекту.
6. Безпека: Django має вбудовані засоби для запобігання багатьом видам атак, таким як внедрення коду та атаки CSRF (Cross-Site Request Forgery).
7. RESTful підтримка: За допомогою додатку Django REST framework, Django спрощує розробку RESTful веб-сервісів та API.
8. Локалізація: Django має вбудовану підтримку для роботи з різними мовами та культурами, що полегшує розробку міжнародних проектів.

Підсумовуючи, Django - це потужний веб-фреймворк для Python, призначений для ефективної розробки веб-додатків. Використовуючи вбудовані інструменти, такі як ORM, адміністративний інтерфейс, система шаблонів та заходи безпеки, Django дозволяє розробникам фокусуватися на основній бізнес-логіці своїх проектів. Він також забезпечує швидке розгортання і підтримку розширюваності, що робить його ідеальним вибором для створення веб-додатків.

Література

[1] Django Packages : Reusable apps, sites and tools directory for Django. [Electronic resource]. – Access mode: https://djangopackages.org

[2] Python Tutorials – Real Python. [Electronic resource]. – Access mode: https://realpython.com

[3] Django REST framework. [Electronic resource]. – Access mode: https://www.django-rest-framework.org

ОГЛЯД ВІДКРИТИХ РІШЕНЬ ДЛЯ DATA LAKE: ВІД МАСШТАБОВАНОСТІ ДО БЕЗПЕКИ

Венгріна О.С.
E-mail: olena.venhrina@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

У сучасному світі обробки великих даних, Data Lake відіграють ключову роль, забезпечуючи користувачам можливість зберігати та аналізувати величезні обсяги інформації в їхньому первісному вигляді. З розвитком технологій відкритого коду, інструменти для створення Data Lake стали доступнішими, пропонуючи більшу гнучкість, масштабованість та інтеграційний потенціал. Дана робота зосереджується на аналізі ключових відкритих рішень для створення Data Lake, і виконана з метою визначення, як відкриті інструменти можуть сприяти створенню ефективних та надійних Data Lake, що є критично важливим для користувачів, які прагнуть використовувати потенціал великих даних для розвитку та інновацій.

Для глибокого розуміння потенціалу рішень з відкритим кодом у створенні та управлінні Data Lake, було проведено детальний аналіз найбільш популярних засобів, які використовуються у цій сфері. В даному досліджені було виділено наступні ключові параметри: наявність відкритого коду, масштабованість, підтримка різних типів даних, можливість аналітики в реальному часі, простота використання, інтеграція з іншими системами, підтримка обчислювальних моделей, надійність, безпека та рівень підтримки спільноти.

У таблиці 1 наведені дані про властивості та функціональність найбільш розповсюджених засобів для роботи з Data Lake.

Таблиця 1 – Характеристики засобів для роботи з Data Lake

Показник	**HDFS**	**Apache Hadoop**	**Apache Spark**	**Apache Hive**	**Presto**	**Apache Kafka**	**MinIO**	**Elasticsearch**
Відкритий код	Так	Так	Так	Так	Так	Так	Так	Так
Масштабованість	Висока	Висока	Висока	Середня	Висока	Висока	Висока	Висока
Підтримка різних типів даних	Так	Так	Так	Так	Так	Обмежено	Так	Так
Аналітика в реальному часі	Ні	Ні	Так	Ні	Так	Так	Ні	Так
Простота використання	Низька	Середня	Середня	Середня	Середня	Низька	Висока	Середня
Інтеграція з іншими системами	Висока	Висока	Висока	Висока	Висока	Висока	Середня	Висока
Підтримка обчислювальних моделей	Обмежено	Обмежено	Висока	Обмежено	Обмежено	Обмежено	Обмежено	Обмежено
Надійність	Висока	Висока	Висока	Середня	Висока	Висока	Середня	Висока
Безпека	Середня	Середня	Середня	Середня	Середня	Висока	Середня	Висока
Підтримка спільноти	Висока	Висока	Висока	Висока	Середня	Висока	Середня	Висока

В контексті розвитку та використання Data Lake, особливо важливим є вибір правильних інструментів, які б відповідали специфічним потребам та вимогам проекту.

Нижче представлена табл. 2, яка, грунтуючись на даних з таблиці 1, відображає детальний порівняльний аналіз кожного з цих інструментів, демонструючи їхні сильні та слабкі сторони. Ця інформація дозволяє користувачам зробити обґрунтований вибір, враховуючи власні потреби та ресурси, а також специфіку конкретного проекту або організації.

Таблиця 2 – Переваги та недоліки інструментів відкритого коду для Data Lakes

Інструмент	Переваги	Недоліки
HDFS	Висока масштабованість, надійність	Складність в налаштуванні та обслуговуванні
Apache Hadoop	Підтримка великої кількості даних, гнучка обробка	Складність у використанні, не найшвидший для обробки даних
Apache Spark	Швидка обробка даних, підтримка машинного навчання	Вище споживання пам'яті, складнощі в масштабуванні
Apache Hive	SQL-подібний інтерфейс, легка інтеграція з Hadoop	Не підходить для обробки в реальному часі, повільність на великих обсягах
Presto	Висока швидкість обробки, інтерактивні запити	Потребує значних ресурсів для оптимальної роботи
Apache Kafka	Ефективна обробка потокових даних, висока пропускна спроможність	Складність у налаштуванні та моніторингу
MinIO	Висока продуктивність, простота використання	Менша функціональність порівняно з комерційними об'єктними сховищами
Elasticsearch	Швидкий пошук та аналітика, масштабованість	Вимагає значних ресурсів для оптимізації та управління

На основі проведеного аналізу відкритих інструментів для створення Data Lake можна зробити наступні висновки:

1. Більшість інструментів, таких як Apache Hadoop, Apache Spark, та Elasticsearch, є дуже масштабованими та підтримують різні типи даних, що робить їх універсальними рішеннями для різноманітних сценаріїв використання.

2. Деякі інструменти мають спеціалізовані застосування; наприклад, Apache Kafka ефективний для обробки потокових даних в реальному часі, тоді як Apache Hive більше підходить для батчевої обробки з використанням SQL-подібних запитів.

3. Інструменти, такі як MinIO, вирізняються своєю простотою у використанні, що може бути важливим для команд з обмеженими ресурсами або досвідом.

4. Більшість інструментів добре інтегруються з іншими системами та платформами, що дозволяє створювати складні системи обробки даних.

5. Інструменти, такі як Apache Kafka та Elasticsearch, мають високий рівень безпеки та надійності, що є критично важливим для бізнес-критичних застосувань.

6. Майже всі розглянуті інструменти мають активні спільноти, які надають значну підтримку та ресурси для навчання та вирішення проблем, що є важливим для сталого розвитку та використання цих технологій.

Вибір конкретного інструменту або комбінації інструментів для створення Data Lake залежатиме від специфічних потреб проекту, вимог до обробки та аналізу даних, а також від доступних ресурсів та технічної експертизи команди.

Литература

[1] Apache Hadoop. [Електронний ресурс]. – Режим доступу до ресурсу: https://hadoop.apache.org/

[2] Apache Spark. [Електронний ресурс]. – Режим доступу до ресурсу: https://spark.apache.org/

[3] Apache Hive – Data Warehouse Software for Reading, Writing, and Managing Large Datasets. [Електронний ресурс]. – Режим доступу до ресурсу: https://hive.apache.org/

[4] Presto Distributed SQL Query Engine for Big Data. [Електронний ресурс]. – Режим доступу до ресурсу: https://prestodb.io/

[5] Apache Kafka. [Електронний ресурс]. – Режим доступу до ресурсу: https://kafka.apache.org/

[6] MinIO | High Performance, Kubernetes Native Object Storage. [Електронний ресурс]. – Режим доступу до ресурсу: https://min.io/

ПІДВИЩЕННЯ ПРОДУКТИВНОСТІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЗАВДЯКИ БЕЗКОШТОВНИМ ІНСТРУМЕНТАМ

Глушко С.О.
E-mail: serhii.hlushko@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

У сфері розробки програмного забезпечення постійно зростає попит на ефективні та водночас економічно вигідні рішення. Одним із шляхів оптимізації процесу розробки є використання безкоштовних інструментів. Важливо розуміти, як безкоштовні інструменти можуть підвищити продуктивність розробників, а також враховувати їхні переваги та обмеження.

Безкоштовні інструменти розробки охоплюють широкий спектр програмних продуктів, включаючи інтегровані середовища розробки (IDE), системи управління версіями, бази даних, фреймворки та інші утиліти. Прикладами популярних безкоштовних інструментів є Visual Studio Code, Eclipse, Git, MySQL, та фреймворки як React та Symfony [1].

Серед переваг безкоштовних інструментів можна виділити [2]:

1. Економічна Вигода: Основна перевага безкоштовних інструментів полягає у їх відсутності прямих витрат. Це робить їх особливо привабливими для стартапів та незалежних розробників.

2. Спільнота та Підтримка: Багато безкоштовних інструментів мають великі активні спільноти, що забезпечує розробникам доступ до широкого спектру ресурсів, таких як документація, напрацьовані рішення популярних задач, форуми для обговорення та підтримки.

3. Гнучкість та Відкритість: Більшість відкритих інструментів пропонують високий ступінь налаштування та гнучкість, дозволяючи розробникам модифікувати інструменти відповідно до своїх конкретних потреб.

Порівнюючи різні підходи до розробки програмного забезпечення можна сказати наступне [3, 4]:

1. Самостійно написаний код:

– Гнучкість та контроль: Максимальна свобода дій, оскільки розробник не обмежений структурою фреймворку.

– Складність та час: Розробка може займати більше часу, особливо для складних функцій, які можуть бути вже реалізовані в фреймворках.

– Крива навчання: Може бути крутою для початківців, оскільки вимагає глибокого розуміння мови програмування і принципів конкретної розробки.

– Підтримка та масштабування: Утримання та масштабування може бути складним, особливо для великих і складних проектів через відсутність структурованого коду та підтримки спільноти.

2. Використання відкритого фреймворку:

– Структурована розробка: для прикладу Symfony відомий своєю структурою і організацією, що робить його підходящим для складних та масштабних застосунків.

– Продуктивність: Зазвичай має обширні функції та опції налаштування, але може бути повільнішим у порівнянні з легшими фреймворками.

– Гнучкість: Пропонує велику гнучкість та модульність з системою пакетів, що дозволяє повторно використовувати компоненти.

– Підтримка спільноти: Має велику спільноту, розгалужену документацію та довгострокову підтримку, що є корисним для підтримки складних проблем і тривалого технічного обслуговування. Втім, може мати складну криву навчання та підвищену складність у деяких випадках.

3. Використання платного фреймворку чи інструменту:

– Вартість: Потребує інвестицій, що може бути бар'єром для малих компаній або індивідуальних розробників.

– Підтримка та Надійність: Зазвичай пропонує вищий рівень підтримки та стабільності, оскільки розробляється комерційними організаціями.

– Обмежена Гнучкість: Може бути менш гнучким у порівнянні з відкритими фреймворками через платну природу.

– Виключні Особливості: Часто пропонує унікальні можливості, яких не знайдете у відкритих фреймворках.

Також, варто враховувати, що розробкою займаються люди, чий стан має суттєвий вплив на продуктивність розробки, а він може значно відрізнятися залежно від використання різних підходів у розробці. При використанні самостійно написаного коду інженери можуть відчувати більше контролю та свободи, але також можуть стикатися зі збільшеним тиском через необхідність вирішення всіх технічних проблем самостійно. А також даний підхід моє викликати негативний стан через отримання навичок, які можуть бути непотрібні на ринку праці. Використання відкритих фреймворків, як-от Symfony, може сприйматися як більш структурований та ефективний підхід, що забезпечує кращу підтримку та менше повторюваності в роботі. З іншого боку, це може обмежувати творчість розробників, в порівнянні з написанням всього коду самостійно. Платні фреймворки можуть забезпечити вищий рівень підтримки та ексклюзивних функцій, але також можуть викликати занепокоєння щодо залежності від постачальника, а також відсутності гнучкості з вибором бібліотек чи рішень. В кінцевому підсумку, вибір підходу залежить від індивідуальних переваг, проектних вимог та специфіки робочого процесу команди.

Також, незважаючи на численні переваги, безкоштовні інструменти також мають свої обмеження. Вони можуть не завжди відповідати специфічним вимогам деяких проектів, а також можуть мати обмежену підтримку порівняно з платними альтернативами.

Безкоштовні інструменти розробки програмного забезпечення пропонують великі можливості для підвищення продуктивності та ефективності розробки. Їх використання дозволяє знизити витрати, сприяє швидкому навчанню та адаптації, та забезпечує доступ до ресурсів великої спільноти. Однак, важливо також зважати на можливі обмеження та вибирати інструменти, виходячи з конкретних потреб та вимог проекту.

Література

[1] Open Source Tools For Software Development: A Comprehensive Guide [Електронний ресурс]. – Режим доступу: https://osssoftware.org/blog/open-source-tools-for-software-development-a-comprehensive-guide/

[2] The Pros and Cons of Open Source Software Development [Електронний ресурс]. – Режим доступу: https://www.freecodecamp.org/news/what-is-great-about-developing-open-source-and-what-is-not/

[3] Front-end Development: Custom Code vs Ready-to-Use Frameworks [Електронний ресурс]. – Режим доступу: https://www.htmlpanda.com/blog/front-end-development-custom-code-vs-ready-to-use-frameworks/

[3] Why you should build your own framework but you shouldn’t use it [Електронний ресурс]. – Режим доступу: https://dev.to/gasner/why-you-should-build-your-own-framework-but-you-shouldnt-use-it-15jf

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ФОРМУВАННЯ ІНФОРМАЦІЙНОЇ СТРІЧКИ ЗА ІНТЕРЕСАМИ КОРИСТУВАЧА

Гребінець О.В., Льовкін В.М.
E-mail: vliovkin@gmail.com
Запоріжжя, Національний університет «Запорізька політехніка»

У сучасному світі, який характеризується надзвичайною швидкістю інформаційних потоків, залишатися в курсі справ надзвичайно важливо. Проблема при цьому ускладнюється тим, що теми, які цікавлять користувача, якими він захоплюється, можуть бути найрізноманітнішими. Відповідно не всі новини однаково привабливі для всіх. У світі, переповненому інформацією, де самі заголовки вимагають уваги, а послідовності новин розгортаються невпинним потоком, відповідно навіть читання заголовків вимагає значного часу, потреба в персоналізованих новинах є великою, а загалом ніколи і не була такою великою. До того ж ринок відповідного програмного забезпечення значно змінився протягом останнього часу, через що програми, які могли частково задовольняти такі потреби раніше, перестають відповідати вимогам користувачів зараз. В якості такого прикладу можна привести сервіс, який раніше був відомий як Twitter.

У цій роботі розроблено програмне забезпечення, яке в якості джерела інформаційної стрічки використовує RSS-стрічку, але персоналізує її під інтереси конкретного користувача. Алгоритм, покладений в основі реалізації програмного забезпечення, має певну послідовність: внесення RSS-стрічки, парсинг RSS-стрічки, перетворення вмісту заголовків та описів у вектор, перетворення інтересів у вектор, визначення косинусної подібності, виведення результату.

Для створення програмного забезпечення було використано мову Python – інтерпретовану об'єктно-орієнтовану мову програмування високого рівня із динамічною типізацією. Також додатково було використано бібліотеку feedparser [1] для аналізу RSS даних та бібліотеку sklearn в частині підпакету feature_extraction.text і класу CountVectorizer для перетворення тексту у вектор на основі частоти появи кожного слова у всьому тексті [2] і підпакету metrics.pairwise й функції cosine_similarity для визначення косинусної подібності, яка використовується для визначення схожості текстів на основі кута між векторами [3].

Для реалізації алгоритму в програмі була створена базова вхідна функція, що приймає в себе два параметри: RSS-стрічку та інтереси користувача, що задаються особисто користувачем у програмі. Принцип роботи функції:

– вноситься RSS-стрічка новинних сайтів, з якої у подальшому формується персоналізована інформаційна стрічка;

– за допомогою методу parse feedparser відбувається збір інформації зі стрічки;

– вміст назв та описів усіх статей вноситься до окремого переліку;

– дані з попереднього переліку та переліку, що містить інтереси користувача, трансформуються у вектори за допомогою CountVectorizer, а результат зберігається в змінних transform, interests_transform;

– за допомогою cosine_similarity визначається косинусна подібність між текстом та інтересами;

– виводиться результат косинусної подібності між текстом, що складається з назви і опису статті, та інтересами.

Література

[1] Python Library for Parsing Rss Feed [Electronic resource]. – Access mode: https://copyprogramming.com/howto/rss-feed-parser-library-in-python

[2] Using CountVectorizer to Extracting Features from Text [Electronic resource]. – Access mode: https://www.geeksforgeeks.org/using-countvectorizer-to-extracting-features-from-text/

[3] Understanding Cosine Similarity in Python with Scikit-Learn [Electronic resource]. – Access mode: https://memgraph.com/blog/cosine-similarity-python-scikit-learn

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ РОЗПІЗНАВАННЯ РЕКЛАМНОГО КОНТЕНТУ ВЕБСАЙТІВ

Грищенков М.С., Льовкін В.М.
E-mail: vliovkin@gmail.com
Запоріжжя, Національний університет «Запорізька політехніка»

Програмне забезпечення для розпізнання рекламного контенту є корисним і важливим інструментом, оскільки дозволяє знизити рекламний потік, який нерідко перевантажує людину у поєднанні з величезним сучасним інформаційним потоком. Такі програми дозволяють заблокувати рекламу на вебсторінках, що дозволяє зробити досвід споживання інформації більш зручним та менш нав'язливим для користувачів. Окрім цього програмне забезпечення для розпізнавання та блокування рекламного контенту на основі результатів розпізнавання має додаткову цінність, пов'язану з тим, що рекламні банери, спливаючі вікна часом пов'язані зі зловмисними діями, одні з яких можуть пошкодити комп'ютер або інший пристрій користувача, а інші – використовуються для слідкування за користувачами [1].

Загалом варто зазначити, що реклама може бути представлена як у вигляді зображень певного типу (зокрема банерів), так і в текстовому вигляді. Якщо розпізнавання реклами у вигляді зображень на даний момент на практиці має достатньо ефективні рішення, то оброблення текстових даних на даний момент є достатньо проблемним. Цим і визначається актуальність даної роботи, в якій акцент зроблено саме на створенні програмного забезпечення, орієнтованого на розпізнавання текстового рекламного контенту.

Користь такого програмного забезпечення полягає в тому, щоб проаналізувати вебсторінку ще до того, як її прочитав користувач, та виділити фрагменти, які містять рекламу, щоб користувач міг оцінити обсяг рекламної інформації в тексті вебсторінки. Це дозволяє далі вирішити, чи варто витрачати увагу на таку статтю, чи одразу перейти до іншої статті.

У даній роботі розроблено програмне забезпечення для розпізнання рекламного контенту. Тренувальна вибірка була сформована у вигляді csv-файлу. Безпосередньо робота програми організована на основі токенайзера, створенного з використання пакету nltk, що дозволило перетворити текстові дані самої вибірки та текстів у подальшому в послідовність чисел. Для розпізнавання рекламного контенту було побудовано модель на основі нейронної мережі, що містить 3 шари:

– перший шар є повнозв'язним шаром з 64 нейронами, що використовує функцію активації Rectified Linear Unit (ReLU), яка знаходить максимум з 0 і самого числа, яке визначає для неї вхідні дані;

– другий шар також є повнозв'язним шаром з 16 нейронами і використовує функцію активації ReLU;

– останній шар є повнозв'язним шаром з одним нейроном і використовує функцію активації sigmoid, яка відображає вхідні дані в діапазон від 0 до 1 на виході, таким чином визначаючи ймовірність наявності реклами в тексті.

При реалізації самої моделі було також використано функції втрат binary_crossentropy, оптимізатор Adam та метрику точності accuracy. Створена таким чином у результаті модель є багатошаровою мережею прямого поширення сигналу (feedforward neural network), тобто сигнал в ній поширюється тільки в одному напрямку – від вхідних до вихідних шарів. Після виконаного навчання моделі створена таким чином програма здатна приймати будь-який текст та виводити ймовірність того, що отриманий на вхід текст включає в себе рекламний контент або є рекламним контентом.

Література

[1] Cox, J. Inside a Global Phone Spy Tool Monitoring Billions. [Electronic resource]. – Access mode: / J. Cox. – Access mode: https://www.404media.co/inside-global-phone-spy-tool-patternz-nuviad-real-time-bidding/

АГРЕГУВАННЯ ДАНИХ З ІНТЕРНЕТ ДЖЕРЕЛ З ВИКОРИСТАННЯМ СТЕКУ БЕЗКОШТОВНИХ ПРОГРАМНИХ РІШЕНЬ

[1]Киблицький Р.Р., [2]Воловщиков В.Ю., [3]Шапо В.Ф.
E-mail: [1]rostyslav.kyblyskyi@cs.khpi.edu.ua, [2]valeriy.volovshchykov@khpi.edu.ua, [3]vladlen.shapo@gmail.com
[1, 2]Харків, Національний технічний університет "Харківський політехнічний інститут", [3]Одеса, Інститут Військово-Морських Сил

Дана робота є логічним продовженням [1] та присвячена питанням формулювання постановки задачі на розробку моделей та програмних рішень для агрегування даних з Інтернет джерел для формування каталогу мобільних пристроїв (МП). Постановку задачі пропонується формалізувати у вигляді діаграми Вігерса К. [2]. У якості стеку безкоштовних програмних рішень реалізації постановки задачі пропонується застосовувати Apache, PHP, Java, MySQL, Bootstrap, JavaScript та інші,

Питання агрегування даних з Інтернет джерел є актуальними і складними. Для вирішення цієї задачі використовують класичні, адаптовані та спеціалізовані розробки.

Технології агрегування даних та підходи до їх реалізації можна розділити на два великих класи: агрегування даних з незареєстрованих та з зареєстрованих ресурсів. При агрегуванні даних з незареєстрованих ресурсів можна стикнутись з структурованими, слабо-структурованими, або неструктурованими даними. В основі агрегації даних з зареєстрованих ресурсів знаходяться переважно агрегатори.

В роботі задача агрегаіції даних ставиться, як задача агрегації структурованих даних або агрегації даних з використанням агрегаторів.

Отже, нехай є ряд Інтернет джерел (Інтернет магазинів), які пропонують у своєму електронному каталозі МП. Нехай є користувачі мережі Інтернет, які мають на меті отримати доступ до електронних каталогів МП. Оскільки Web-ресурси Інтернет джерел за умовчанням не формують агреговані електронні каталоги МП, то користувачу для дослідження цієї інформації необхідно було би послідовно проаналізувати електронні каталоги кожного Інтернет джерела. Це є не зовсім зручним. Більш ефективним було би виконати агрегування інформації щодо МП в одному місці у вигляді агрегованого електронного каталогу. У ролі такого майданчика пропонується тематичний агрегатор. Процедура агрегації даних, яка пропонується, глобально реалізується двома послідовними кроками: вивантаження даних з боку Інтернет джерела та їх агрегація та відображення на агрегаторі. Дана процедура відповідає необхідності реалізації двох програмних рішень, які відповідно зосереджені на боці Інтернет джерела та агрегатору. Програмне рішення на боці Інтернет джерела повинно вирішувати задачу з контролю змін в базі даних (БД) каталогу МП. У випадку виконання таких змін, останні повинні фіксуватися в БД Інтернет джерела. На основі зафіксованих змін повинна бути виконана вибірка змінених (оновлених, доданих) даних, фіксація їх в локальному сховищі *.json та подальше вивантаження їх на агрегатор. Подальша робота із агрегації та фіксації змін виконується на боці агрегатора відповідними програмними рішеннями. А саме, дані, отримані Web-сервером агрегатора, перенаправляються для закінчення процедури агрегації даних на сервер БД агрегатора, ооновлюючи останні. В результаті такої процедури користувач буде мати можливість доступу до актуальної агрегованої інформації. Представлена постановка задачі ілюструє взаємодію Інтернет джерела та агрегатора, як взаємодію backend та frontend відповідно.

Вимоги до програмних рішень, які стануть базою до їх реалізації наведені на рис. 1 у формі діаграми Вігерса К.

Подальша робота буде присвячена детальному проектуанню програмних рішень для агрегування даних з Інтернет джерел для формування каталогу МП шляхом розробки відповідних моделей та їх програмної реалізації. Верифікацію та валідацію програмних рішень буде запропоновано виконувати зокрема з використанням вільнорозповсюджуваних засобів PHPUnit та Apache JMeter.

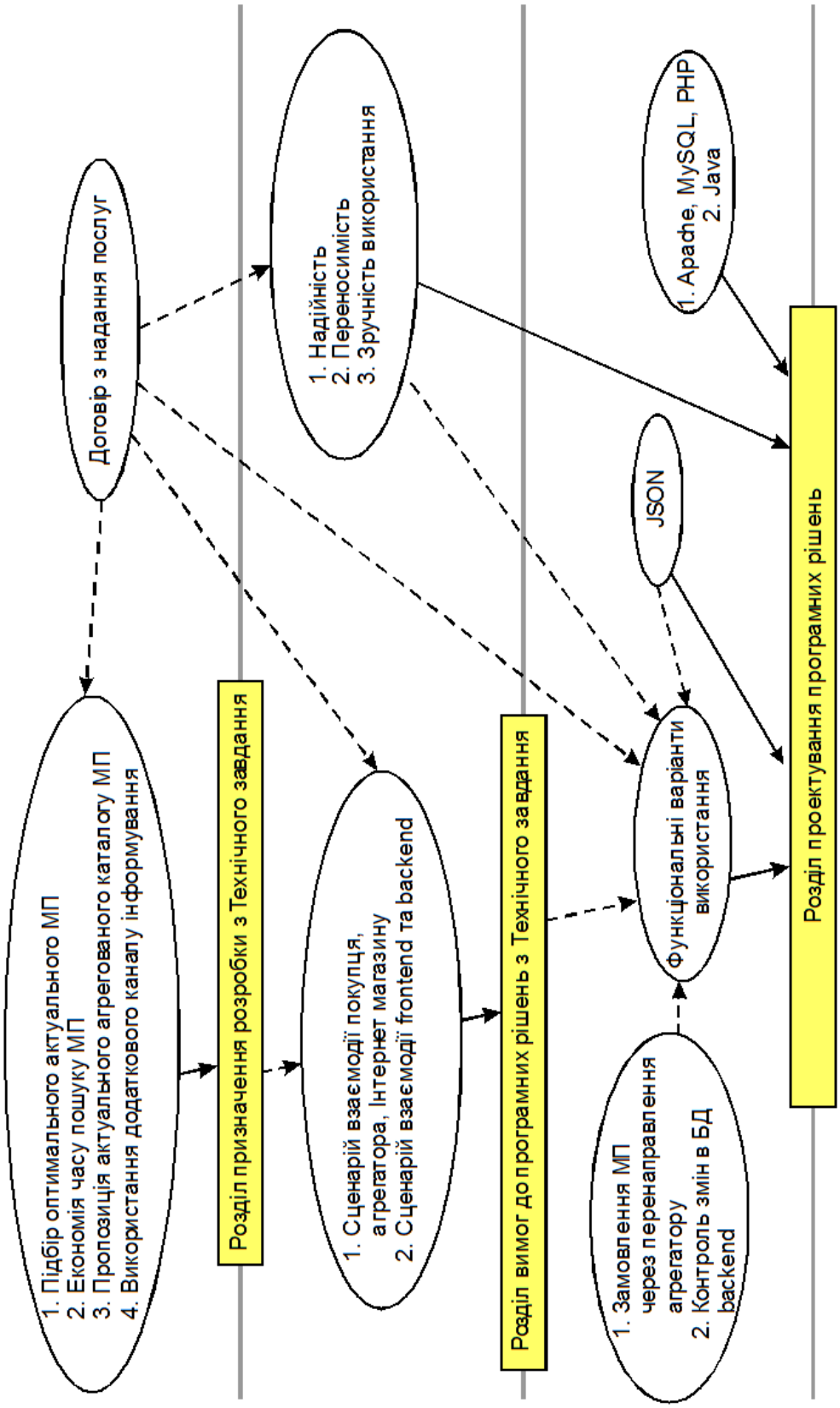


Рисунок 1 – Вимоги до програмних рішень за Вігерсом К.

Література

[1] Чухрій В.С., Воловщиков В.Ю., Шапо В.Ф. Стек програмних рішень для агрегування даних з глобальної мережі Internet // Матеріали XIV-ої Міжнародної науково-практичної конференції «Free and Open Source Software», Харків, 14-16 лютого 2023 р. – Харків: ХНЕУ імені Семена Кузнеця, 2023. с. 65-66.

[2] Karl Wiegers, Joy Beatty. Software Requirements (Developer Best Practices). – Redmond: Microsoft Press, 2013. – 672 p.

ПОРІВНЯННЯ МОЖЛИВОСТЕЙ JUPYTER І GOOGLE COLAB ДЛЯ ВИРІШЕННЯ ЗАДАЧ МАШИННОГО НАВЧАННЯ ТА ШТУЧНОГО ІНТЕЛЕКТУ

Жилін М.Ю.
E-mail: mykhailo.zhjylin@nure.ua
Харків, Харківський національний університет радіоелектроніки

У сучасному світі об’єми інформації стрімко збільшуються і необідні зручніші інструменти для її аналізу. У цьому контексті виділяються два потужних інтерактивних середовища Jupyter [1] та Google Colab [2].

Jupyter є повністю безкоштовним інструментом, він поширюється під ліцензією BSD, яка є однією з ліцензій відкритого програмного забезпечення. Це означає, що Jupyter можна вільно використовувати, змінювати і поширювати як для особистих, так і для комерційних цілей.

Доступ до Google Colab і його використання регулюються умовами сервісу Google і додатковими політиками конфіденційності та умовами використання, встановленими компанією Google. Користувачі Google Colab повинні погодитися з цими умовами для використання сервісу. Ці умови містять положення про використання наданих Google ресурсів, обмеження на використання сервісу для певних цілей, а також правила обробки та зберігання даних користувача. Хоча сама платформа Colab надається користувачам безкоштовно (з опціональною платною підпискою Colab Pro для розширеного доступу до ресурсів), вона не є "відкритим програмним забезпеченням" у традиційному розумінні. Сервіс заснований на Jupyter Notebooks, як зазначено вище, є відкритим проектом. Однак інфраструктура і додаткові функції, пропоновані Google в рамках Colab, унікальні для сервісу і регулюються корпоративними політиками Google.

Jupyter - це відкрите, інтерактивне обчислювальне середовище, що дає змогу створювати та ділитися документами, які містять код, візуалізації та пояснювальний текст. Ці документи, відомі як Jupyter Notebooks, можуть служити для різних цілей, від оброблення даних і статистичного моделювання до машинного навчання, візуалізації даних і багато іншого.

Основна мета Jupyter полягає в підтримці інтерактивного розроблення та представлення даних і аналітики. Це робить його потужним інструментом для дослідників, науковців, інженерів та аналітиків даних для експериментування з кодом у реальному часі, візуалізації даних і результатів аналізу, створення навчальних матеріалів, включно з підручниками та книгами, спільної роботи та обміну дослідженнями з колегами, представлення досліджень та аналізів в інтерактивному та візуально привабливому форматі.

Jupyter підтримує понад 40 мов програмування, завдяки чому він стає універсальним інструментом для роботи з даними та кодом. Серед мов, які найчастіше використовуються в Jupyter Notebooks: Python, R, Julia, Scala. Крім того, Jupyter підтримує безліч інших мов, включно з JavaScript, Ruby, Haskell і багато інших, завдяки використанню спеціальних "kernels" – компонентів, які виконують код, написаний на відповідній мові програмування, всередині Jupyter Notebook. Jupyter забезпечує гнучку, потужну та зручну платформу для досліджень, аналізу даних та освітніх проектів, підтримуючи широкий спектр мов програмування та інструментів.

Google Colab, або Colaboratory, являє собою безкоштовний хмарний сервіс від Google, який дає змогу користувачам писати і виконувати код через браузер без необхідності будь-якої конфігурації.

Це інструмент, заснований на концепції Jupyter Notebooks, який значно спрощує спільну роботу, а також розробку проєктів у галузі машинного навчання, аналізу даних і досліджень. Оскільки Colab використовую концепцію Jupyter, то він успадковую основні функціональність Jupyter але має ще такі такі додаткові можливості:

Colab надає доступ до обчислювальних ресурсів, включно з GPU і TPU, що робить його ідеальним інструментом для навчання моделей машинного навчання;

– зберігає інформацію на Google Disk;

– дає змогу легко ділитися ноутбуками, коментувати та редагувати код у реальному часі, що полегшує колаборативну розробку та дослідження.

На поточний момент Google Colab підтримує код мовою Python, яка є однією з найпопулярніших мов у сфері аналізу даних, машинного навчання та наукових обчислень. Colab підтримує всі основні бібліотеки і фреймворки Python, які використовуються в цих галузях, включно з TensorFlow, PyTorch, Keras, Scikit-learn і багато інших. Хоча основна мова програмування в Colab - Python, користувачі також можуть вбудовувати в ноутбуки код іншими мовами, наприклад, використовуючи команди для запуску коду мовами, такими як Shell (bash) і SQL, а також вбудовувати HTML і JavaScript безпосередньо в документи для створення інтерактивних елементів.

Google Colab пропонує простий у використанні, потужний і гнучкий інструмент для розроблення та спільної роботи над проектами в галузі науки про дані та машинного навчання, доступний абсолютно безкоштовно, який не потребує від користувачів налаштування власної обчислювальної інфраструктури.

Jupyter і Google Colab обидва являють собою інтерактивні середовища для роботи з кодом, візуалізацією даних і науковими обчисленнями. Вони мають багато спільного, але й низку ключових відмінностей.

І Jupyter, і Colab дають змогу користувачам створювати інтерактивні ноутбуки, що містять живий код, візуалізації і текст. Обидві платформи широко використовують Python, найпопулярнішу мову в науці про дані та машинному навчанні. Вони активно застосовуються для навчання, досліджень, аналізу даних і розробки машинного навчання.

Але Jupyter та Colab мають відмінності у розгортанні, обчислювальних ресурсах, спільній роботі, інтеграції з хмарним сховищем:

– Jupyter можна встановити і запускати локально на власному комп'ютері або сервері, в той час як Google Colab працює в хмарі і доступний через веб-браузер без необхідності установки;

– Colab надає доступ до обчислювальних ресурсів Google, включно з безплатним доступом до GPU і TPU, що може бути особливо корисно для тренування моделей машинного навчання. Jupyter, запущений локально, обмежується ресурсами комп'ютера, хоча його також можна налаштувати на роботу із зовнішніми обчислювальними кластерами;

– Colab забезпечує більш зручні засоби для спільної роботи в реальному часі, оскільки ноутбуки зберігаються в хмарі і можуть бути легко поділені з іншими користувачами. У Jupyter для спільної роботи зазвичай потрібні додаткові налаштування або використання JupyterHub;

– Colab тісно інтегрований з Google Drive, що полегшує збереження і доступ до ноутбуків. Jupyter може працювати з різними системами зберігання, але вимагає додаткового налаштування для інтеграції з хмарними сховищами;

– Colab пропонує безкоштовний доступ до деяких ресурсів GPU і TPU, що робить його доступним для початківців і студентів. Jupyter сам по собі безкоштовний, але доступ до потужних обчислювальних ресурсів може потребувати додаткових витрат.

Обидві платформи популярні серед наукової спільноти, аналітиків даних і розробників машинного навчання.

На рисунку 1 наведена графік, що відображає інтерес до платформ Jupyter і Colab за останні п’ять років на основі частоти запитів у Google пошуковику [3], де можна побачити, що в Україні більш популярний інструмент Jupyter ніж Colab. Також рисунок 3 показує, що найбільша кількість запитів щодо обох платформ йде з Китаю [3].

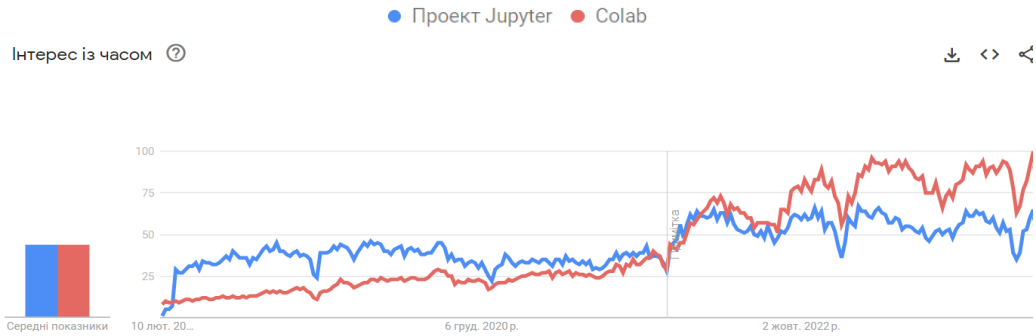


Рисунок 1 – Інтерес до платформ Jupyter і Colab з 2019 до 2023 включно [3]

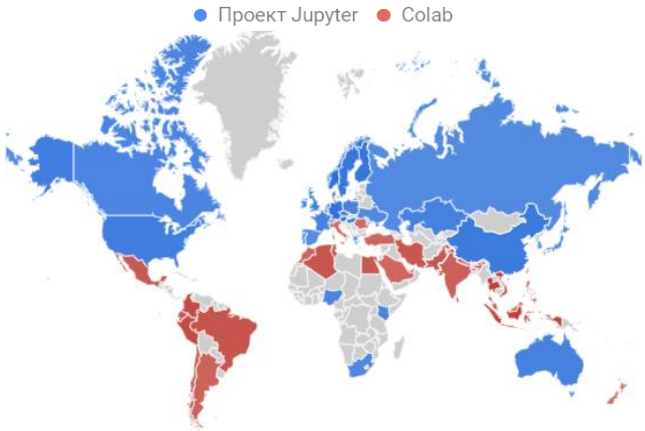


Рисунок 2 – Домінування інструменту в регіоні [3]

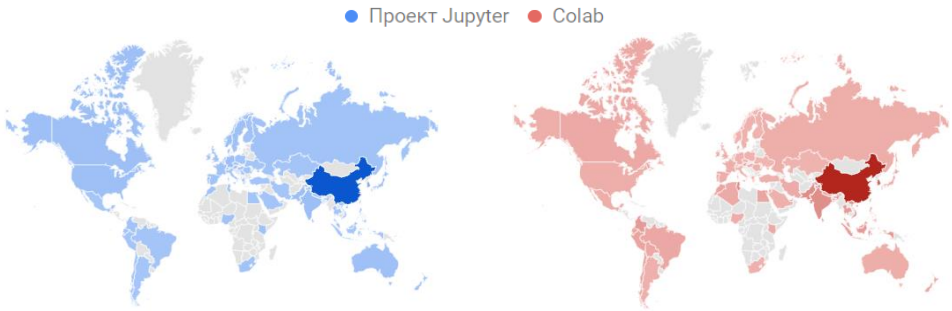


Рисунок 3 – Кількість запитів для Jupyter та Colab за регіонами [3]

Таким чином, часто віддають перевагу під час вирішення задач машинного навчання та штучного інтелекту Google Colab за зручність використання в хмарі, доступ до GPU/TPU та спрощену спільну роботу [4], в той час як Jupyter цінується можливість працювати без інтернету, за гнучкість і можливість повного налаштування середовища, за підтримку безлічі мов програмування.

Література

[1] Jupyter [Електронний ресурс]. – Режим доступу до ресурсу: https://jupyter.org/

[2] Google Colaboratory [Електронний ресурс]. – Режим доступу до ресурсу: https://colab.google/

[3] GoogleTrend [Електронний ресурс]. - Режим доступу до ресурсу: https://trends.google.com/trends/explore?date=today%205-y&q=%2Fg%2F11f60xvtsb,Colab&hl=uk

[4] Yakovleva O., Nebeský L., Kirichenko A. Using the GPT models for responses based on custom content to develop neural consultant for university applicants. Abstracts of V International Scientific and Practical Conference. Madrid, Spain. pp. 172-178.

ВИКОРИСТАННЯ ВІДКРИТОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ МІНІМІЗАЦІЇ РИЗИКІВ ВІРТУАЛЬНИХ ВАЛЮТНИХ ТРАНЗАКЦІЙ

Залевська А.О.
Керівник: Венгріна О.С.
E-mail: Nastacumacenko86@Gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

В контексті стрімкого розвитку інформаційних технологій та поширення криптовалют, питання безпеки і ризиків, пов'язаних із використанням віртуальної валюти, набувають особливої актуальності. Анонімність та відсутність централізованого контролю в світі криптовалют породжують унікальні виклики і ризики для користувачів. В цій роботі акцент зроблено на ідентифікації та оцінці ризиків, що виникають при виконанні транзакцій з віртуальною валютою. Важливу роль у цьому контексті відіграє зростаюча популярність використання відкритого програмного забезпечення, яке сприяє прозорості та доступності інструментів для аналізу та захисту.

Серед використаних інструментів:

1. Blockchain Explorer – це життєво важливий інструмент для вивчення та аналізу блокчейн-мереж, який надає користувачам детальну інформацію про транзакції, блоки та адреси, що є ключовим для розуміння механізмів блокчейну та ідентифікації потенційних ризиків.
2. OWASP ZAP – це відкритий програмне забезпечення для виявлення та усунення вразливостей веб-додатків, який забезпечує комплексний аналіз та захист, необхідні для безпечного обігу віртуальної валюти.
3. Wireshark – це рішення для аналізу мережевого трафіку, який дозволяє вивчати деталі комунікацій в мережах розповсюдження криптовалют, сприяючи глибшому розумінню безпекових аспектів і потенційних вразливостей.
4. Tableau Public – це потужний інструмент для візуалізації даних, що дозволяє користувачам ілюструвати та аналізувати інформацію, пов'язану з транзакціями криптовалют.

Завдяки відкритому характеру цих інструментів, вони не тільки сприяють забезпеченню безпеки інформаційних систем, але й надають можливості для адаптації до постійно змінюваних умов кіберпростору. Їх активне використання та обмін знаннями та досвідом серед спільноти експертів з кібербезпеки відіграє ключову роль у розвитку ефективних методів аналізу

Література

[1] Bitcoin Core. [Електронний ресурс]. – Режим доступу до ресурсу: https://bitcoincore.org/

[2] Electrum Bitcoin Wallet. [Електронний ресурс]. – Режим доступу до ресурсу: https://electrum.org/

[3] Go Ethereum (Geth). [Електронний ресурс]. – Режим доступу до ресурсу: https://geth.ethereum.org/

[4] MetaMask. [Електронний ресурс]. – Режим доступу до ресурсу: https://metamask.io/

Wireshark. [Електронний ресурс]. – Режим доступу до ресурсу: https://www.wireshark.org/

[5] OWASP ZAP (Zed Attack Proxy). [Електронний ресурс]. – Режим доступу до ресурсу: https://www.zaproxy.org/

[6] Tableau Public. [Електронний ресурс]. – Режим доступу до ресурсу: https://public.tableau.com/en-us/s/

[7] Blockchain Explorer. [Електронний ресурс]. – Режим доступу до ресурсу: https://blockchain.info/

ОГЛЯД МОЖЛИВОСТЕЙ ЩОДО ДЕТЕКЦІЇ РУХУ ЗАСТОСУНКУ ДЛЯ ВІДЕОСПОСТЕРЕЖЕННЯ ISPY

Ісаєв Є.А.
Керівник: Яковлева О. В.
E-mail: yevhenii.isaiev@nure.ua
Харків, Харківський національний університет радіоелектроніки

Застосунки для моніторингу камер в сучасному світі набувають все більшої популярності і важливості. Швидкий розвиток технологій і зростаюча потреба в безпеці призвели до широкого застосування таких застосунків у різних сферах, включаючи домашню безпеку, комерційні об'єкти та громадські місця. Дана робота присвячена огляду застосунка iSpy [1] для відеоспостереження. Особлива увага приділена можливостям iSpy щодо детекції руху.

iSpy є одним з найкращих застосунків для відеоспостереження. iSpy поширюється безкоштовно під ліцензією GPL (General Public License), що означає, що користувачі можуть вільно завантажувати і використовувати програмне забезпечення, а також вивчати і змінювати вихідний код. Однак, незважаючи на те, що базова версія програми безкоштовна, розробники пропонують додаткові платні послуги та функції в рамках підписки. Це може включати розширені можливості для віддаленого доступу, підтримку хмарного зберігання, SMS-сповіщення та інші розширені функції, які вимагають підписки або одноразової оплати для активації. Таким чином, iSpy пропонує гнучку модель використання: базова версія доступна безкоштовно, але для доступу до додаткових функцій і поліпшеної підтримки може знадобитися платна підписка.

iSpy - це програмне забезпечення для моніторингу відео та аудіо за допомогою веб- та IP-камер, мікрофонів, відео реєстраторів і навіть веб-камер на комп'ютерах. Воно стоїть на першому місці в багатьох рейтингах завдяки своїм розширеним можливостям. Застосунок iSpy має детектор руху, який дозволяє виявляти будь-які рухи перед камерою. Він має широкі можливості налаштування для відповідності конкретним потребам користувача. Після виявлення руху, iSpy може автоматично запускати запис відео, сповіщати користувача, або запускати інші зазначені дії, що робить його надзвичайно корисним для забезпечення безпеки та моніторингу. iSpy має підтримку камер високої чіткості, звукові сповіщення і архівацію відео. Оскільки iSpy є проектом з відкритим вихідним кодом, користувачі можуть адаптувати програму під свої потреби. Також, завдяки тому, що застосунок iSpy має активну спільноту користувачів, можливо розширити функціонал застосунка, за допомогою готових плагінів, такі як: плагін для розпізнавання автомобільних номерних знаків, плагін для накладеного тексту, або плагін для сканування штрих коду. Як вже зазначалося вище основна версія є безкоштовною, що робить застосунок доступним для широкого кола користувачів, але якщо потрібно його використовувати для бізнесу, або використовувати віддалений доступ, то необхідно купляти ліцензію.

У програмі iSpy вбудований ряд різних детекторів руху. Кожен детектор руху має різні характеристики і підходить для різних цілей. Існує чотири різних режими для детектування руху:

1) два кадри (Two Frames) – найпоширеніший тип, який використовують більшість користувачів. iSpy просто порівнює останній кадр з поточним;
2) користувацький кадр (Custom Frame) – iSpy зберігає один кадр у пам'яті і порівнює його з наступними кадрами;
3) моделювання фону (Background Modelling) – iSpy бере користувацький кадр, але з часом адаптує його, щоб він змінювався до поточного кадру. Це корисно, коли у сцені є щось, що постійно рухається - iSpy вивчить, як ігнорувати це.
4) немає (None) – це відсутність виявлення руху, що має бути використаним, якщо потрібно просто записати відеоспостереження за розкладом, наприклад, записати відео у вигляді таймлапсу, або за запитом.

Іншим важливим параметром для детекції руху є діапазон спрацювання – він є одним із найважливіший параметрів контролю в програмі iSpy. Він встановлює мінімальні та максимальні рівні спрацювання для виявлення руху, які контролюють всі функції запису та сповіщення.

Як показано на рисунку 1, камера надає прямий перегляд того, скільки руху виявляється, разом з точним місцем точок спрацювання. Контроль дозволяє встановити мінімальне та максимальне значення для спрацювання (у відсотках). Встановлення максимального значення добре підходить для ігнорування змін у всій сцені, наприклад, раптових змін яскравості через погодні умови. Залежно від моделі камери та умов освітлення, необхідно буде проекспериментувати з цими налаштуваннями, щоб переконатися, що сповіщення про рух не надходять постійно через сигнальний шум. Цей процес може потребувати деякого часу. Регулювання здійснюється в реальному часі, і індикатор виявлення блиматиме під час активації, тому необхідно налаштувати діапазон чутливості, щоб досягнути потрібного рівня руху, необхідного для спрацювання сповіщення.

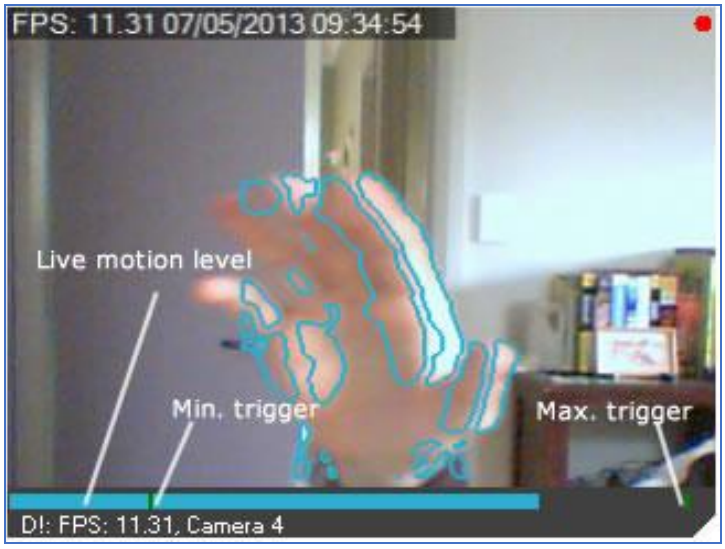


Рисунок 1 – Застосування параметру діапазону спрацювання [2]

Додаток iSpy має інший важливий параметр - фільтрація HSL (відтінок, насиченість і яскравість) – цей параметр дозволяє iSpy ігнорувати певні діапазони кольорів або яскравості при пошуку руху. Існує багато прикладів, в який ситуаціях цей параметр може бути корисним, наприклад: виявлення диму, виявлення вогню, виявлення підвищеної температури (за допомогою використання тепловізійних камер), ігнорування постійно рухомих об'єктів (таких як дерева або хмари), відстеження об'єктів певного кольору. Колірне коло ліворуч, яке показане на рисунку 2, використовується, щоб вибрати діапазон кольорів для ігнорування або відстеження (залежно від вибору у випадаючому списку "Тип заливки"). Потім можна вказати мінімальний і максимальний рівні насиченості та яскравості.

Попередній перегляд того, що бачить датчик руху, доступний у верхньому правому куті. На рисунку 2 видно, що детектор налаштовано на пошук лише червоних кольорів та ігнорування всіх інших - це означає, що він фіксуватиме та сповіщатиме, якщо в кадр потрапить щось червоного кольору. Потрібно зауважити, що фільтрація HSL додає значну кількість обробки і може сповільнити частоту кадрів (залежно від можливостей комп'ютера).

Ще одним корисним параметром для відстеження руху в застосунку iSpy є виявлення зон (рис.3). За допомогою параметра спеціального параметру, iSpy може стежити за певними областями в полі зору камери і ігнорувати інші. Ви можете додати скільки завгодно прямокутних зон виявлення (iSpy буде стежити тільки за областю в межах цих зон). Щоб додати зону, натисніть і перетягніть напівпрозорий прямокутник. Щоб очистити всі зони, натисніть "Очистити зони". Якщо зони не визначені, iSpy буде контролювати весь огляд камери.

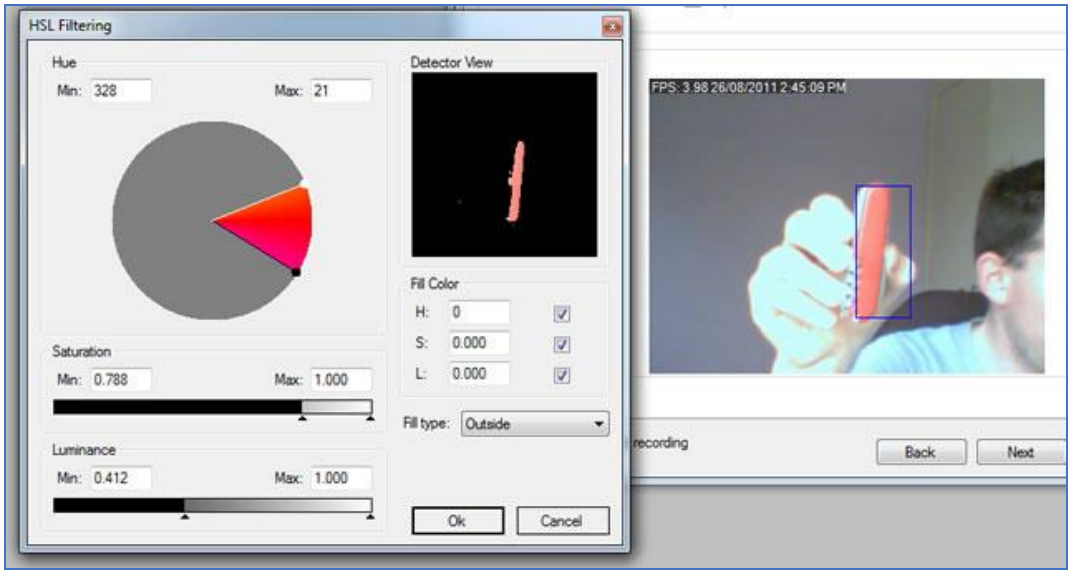


Рисунок 2 – Налаштування HSL у застосунку iSpy [2]

Рисунок 3 – Налаштування зон виявлення у застосунку iSpy [2]

Таким чином, застосунок iSpy має потужну функціональність для виявлення руху під час відеоспостереження та може використовуватися з різною метою, наприклад, в системах безпеки підприємства, де потрібно проводити розпізнавання облич співробітників [3]. В таких системах iSpy може використовуватися як на основі своєї базової функціональності, так і в поєднанні з додатково створеними розробниками модулями для вирішення специфічних задач в кожному окремому випадку.

Література

[1] iSpy Software Review (Best Free IP Camera Software) [Електронний ресурс]. – Режим доступу до ресурсу: https://learncctv.com/ispy-software-review/

[2] Motion detection (sensitivity) in iSpy [Електронний ресурс]. – Режим доступу до ресурсу: https://www.ispyconnect.com/docs/ispy/motion-detection#

[3] Yakovleva, O., Kovtunenko, A., Liubchenko, V., Honcharenko, V., & Kobylin, O. (2023). Face Detection for Video Surveillance-based Security System (COLINS-2023). In CEUR Workshop Proceedings (Vol. 3403). pp. 69-86.

СИСТЕМА КОНТРОЛЮ ЯКОСТІ ЗВАРЮВАННЯ НА БАЗІ OPENCV

Луценко В.А.
Керівник: Пузирьов С.А.
E-mail: walker7732@gmail.com
Миколаїв, Чорноморський національний університет імені Петра Могили

OpenCV (Open Source Computer Vision Library) - це відкрите програмне забезпечення, призначене для обробки зображень та комп'ютерного зору. Воно надає набір інструментів та бібліотек для розробки програмного забезпечення, що працює з зображеннями та відео, включаючи функції розпізнавання облич, виявлення об'єктів, вимірювання розсташування об'єктів, відстеження руху, а також багато інших завдань в області комп'ютерного зору та обробки зображень. OpenCV широко використовується у наукових дослідженнях, промисловості та різноманітних додатках, де потрібно аналізувати зображення та робити рішення на їх основі.

Система контролю якості зварювання на базі OpenCV є ключовим інструментом у вирішенні низки проблем, пов'язаних з якістю та безпекою в зварювальній промисловості. За допомогою комп'ютерного зору ця технологія дозволяє автоматизувати процес контролю, ідентифікувати потенційні дефекти та забезпечити вчасну реакцію на них. Спираючись на алгоритми машинного навчання та високоточні обробники зображень, системи на базі OpenCV можуть виявляти навіть мікроскопічні дефекти, що дозволяє забезпечити високу якість зварювального процесу та уникнути потенційних аварій та нещасних випадків. Такий підхід допомагає знизити витрати на ремонт та відновлення обладнання, покращити безпеку працівників та підвищити продуктивність виробництва. У цьому контексті використання системи контролю якості зварювання на базі OpenCV є не лише доцільним, але й незамінним для підтримання високих стандартів якості та ефективності в зварювальній галузі:

– використання комп'ютерного зору для аналізу якості зварювання дозволяє підвищити ефективність і точність процесу контролю;

– автоматизовані системи на базі OpenCV можуть виявляти дефекти зварювання, такі як тріщини, неповне з'єднання або вириви, забезпечуючи вчасну реакцію на них;

– використання OpenCV дозволяє інтегрувати алгоритми машинного навчання для вдосконалення процесу виявлення дефектів та адаптації до нових умов зварювання;

– застосування систем контролю якості на базі OpenCV може знизити витрати на відновлення та ремонт зварювального обладнання за рахунок запобігання появі серйозних дефектів;

– інтеграція системи контролю якості зварювання на базі OpenCV може сприяти створенню більш безпечного та надійного виробничого середовища;

– використання відкритого програмного забезпечення, такого як OpenCV, забезпечує доступність та гнучкість у розробці та налаштуванні систем контролю якості зварювання.

Використання OpenCV для контролю якості зварювання відкриває перед промисловими підприємствами широкі можливості в удосконаленні цього процесу. Однією з головних технологічних переваг є висока точність аналізу зображень, яку забезпечують розширені алгоритми обробки OpenCV. Це дозволяє виявляти навіть мікроскопічні дефекти та забезпечує високу якість зварювального з'єднання.

Автоматизація процесу контролю також є важливим аспектом. Системи на базі OpenCV можуть працювати у режимі реального часу, автоматично виявляючи та класифікуючи дефекти без прямого втручання оператора. Це допомагає збільшити продуктивність та знизити ймовірність людських помилок.

Підвищена швидкість обробки зображень OpenCV робить можливим виконання контролю якості навіть на великих виробничих лініях без значних затримок. Це дозволяє забезпечувати безперервний процес виробництва та своєчасну реакцію на виявлені дефекти.

Більш того, гнучкість та адаптивність OpenCV дають можливість створювати і налаштовувати різноманітні алгоритми контролю якості, що дозволяє враховувати різні

умови зварювання та виявляти навіть найменш очевидні дефекти. Такий підхід робить системи контролю якості на базі OpenCV незамінними інструментами для підтримання високих стандартів якості в зварювальній промисловості.

Література

[1] OpenCV. Open Source Computer Vision Library [Електронний ресурс]. – Режим доступу: https://opencv.org/

[2] Puthran, A. V., & Rao, K. P. (2019). Real-Time Weld Quality Monitoring System using Computer Vision and Deep Learning Techniques. In 2019 IEEE 17th International Conference on Industrial Informatics (INDIN) (pp. 696-701). IEEE.

[3] Huang, G., & Liu, Z. (2018). Research on welding seam quality detection system based on computer vision. Journal of Physics: Conference Series, 1066(5), 052024

СТВОРЕННЯ БАЗОВИХ МОЖЛИВОСТЕЙ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ МОНІТОРИНГУ І ПРОГНОЗУВАННЯ АВТОМОБІЛЬНОГО ТРАФІКУ

Льовкін В.М.
E-mail: vliovkin@gmail.com
Запоріжжя, Національний університет «Запорізька політехніка»

Автомобільний трафік, особливо у місті, потребує обов'язкового ефективного керування. Якщо таке керування не здійснюється, то це впливає на якість життя у місті, значно його погіршуючи. Безпосередньо керування пов'язано з цілим рядом питань, частина з яких є довгостроковими, але інші здійснюються в короткостроковому періоді та можуть представляти собою, наприклад, тимчасові обмеження, що дозволяють врегулювати поточну ситуацію. Відповідно для того, щоб розуміти поточну ситуацію, необхідно створити програмне забезпечення, яке дозволить виконувати моніторинг автомобільного трафіку. За наявності засобів прогнозування автомобільного трафіку таке програмне забезпечення має дозволити проактивно реагувати на ситуацію, не чекаючи моменту, коли відповідний стан з негативним впливом реалізується. Тож створення програмного забезпечення, яке надає можливості і моніторингу, і прогнозування трафіку, можна вважати важливим.

Безпосередньо прогнозування автомобільного трафіку запропоновано виконувати на основі створення моделей за архітектурою LSTM для прогнозування трафіку на наступні 6 годин [1]. Враховуючи описану логіку та обране рішення для прогнозування, логічним є використання для реалізації програмного забезпечення наступних інструментів:

– мова програмування python, враховуючи, що вона дозволяє не тільки отримувати доступ до якісних засобів, здатних створювати моделі в результаті машинного навчання, підтримуючи цілий стек варіантів, але і також обробляти дані, що передбачає моніторинг автомобільного трафіку, забезпечуючи до того ж можливість реалізації вебдодатку, що може спростити доступ до використання програми загалом;

– бібліотеки TensorFlow [2] і Keras [3] для створення, навчання та за необхідності тестування моделей прогнозування автомобільного трафіку: TensorFlow при цьому використовується як бекенд, а Keras використовується для підвищення зручності роботи над створенням моделей;

– бібліотека Pandas [4] для структурування даних через забезпечення багатовимірних структур даних зі зручною можливістю вибору даних у подальшому для забезпечення різного роду інструментів моніторингу трафіку;

– бібліотека scikit-learn [5] для забезпечення супутніх математичних обчислень при роботі з даними і моделями.

Використання даного стеку інструментів спрощує і прискорює сам процес розробки, при цьому забезпечуючи в підсумку створення результуючого програмного рішення, яке здатне ефективно вирішувати практичні задачі, пов'язані з моніторингом та прогнозуванням

автомобільного трафіку як щодо організації відповідних обчислень, так і щодо надання відповідного гнучкого інтерфейсу для взаємодії користувачів з програмою.

Література

[1] Льовкін, В. М. Моделі прогнозування автомобільного трафіку на основі LSTM при розробці прикладних програм / В. М. Льовкін // Технічна інженерія. – 2023. – № 2 (92). – С. 152-157. – DOI: https://doi.org/10.26642/ten-2023-2(92)-152-157.

[2] TensorFlow [Electronic resource]. – Access mode: https://www.tensorflow.org/

[3] Keras: Deep Learning for humans [Electronic resource]. – Access mode: https://keras.io/

[4] Pandas documentation [Electronic resource]. – Access mode: https://pandas.pydata.org/docs/index.html

[5] scikit-learn: machine learning in Python – scikit-learn 1.4.0 documentation [Electronic resource]. – Access mode: https://scikit-learn.org/

BITO – ШТУЧНИЙ ІНТЕЛЕКТ ДЛЯ ДОПОМОГИ РОЗРОБНИКАМ ПРОГРАМНОГО КОДУ

Сівіцкий В.В.
Керівник: Сажко Г.І.
E-mail:vovasiv@outlook.com
Харків, Українська інженерно-педагогічна академія

Із стрімким розвитком технологій та поширенням штучного інтелекту, виникає необхідність в пошуку ефективних інструментів для полегшення роботи розробників програмного забезпечення. У цьому контексті виникає актуальність вивчення та аналізу інноваційного програмного продукту під назвою BITO (Brain-Inspired Tool for Optimization), який пропонує використання штучного інтелекту для поліпшення процесу розробки програм.

Літературний аналіз свідчить про те, що на сьогоднішній день інтелектуальні системи стають невід'ємною частиною розробки програмного забезпечення. Відкриття BITO відображає спробу знайти новаторський підхід до використання штучного інтелекту для вирішення завдань, що виникають у сучасному програмуванні [1].

Метою цієї наукової статті є вивчення можливостей та ефективності BITO в контексті розробки програмного забезпечення. BITO пропонує інтеграцію штучного інтелекту для оптимізації процесів програмування, але важливо визначити, наскільки цей продукт відповідає вимогам та чи може він стати корисним для розробників програм.

У ході нашого дослідження ми з'ясували, що BITO виявляється дієвим інструментом для розробників програмного коду. Він використовує нейронні мережі та алгоритми машинного навчання для автоматизації рутинних завдань, таких як виявлення помилок у коді, оптимізація алгоритмів та прискорення процесу вирішення завдань.

Дослідження також дозволило визначити переваги та недоліки використання BITO. З одного боку, він забезпечує значний приріст продуктивності розробників, спрощуючи їхню роботу та допомагаючи у підвищенні якості програм. З іншого боку, можливі питання щодо конфіденційності та безпеки даних, які обробляються інтелектуальними алгоритмами [2].

Для подальшого вдосконалення BITO рекомендується розширювати його набір функціональностей, зокрема, покращуючи алгоритми машинного навчання для більш точного визначення та виправлення помилок у програмному коді. Також, важливо акцентувати увагу на вдосконаленні алгоритмів оптимізації, щоб забезпечити ефективність в роботі з різноманітними завданнями розробки програм.

Однак, під час інтеграції BITO у розробку програм, важливо враховувати етичні питання та прозорість використання штучного інтелекту. Розробники повинні гарантувати безпеку та конфіденційність даних, які обробляються програмою, та віддавати перевагу відкритим стандартам та аудитам, щоб забезпечити довіру користувачів [3].

У нашому дослідженні ми розглянули інноваційний продукт BITO, який використовує штучний інтелект для оптимізації розробки програмного забезпечення. Виявлено, що BITO

може ефективно полегшити роботу розробників, забезпечуючи автоматизацію рутинних завдань та підвищуючи якість програм.

За результатами дослідження ми висловлюємо рекомендації щодо подальшого вдосконалення BITO, а також підкреслюємо важливість етичних аспектів використання штучного інтелекту в розробці програм. Інтеграція BITO може принести значний внесок у покращення продуктивності та якості програм, але це має відбуватися в контексті високих стандартів безпеки та етики. [1].

Майбутні дослідження можуть фокусуватися на розширенні можливостей BITO для роботи з різними мовами програмування та платформами. Також важливо вивчити вплив використання BITO на процеси командної розробки та визначити оптимальні стратегії впровадження великих команд розробників.

Загалом, BITO може стати важливим інструментом для розробників програмного коду, привносячи новаторські можливості у сферу штучного інтелекту. Однак, ключовою умовою є ретельне вивчення та вдосконалення його функціональностей, забезпечення безпеки та етичності використання, щоб він міг відповідати високим стандартам у сфері розробки програмного забезпечення.

Література

[1] Bito – дозволяє легко писати код [Електронний ресурс]. – Режим доступу до ресурсу: https://aiguru.com.ua/pysaty-kod-za-dopomohoyu-shtuchnoho-intelektu/

[2] Bito [Електронний ресурс]. – Режим доступу до ресурсу: https://itest.com.ua/instrumenty/bito/

[3] Bito AI understands your code like no other AI [Електронний ресурс]. – Режим доступу до ресурсу: https://bito.ai/

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СТИСКАННЯ ФОТОГРАФІЙ НА ОСНОВІ ЗАСОБІВ КЛАСТЕРИЗАЦІЇ

Скорик С.С., Льовкін В.М.
E-mail: vliovkin@gmail.com
Запоріжжя, Національний університет «Запорізька політехніка»

Надмірний обсяг фотографій та їхня висока роздільна здатність стають викликом для ефективного зберігання та передавання цифрового контенту з постійним ростом обсягу даних, який зберігається в мережі. Програмне забезпечення для стискання фотографій є важливим інструментом для оптимізації розміру файлів, включаючи засоби без втрати якості та з втратою. У даній роботі розроблено програмне забезпечення, яке дозволяє виконувати стискання фотографій на основі різних засобів кластеризації, обираючи в підсумку той, який користувач вважає найбільш придатним.

Для розробки програмного застосунку було обрано мову Python через її високий рівень абстракції та простоту роботи, що дозволило при реалізації зосередитись на наданні широкого вибору засобів кластеризації. Особливо важливим фактором є можливість легкості інтеграції з засобами машинного навчання загалом [1], що відкриває широкий простір для подальших покращень та розвитку.

Базовим механізмом стискання фотографій було використано метод кластеризації K-Means [2]. Застосування даного методу кластеризації для стискання фотографій передбачає консолідацію фрагментів зображення шляхом їхнього об'єднання відповідно до найближчого середнього кольору. Робота алгоритму полягає у визначенні груп пікселів, які спільно поділяють подібні характеристики кольору й у їхньому подальшому об'єднанні в кластери, враховуючи при цьому як фактор прийняття рішень необхідність мінімізації внутрішньокластерної дисперсії. Виконання даного процесу призводить до того, що зменшується кількість унікальних кольорів у зображенні, а це відтак забезпечує ефективне стискання з врахуванням кольорових аспектів. Обсяг файлів, у яких зберігаються дані фотографії, у підсумку зменшується.

Враховуючи наявність інших методів кластеризації, при реалізації програмного забезпечення було запропоновано альтернативні рішення для стискання фотографій, включаючи зокрема наступні:

– Hierarchical Clustering – це метод, що базується на ієрархічній структурі кластерів, де пікселі спочатку об'єднуються в певний набір кластерів, які в свою чергу далі об'єднуються в більші, утворюючи ієрархічну структуру, що дозволяє при стисканні зображень враховувати просторове розташування пікселів [2];

– DBSCAN (Density-Based Spatial Clustering of Applications with Noise) – це метод, який особливий тим, що використовує густину даних для визначення кластерів, при цьому працюючи з зображеннями, групує пікселі, які розташовані щільно один до одного, і виділяє області низької густини як шум чи окремі елементи, що в підсумку дає змогу впроваджувати більш гнучкий підхід до групування, особливо в зображеннях зі змінною густиною об'єктів [3].

Література

[1] Щербина, Д. С., Н. А. Потапова. Роль чисельних методів в машинному навчанні [Електронний ресурс] / Д. С. Щербина, Н. А. Потапова // Матеріали IV Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених «Прикладні інформаційні технології»: збірник наукових праць. – Вінниця: ДонНУ імені Василя Стуса, 2023. – С. 313-314.

[2] Clustering – scikit-learn 1.4.0 documentation [Electronic resource]. – Access mode: https://scikit-learn.org/stable/modules/clustering.html

[3] sklearn.cluster.DBSCAN – scikit-learn 1.4.0 documentation [Electronic resource]. – Access mode: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

СЕРВЕРНА АРХІТЕКТУРА ДЛЯ STREAMING VR

Сюсько К.Ю.
Керівник: Чуйко Г.П.
E-mail: kirill.syusko17@gmail.com
Миколаїв, Чорноморський національний університет імені Петра Могили

VR або віртуальна реальність - це інтерактивна технологія, що забезпечує користувачеві враження від поглиблення в імітованому середовищі, яке відтворює реальні або уявні об'єкти та ситуації. Зазвичай ця технологія використовується за допомогою спеціальних гарнітур, які користувач надягає на голову, щоб отримати візуальні, аудіо та інші сенсорні враження. Головна мета віртуальної реальності - створити іммерсивне середовище, яке дозволяє користувачам відчувати себе, ніби вони перебувають у цьому віртуальному просторі, взаємодіючи з ним за допомогою рухів, голосу та інших введених команд. Віртуальна реальність застосовується в різних галузях, включаючи розваги, навчання, медицину, архітектуру, військові технології та багато інших. Вона відкриває нові можливості для взаємодії з інформацією та оточуючим середовищем, розширюючи межі того, що можливо зробити в цифровому просторі. [1]

VR streaming - це процес передачі відео- або аудіоконтенту в реальному часі з використанням віртуальної реальності (VR) як формату. Це означає, що користувачі можуть переглядати або споживати контент у форматі VR безпосередньо на своїх пристроях, таких як віртуальні реальності гарнітури або інші спеціалізовані пристрої.

VR streaming може бути використаний для передачі різних типів контенту, таких як відеоігри, спортивні події, концерти, подорожі, трансляції подій та інші віртуальні досвіди.

Одним із важливих аспектів VR streaming є здатність транслювати контент у 360-градусному форматі, що дозволяє користувачам отримувати максимально іммерсивний досвід, відчуваючи, що вони знаходяться в самому центрі події або сцени, яка транслюється. Це створює унікальні можливості для споживачів контенту відчути себе частиною дійсності за допомогою технології віртуальної реальності.

VR стрімінг має кілька переваг і вирішує деякі проблеми, які пов'язані з розгортанням та споживанням віртуальної реальності. [2]

Плюси VR стрімінгу:

– Доступність для різних платформ. VR стрімінг дозволяє користувачам отримувати доступ до віртуального контенту через різні пристрої, такі як гарнітури віртуальної реальності, комп'ютери, смартфони або планшети.

– Спрощена розробка контенту. Розробники можуть створювати віртуальний контент для однієї платформи, а потім транслювати його через сервери на різні пристрої, що спрощує розробку та розгортання VR додатків.

– Масштабованість. VR стрімінг дозволяє одночасно обслуговувати багато користувачів, не потребуючи значних обчислювальних потужностей на стороні кожного пристрою.

– Оновлення та підтримка. Оновлення та поліпшення можуть бути виконані централізовано на серверах, що полегшує управління та підтримку для розробників.

– Низька вимоги до обладнання. Користувачі можуть отримати доступ до віртуального контенту навіть на пристроях з обмеженими обчислювальними можливостями, оскільки весь важкий обчислювальний процес відбувається на сервері.

Проблеми, які вирішує VR стрімінг:

– Високі вимоги до обладнання. Віртуальна реальність вимагає потужних обчислювальних ресурсів, які не завжди доступні на кожному пристрої. VR стрімінг дозволяє обробку складних обчислювань на сервері, зменшуючи навантаження на клієнтські пристрої.

– Кросс-платформена сумісність. VR стрімінг дозволяє користувачам отримати доступ до контенту незалежно від типу їхніх пристроїв або платформи.

– Оновлення та підтримка. Оновлення програмного забезпечення можуть бути виконані централізовано на серверах, забезпечуючи однакові умови для всіх користувачів.

Побудова серверної архітектури для VR стрімінгу з використанням технології NVIDIA CloudXR включає кілька ключових кроків [3]:

– Вибір хмарного провайдера. Перший крок - вибір хмарного провайдера, який підтримує NVIDIA CloudXR. Популярні хмарні платформи, такі як Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP), можуть бути використані для розгортання серверів.

– Налаштування інфраструктури. Налаштуйте необхідні обчислювальні та мережеві ресурси на вибраному хмарному провайдері. Виберіть оптимальний тип інстансів та кількість ресурсів відповідно до потреб вашого застосунку.

– Встановлення NVIDIA CloudXR SDK. Завантажте та встановіть NVIDIA CloudXR SDK на сервері. Це набір інструментів, який надає необхідні API та бібліотеки для розгортання VR стрімінгу.

– Налаштування мережі та мережевого роутінгу. Налаштуйте мережу для оптимального обміну даними між сервером та клієнтськими пристроями VR. Забезпечте достатню пропускну здатність та низький пінг для мінімізації затримок та покращення якості стрімінгу.

– Розробка клієнтської сторони додатку. Розробіть додаток або інтегруйте клієнтську частину VR стрімінгу з використанням NVIDIA CloudXR SDK у вашій VR програмі або гарнітурі.

Література

[1] Википедия. Virtual reality [Электронный ресурс]. – Режим доступа к ресурсу: https://en.wikipedia.org/wiki/Virtual_reality

[2] VR streaming system archetecture [Электронный ресурс]. – Режим доступа к ресурсу: https://www.researchgate.net/figure/360-VR-streaming-system-architecture_fig1_328815782

[3] NVIDIA CloudXR [Электронный ресурс]. – Режим доступа к ресурсу: https://www.nvidia.com/ru-ru/design-visualization/solutions/cloud-xr/

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ПРОГНОЗУВАННЯ ВАРТОСТІ ТУРИСТИЧНИХ ПОДОРОЖЕЙ

Тарасов Я.К., Льовкін В.М.
E-mail: vliovkin@gmail.com
Запоріжжя, Національний університет «Запорізька політехніка»

За сучасного стану економічного середовища велике значення має розробка і використання програмних засобів прогнозування, які ґрунтуються на методах навчання та виявленні взаємозв'язків між різноманітними характеристиками. Так, враховуючи широке застосування аналізу даних у сфері економіки, виникає необхідність у розробці ефективних алгоритмів, які здатні достатньо точно передбачати зміни вартості туристичних подорожей в залежності від різних параметрів. Тому вирішення завдань, пов'язаних із розробкою програмних засобів прогнозування, зокрема вартості туристичних подорожей, є важливим у сучасному економічному контексті.

Існує ряд методів, які можуть використовуватися для вирішення цієї задачі, включаючи лінійну регресію, метод опорних векторів, дерева рішень, нейронні мережі тощо. Ідея, взята за основу в даній роботі, полягає у використанні Random Forest для навчання моделі на існуючих даних, враховуючи, що більшість з перелічених методів може бути менш стійкими до перенавчання або менш гнучкими при моделюванні складних нелінійних залежностей. Головна перевага полягає у використанні ансамблю дерев рішень для аналізу та узагальнення даних, що дозволяє моделі автоматично адаптуватися до варіацій у вхідних даних та уникати перенавчання. Також алгоритм відомий своєю стійкістю до шуму та здатністю працювати ефективно навіть за наявності складних взаємозв'язків у вхідних даних.

Алгоритм, покладений в основу прогнозування вартості туристичних подорожей, має наступну послідовність етапів: обробка та підготовка даних, розділення даних на навчальний та тестовий набори, навчання моделі на навчальних даних, використання моделі на тестових даних для обчислення середньоквадратичної помилки, використання навченої моделі на нових даних з виведенням значення ціни користувачу. Для перетворення категоріальних даних у формат, придатний для моделювання, використано техніку Label Encoding, оскільки алгоритм RandomForestRegressor вимагає, щоб всі характеристики були числовими.

Програмне забезпечення розроблювалось з використанням мови програмування Python, яка є високорівневою та орієнтованою на ефективне вирішення різноманітних завдань з обмеженим використанням ресурсів завдяки наявності широкого спектру пакетів, розроблених спільнотою розробників. У підсумку для створення застосунку використано: пакет pandas для зручної обробки табличних даних у форматі DataFrame, а також для збереження даних у CSV-файл і їх подальшого зчитування [1], пакет sklearn для машинного навчання та аналізу даних [2], пакет bs4 для парсингу коду HTML-сторінки [3].

Алгоритм розпочинається ітерацією по HTML-сторінках подорожей, після чого ця інформація обробляється та збирається в словнику, що об'єднується у DataFrame. Результат зберігається у CSV-файл. Всі дані розділяються на тренувальний та тестовий набори за допомогою функції train_test_split. Створюється модель RandomForestRegressor, яка навчається на тренувальному наборі. Оцінка точності моделі здійснюється при визначенні середньоквадратичної помилки за допомогою функції mean_squared_error на тестовому наборі. Далі модель використовується для прогнозування вартості для нових даних.

Література

[1] Pandas documentation [Electronic resource]. – Access mode: https://pandas.pydata.org/docs/index.html

[2] sklearn API [Electronic resource]. – Access mode: https://scikit-learn.org/stable/modules/classes.html

[3] bs4 documentation [Electronic resource]. – Access mode: https://www.crummy.com/software/BeautifulSoup/bs4/doc/

ОГЛЯД МОЖЛИВОСТЕЙ ЗАЛУЧЕННЯ ШТУЧНОГО ІНТЕЛЕКТУ В БУДІВЕЛЬНІЙ ГАЛУЗІ

Тоотс Р.В., Шаповалова О.О.
E-mail: toots.robert@hneu.net, olena.shapovalova@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Будівельна галузь на всіх етапах проектування та реалізації проектів поки ще далека від повної діджиталізації. Серед перспективних напрямів підвищення її ефективності є залучення засобів штучного інтелекту (ШІ) до різних фаз роботи над будівельними проектами. Комплексний підхід в будівництві та реконструкції зруйнованих війною об'єктів дозволить раціонально використати наявні ресурси та, за умови застосування новітніх технологій, узгодити часові та ресурсні витрати для своєчасного та якісного виконання проекту. ШІ має бути залученим на всіх етапах будівництва, починаючи з аналізу потенційних потреб до супроводу експлуатації будівель і споруд.

На етапі аналізу потенційних потреб конкретного будівельного проекту ШІ може стати у нагоді для розв'язання задач автоматизації планування та оптимального розподілу ресурсів, прогнозування тривалості виконання етапів проекту, підбору матеріалів тощо. На етапі планування та проектування забудови застосування такого інструменту ШІ як нейронна мережа попередньо потребує збору та аналізу даних про аналогічні будівельні проекти (бюджет, тривалість етапів проекту, деталізація технологічних та геометричних характеристик будівель тощо) для її навчання та тестування.

Етап автоматизації проектування також може отримати підтримку з боку нейронної мережі, яку завчасно слід навчити та протестувати на релевантних датасетах виконаних проектів, що мають містити геодезичні дані, геологічні та кліматичні особливості ділянок забудови, містобудівні обмеження, технічні умови тощо.

Важливим моментом гнучкості керування проектом та його успішної реалізації з оптимальними витратами є оцінка ризиків та перспективи на кожному етапі з постійним коректуванням в залежності від динаміки виробничих процесів. ШІ має багатий арсенал алгоритмів для прогнозування та розпізнавання можливих ризиків, прийняття рішень щодо їх зниження та визначення стратегій моніторингу та позбавлення від «вузьких місць» [5-10].

В подальшому на етапі управління ресурсами та постачанням ШІ може бути корисним в логістиці, при моніторингу потреб у будівельних матеріалах та керуванні ланцюгом постачання, а також для своєчасного забезпеченням потреб у робочій силі та оптимізації графіків роботи. На завершальному етапі будівництва ШІ може використовуватись для перевірки якості виконаних робіт та їх відповідності стандартам.

Весь перелічений вище інструментарій для створення та організації роботи з нейронними мережами можна знайти в таких спеціалізованих бібліотеках та фреймворках як TensorFlow, Theano, PyTorch, MXNet, Caffe (бібліотеки), Keras, FastAI, Chainer (фреймворкі), що дозволяє розробникам ефективно створювати, навчати та використовувати нейромережі. Так, розроблений Google, TensorFlow є однією з найпопулярніших та широко використовуваних бібліотек для машинного та глибокого навчання зі зручним для роботи з нейромережами інтерфейсом та значною спільнотою користувачів. Аналогічною розробкою є бібліотека Theano, яку розробляли в університеті Монреаля, і яка свого часу вона відігравала важливу роль у розвитку глибокого навчання. Для спрощення створення та навчання нейромереж з TensorFlow та Theano доцільно використовувати Keras, високорівневий інтерфейс для роботи з нейромережами, який працює поверх них, роблячи код більш зрозумілим та лаконічним. В межах роботи з означеними бібліотеками створюються ШІ-моделі, які базуються на зібраних даних (будівельні проекти, інформація про матеріали та технічні характеристики, фінансові показники тощо) та передбачають створення прогностичних, класифікаційних, оптимізаційних та інших алгоритмів.

ОГЛЯД БІБЛІОТЕКИ PUPPETEER ДЛЯ ВИРІШЕННЯ ЗАДАЧІ DATA SCRAPING З ЦІЛЛЮ ЗБОРУ ЦІННИХ ДАНИХ

Топчій М.А.
Керівник: Яковлева О.В.
E-mail: mykyta.topchii@nure.ua
Харків, Харківський національний університет радіоелектроніки

У постійно змінюваній галузі веб-розробки та вилучення даних необхідні ефективні та надійні інструменти для вирішення проблем веб-аналізу. Одним із таких потужних інструментів є бібліотека Puppeteer [1]. Puppeteer є безкоштовною і поширюється під ліцензією Apache License 2.0. Ліцензія Apache 2.0 є однією з популярних ліцензій відкритого програмного забезпечення, яка дає змогу користувачам вільно використовувати, змінювати, розповсюджувати та інтегрувати програмне забезпечення у свої проєкти, включно з комерційними.

Puppeteer — це бібліотека Node.js, яка надає API високого рівня для керування Chrome/Chromium через протокол DevTools. Puppeteer за замовчуванням працює в безголовому режимі (без графічного користувацього інтерфейсу), але його можна налаштувати на повний Chrome/Chromium.

Puppeteer дозволяє автоматизувати різноманітні процеси для роботи з вебсайтами, мануальне виконання яких, займає купу часу. Наприклад, серед таких процесів можна виділити: сканування веб-сторінок, створення скріншотів, створення PDF-файлів, автоматизація відправки форм та інших взаємодій користувача з веб-сторінкою, виконання автоматизованих тестів або діагностика вашого вебсайту на предмет проблем з продуктивністю. При встановленні бібліотеки, Puppeteer додатково завантажує Chrome for Testing, який займає приблизно 170МБ для macOS, 282МБ для систем Linux та 280МБ для Windows.

Однією з популярних мет використання цієї бібліотеки є збір значущої інформації з певної веб-сторінки для подальшого використання в своїх цілях або Data scraping [2].

Data scraping - це техніка, за якої комп’ютерна програма витягує дані з результатів, зрозумілих людині, які надходять з іншої програми. Як правило, передача даних між програмами відбувається за допомогою таких структур даних, які є придатними тільки для обробки комп’ютерами, але не людьми. Ці формати передачі даних зазвичай жорстко структуровані та мінімізують неоднозначність та вірогідність помилок. Тому такі протоколи здебільшого взагалі не доходять до людини.

Отже різниця між data scraping та звичайним парсінгом полягає у тому, що в першому випадку, дані будуть відображені користувачу, а не являтимуться вхідним потоком для іншої програми. Data scraping часто передбачає ігнорування бінарних даних (зазвичай зображень або мультимедійних даних), зайвих коментарів та іншої інформації, яка є або нерелевантною, або перешкоджає автоматизованій обробці.

Таким чином, переваги Puppeteer у контексті data scraping полягають у тому, що Puppeteer дозволяє викликати безголовий браузер, щоб імітувати поведінку реального користувача та автоматизувати задачі браузера. Крім того, оскільки існує перехід зі сторінок і результат сканується після завантаження веб-сторінки, можна обробляти не лише статичні веб-сторінки, але й динамічні. Також є можливість автоматизувати виконування дій користувача на веб-сторінці, такі як натискання посилань, кнопок, заповнення форм або прокручування.

Література

[1] Puppeteer | Puppeteer [Електронний ресурс]. – Режим доступу до ресурсу: https://pptr.dev

[2] Data scraping (2023) [Електронний ресурс]. – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/Data_scraping

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ОБРОБЛЕННЯ ПРИРОДНОЇ МОВИ ДЛЯ ВИЗНАЧЕННЯ ТЕМ СТАТЕЙ БЛОГУ

Харитонов Д. О., Льовкін В.М.
E-mail: vliovkin@gmail.com
Запоріжжя, Національний університет «Запорізька політехніка»

Засоби оброблення природної мови стають дедалі більш популярними і головне важливими у сучасному світі. Аналіз статей, великих текстів, книг є важливим інструментом для аналітиків, бізнесу, журналістики тощо для аналізу контенту, який зокрема буде популярним у майбутньому. Аналіз людиною такої великої кількості інформації є витратним як за часом виконання, так і за витраченими коштами, тому для таких задач ефективнішим рішенням є створення програмного аналізатору тексту для визначення головних тем зі статті блогу.

З наявних технологій для цього було використано вільні рішення, зокрема мову программування Python [1] та бібліотеку NLTK [2]. Python володіє великою кількістю готових до використання пакетів та модулів, що робить мову ідеальним вибором для швидкої розробки та прототипування проєктів, також код на Python лаконічний та одразу зрозумілий, що збільшує шкидкість розробки. Такі програмні засоби нерідко знаходяться у платному доступі, а ці програмні рішення є абсолютно безкоштовними.

У роботі було реалізовано програму для обробки статей блогів, яка аналізує введений користувачем текст статті, а далі на основі введеного тексту пропонує співставити його з популярними темами статей, що співвідносяться з цим текстом. У процесі аналізу статей блогу використовувалися наступні бібліотеки: NLTK як бібліотека для роботи з природною мовою, яка містить інструменти для токенізації, лематизації та роботи зі стоп-словами, scikit-learn як бібліотека для машинного навчання, включаючи зокрема модуль TfidfVectorizer, функцію NMF.

При розробці було реалізовано наступні функції для обробки тексту:

– токенізація тексту: для розділення тексту на окремі слова використовується функція word_tokenize з бібліотеки NLTK, що дозволяє отримати набір токенів, які стануть базовими одиницями для подальшої обробки;

– перетворення тексту в нижній регістр: функція lower використовується для перетворення всіх символів тексту у нижній регістр, що допомагає уніфікувати тексти та запобігти виникненню проблем через різницю у великих та малих літерах;

– вилучення стоп-слів: використання набору стоп-слів, таких як "the", "and", "is", допомагає у вилученні загальнопоширених слів, які не несуть значущої інформації, зокрема для визначення теми статті, що реалізовано у функції preprocess_text за допомогою програмного коду set (stopwords.words ("english"));

– лематизація слів: функція WordNetLemmatizer здійснює лематизацію слів, тобто приведення їх до їхніх базових форм, це сприяє уніфікації слів та полегшує подальший аналіз тексту;

– TF-IDF векторизація: застосування TfidfVectorizer дозволяє перетворити набір текстів у матрицю TF-IDF, де кожен рядок відповідає одній статті, а кожен стовпчик представляє слово з урахуванням його важливості в статті;

– NMF (Non-Negative Matrix Factorization): використання NMF дозволяє безпосередньо виконувати виділення тем у статті, цей метод розкладає матрицю TF-IDF на добуток двох матриць, що дозволяє отримати представлення тексту статті у вигляді тем.

Література

[1] Python Documentation. Python Software Foundation [Electronic resource]. – Access mode: https://www.python.org/doc/

[2] NLTK: Natural Language Toolkit [Electronic resource]. – Access mode: https://www.nltk.org/

SPRING BOOT: ЛЕГКІСТЬ ТА ПОТУЖНІСТЬ РОЗРОБКИ НА JAVA

Широкорад К.А.
Керівник: Яковлева О.В.
E-mail: kseniia.shyrokorad@nure.ua
Харків, Харківський національний університет радіоелектроніки

Завершеність та ефективність у розробці програмного забезпечення є ключовими аспектами в успішному проекті. У світі Java-розробки фреймворк Spring Boot визнаний як один з провідних інструментів, який дозволяє розробникам зосередитися на реалізації бізнес-логіки, мінімізуючи при цьому витрати на складну конфігурацію та інфраструктуру. Робота присвячена огляду основних архітектурних принципів Spring Boot, його ролі у модульній організації проектів, а також його ключовому внеску у розробку мікросервісних архітектур.

Spring Boot поширюється під ліцензією Apache License 2.0. Це вільна і відкрита ліцензія, що надається Apache Software Foundation, яка дає змогу користувачам вільно використовувати, змінювати, розповсюджувати та інтегрувати Spring Boot у свої проекти, як комерційні, так і некомерційні, без необхідності платити за ліцензію. Ліцензія Apache 2.0 вимагає, щоб у модифікованих версіях програмного забезпечення та в похідних роботах було зазначено авторство вихідного коду, а зміни в коді були відзначені. Вона також надає юридичний захист від патентних позовів для користувачів і розробників.

Spring Boot - це проєкт в екосистемі Spring, призначений для спрощення процесу конфігурації та розгортання додатків на базі Spring. За останні роки Spring Boot визначився не лише як технологічний інструмент, а й як каталізатор для прискорення розвитку Java-спільноти. Запроваджуючи новаторські підходи до розробки, цей фреймворк продовжує привертати увагу та визначати стандарти в індустрії. Розглянемо більше деталей щодо архітектури, особливостей та впливу Spring Boot на розробку програмного забезпечення. Spring Boot, як фреймворк для розробки Java-додатків, визначається не тільки своєю ефективністю, але й особливою архітектурою, яка враховує потреби розробників та забезпечує гнучкість в конфігурації [1].

Основни архітектурні принципи Spring Boot:

1. Інверсія управління (IoC). Spring Boot використовує принцип інверсії управління для зменшення залежностей між компонентами додатку. Це означає, що контейнер Spring вирішує, які класи інстанціювати та як вони будуть взаємодіяти, щоб розробникам не доводилось вручну управляти цим процесом.

2. Використання залежностей (DI). Спрощення розробки досягається завдяки механізму використання залежностей. Spring Boot автоматично управляє залежностями між компонентами, забезпечуючи розробникам зручний та ефективний спосіб створення взаємодіючих об'єктів.

3. Угода головніша за конфігурацію. Фреймворк активно використовує Convention Over Configuration, що означає, що для багатьох речей не потрібне явне налаштування. Спрощений підхід до конфігурації дозволяє швидко стартувати проекти та зосереджуватися на розробці функціональності.

Spring Boot використовує концепцію модульної структури для організації проектів. Модульність сприяє розділенню функціональності на логічні блоки, що полегшує розробку, тестування та підтримку. Завдяки модульності та гнучкості Spring Boot, розробники можуть легко інтегрувати моделі штучного інтелекту, використовуючи бібліотеки та фреймворки для машинного навчання. Підтримка вбудованих серверів дозволяє легко розгортати та взаємодіяти із ШІ-моделями в реальному часі. Spring Boot становить ідеальний інструмент для розробки чат-ботів з використанням Штучного Інтелекту (AI), завдяки своїй гнучкості та великому спектру інтеграційних можливостей. Spring Boot легко інтегрується з різноманітними бібліотеками штучного інтелекту, такими як TensorFlow, PyTorch, або OpenAI. Це відкриває широкі можливості для застосування алгоритмів машинного навчання у чат-ботах. За допомогою Spring Boot, розробники можуть ефективно впроваджувати моделі

машинного навчання для передбачення потреб користувачів та автоматизації відповідей чат-бота. Spring Boot дозволяє легко обробляти та адаптувати вхідні дані, що отримані від користувачів чат-бота, і використовувати їх для покращення роботи AI-алгоритмів.

Spring Boot продовжує демонструвати стабільний ріст популярності [2]. За останні роки кількість проектів, використовуючих Spring Boot, перевищила мільйони. Регулярні оновлення та покращення свідчать про активний розвиток фреймворка (рис.1) [3]. Google Trends демонструє початковий інтерес до технології з перебігом часу (рис.2) [3].

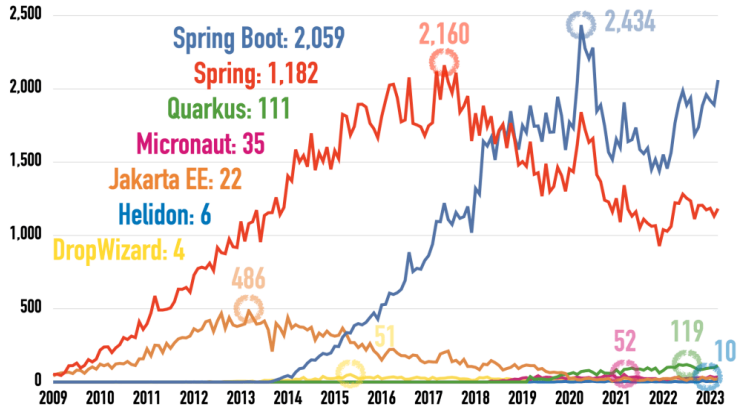


Рисунок 1 – Кількість запитів до бази даних щодо запитань, відповідей і коментарів у Stack Overflow за допомогою StackExchange Data Explorer [3]

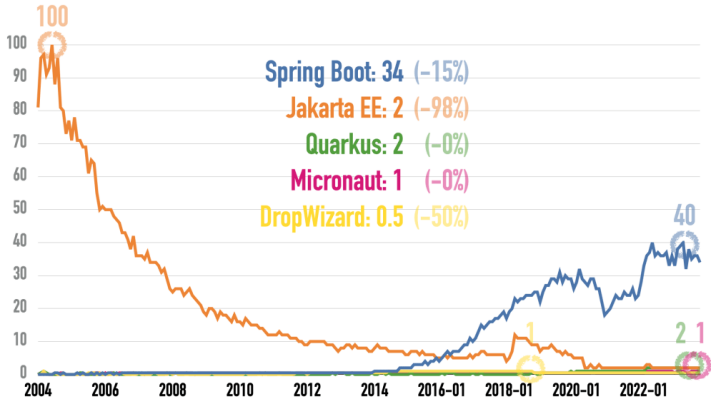


Рисунок 2 – Початковий інтерес до технології з перебігом часу від Google Trends [3]

Таким чином, Spring Boot - це не лише фреймворк, але й екосистема, що визначає стандарти у розробці Java-додатків. Його модульність, автоматична конфігурація та підтримка мікросервісної архітектури дозволяють розробникам створювати надійні та масштабовані рішення. Статистика свідчить про активну спільноту та стійкий ріст популярності, роблячи Spring Boot ключовим інструментом для вимогливих розробників.

Література

[1] Spring Boot [Електронний ресурс]. - Режим доступу до ресурсу: https://spring.io/

[2] Java Tech Popularity Index Q2/2023: Back-End Frameworks [Електронний ресурс]. - Режим доступу до ресурсу: https://betterprojectsfaster.com/

[3] Java Tech Popularity Index Q4/2023: Back-End Frameworks [Електронний ресурс]. - Режим доступу до ресурсу: https://betterprojectsfaster.com/java-tech-popularity-index-2023-q4/be/

ОГЛЯД ІНСТРУМЕНТІВ ДЛЯ ГНУЧКОГО УПРАВЛІННЯ ПРОЄКТАМИ

Шишкін М.С., Назаров Д.Л.
Керівник: Старкова О.В.
E-mail: maksim240600@gmail.com, unrealist@ukr.net
Харьков, Харківський національний економічний університет імені Семена Кузнеця

На сьогодні в практиці управління проєктами склалися два підходи до управління – традиційний та гнучкий. За першого – project-менеджер спільно з командою відпочатку визначають основні орієнтири: кінцеві цілі та проміжні результати. Такий підхід має назву Waterfall. Найефективніше він застосовується у командах, в яких кожен учасник знає власну частину роботи, а замовник готовий чекати на результат до фінальної дати і не сумнівається щодо його якості. Waterfall не може ефективно застосовуватися у проєктах з умовами, що постійно змінюються, адже специфіка підходу, зміни на ринку чи особливості бізнес-галузі не дають замовнику в процесі роботи уточнювати або змінювати характеристики продукту [1, 2].

Крім того, традиційний підхід не дозволяє внесення істотних змін після початку, тому існує значний ризик виявлення невідповідностей продукту на фінальній стадії роботи, коли внесення змін можливе, але таке внесення може бути вартісним та займе багато часу [1].

Методи управління проєктами, що передбачають адаптивність, називають гнучкими – Agile [3]. У гнучких методах проєкт поділяється на менші відрізки роботи – підпроєкти або спрінти. Про результати кожного з таких етапів команда звітує керівнику або напряму замовнику, які, у свою чергу, надають уточнення та окреслюють загальне враження щодо здобутих результатів етапу [1, 3]. Для гнучного управління проєктами існує ряд програмних рішень, які можуть стати наочним, доступним і адаптивним робочим інструментом команди (табл. 1) [1].

Таблиця 1

Інструменти для гнучкого управління проєктами [1]

Трекер	Платформа	Особливості	Недоліки	Тарифи
1	2	3	4	5
Worksection	Web, Windows, iOS, Android	Гнучкий інструмент для роботи з задачами, який підійде командам різних розмірів. З ним зручно працювати над кількома проєктами одночасно. Є діаграми Ганта, дошки Канбан, тайм-трекер тощо. Також передбачені вбудовані інтеграції з GoogleDocs, Slack, Telegram, Viber, CRM-системами	Відсутність офлайн-версії і не завжди ідеальна робота мобільної версії	Є безкоштовний варіант для 5 користувачів і 2 проєктів. Платні пакети коштують від 29 до 199 дол. США на місяць. Можна протестувати обраний пакет протягом 14 днів без оплати
ClickUp	Web, iOS, Android, Windows, Mac, Linux	Функціональний продукт, що має багато інструментів, корисних у гнучкій розробці: чати, вікі-документи, інтеграції з Google Workspace, Dropbox, Zapier. Можна налаштувати фільтрацію за календарем, таймлайнами задач	Обмежені можливості роботи мобільної версії	За продукт можна не платити лише при використанні 3 дошок. Більш просунуті тарифи коштують від 5 дол. США на місяць. Є пробний період на 14 днів

Продовження таблиці 1

1	2	3	4	5
Smartsheet	Web, iOS, Android	Серед гнучких інструментів управління проєктами цей обирають ті, кому подобається працювати з таблицями. Також тут є діаграми Ганта, календар і нагадування, інтеграція з Jira, продуктами Google та Microsoft	Сповіщення працюють не завжди коректно. Обмежено візуальні можливості для роботи з проєктами	Продукт коштує від 7 дол. США за користувача на місяць. Є безкоштовна тріал-версія
Teamwork	Web, iOS, Android, Windows, Mac, Linux	Календар, дошка Канбан, діаграми Ганта, задачі і підзадачі, чат, тайм-трекер – те, що робить цей продукт зручним для управління проєктами	Занадто складний інтерфейс. Є проблеми в роботі API	Для користувачів, що працюють з 1 чи 2 проєктами є безоплатна версія. Також надається тріал-версія на 30 днів. Платні тарифи обійдуться від 10 до 18 дол. США за користувача на місяць
Atlassian Jira	Web, iOS, Android, Windows, Mac, Linux	Один з найпопулярніших серед розробників інструмент для agile-менеджменту. Дошки Канбан і Scrum, зручні звіти, адаптовані для роботи по спринтам функції, задачі з різними рівнями пріоритету та багато іншого	Через велику кількість функцій користувачам іноді складно розібратися в інтерфейсі і знайти те, що їм потрібно	Команди, в яких не більше 10 працівників, можуть використовувати інструмент без оплати. Всі інші мають платити $7 на місяць. Щоб оцінити функціонал можна підключити тріал-версію

Наведний в табл. 1 перелік інструментів не є вичерпним. Відповідно до потреб розробників можна обрати рішення для гнучкого управління проєктами, що стане оптимальним поєднанням ціни, якості, а також зручності для команди розробників. Крім того, при виборі продукту треба міркувати про перспективи та одразу обирати той продукт, який можна буде використовувати по мірі масштабування бізнесу.

Литература

[1] Кращі інструменти для Agile управління проєктами у 2022 році [Електроний ресурс]. – Режим доступу до ресурсу: https://worksection.com/ua/blog/best-tools-for-agile-project-management.html.

[2] Методології управління проєктами, або що таке Waterfall, Agile та Scrum [Електроний ресурс]. – Режим доступу до ресурсу: https://devisu.ua/uk/stattia/metodologii-upravlinnya-proktami-abo-shcho-take-waterfall-agile-ta-scrum.

[3] Agile як невід'ємна частина успішних проєктів [Електроний ресурс]. – Режим доступу до ресурсу: https://foxminded.ua/shcho-take-agile/.

СЕКЦІЯ 3

R-STUDIO CAPABILITIES FOR BIG DATA ANALYSIS

Dolgova N.G.
E-mail: natalya.dolgova@hneu.net
Kharkiv, Simon Kuznets Kharkiv National University of Economics

RStudio, the leading integrated development environment (IDE) for R, a programming language recognized for its statistical computing and graphics capabilities, provides efficient execution of complex data analysis and visualization tasks for data scientists and analysts.

Due to its open source nature and active community, RStudio is constantly being extended with new packages and extensions, allowing the environment to be customized to meet the specific needs of big data analysis [1].

RStudio's adaptability is demonstrated by its extensive integration with various data sources: relational databases (MySQL, PostgreSQL), big data platforms (Apache Hadoop, Apache Spark), cloud storage (Amazon S3, Google Cloud Storage), and APIs for extracting data from the web, facilitated through a diverse set of R packages [2].

The power of RStudio in big data analysis is emphasized by its loose integration with big data technologies, enabling distributed computing and efficient data management, which greatly increases productivity in data analysis tasks. RStudio offers extensive capabilities for big data analytics. Direct integration with big data platforms such as Apache Spark, providing advanced capabilities for distributed data processing and analysis through the sparklyr package.

Efficient data management through packages such as dplyr and data.table to manipulate large amounts of data, providing high-speed processing and ease of use. Data visualization using ggplot2 and other tools to create informative and expressive graphs of large datasets.

RStudio's integration with Apache Hadoop provides R users with the ability to work with the Hadoop ecosystem, including HDFS (Hadoop Distributed File System) and MapReduce, as well as other components of the Hadoop ecosystem such as Hive and HBase. This integration allows you to analyze large amounts of data stored in Hadoop directly from RStudio using various packages and interfaces.

The rhdfs package provides an interface to work with the HDFS file system from R. It can be used to upload, download, and manage files in HDFS, which is the foundation for working with data stored in Hadoop. Analysts can use the familiar R syntax to work with data stored in Hadoop without going into the details of MapReduce or HDFS.

The rmr2 package allows MapReduce tasks to be written in R, providing the ability to process data stored in Hadoop using Hadoop cluster resources for distributed data processing.

The rhbase package is designed to work with HBase, a distributed, scalable database that runs on top of HDFS. This package allows you to perform read and write operations to the HBase database directly from R.

There may not be a specialized package to work with the Hive package from R, but it is possible to use JDBC to connect to Hive and execute SQL queries to analyze the data stored in Hive.

Ability to harness the power of Hadoop to process large amounts of data directly from RStudio.

RStudio's applications span many fields. In the financial sector for risk analysis and portfolio optimization, in healthcare for epidemiological studies, and in marketing and social sciences for statistical modeling due to its ability to integrate with various data sources and technologies.

References

[1] The R Project for Statistical Computing [Electronic resource]. – Access mode: https://www.r-project.org/

[2] Data Analysis with RStudio: An Easygoing Introduction. DOI: 10.1007/978-3-662-62518-7. [Electronic resource]. – Access mode: https://www.researchgate.net/publication/348131081

IMPROVING DATA QUALITY FOR A THREE-FACTOR NONLINEAR REGRESSION MODEL FOR ESTIMATING THE SIZE OF WEB APPLICATIONS CREATED USING THE REACT FRAMEWORK

Makarova L., Hashko D.
E-mail: lidiia.makarova@nuos.edu.ua, dimagashko@gmail.com
Mykolaiv, Admiral Makarov National University of Shipbuilding

Software development is a complex system of different steps and stages from initial planning, analysis and design to development, testing and maintenance. Estimating labor intensity at each step, especially in development, is crucial for accurately allocating resources, setting realistic timelines, and managing workloads effectively. There are many ways to estimate the labor intensity of development, but most of them are based on the size of the application, for example, COCOMO II (Constructive Cost Model II) [1].

But to effectively apply these methods, first we need to estimate the size of the application being developed. One of the most accurate and powerful approaches for this purpose involves using a regression model [2] specifically designed for estimating the size of such applications.

In conference thesis of VI All-Ukrainian Scientific and Practical Internet Conference of Students, Postgraduates, and Young Scientists "Modern Information Systems and Technologies" (Kherson, 30.11.2023) we have presented our three-factor nonlinear regression model for estimating the size of web applications created using the React framework [3].

React is an open-source, front-end JavaScript framework for building web applications using component-based architecture [4]. It's one of the most popular frameworks in web development that allows developers to create scalable applications of any size with high performance.

The presented model is shown below:

$$Y = 10^{\varepsilon - 1,4302} X_1^{0,3568} X_2^{0,6256} X_3^{-0,1419}, \quad (1)$$

where: X_1 – the number of components,

X_2 – the average number of methods,

X_3 – reusability index (how often components are used in other components).

The model shows acceptable results on test data, but we have concerns about its quality metrics so we want to improve the quality of our model.

To measure the quality of the model we use such metrics as the coefficient of determination (R^2), the Mean Magnitude of Relative Error (*MMRE*) and *PRED* (0.25). For the initial model these metrics are 0.7862, 0.3213 and 0.5526 respectively.

First, it's important to note that the main difficulties in building a model for estimating the size of web applications created using the React framework is that React and JavaScript itself allows significant variability in how to write components and the code in general, so it makes it harder to build a quality model compared to more strict languages and technologies.

One of the possible ways to improve the quality of our model is to use another type data transformation. Currently, we're using logarithmic transformation to normalize source data, but we could try to use other transformations like Johnson's or Box-Cox transformations and see how it would affect the quality of our model.

But before changing how we process our data, we need to make sure that the data is clean and representative, which allows us to improve the quality of our model and have more solid base for further improvements. Because if the data is inconsistent or lacks representativeness then the model's quality may not reach its full potential whatever methods we use to process the data.

There are two ways to improve the quality of our data – first we can rethink what type of metrics we use to build our model (use other metrics or add more) or we can improve the way we take this data, for example, by adding more restrictions for source projects so the data we get from them is more consistent, relevant and representative.

For now, we don't want to change the metrics we use, because all the current metrics can be retrieved from UML Class Diagram and so the model is clear and easy to use. Also, the metrics were chosen based on specifics of the React framework and JavaScript language so they should be representative enough, but by using other metrics or adding more we can increase complexity of our model and make it harder to use.

First, we decided to add more restrictions for source projects, so we get more consistent data from them and have more solid base for further improvements.

Initially we wanted to make our model more universal, so it fits for estimating the size of any web application created using the React framework and because of this for our data we used any open-source projects created using the React framework available on GitHub [5] with no any other requirements. But the problem here is that both React and JavaScript evolve rapidly and so over time developers are using different methodologies, styles of code, best practices, etc. So, three metrics to build a universal model for projects created using the React framework may not be enough to build a model with a higher quality. But at the same time there also might be no reason for this – the model is intended to be used to estimate the size of newly created applications and its more likely that they would develop their applications using the latest practices. So, including older projects in the data just makes the data inconsistent and less representative for real-life usage.

The other problem with initial data is that it includes projects of different types. Mostly it contains web applications, but also some libraries, small games and test applications from some tutorials. But all of them have their own specifics that can affect their size significantly.

We want to build our model for estimating the size of real-life web applications, so we should not use other types of projects and even test web applications for our data. What makes it harder to find enough open-source projects, is that most of real-life web applications are proprietary and are not available for public use. We could also just build our model for estimating the size of libraries instead, because they are usually open-source and there are mostly unlimited number of them available, but they can be harder to analyze and be less consistent.

So, considering the above, we decided to add the following requirements to our data:

1) project age: created less than 2 years ago;

2) last commit: less than 6 months ago;

3)project type: real-life web applications.

Then based on these new requirements we found a new set of projects for our source data, and rebuilt the model with updated data. We also calculated the quality metrics for updated model and all of them showed that the quality of the model increased. The coefficient of determination (R^2), the Mean Magnitude of Relative Error (*MMRE*) and *PRED* (0.25) of the updated model are 0.8132, 0.3012 and 0.6527 respectively.

So, by introducing more requirements for source data we actually improved the quality of our model. Now, with a more solid base, we can use other methods to improve it ever further.

References

[1] Boehm B. Software Cost Estimation with Cocomo II. New Jersey, Prentice Hall. 2000. 544 p.

[2] Douglas C. Montgomery, Elizabeth A. Peck, G. Geoffrey Vining. Introduction to Linear Regression Analysis. John Wiley & Sons, Inc., Hoboken, New Jersey. 2012. 872p.

[3] Dmytro Hashko, Lidiya Makarova. A three-factor nonlinear regression model for estimating the size of web applications created using the React framework. Proceedings of the VI All-Ukrainian Scientific and Practical Internet Conference of Students, Postgraduates, and Young Scientists "Modern Information Systems and Technologies" (November 30, 2023, Khmelnytskyi, Kherson). Edited by Hryhorova A. Kherson: FOP Vyshemyrskyi V. Publishing House, 2023. Pp. 147-148.

[4] React. The library for web and native user interfaces. [Electronic resource]. – Access mode: https://reactjs.org/

[5] GitHub. [Electronic resource]. – Access mode: https://github.com.

FREE EDUCATION SERVICES WITH INTERACTIVE CONTENT

Mikhieiev I.
E-mail:ivan.mikhieiev@hneu.net
Kharkiv, Simon Kuznets Kharkiv National University of Economics

Interactive content is any type of material that conveys its content, encouraging the participation of users: pupils, students, training listeners, etc. Thanks to this, the interaction with the content develops from passive consumption (the user simply reads a book, or listens to an audio recording, or watches a video) to active engagement (to get some part of the educational material, the user needs to interact with the content).

The interactive approach is becoming increasingly popular in the field of education and professional development. Large educational content providers are developing and implementing such approaches, ranging from specific educational institutions to leading educational content providers such as LinkedIn Learning [1].

Today, the main types of content in the educational process of educational institutions are: books and other options of educational, scientific and methodical literature, audio and video content. Each of them has its strengths. So, for example, books provide a deep analytical approach and the ability to present detailed concepts and ideas; readers can work with books at their own pace, pausing to think and process information; for people who learn information better by hearing, audio recordings can be an effective means of obtaining information; it is also worth noting that listening to audio recordings is easy to combine with other activities; video allows the use of visual effects, graphics, demonstrations, which facilitate the understanding of complex concepts; it is important that audio and video can convey not only information, but also emotions through the body language and voice of the speaker. Interactive content allows you to combine the positive aspects of text, audio and video presentation of information.

The creation of professional interactive educational content requires compliance with the following key principles [1, 2]. First, there are clear learning objectives: it is necessary to specifically define the learning outcomes in order to effectively develop and manage the design of interactive content, it is necessary to formalize and declare the knowledge and skills that students should acquire during the study. The second important factor is the variety of formats and interactive elements, such as formatted text, links, tables, flip cards, drop-down elements, "hot" buttons on pictures and diagrams, highlighting of program code and hints for explanations, screencasts, etc. A third important component is feedback and progress tracking: interactive content should provide immediate feedback to students based on their interactions, including reinforcing correct answers and offering guidance on incorrect answers. Mechanisms to track progress, such as progress bars or badges, motivate students and create a sense of achievement. The most difficult, but very important basis for interactive content is personalization. In order to meet the needs of individual students, taking into account their prior knowledge, position and skill level, it is necessary to provide opportunities to customize your interactive interaction, as well as opportunities to personalize content on a human level, even if it is just by addressing the listener by name.

Referances

[1] From Passive to Active: The Training Advantages of Interactive Content [Electronic resource] - Access mode: https://www.linkedin.com/pulse/from-passive-active-training-advantages-interactive-content/.

[2] Neelakandan N. Interactive Learning Content In eLearning: How Effective Is It? [Electronic resource] - Access mode: https://elearningindustry.com/interactive-learning-content-elearning-how-effective-is-it.

ANALISIS OF FREE INSTRUMENTS FOR GENERATING AND MANIPULATING STATISTICAL DATA

Zhuravka A.V., Ivanov A., Laetitia Villeneuve
E-mail: andrii.zhuravka@nure.ua
Kharkiv, Kharkiv National University of Radioelectronics

When conducting an analysis of free instruments for generating and manipulating statistical data, it's important to consider various aspects to ensure the tools align with your specific requirements. Here's an overview of key considerations:

Functionality. Evaluate the tools' statistical capabilities, including the range of analyses they support (descriptive statistics, hypothesis testing, regression analysis, etc.).

Check if the tools offer advanced features such as data visualization, correlation analysis, and the ability to handle different types of statistical distributions.

Ease of Use. Consider the user interface and overall user experience. Look for tools that are intuitive and have a low learning curve.

Assess the availability of tutorials, documentation, and user guides to facilitate easy adoption.

Data Import and Export. Check the compatibility of the tools with various data formats. Ensure they can easily import data from common sources (CSV, Excel, databases) and export results to desired formats.

Evaluate how efficiently the tools handle large datasets and if they provide options for data cleaning and preprocessing.

Community Support. Investigate the size and activity of the user community. A vibrant community often indicates ongoing development, support, and a wealth of shared knowledge.

Check for forums, discussion groups, or online communities where users can seek help and share experiences.

Open-Source Nature. Verify the open-source licensing of the tools. Open-source tools often have transparent development processes and allow for customization.

Consider the availability of source code, as it can be important for organizations with specific needs or security requirements.

Interoperability. Assess whether the tools can integrate seamlessly with other software commonly used in your workflow, such as databases, data visualization tools, or programming languages.

Check for compatibility with common data science libraries if you plan to use the tools in conjunction with programming languages like Python or R.

Scalability. Evaluate the tools' ability to scale with larger datasets or more complex analyses. Consider whether they can meet your requirements as your data and analysis needs grow.

Customization and Extensibility. Check if the tools support customization through scripting or plugins. This allows users to tailor the tools to specific needs or extend their functionality.

Performance. Assess the performance of the tools, especially when handling computationally intensive tasks or large datasets. Consider factors such as processing speed and resource utilization.

Security and Privacy. Ensure that the tools adhere to security and privacy standards, especially if dealing with sensitive or confidential data. Consider encryption, access controls, and compliance with relevant regulations.

By thoroughly analyzing these factors, you can make informed decisions about which free statistical tools align best with your analytical goals and organizational requirements.

Documentation. Assess the quality and comprehensiveness of documentation provided by the tools. Well-documented tools make it easier for users to understand functionalities, troubleshoot issues, and effectively utilize features.

Learning Resources. Check for the availability of tutorials, online courses, or other educational resources. A rich set of learning materials can facilitate the onboarding process for users with varying levels of statistical and technical expertise.

User Support. Evaluate the level of user support offered by the tool's community or development team. This could include forums, mailing lists, or dedicated support channels. Consider the responsiveness and helpfulness of the support system.

Update Frequency. Look into the frequency of updates and releases. Regular updates can indicate active development and the addition of new features, bug fixes, and security patches.

Platform Compatibility. Check if the tools are compatible with various operating systems (Windows, macOS, Linux) and consider whether they support both desktop and server-based installations.

Data Visualization Capabilities. Evaluate the tools' data visualization features. Effective visualization is essential for interpreting statistical results. Consider whether the tools provide customizable charts, graphs, and other visualization options.

Data Transformation and Cleaning. Assess the tools' capabilities for data transformation and cleaning. Look for features that simplify the process of handling missing values, outliers, and data imputation.

Statistical Libraries and Algorithms. If the tool is scriptable or programmable, check for the availability of statistical libraries and algorithms. This is particularly important for users who prefer to write custom scripts for their analyses.

Collaboration Features. Consider whether the tools offer collaboration features, such as version control for statistical analyses, sharing capabilities, and collaborative editing. These features are valuable for team-based projects.

Licensing Limitations. Review the licensing terms and limitations associated with the free tools. Some tools may have restrictions on commercial use, redistribution, or modifications, so it's essential to be aware of any licensing constraints.

Collection of statistical data, their analysis, processing, visualization, evaluation and development of propositions or creation of revisions It is a daily task of any kind of direct activity. The tool for creating and processing statistical data is a database. Since the last century, the standard workflow for processing material has evolved primarily from collecting information to feed a database. With the development of technology, software systems have begun to appear on the market that allow analysis and visualization based on collected data in the form of tables, graphs or diagrams. The current analysis of the database allows us to evaluate the collected information and provide recommendations for improving the efficiency of work directly. Analysis of light data showed that the simplest and most accessible software systems are Second Prism, DataMarket, OfficeReports, Q Research Software. The stench can become progressive at the current stage. Q Research Software is a valuable alternative to SPSS, SAS and other systems. Splintered by the Australian company Numbers, Inc. Q offers a robust and easy-to-use interface, as well as features such as selection analysis, rich analysis, predictive modeling and an interactive toolbar for creating charts and working with data that you can I visualize the results and prepare follow-up reports.

FREE TECHNOLOGIES FOR DEVELOPMENT OF USER INTERFACES

Zhuravka A.V., Snihurov A.V, Ivanov A.
E-mail: andrii.zhuravka@nure.ua
Kharkiv, Kharkiv National University of Radioelectronics

he interface holds significant importance for any software system and is an integral component primarily oriented towards end-users. It is through the interface that users form judgments about the overall usability of an application. Moreover, users often decide to use an application based on how comfortable and understandable its interface is to them. However, the design and development of interfaces are notably labor-intensive.

Importance of User Interface (UI). The user interface is a critical component as it serves as the bridge between users and software applications. A well-designed UI enhances user experience, making applications more accessible, intuitive, and enjoyable.

User judgments about the quality of an application often heavily rely on their interaction with the UI. A user-friendly interface can positively influence user satisfaction and the overall success of a software product.

Batch Processing Technology. Batch processing involves the execution of a series of programs or tasks without direct user interaction. This technology has historical significance, especially in early computing systems where users submitted batches of tasks to be processed sequentially.

Common input formats for batch processing include punched tapes, stacks of punch cards, or sequences of key presses on electric typewriters. It was widely used for tasks like data processing and running scheduled jobs.

Command Line Interface (CLI) Technology. CLI relies on text-based commands entered by users through a keyboard. The computer responds with text outputs displayed on a monitor.

The CLI is known for its efficiency and precision, often preferred by advanced users and administrators. It was prevalent in the early days of computing and is still widely used in many operating systems and development environments.

Biometric Technology (Facial Interface). Biometric technology uses unique physiological or behavioral characteristics for user identification. Facial recognition is a prominent example where a person's facial features are captured and analyzed for authentication.

The technology has applications in security systems, access control, and user authentication. It provides a convenient and secure way to interact with devices, especially in mobile and computing environments.

Text Files in Command Line Interfaces. In CLI environments, text files are fundamental for storing and manipulating data. These files typically contain plain text and are easy to create using command-line commands.

Text files play a crucial role in scripting, configuration, and data interchange in command-line-driven workflows.

Ongoing Challenges in UI Design and Development. Despite advancements, UI design and development remain challenging due to the diverse user base, evolving technologies, and the need for constant innovation.

The effort involved in creating user-friendly interfaces extends beyond the initial design phase, requiring ongoing updates, improvements, and adaptations to changing user needs.

Future Trends in User Interface Technology. Emerging technologies such as voice interfaces, gesture recognition, and augmented reality are shaping the future of user interfaces.

Designers are exploring ways to create more immersive and natural interactions, reducing the reliance on traditional input devices.

Human-Computer Interaction (HCI). HCI is a multidisciplinary field that studies the interaction between humans and computers. It encompasses the design, evaluation, and implementation of user interfaces to ensure a positive user experience.

Understanding human behavior, cognitive processes, and ergonomic principles is crucial in HCI to create interfaces that align with users' mental models and expectations.

In summary, the evolution of user interfaces reflects the ongoing quest to make technology more accessible, efficient, and user-friendly. The journey includes diverse technologies and methodologies, each with its unique contributions and challenges.

Every time you interact with a computer, including turning it on, you engage with a user interface (UI) designed for users. This interface may seem simple and straightforward, but the industry has invested a considerable amount of effort to make it so.

Technologies for developing user interfaces (UI) are a crucial domain in information technology, focused on creating user-friendly and efficient interfaces for interacting with software products. Here are several key technologies in this field.

HTML, CSS, and JavaScript. HTML (Hypertext Markup Language): Used for structuring web pages and defining their elements.

CSS (Cascading Style Sheets). Responsible for the appearance and style of web pages, including colors, fonts, and element positioning.

JavaScript. Provides dynamism to websites, allowing interaction with page elements without the need for reloading.

React, Angular, and Vue.js. React: A user interface library developed by Facebook, using a virtual DOM for efficient element updates.

Angular. A framework developed by Google, facilitating the creation of dynamic single-page applications (SPAs).

Vue.js. A lightweight framework for building interfaces, allowing incremental implementation of functionality.

Flutter and React Native. Flutter: Developed by Google, this framework enables the creation of cross-platform mobile applications from a single codebase using the Dart programming language.

React Native: From Facebook, this framework allows the development of mobile applications for both Android and iOS using React and JavaScript.

Adobe XD, Sketch, and Figma. Adobe XD: Software for designing user interfaces and prototyping. Sketch: A graphic editor specialized in designing interfaces for mobile and web applications. Figma: An online design and prototyping tool that enables collaborative work on projects.

GraphQL. Used to optimize client-server interaction, providing a flexible query interface to obtain necessary data, reducing network loads.

AR/VR Technologies. Integration of augmented and virtual reality into application interfaces, expanding interaction possibilities.

These technologies interact to create modern, efficient, and user-friendly interfaces for various platforms and devices. Developers utilize these tools to craft interfaces that not only meet functional requirements but also provide a pleasant and productive user experience.

ФЕНОМЕН API ЯК ФАКТОР СИНЕРГІЇ У МІЖСИСТЕМНІЙ КОМУНІКАЦІЇ ІНТЕЛЕКТУАЛЬНИХ КОМПОНЕНТІВ БІЗНЕС-СИСТЕМ

Андрейчіков О.О.
Керівник: Старкова О.В.
E-mail: andreychikov.oleksandr@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

У сучасному світі ведення бізнесу невіддільно пов'язане з використанням різноманітних інформаційних та комунікаційних технологій. Найпоширенішими аспектами такої бізнес діяльності є управління сторінками у соціальних мережах, ведення власних веб-сайтів та мобільних додатків або магазинів на маркетплейсах (Rozetka, Amazon тощо), використання інтернет-банкінгу, а для більш технологічно складних типів бізнесу ще й систем CRM/ERP, хмарних сховищ та інших сервісів, які забезпечують обмін даними між різними програмами та їх компонентами в реальному часі та дають можливість отримувати цінні інсайти і знання з накопичених даних. При чому більшість цих IT-складових пов’язані між собою за допомогою технології API (Application Programming Interface) – інтерфейс програмного забезпечення, який надає можливість двом або більше програмам (додаткам) взаємодіяти між собою за допомогою певних наборів функцій (інструкцій у вигляді блоків коду, які можуть бути викликані іншими програмами, в тому числі віддалено). Іншими словами API – це контракт взаємодії (специфікація) між постачальником API та його користувачами, в якому детально описуються наявні методи API, підтримувані формати та структури даних, очікувані вхідні дані, можливі відповіді, коди помилок та інші важливі

інструкції. Дотримуючись цього узгодженого контракту як постачальник API, так і споживач можуть забезпечити безшовну інтеграцію та взаємодію між відповідними програмними компонентами.

Технологія API вже настільки широко впроваджена у різні цифрові бізнес сервіси, що без неї вже важко уявити наше звичне життя. Так завдяки цій технології, через яку між собою комунікують сервіси, додатки та програми, мобільний застосунок інтернет-банкінгу показує стан рахунків (баланси, історії транзакцій тощо), служби доставки їжі через свої застосунки показують пропозиції від ресторанів та магазинів, служби таксі завдяки API розраховують вартість та маршрути поїздок, макретплейси та інтернет-магазини можуть онлайн продавати та доставляти товари через поштові служби, більшість державних е-сервісів в Україні також побудовані на технології API, завдяки якій відбувається обмін даними між реєстрами, базами даних тощо. Рекламні компанії в інтернет, туристичні послуги, покупка авіа квитків або у кінотеатри, авторизація на сайтах за допомогою аккаунтів Google або Facebook (SSO, Single Sign-On), взаємодія з голосовими помічниками, чат-ботами, хмарними сховищами тощо також побудовані на базі API – і це лише невелика частина прикладів. Таким чином, комунікація через API породжує складні бізнес-системи, які є сукупністю великої кількості суб'єктів у взаємодії, що дозволяє через інтеграцію завершити загальну місію. Утворені таким чином складні системи характеризуються новими властивостями, які існують лише на системному рівні і не можуть спостерігатись на рівні цих складових.

Формалізуючи викладене, можна сказати, що завдяки технології API відбувається міжсистемна комунікація (обмін даними) між інтелектуальними компонентами (програмами з їх бізнес-логікою, сховищами та базами даних, реєстрами тощо) різних бізнес-систем, які завдяки такій комунікації у режимі реального часу утворюють екосистемні інтеграційні мережеві (end-to-end) продукти та сервіси. Для бізнесу це означає можливість знаходити клієнтів, надавати їм послуги онлайн, розширювати асортимент та створювати принципово нові види продуктів та послуг, покращувати їх якість, замість персоналу залучати клієнта до самообслуговування на певних етапах надання послуг в режимі 24/7/365, що значно знижує витрати і т.д. Всі ці переваги бізнес отримує від комунікації з іншими бізнес-одиницями в процесі об'єднання зусиль та ресурсів потрібних для виконання необхідних бізнес-процесів. Це і є проявом синергії, коли результат складання (об’єднання) частин більший за їх суму.

Окрім наведених переваг комунікацій між бізнес-системами за допомогою API, треба також детальніше зупинитись і на значенні для бізнесу комунікацій як таких. Комунікація (від лат. communicatio – єдність, передача, пов'язаного з дієсловом лат. communico – роблю спільним, з'єдную, яке в свою чергу походить від лат. communis – спільний) – це в першу чергу процес обміну (даними, інформацією тощо), без якого успішне функціонування сучасного бізнесу вже майже неможливе. Так в роботі [1] зазначено, що комунікації для бізнесу – це все, бо комунікаціями просякнуті всі бізнес-процеси та робота компаній, тому так важливо приділити їм багато уваги. Підтвердження даної тези міститься також в роботі [2, ст. 86], де автором була розроблена математична модель інтелектуального капіталу як множини властивостей, якими володіють компанії та завдяки яким вони можуть встановлювати ті чи інші зовнішні зв’язки з ринком як надсистемою. Дане твердження також цілком корелюється і з основною функцією API – надавати доступ до певних даних, тобто робити їх спільними в рамках певних бізнес-процесів за визначеною логікою та правилами. При чому процес обміну даними та інформацією важливий для сучасного бізнесу не лише із зовнішнім середовищем, як показано вище, а ще й для внутрішнього середовища, тобто всередині компанії на різних рівнях (горизонтальному, вертикальному, діагональному), що через систему управління компанії врешті впливає на результати її діяльності. Комунікації в процесі їх реалізації формують капітал відносин, який є частиною інтелектуального капіталу компаній. Зокрема цей аспект був розглянутий автором в роботі [3, ст. 11], де за допомогою системологічного підходу була розроблена онтологія інтелектуального капіталу, в якому як окремий вид (підсистема) виділяється саме комунікативний капітал (зовнішній і внутрішній),

який визначає якість взаємодії компанії відповідно у зовнішньому та внутрішньому середовищі. Тобто було визначено, що система комунікацій та відносин для компанії є важливим елементом її інтелектуального капіталу, завдяки якому, як показано у роботі [4, ст. 20], компанії отримують значно більшу додану вартість та конкурентну перевагу на відміну від фізичного капіталу через те, що в сучасному світі змістився фокус з фізичних активів на активи, засновані на знаннях, які все більше стають рушійною силою економічного зростання та інновацій.

Окрім прояву синергічного ефекту, феномен API також корелюється і з моделлю Барабаші-Альберт (БА) [5, ст. 8] – алгоритмом генерації випадкових безмасштабних мереж з використанням принципу переважного приєднання. Переважне приєднання означає, що чим підключений вузол (концентратор) більший, тим більша ймовірність, що він отримає нові посилання. Вузли з більш високим ступенем мають сильнішу здатність захоплювати посилання, додані в мережу. Іншими словами, коли бізнес-система починає використовувати технології API і стає досить масштабною, тобто збільшує кількість користувачів свого API у порівнянні з іншими вузлами, то вірогідність її масштабування (за рахунок приєднання нових користувачів) також збільшується відносно менших вузлів, що призводить до нових інтеграцій (масштабування) і, як результат, до збільшення прибутку. І в цьому зв’язку, описаний у роботі [1], принцип «спочатку стійкість – потім масштабування», виглядає виправданим та логічним при побудові компаній та бізнес-процесів.

Таким чином можна зробити висновок, що технологія API є втіленням формалізованих зв’язків компаній у капіталі відносин як важливого елемента інтелектуального капіталу, завдяки якому можна отримувати значно більший ефект від своєї діяльності, що й уособлює принцип синергії в бізнесі на прикладі інформаційно-комунікаційних технологій.

Література

[1] Стійке масштабування бізнесу: чому без комунікацій нічого не вийде – колонка. [Electronic resource]. – Access mode: https://ain.ua/2024/02/02/chomu-bez-komunikaczij-nichogo-ne-vyjde-kolonka/

[2] Андрейчіков О. О. Розробка математичної моделі ІК як системи інформаційних зв'язків організації. Вісник національного технічного університету «ХПІ». Тематичний випуск: Нові рішення в сучасних технологіях. Харків: НТУ «ХПІ» – 2011. – №54. – С. 81-86. – 156 с. [Electronic resource]. – Access mode: https://repository.kpi.kharkov.ua/server/api/core/bitstreams/8b8a033a-01c2-4d56-9700-38830e89b9cd/content

[3] Андрейчіков О. О. Онтологічна модель інтелектуального капіталу. Східно-Європейський журнал передових технологій №5/2(53). Харків, 2011. – С. 8-11. – 62 с. [Electronic resource]. – Access mode: https://journals.uran.ua/eejet/article/view/510

[4] Старкова О.В., Андрейчіков О.О., Роль інтелектуального капіталу в контексті розвитку ІТ-компаній. Інформаційні технології та інженерія : Всеукраїнська науково-практична конференція молодих вчених, аспірантів і студентів : тези доп., 31 січня – 2 лютого 2024 р. / ЧНУ імені Петра Могили. Миколаїв, 2024. – С. 20-22. – 140 с. [Electronic resource]. – Access mode: https://dspace.chmnu.edu.ua/jspui/handle/123456789/1661

[5] Albert-Laszlo Barabasi Network Science The Barabasi-Albert Model. [Electronic resource]. – Access mode: https://barabasi.com/f/622.pdf

ЗАСТОСУВАННЯ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗАЦІЇ ЙМОВІРНІСНОГО ПІДХОДУ ДО ВИЗНАЧЕННЯ ПРАЦЕЗДАТНОСТІ СИСТЕМИ

Бугай І.С.
Керівник: Солодовник Г.В.
E-mail: ganna.solodovnyk@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Актуальність. В сучасному технологічному світі застосування вільного програмного забезпечення надає значні переваги для наукових, інженерних та комерційних робіт, а зокрема, у сфері аналізу та оцінки функціонування систем. Використання вільного програмного забезпечення та забезпечення з відкритим кодом для аналізу систем знижує загальні витрати на розробку програм, оскільки відсутня потреба в придбанні ліцензій, шо робить його особливо привабливим для стартапів, наукових організацій та освітніх закладів з обмеженим бюджетом.

З іншого боку гнучкість відкритого вихідного коду дозволяє користувачам адаптувати програмне забезпечення під свої специфічні потреби, а це в свою чергу сприяє інноваційним дослідженням та розробкам у галузі визначення працездатності систем. До того ж вільне програмне забезпечення дозволяє ефективно реалізовувати складні математичні моделі та алгоритми для ймовірнісного аналізу, що є критично важливим для визначення надійності та працездатності систем.

Підсумовуюче сказане можна зробити висновок, що автоматизація визначення ймовірностей працездатності системи з використанням програмного забезпечення з відкритим вихідним кодом є актуальною проблемою, рішення якої сприяє підвищенню точності та ефективності отриманих оцінок ймовірностей.

За об’єкт дослідження інформаційних ризиків та вирішення завдань щодо інформаційної безпеки можна вважати компоненти, тобто засоби автоматизації та інформаційні технології та інформаційні ресурси, які накопичуються та обробляються такими компонентами.

Метою роботи є автоматизація розв’язання задачі знаходження ймовірностей роботи та виходу з ладу компонент устаткування за відомих середніх значень часу безвідмовної роботи та ремонту окремих компонент.

Формальна постанова проблеми полягає в тому, що в інформаційній системі компанії є устаткування, яке складається з двох компонент. Елемент цього устаткування може перебувати в таких станах: S_0 – елемент справний і вільний (простій), S_1 – елемент є справним і зайнятим (обробка), S_2 – елемент є справним, але відбувається налагодження програмного забезпечення, S_3 – елемент є несправним. Потрібно розрахувати фінальні ймовірності перебування елемента в кожному зі станів, враховуючи такі значення інтенсивностей відповідних потоків подій: закінчення ремонту (λ_0); замовлення на обробку (λ_1); виникнення потреби в оновлені ПЗ (λ_2); вихід елементу з ладу (λ_3); закінчення оновлення ПЗ (λ_4).

В такій постановці задачі описано марковський випадковий процес, тому що час виходу з ладу та час, затрачений на ремонт компоненти, у майбутньому не залежать від її станів у минулому, а тільки від того в якому стані вона перебуває в теперішній час. Крім того, якщо прийняти припущення про те, що перехід компоненти з одного стану в інший відбувається практично миттєво у випадкові моменти часу даний марковський процес можна вважати процесом з дискретними станами та неперервним часом.

Розв’язання задачі передбачає виконання наступних пунктів:

1. Побудови розміченого графа станів для компоненти згідно з умовами задачі (рис. 1).
2. Складання рівнянь Колмогорова для побудованого графа станів компоненти.

3. Переходу до фінальних ймовірностей станів, які можна інтерпретувати як середній відносний час перебування компоненти в певному стані. Фінальні ймовірності – це ймовірності станів системи за t→∞, для системи, яка перебуває в граничному стаціонарному режимі, в якому випадковим чином змінюються її стани, але ймовірності вже не залежать від часу.

Отримання значень фінальних ймовірностей роботи та ремонту компоненти шляхом розв’язання системи рівнянь методом Гауса.

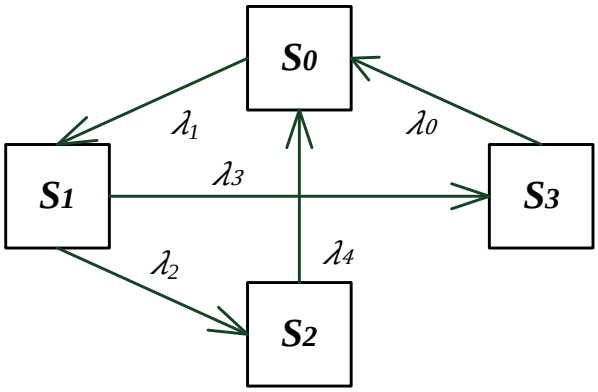


Рисунок 1 – Граф станів компоненти

Систему рівнянь Колмогорова для даної задачі можна розв’язати засобами електронних таблиць Calc одного з офісних пакетів з відкритим кодом LibreOffice або ApacheOpenOffice [2]. Ефективнішим рішенням є розв’язання задачі за допомогою однієї з математичних програм, таких як Python з бібліотеками NumPy або MATLAB, або за допомогою калькуляторів для матричних операцій. В даній роботі будо використано бібліотеку NumPy.

NumPy – це проект з відкритим кодом, розроблений групою учасників, які взяли на себе зобов'язання створити відкриту, інклюзивну та позитивну спільноту. NumPy забезпечує: потужний N-вимірний об'єкт масиву, складні (мовлення) функції, інструменти для інтеграції коду C/C + + і Fortran, реалізацію функцій лінійної алгебра, перетворення Фур'є та можливості роботи з випадковими числами [3].

Перевагами NumPy є швидкодія: надання оптимізованих функцій та методів для роботи с масивами; зручність використання: можливості роботи з простими та зрозумілими функціями обробки масивів та матриць; інтеграція іншими бібліотеками, оскільки NumPy є основою для багатьох інших наукових та аналітичних бібліотек: pandas, matplotlib, scikit-learn тощо [4].

Висновки. Використання бібліотеки NumPy для автоматизації ймовірнісного підходу до визначення працездатності технічної системи дозволило підвищити продуктивність, облегшити розробку програмного забезпечення, а також надати можливості інтегрування з іншими популярними бібліотеками аналітики даних для подальшого розвитку системи.

Література.

[1] Сайт Stud.com.ua Рівняння Колмогорова. граничні ймовірності станів [Електронний ресурс]. – Режим доступу до ресурсу: https://stud.com.ua/80825/ekonomika/rivnyannya_kolmogorova_granichni_ymovirnosti_staniv

[2] Сайт TopTut // 30 найкращих прикладів програмного забезпечення з відкритим кодом (список готових до 2024 року) [Електронний ресурс]. – Режим доступу до ресурсу: https://www.toptut.com/uk/list-of-open-source-software/

[3] Сайт NumPy // numpy 1.26.4 [Електронний ресурс]. – Режим доступу до ресурсу: https://pypi.org/project/numpy/

[4] Сайт ElaKpi // Мова програмування Python. NumPy в Python [Електронний ресурс]. – Режим доступу до ресурсу: https://ela.kpi.ua/items/fba3621a-09ab-4f4a-8af0-2331e953ef94

ЗАСТОСУВАННЯ ЗАСОБІВ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У ПРИЙНЯТТІ РІШЕНЬ В УМОВАХ НЕВИЗНАЧЕНОСТІ

Вертебний М.М.
Керівник: Солодовник Г.В.
E-mail: ganna.solodovnyk@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Актуальність. Для сучасних умов функціонування економіки є характерними високий рівень невизначеності та перманентна стохастичність процесів реалізації рішень. Такий стан речей обумовлює зростання ризиків та ступеня невизначеності під час прийняття управлінських рішень. Невизначеність економічних процесів формує нові вимоги до управління соціально-економічними системами всіх рівнів та сфер діяльності. В такій ситуації застосування класичних методів й моделей без подальшої автоматизації є малоефективним. Тому актуальною є необхідність вивчення джерел ризику та невизначеності, врахування невизначеності в процесі підготовки та прийняття рішень, а також автоматизація цих процесів шляхом створення та впровадження інформаційних систем, які реалізують математичні моделі управління соціально-економічними системами в умовах неповноти інформації [1].

Іншим аспектом, який обумовлює актуальність роботи є динамічність самих соціально-економічних систем, що вимагає моделювання та автоматизації процесів управління складними організаційними системами. Функціонування та розвиток таких систем неможливий без досліджень законів управління соціально-економічними об’єктами, що передбачає розробку математичних моделей та комп’ютерні експерименти з ними. Предметом досліджень та автоматизації під час управління соціально-економічними системами є процеси структурної організації та функціонування, а також механізми їх поведінки.

Метою роботи є автоматизація процесу прийняття рішень в умовах ризику та невизначеності із застосуванням теоретико-ігрової концепції засобами вільно розповсюдженого програмного забезпечення.

Моделювання процесу прийняття рішень в умовах неповноти інформації, конфліктності та обумовленого цими факторами ризику ґрунтуються на різних концепціях. Однією з яких є концепція теорії гри та статистичних рішень. Теоретико-ігрова концепція є відомою, добре дослідженою та широко використовується у практиці розробки та прийняття управлінських рішень.

Теорія гри є розділом математики, який вивчає математичні моделі прийняття рішень в умовах конфліктності або невизначеності, коли інтереси сторін є протилежними (антагоністичні ігри), або не співпадають (ігри з непротилежними інтересами) [2]. Найчастішу ситуацію прийняття рішень в умовах ризику та невизначеності розглядають як гру з природою. За такої постанови задача формулюється як необхідність вибору найкращої альтернативи у ситуаціях, коли свідомо діючий гравець (перший гравець або суб’єкт управління) діє проти природи, яка проти першого гравця не діє свідомо, а обирає власні стани випадковим чином. В даній концепції під природою розуміють сукупність економічних, політичних, кліматичних та інших чинників, під дією яких складаються умови функціонування об’єкта управління.

Згідно з моделлю гри з природою ситуація прийняття рішень описується трійкою множин: множина рішень першого гравця; множина станів природи або зовнішнього середовища; функціонал оцінювання, який визначається на декартовому добутку множин рішень першого гравця та станів природи. Якщо елементи функціоналу оцінювання є функціями виграшу першого гравця, то вся ситуація прийняття рішення (в дискретному випадку) описується матрицею виграшу першого гравця [3].

Вибір найкращого управлінського рішення здійснюється згідно критеріїв прийняття рішення з множини критеріїв, які є характерними для певної інформаційної ситуації. В даній роботі виокремлено дві інформаційні ситуації: ситуація ризику, для якої характерним є

наявність апріорних ймовірностей настання кожного зі станів природи; ситуація невизначеності, для якої характерним є відсутність апріорних ймовірностей настання кожного зі станів природи на момент ухвалення рішення. Для першої ситуації використано критерій максимуму очікуваного середнього виграшу, для другої – критерії максимаксу, Вальда, Севіджа, Гурвиця.

Підвищення ефективності управлінських рішень може бути досягнуто за допомогою використання систем підтримки прийняття рішень (СППР), що є комплексом програмних засобів, які за використання технічного та програмного забезпечення, даних та набору моделей забезпечують підтримку всіх етапів процесу прийняття рішень.

Прикладами таких систем можуть слугувати системи родини Expert. Так система Marketing Expert призначена для стратегічного планування та аудиту маркетингу. Це інструмент розробки та прийняття стратегічних та тактичних рішень, яка дозволяє реалізовувати аналітичні операції, будувати моделі компанії.

Іншим прикладом СППР є Business Analytics – програмне забезпечення для аналізу та моделювання бізнес процесів та прийняття рішень на основі статистичних даних у режимі реального часу

Наведені приклади не належать до вільно розповсюдженого програмного забезпечення, проте можуть мати можливості безкоштовного використання протягом певного часу. Проте, як правило, в СППР не враховані умови ризику та невизначеності, тому в роботі використано табличний процесор для автоматизації теоретико-ігрових моделей.

Найпоширенішими табличними процесорами, які надаються за умовами вільних ліцензій є: LibreOffice Calc, OpenOffice.org Calc, Gnumeric, WPS Spreadsheets. За інструментарій автоматизації обрано LibreOffice Calc, який є частиною офісного пакету LibreOffice [5] і розповсюджується за ліцензією Mozilla Public License v2.0. Можливості цього програмного засобу складають: обробка та візуалізація табличних даних, аналіз даних, прийняття рішень, експорт електронних таблиць в Adobe PDF та в HTML. Після форкінгу з OpenOffice.org в 2010 році LibreOffice Calc зазнав значного вдосконалення з метою виправлення багатьох дефектів у розрахунках формул, які включають зовнішні посилання, підвищення продуктивність кешування даних.

Висновок. Засобами табличного процесора LibreOffice Calc було автоматизовано процес прийняття рішень в умовах ризику та невизначеності за використання теоретико-ігрової концепції. Використання програмного забезпечення, яке надержить до безкоштовних застосунків з відкиним кодом може бути ефективно використано в процесі розробки та прийняття управлінських рішень. Автоматизація таких процесів дозволяє підвищити наукову обґрунтованість управлінських рішень та забезпечити конкурентоспроможність та розвиток об’єктів управляння.

Література.

[1] Акулов М.Г., Тютюніков І.Є., Куперштейн Л.М., Ткаченко М.І. Моделювання економічної динаміки / Навч. посібник. Під ред. М.Г. Акулова – Вінниця.: ВФЕУ, 2017. – 310 с.

[2] Моделі та методи прийняття рішень: навчальний посібник / Л. Нікітіна І. Яценко. – Харків: НТУ «ХПІ», 2023. – 179 с.

[3] Солодовник Г.В. Управління економічним та інформаційним ризиком: навчальний посібник. – Х.: ТОВ «ДІСА ПЛЮС», 2018. -152 с.

[4] Сайт Dynatrace // Business Analytics [Електронний ресурс]. – Режим доступу до ресурсу: https://www.dynatrace.com/monitoring/platform/business-analytics/?utm_source=google&utm_medium=cpc&utm_term=business%20analytics%20platform&utm_campaign=emea-south-dem-dem&utm_content=none&utm_campaign_id=10867142597&gclsrc=aw.ds&gad_source=1&gclid=CjwKCAiAq4KuBhA6EiwArMAw1JLK8dBORKiuQRyDL7EaITPa2mwRVDvSTu47IhdSV-fU1lN9CBg0aBoCKGcQAvD_BwE

[5] Офіційний сайт Open Office: [Електронний ресурс] / Режим доступу: http://www.openoffice.org.

ОГЛЯД МОЖЛИВОСТЕЙ ЗАСТОСУНКУ BLENDER ДЛЯ ПРОЦЕДУРНОГО МОДЕЛЮВАННЯ

Гречишкін Д.С.,
Керівник: Яковлева О.В.
E-mail: danylo.hrechyshkin@nure.ua
Харків, Харківський національний університет радіоелектроніки

Комп'ютерна графіка завжди були галуззю, яка потребувала глибоких знань у комп'ютерних науках та значних розрахункових потужностей. Однією з підгалузей комп'ютерної графіки є тривімірна комп'ютерна графіка, яка використовується при створенні статичних зображень, сцен для кінематографу, відеоігор, та моделюванню симуляцій тканин, рідин, вогню та диму. На ринку існує декілька продуктів, які дозволяють вирішувати такі задачі, але всі вони розповсюджуються за платною моделлю, окрім Blender [1]. Blender розробляється Blender Foundation і спільнотою волонтерів з усього світу. Програма є кросплатформною і доступна для різних операційних систем, включно з Windows, macOS і Linux. Blender поширюється під ліцензією GNU General Public License (GPL), версія 3 або пізніша. GPL - одна з основних ліцензій вільного програмного забезпечення, що підкреслює важливість свободи використання, вивчення, розповсюдження та зміни програмного забезпечення. Вона гарантує, що всі користувачі мають свободу ділитися програмою та її поліпшеннями із суспільством, підтримуючи таким чином принципи відкритого програмного забезпечення.

Blender покриває всі базові вимоги, як і конкуренти, а саме:

моделювання, при якому користувач маніпулює вершинами, ребрами та полігонами, їх положенням;

скульптінг – режим який дозволяє працювати з моделями завдяки пензликам, які дозволяють робити високо деталізовану геометрію, що може підходити для фото реалістичних сцен (кіно, відеоігри, симуляції та моделювання);

UV-розгортка – позиціонування зображень на геометрію, в процесі якого створюється розгортка тривимірної моделі на площину;

фото реалістичний рендерінг – процес створення сцен, які в питаннях освітлення та матеріалів відповідають поведінці матеріалів і освітлення в житті;

симуляції – процес створення динамічних симуляцій рідин, тканин диму та вогню, користувач може обирати точність цим симуляцій, від чого відповідно буде збільшуватися вимога до ресурсів;

процедурне моделювання – моделювання яке використовує процедурні операції над геометрією та матеріалами (geometry nodes [2], material nodes [3]);

Операції по маніпуляції геометрією є необхідним мінімумом для подібного програмного забезпечення і вони запроваджені в належному стані [4]: користувач може створювали сцени які будуть налічувати мільйони трикутників і за умови достатньої кількості ресурсів, сцена буде опрацьовуватися швидко. І якщо фахівці, які використовують подібні програми, як правило, гарно розуміють принципи роботи базових функцій, необхідних при створенні сцен і моделей, то процедурне моделювання завжди вимагало більшого розуміння механізмів і понять, які в класичному моделюванні не присутні. Тому метою цього огляду є висвітлення аспектів процедурного моделювання, яке можна виконувати завдяки цьому програмному забезпеченню.

Щоб пояснити як процедурне моделювання працює і де воно може використовуватися звернемося до деяких прикладів та спочатку визначимось з поняттями. Рендерінг – візуалізація сцени при якій всі налаштування сцени та штучної камери використовуються при обрахуванні вихідного зображення. В залежності від рушія рендерінгу результат може бути фото реалістичним або стилізованим. Шейдери – це набір інструкцій за якими при рендерінгу буде відображатися поверхня матеріалу. Таким чином можна налаштовувати

типові матеріали: пластик, метал, скло. Ще в перших версіях програми були елементи вузлового програмування саме для шейдерів (material nodes).

На ряду з використанням текстурних карт користувач може використовувати карти процедурні. Зазвичай це представники різних алгоритмів генерації мап шумів (voronoi, magic, perlin, musgrave). Використовуючи ці мапи користувач може створити процедурні матеріали, які будуються виключно на процедурно сгенерованих значеннях.

Опускаючи деталі обробки цих мап з виділенням різних каналів, фільтрації зображення – це і є базова концепція процедурних матеріалів. На рисунку 1 приведено приклад таких матеріалів, при певних зусиллях вони можуть виглядати реалістично, яка показано на рисунку 2.

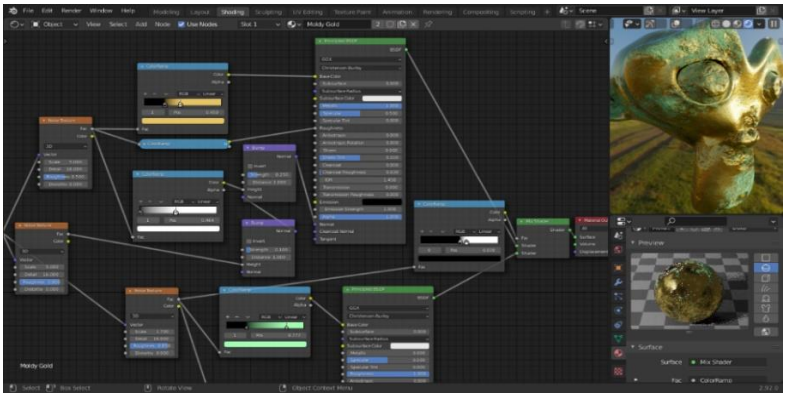


Рисунок 1 – Вузловий інтерфейс редагування матеріалів

Рисунок 2 – Реалістичні процедурні матеріали

До певного моменту це було все, що blender міг запропонувати з процедурного моделювання, але у 2021 році з версією 2.92 з’явилися geometry nodes, які дозволяли редагувати геометрію моделей.

Таким чином користувач може робити комплексні об’єкти, які будуть складатися з інших об’єктів, він може збирати моделі за якимись визначеними правилами (наприклад, процедурна будівля, яка має обмежений набір фасадів, але їх комбінації використовуються щоб створювати унікальні будівлі), розподіл незначних моделей по поверхні іншої моделі (створення більш коректних реалістичних поверхонь для фото реалістичних сцен) або створення цілих сцен лише завдяки геометричним вузлам.

Головним принципом в створенні процедурних геометричних моделей або сцен є конкретизація задачі і опис її через набір певних послідовних операції. Наприклад, треба створити певну кількість повторюваних коридорів. Для цього користувач спочатку створить окрему стіну, стелю підлогу та ліхтар для стелі. Після чого на власний розсуд треба вигадати сам алгоритм по якому ці повторювані елементи будуть збиратися. Одним з варіантів є генерація сітки, з якої довільно прибирають певні квадрати після чого контур, який лишився

заповнюють повторюваними елементами. Інакшим способом можна лишити користувачу можливість генерувати сам контур кімнат і вже потім заповнювати його. Обидва варіанти реалізації поставленої задачі можна побачити нижче, на рисунку 3-4.

Рисунок 3 – Рендер процедурно згенерованої кімнати першим алгоритмом

Рисунок 4 – Рендер процедурно згенерованої кімнати другим алгоритмом

Таким чином, геометричні вузли – це інструмент, який потребує дійсно глибоких знань і розуміння певних абстракцій, але при опануванні дозволяє пришвидшувати роботу при створенні певних моделей, сцен та візуалізацій.

Blender також може використовуватися під час наукових досліджень в області комп'ютерного зору для створення моделей, які максимально наближені до реальних об'єктів. Наприклад, за допомогою Blender можна створили датасет з тривимірних моделей облич та дослідити властивості різних детекторів облич, або методів для нормалізації та розпізнавання облич [5]. Також за допомогою Blender можна створити датасет реалістичних сцен з корегуємою щільністю натовпу, різнем рівнем освітлення та інших деталей, та використовувати створені моделі для вирішення задачі задачі підрахунку людей у натовпі.

Література

[1] Blender Foundation. Home of the blender project - free and open 3D creation software [Електронний ресурс]. – Режим доступу до ресурсу: https://www.blender.org/

[2] Geometry nodes [Електронний ресурс]. – Режим доступу до ресурсу: https://docs.blender.org/manual/en/latest/modeling/geometry_nodes/index.html

[3] Introduction to Nodes [Електронний ресурс]. – Режим доступу до ресурсу: https://docs.blender.org/manual/en/2.79/render/blender_render/materials/nodes/introduction.html

[4] The Freedom to Create [Електронний ресурс]. – Режим доступу до ресурсу: https://www.blender.org/about/#:~:text=Blender%20is%20the%20free%20and,video%20editing%20and%20game%20creation.

[5] Yakovleva, O., Kovtunenko, A., Liubchenko, V., Honcharenko, V., & Kobylin, O. (2023). Face Detection for Video Surveillance-based Security System (COLINS-2023). In CEUR Workshop Proceedings (Vol. 3403). pp. 69-86.

АВТОМАТИЗАЦІЯ ВИБОРУ ЗАХОДІВ БЕЗПЕКИ В УМОВАХ НЕВИЗНАЧЕНОСТІ

Карабан О.Д.
Керівник: Солодовник Г.В.
E-mail: ganna.solodovnyk@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Проблеми інформаційного захисту в автоматизованих системах організацій та підприємств в різних сферах людської діяльності виникли майже одразу після початку використання електронної обчислювальної техніки для регулярної обробки інформації. Під час застосування засобів автоматизації для обробки інформації основна увага приділяється забезпеченню фізичної цілісності, достовірності (надійності) та доступності інформації.

Тому актуальними є питання розробки та вибору засобів інформаційної безпеки, які забезпечують такий стан інформаційних ресурсів, об’єктів або систем, за якого вони будуть захищені від негативних впливів, які можуть завдати шкоди або інформації або засобам її передачі та обробки. Оскільки сучасні інформаційні ресурси та системи не можуть розглядатися окремо від комплексу факторів забезпечення інформаційної безпеки, тобто загроз для інформаційних ресурсів, різних засобів і заходів захисту та вразливостей в системах захисту інформації, то в загальному сенсі під інформаційну безпеку можна розглядати як набір засобів, заходів та процедур забезпечення захисту інформаційних активів. З метою забезпечення підприємству розвитку слід розробити та впровадити систему управління інформаційною безпекою, яка б передбачала: етап планування (формування переліку інформаційних активів, оцінку ризиків та вибору системи заходів безпеки інформації); етап впровадження відповідної системи заходів безпеки; етап оцінювання ефективності та продуктивності ; етап поліпшення тобто реалізацію превентивних та коригуючих заходів [1].

Метою роботи є автоматизація визначення найкращого варіанту засобів інформаційної безпеки за критеріями ефективності та ризикованості.

В якості математичного інструментарію було використано апарат теорії ймовірності, в якості інструментарію автоматизації – інтерпретована об'єктно-орієнтована мова програмування високого рівня Python.

Мова Python належить до вільно розповсюдженого програмного забезпечення, а отже її інтерпретатор так само як і широка стандартна бібліотека можуть бути отримані з офіційного сайту Python і надалі вільно розповсюджуватися. Цей самий сайт має дистрибутиви та посилання на численні модулі, програми, утиліти, а також додаткову документацію.

Серед чисельних переваг Python є: наявність у стандартній бібліотеці вихідних текстів і бінарних дистрибутивів для всіх основних операційних систем; наявність ефективних структур даних високого рівня; ефективний підхід до об'єктно-орієнтованого програмування; простий синтаксис; динамічна обробка типів,

Можливості інтерпретатора Python можуть бути розширені за допомогою функцій та типів даних, розробленими на C або C++, або на будь-якій іншій мові, яку можна викликати з C [2].

В ході виконання роботи було сформульовано задачу визначення засобів інформаційної безпеки з найбільшою ефективністю та найнижчою ризикованістю. Множина варіантів заходів безпеки складається з двох абстрактних варанів, для кожного з яких було надано значення ефективності та ймовірність досягнення цього значення за песимістичною, стриманою та оптимістичною оцінками. Для розв’язання задачі було сформульовано наступні підзадачі:

визначення середньозваженої за ймовірностями ефективності для обох варіантів заходів інформаційної безпеки на підставі початкових даних;

визначення ризикованості, як середньоквадратичного відхилення ефективності для обох варіантів на підставі початкових даних;

визначення найкращого варіанту заходів інформаційної безпеки шляхом порівняння значень ризикованість та середньозваженої за ймовірностями ефективності для обох варіантів.

На рисунку 1 наведено результати автоматизації розв'язання описаної вище задачі засобами мови Python.

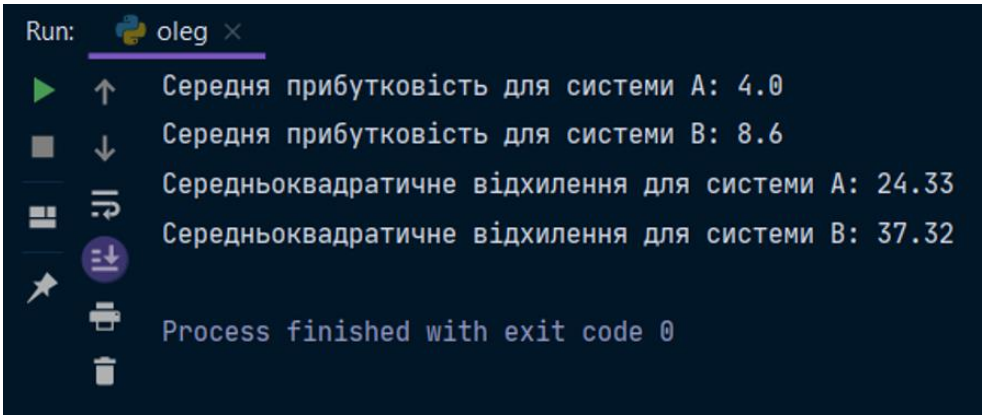


Рисунок 1 – Екрана форма виведення результатів розрахунків

Порівняльний аналіз отриманих даних дозволяє зробити висновок, що за показником середньозваженої за ймовірностями ефективністю кращою є друга система заходів, оскільки вона має більше значення цього показника (8.6 > 4.0). Проте за показником ризикованості кращим є перший варіант, оскільки він має менше значення (24.33 < 37.32). За такого співвідношення значень показників відповідь за задачею буде: залишити остаточний вибір на особою, яка приймає рішення.

Наведена в роботі автоматизація визначення найкращого варіанту засобів інформаційної безпеки за критеріями ефективності та ризикованості може бути вдосконалена за рахунок автоматизації наступних задач:

1 розрахунок коефіцієнта варіації для визначення того чи компенсується підвищений ризик підвищеною ефективністю;

2 виведення остаточної відповіді стосовно кращого варіанту засобів інформаційної безпеки в текстовому вигляді, що покращить користувацький інтерфейс.

Середньоквадратичне відхилення ефективності використовується в якості критерію ризикованості, що не зовсім точно відбиває реальне положення, оскільки виникають ситуації, за яких варіанти забезпечують приблизно однакову ефективність та мають однакові середньоквадратичні відхилення, проте вони не є в однаково ризикованими [3].

Висновок. Автоматизація визначення найкращого варіанту засобів інформаційної безпеки за критеріями ефективності та ризикованості може бути впроваджена на підприємствах, які функціонують в умовах високого рівня конкурентності та стохастичності зовнішнього середовища. Ефект від впровадження полягає у підвищенні наукової обґрунтованості управлінських рішень в сфері захисту інформації.

Література.

[1] Управління інформаційною безпекою: конспект лекцій [Електронний ресурс] : навч. посіб. для студ. спец. 125 «Кібербезпека» / КПІ імені Ігоря Сікорського; уклад.: С. О. Носок, О. М. Фаль, В. М. Ткач. – Електронні текстові дані (1 файл: 1114 Кбайт). – Київ : КПІ імені Ігоря Сікорського, 2021. – 258 с.

[2] Офіційний сайт Python: [Електронний ресурс]. – Режим доступу: https://www.python.org/

[3] Солодовник Г.В. Управління економічним та інформаційним ризиком: навчальний посібник. – Х.: ТОВ «ДІСА ПЛЮС», 2018. – 152 с.

АВТОМАТИЗАЦІЯ ПРИЙНЯТТЯ БАГАТОЕТАПНИХ РІШЕНЬ В УМОВАХ РИЗИКУ

Карабан О.Д.
Керівник: Солодовник Г.В.
E-mail: ganna.solodovnyk@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Актуальність. Кожного дня, як в професійній діяльності так і в пересічному житті, людина стикається з ситуацією прийняття рішення. Така ситуація вникає за наявності наступних елементів: суб'єкта та об'єкта управління; бажання та повноважень у суб'єкта управління щось змінити у стані об'єкта управління; кількох альтернативних рішень та інформації, на підставі якої суб'єкт управління може порівняти альтернативи рішень для вибору найкращої з них. Порівняння альтернатив рішень здійснюється з позиції досягнення певної мети. Слід зауважити, що інформація, на яку спирається суб'єкт управління під час розробки та прийняття рішення є неповною та недостовірною через мінливість умов функціонування об'єкта управління (як зовнішніх так і внутрішніх).

Ситуацію прийняття рішень в умовах ризику формують наступні умови [1]: наявність неповноти даних; необхідність вибору одного з альтернативних рішень (при цьому відмова від вибору також є різновидом вибору); можливість оцінити ймовірність настання результатів альтернативних рішень. Ситуація ризику принципово відрізняється від ситуації невизначеності тим, що у ситуації невизначеності суб'єкту управління невідомі ймовірності настання результатів рішень.

Основними чинниками невизначеності тобто джерелами ризику вважають: спонтанність природних процесів і явищ, а також стихійні лиха; випадковість; наявність протиборчих тенденцій, зіткнення суперечливих інтересів (проявів даного джерела ризику існує досить багато – від міжнаціональних конфліктів до конкуренції і звичайної розбіжності інтересів); неповнота та недостатність інформації про об'єкт, процес, явище, стосовно яких приймається рішення, обмеженість можливостей людини у зборі й переробці інформації, постійна зміна такої інформації [2].

Метою роботи є автоматизація процесу прийняття багатоетапних рішень в умовах ризику та невизначеності із застосуванням методу побудови дерева рішень засобами вільного програмного забезпечення.

На початку роботи сформульовано задачу про прийняття рішення щодо замовлення додаткового обстеження ринку за відомих значень можливого прибутку за настання сприятливого ринку та можливих збитків за настання несприятливого ринку. Також слід ухвалити рішення щодо випуску великої або невеликої партії продукції на підставі значень прибутку та збитків за настання сприятливого та несприятливого станів ринку. За такої постанови задачі прибуток (або збитки як від'ємне значення прибутку) є випадковою величиною, значення якої залежить від ринкового стану, який обирає один з множини своїх станів з певними ймовірностями. Апріорні значення цих ймовірностей можна уточнити за проведення додаткового обстеження ринку, яке коштуватиме певну суму.

Розв'язання задачі вимагає ухвалення низки рішень, причому результати попередніх рішень впливає на процес ухвалення наступних, тобто сформульована задача передбачає прийняття позиційних або багатоетапних рішень в умовах ризику. В практиці прийняття управлінських рішень в цьому випадку використовують метод побудови дерева рішень. На рисунку 1 наведено схему дерева рішень для сформульованої в роботі задачі.

Процес прийняття рішень за допомогою дерева рішень передбачає наступні етапи: формулювання умов задачі, побудова дерева рішень, оцінка ймовірностей станів середовища, встановлення виграшів для кожної можливої комбінації альтернатив (дій) і станів середовища, розв'язання задачі.

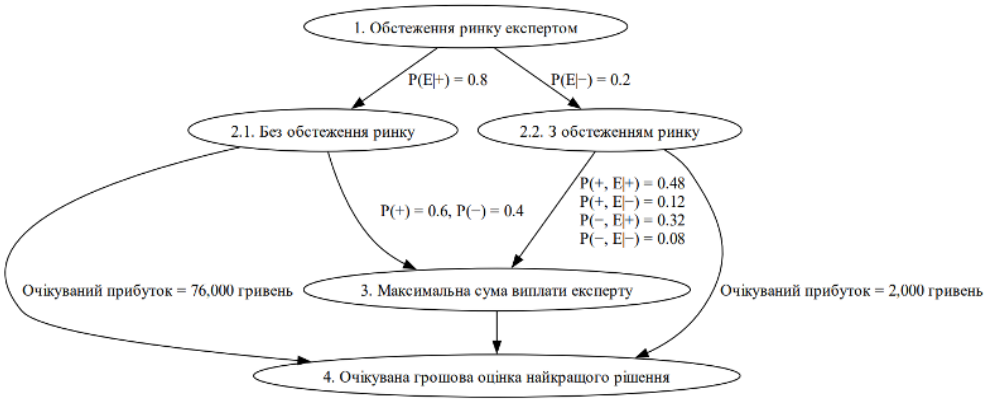


Рисунок 1 – Дерево рішень

На схемі використано наступні позначення: P (+) – ймовірність отримання прибутку за настання сприятливого стану ринку, P(−) – ймовірність виникнення збитків за настання несприятливого стану ринку, P (+, E|+) – ймовірність отримання прибутку за настання сприятливого стану ринку та сприятливого прогнозу експерта, P (+, E|−) – ймовірність отримання прибутку за настання сприятливого стану ринку та несприятливого прогнозу експерта, P (−, E|+) – ймовірність виникнення збитків за настання несприятливого стану ринку але сприятливого прогнозу експерта, P (−, E|−) – ймовірність виникнення збитків за настання несприятливого стану ринку та несприятливого прогнозу експерта.

Очікувана грошова оцінка визначається як сума добутків результатів впровадження рішення (в даній задачі це можливі значення прибутку або збитків) та ймовірностей настання цих результатів (для даної задачі позначено як P()).

За інструментарій автоматизації розв’язання було використаго мову програмування Python, яка належить до вільного програмного забезпечення [3].

Результати розрахунків наведено на рисунку 2.

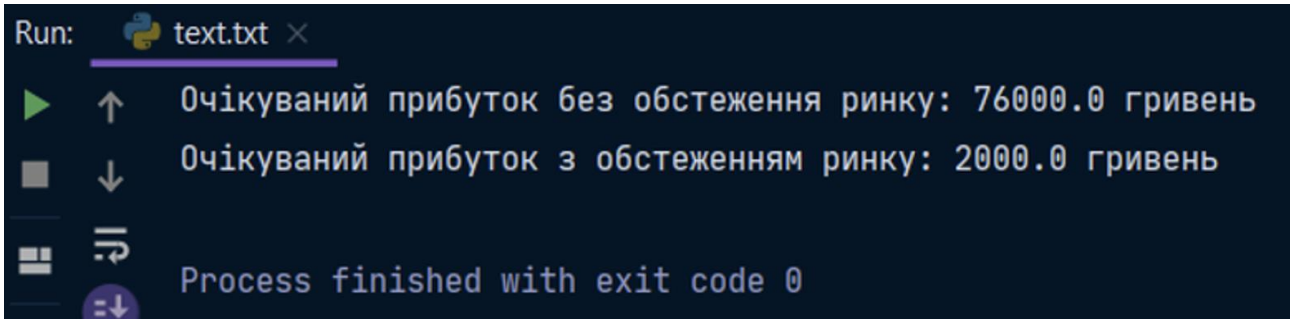


Рисунок 2 – Екрана форма виведення результатів розрахунків очікуваної грошової оцінки альтернативних рішень

Вибір найкращого рішення здійснюється за максимом очікуваної грошової оцінки.

Література.

[1] Решетило В.П., Федотова Ю.В. Невизначеність та ризик: співвідношення понять та специфіка прийняття рішень. Економіка та управління підприємствами. 2020. Випуск №3 (77). С. 149-154.

[2] Сайт ТріумфІТ // Прийняття рішень в умовах повної невизначеності [Електронний ресурс]. – Режим доступу до ресурсу: https://dss.tg.ck.ua/decision-uncertainty-help

[3] Офіційний сайт Python: [Електронний ресурс] / Режим доступу: https://www.python.org/

РОЗРОБКА ПІДСИСТЕМИ ЗАХИСТУ ПЕРСОНАЛЬНИХ ТА ІГРОВИХ ДАНИХ КОРИСТУВАЧА МОБІЛЬНОЇ ГРИ

Кошаренко Д.С.
Керівник: Мерлак О.В.
E-mail: dmitrijkosarenko153@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

У сучасному цифровому світі, коли мобільні ігри стають неодмінною частиною розваг та відпочинку, виникає актуальна проблема захисту персональних та ігрових даних користувачів. Масовий розвиток технологій та поширення мобільних пристроїв спричиняє загрози конфіденційності та цілісності інформації, що потребує компетентного та ефективного рішення.

У сучасному світі, де мобільні ігри стали неодмінною частиною нашого повсякдення, питання безпеки особистих та ігрових даних користувачів набуває надзвичайної актуальності. З постійним розширенням аудиторії гравців, широким розповсюдженням мобільних платформ та зростанням загроз кібербезпеки, необхідно створювати надійні та ефективні механізми захисту, щоб убезпечити особисті дані і забезпечити невідмовну довіру споживачів до ігор на мобільних пристроях[2]. Моя робота з розробки підсистеми захисту персональних та ігрових даних користувача мобільної гри стає ключовою ініціативою, спрямованою на вирішення цієї актуальної проблеми в епоху цифрової трансформації та глобального використання мобільних ігор.

Захист персональних та ігрових даних повинен складатись з декількох не суміжних систем або алгоритмів та повинен передбачати людський фактор та бути розрахованим на будь яку вікову, або цільову аудиторію ігрового мобільного бізнесу.

Робота присвячена розробці підсистеми захисту даних для мобільних ігор, яка враховує особливості сучасних тенденцій у галузі розробки ігор та високих вимог щодо безпеки інформації. У контексті роботи проводиться аналіз існуючих методів захисту даних у мобільних іграх та розглядається їхня вразливість.

Засобами розробки обрано один з найпопулярніших для клієнтської частини застосунку рушій Unity[1]. Застосування Unity дозволить легко інтегрувати розроблену підсистему захисту для забезпечення безпеки ігрових та особистих даних користувачів. Такий підхід дозволяє не лише забезпечити безпеку, але і надати користувачам неперевершений розважальний враження від взаємодії з мобільною грою. Мовою програмування обрано класичну для цього рушія це C#. Зберігання даних користувачів буде проводитись у реляційній базі даних SQLite[3], адже вона легко розгортається, як на клієнтській, так і на серверній частині додатку та має вбудовані рішення криптографічної стійкості даних, що зберігаються.

Проєкт передбачає реалізацію ефективної клієнт-серверної взаємодії для обміну даними між мобільним клієнтом гри та серверною інфраструктурою. Застосовуватимуться передові технології мережевого програмування для забезпечення стабільної та безпечної передачі ігрової інформації, включаючи особисті дані користувачів. Основна увага приділяється визначенню вимог до системи захисту даних, враховуючи потреби та очікування користувачів. Розглядаються технічні та функціональні аспекти розробки, вибір оптимальних методів шифрування, аутентифікації та забезпечення конфіденційності ігрових і особистих даних. Також вивчаються можливості реалізації механізмів взаємодії з ігровим сервером для забезпечення постійного моніторингу та оновлення заходів безпеки, що відповідають змінюючимся вимогам кібербезпеки та динаміці ігрової індустрії.

Література

[1] Unity [Electronic resource]. – Access mode: https://unity.com/ru/products/unity-engine

[2] Digitalconnectmag [Electronic resource]. – Access mode: https://www.digitalconnectmag.com/how-to-protect-your-privacy-while-playing-mobile-games/

[3] Sqlite [Electronic resource]. – Access mode: https://www.sqlite.org/docs.html

ОСОБЛИВОСТІ ОЦІНЮВАННЯ МЕТРИК НАТИВНИХ ANDROID ЗАСТОСУНКІВ

Макарова Л.М., Татаренко М.А.
E-mail: lidiia.makarova@nuos.edu.ua, multi3volume@gmail.com
Миколаїв, Національний університет кораблебудування імені адмірала Макарова

ОС Android є однією із найпопулярніших ОС для мобільних пристроїв [1], не враховуючи інші пристрої (електронні книги, програвачі, спеціалізоване обладнання). Протягом багатьох років був і залишається попит бізнесу різного масштабу на створення мобільних додатків для цієї ОС.

Будь-які мобільні додатки можна розділити на кілька типів: Нативний (Native mobile application), Веб-мобільний (Web Mobile application) і Гібридний (Hybrid Mobile Application), Крос-платформний (Cross-platform mobile applications) [2]. Нативні мобільні додатки розробляються спеціально для однієї платформи (наприклад тільки для ОС Android), мають високу продуктивність і можуть використовувати всі програмні та апаратні можливості платформи та пристрою відповідно. Для ОС Android застосовується стек SDK, який надається компанією Google. Мови програмування, які застосовуються для написання таких додатків: Java і Kotlin.

Актуальність дослідження полягає в оцінюванні розміру та трудовитрат на початкових етапах проектування нативних мобільних додатків для ОС Android з урахуванням типу ПЗ, а також мови програмування, на якій мобільний додаток буде реалізовано. Врахування цих додаткових особливостей при побудові математичної моделі для оцінювання розміру мобільних додатків та трудовитрат на їх створення для ОС Android може підвищити достовірність математичної моделі у кінцевому підсумку. Це актуально як для комерційних продуктів, так і для ПЗ з відкритим вихідним кодом. В дослідженні будуть враховані лише нативні мобільні додатки, для яких основними мовами програмування є Java та Kotlin.

Існують роботи з побудови математичних моделей для оцінювання розміру та трудовитрат мобільних додатків, наприклад, [3 - 5]. В цих роботах враховується мова програмування, фреймворк, але не завжди враховується тип ПЗ або ОС.

Зокрема, робота [4] присвячена побудові нелінійної регресійної моделі з метою оцінювання розміру мобільних додатків типу персональних органайзерів, створених за допомогою мови програмування Kotlin. У цій роботі враховується як тип додатків, а саме додатків-органайзерів, а також мова програмування. Програми були відібрані, що написані виключно мовою програмування Kotlin. Також у роботі акцентується увага на важливості розробки математичних моделей саме під певний тип проектів.

Перевагою роботи можна назвати врахування типу проекту та мови програмування.

Недолік роботи – досить простий набір метрик для побудови моделі. Для кожного проекту було зібрано метрики коду, такі як кількість рядків коду програми (KLOC) та загальна кількість класів (NC). Як наслідок – побудова однофакторної нелінійної регресійної моделі. Крім того автори відзначають, що модель вимагає покращення і можна застосувати, наприклад, інший вид нормалізуючого перетворення або виконати побудову багатофакторної моделі. Проте, після перевірки якості отриманої моделі, результати виявилися непоганими.

У роботі [5] автори виконали побудову трифакторної математичної моделі на основі рівняння нелінійної регресії із застосуванням чотиривимірного перетворення Джонсона сімейства SB . Модель розроблена з метою оцінювання трудомісткості розробки мобільних додатків на етапі планування.

За основу було взято наступні незалежні змінні: фактична трудомісткість розробки, кількість екранів, кількість функцій, кількість файлів.

Також було виконано порівняння побудованої моделі з моделлю лінійної регресії та моделями нелінійної регресії, заснованими на одновимірних нормалізуючих перетвореннях.

Запропонована авторами модель, порівняно з іншими регресійними моделями, має кращу якість (більший множинний коефіцієнт детермінації R2, менше значення середньої

величини відносної помилки MMRE, більше значення відсотка прогнозування PRED(0,25) та менші ширини довірчого інтервалу та інтервалу прогнозування регресії). Але автори не конкретизують для якої сааме мобільної платформи було взято вихідні дані.

Прогнози щодо оцінювання розміру та трудовитрат розробки ПЗ існуючими методами та моделями зазвичай справедливі лише для певного типу проектів. Немає універсальних методів чи моделей, які є оптимальними для всіх типів проектів ПЗ. Тому необхідно вдосконалювати існуючі та/або розробляти нові моделі для оцінювання розміру та трудовитрат при створенні проекту обраного типу.

Будь-який мобільний додаток для ОС Android можна віднести до того чи іншого типу ПЗ. Кожен тип проекту має свої особливості. Тому, якщо модель, що розробляється, їх враховуватиме, це може підвищити її достовірність в кінцевому підсумку. У такому разі перед побудовою моделі, інформація з метрик програмних проектів (мобільних додатків) має бути згрупована за типом проекту.

Крім того, важливою складовою при побудові математичних моделей для оцінювання розміру мобільного додатку є мова, за допомогою якої він створюється. В нашому випадку буде обрано однакову кількість додатків для ОС Android, одного типу, розроблених двома мовами програмування: Java і Kotlin.

Далі наведено етапи робіт:

– знайти мобільні додатки з відкритих джерел, наприклад з: github, gitlab, subversion;

– вибір додатків лише одного типу. Наприклад тільки файлових менеджерів;

– для оцінювання впливу мови програмування має бути дві групи додатків: одна для додатків написаних на Java, друга для додатків написаних на Kotlin;

– перевірка отриманих вибірок на нормальність закону розподілу;

– перевірка незалежних змінних на мультиколінеарність;

– перевірка вибірок на викиди;

– вибір нормалізуючого перетворення;

– побудова нелінійної регресійної моделі;

– перевірка якості побудованої моделі;

– прогнозування розміру та трудовитрат.

Література

[1] Global market share held by mobile operating systems from 2009 to 2023, by quarter [Електронний ресурс]. – Режим доступу до ресурсу: https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/

[2] Native vs. Hybrid vs. Web App: What's the Difference? [Електронний ресурс]. – Режим доступу до ресурсу:https://www.upwork.com/resources/native-hybrid-web-app-differences

[3] Рябков С.І., Приходько С.Б. Удосконалення математичної моделі для оцінювання розміру мультимедійних Android додатків, написаних на Java. Інформаційні технології: моделі, алгоритми, системи: зб. матеріалів І Всеукраїнської науково-практичної інтернет конференції “ITMAS – 2019” (28-29 жовтня 2019 р., м. Миколаїв). Миколаїв: НУК імені адмірала Макарова, 2019. С. 10-12.

[4] Телехан А.М., Макарова Л.М. Нелінійна регресійна модель для оцінювання розміру персональних застосунків-органайзерів, що створюються мовою Kotlin для Андроїд. Інформаційні технології: моделі, алгоритми, системи: зб. матеріалів ІІІ Всеукраїнської науково-практичної інтернет конференції “ITMAS – 2022” (26-28 жовтня 2022 р., м. Миколаїв). Миколаїв: НУК імені адмірала Макарова, 2022. С. 27-29.

[5] Prykhodko S., Prykhodko N., Knyrik K., Pukhalevych A. Mathematical Modeling of Effort of Mobile Application Development in a Planning Phase. International Workshop on Information-Communication Technologies & Embedded Systems. 2019. Vol-2516. [Електронний ресурс]. – Режим доступу до ресурсу: https://ceur-ws.org/Vol-2516/paper7.pdf

PEP8 ТА «CLEAN CODE» В ОЦІНЮВАННІ ЯКОСТІ ЗАСТОСУНКІВ НА PYTHON

Макарова Л.М., Штаба В.Г.
E-mail: lidiia.makarova@nuos.edu.ua, shtaba09@gmail.com
Миколаїв, Національний університет кораблебудування імені адмірала Макарова

Стандартизація певних процесів написання коду ІТ-проектів відіграє дуже важливу роль в оцінювання якості коду, з метою забезпечення його розвитку в майбутньому. Стандартизація дозволить більшій кількості поколіннь розробників вдосконалювати та розвивати код. Розвиток та удосконалленя коду - це одне із найважливіших завдань розробників на довгострокових і масштабних проектах. Загалом для розуміння придатності початкового та напрацьованого коду для проектів, потрібно виконати якісну оцінку цього коду, визначити його зрозумілість та відповідність певним стандартам, та дати оцінку якості і придатності для розвитку, що дозволить в майбутньому більш гнучко та якісно застосовувати бюджет, та виконувати покращення сервісів та продуктів, які використовують програмне забезпечення, що написано на основі цього коду. У цій роботі поставлено завдання визначити фундаментальні стандарти та вимоги для коду мовою Python.

Концепція «Clean Code» - це концепція фундаментального підходу до правил оформлення та написання програмного коду. Ця концепція максимально узагальнена та має певні правила, що підходять до будь-якої високорівневої мови програмування. Дотримання правил «Clean Code» - це ключовий компонент успішної розробки програмного забезпечення. Він покращує продуктивність команди, полегшує процес супроводу і підтримки застосунків, знижує кількість помилок [1, 2]. Тож зрозуміло, що якість коду безпосередньо пов’язана з якістю продукту. Чистий, структурований і зрозумілий код робить застосунок надійнішим, покращує його продуктивність і полегшує внесення змін. У підсумку, він сприяє створенню продукту, який задовольняє потреби користувачів і клієнтів.

Основні засади концепції:

1. Читабельність коду. Код повинен бути написаний так, щоб його було легко читати та розуміти. Це означає використання зрозумілих назв змінних та функцій, а також написання коротких та зрозумілих коментарів.

2. Простота дизайну. Уникнення надмірно складних конструкцій. Часто простіші рішення є надійнішими та легшими для підтримки.

3. DRY (Don't Repeat Yourself). Уникнення дублювання коду. Використання функцій та модулів для уникнення повторення однакових або дуже схожих блоків коду.

4. Тестування. Написання тестів для свого коду, включаючи юніт-тести, щоб забезпечити його надійність та полегшити подальшу розробку та рефакторинг.

5. Рефакторинг. Регулярний перегляд та вдосконалення коду, щоб він залишався чистим та ефективним. Рефакторинг включає в себе виправлення "кодового боргу" (technical debt), оптимізацію та видалення зайвого коду.

6. Відповідність SOLID принципам. Це п'ять принципів об'єктно-орієнтованого програмування, які сприяють створенню більш зрозумілого, гнучкого та підтримуваного коду.

7. Обробка помилок. Написання чіткого та інформативного коду обробки помилок, уникання загальних виключень, де це можливо, і забезпечення того, щоб помилки не залишалися непоміченими.

8. Форматування коду та стандарти. Дотримпння усталених стандартів форматування, таких як PEP8 для Python, щоб код був послідовним і легко читався іншими розробниками.

PEP8 - «Style Guide for Python Code» - є документом, який визначає засади стилю написання комп'ютерного коду мовою Python. Цей стандарт був розроблений Гвідо ван Россумом, Баррі Варсоу і Ніком Когланом і вперше опублікований у 2001 році. Він містить рекомендації щодо того, як форматувати Python код, щоб він був більш читабельним та уніфікованим [3].

Основа гайду полягає в наступних елементарних правилах:

1. Відступи. Використання 4 пробілів на кожен рівень відступу для візуального розмежування блоків коду.

2. Довжина рядка. Обмеження максимальної довжини рядка 79 символами для коду і 72 для коментарів та докстрінгів (блоків документації). Це забезпечує зручність читання коду на різних пристроях та редакторах.

3. Імпорти. Імпорти мають бути на окремих рядках і групуватися в наступному порядку: стандартна бібліотека, сторонні бібліотеки, локальні модулі програми. Крім того, рекомендується використовувати абсолютні імена для імпорту, де це можливо.

4. Пробіли в виразах та інструкціях. Уникання використання великої кількості пробілів.

5. Коментарі. Коментарі, що не відносяться до коду, повинні бути повністю виправдані. Вони повинні пояснювати контекст або причину для того, що робить код, а не описувати очевидні речі.

6. Іменування конвенції. Використовуйте конвенції іменування для класів, функцій, методів, змінних та інших елементів коду. Наприклад, імена класів повинні бути в CamelCase, а імена функцій та змінних - в lowercase_with_underscores.

7. Строки документації (Docstrings). Використання потрійних лапок для рядків документації. Вони повинні пояснювати, що робить функція, метод або клас, і повинні слідувати специфікації Docstring PEP257.

PEP257 – це стандарт розміщення документації (docstrings) в Python коді, котра розміщається на початку модулів, класів, функцій або методів та дає коротке, але зрозуміле представлення про вхідні параметри, повернення значень, функції та можливе застосування методу та іншу інформацію. Дотримання такого стандарту дозволить новим програмістам, що тільки нещодавно прийшли на проект, краще і швидше розібратися з написаним кодом та вдосконалити його.

Виходячи з розгялнутих визначеннь стандартів та концепцій, можна зробити висновок, що застосування розглянутих особливостей стандартів та їхніх правил дозволить вибудувати та обрати актуальні метрики для мови програмування Python. Це дозволить розробити математичну модель для оцінювання якості коду, придатності його до удосконалення та планування розробки нового функціоналу застосунку, що написаний з його використанням.

Головною метою роботи є проведення практичних досліджень реального програмного коду з застосуванням існуючих математичних моделей та застосунків для збирання метрик. Можливо виявиться необхідним написання власних застосунків та програмних інструментів для збору метрик. Важливим аспектом роботи є розробка математичних моделей або адаптація існуючих моделей задля отримання більш достовірних результатів обробки інформації з метрик Android застосунків та їх відповідність стандартам розробки сучасного програмного забезпечення.

Результати досліджень і розробок математичних моделей можуть бути застосовані для планування етапів та розрахунку часу розробки, визначення кошторису та можливих ризиків на етапах розробки Android застосунків мовою програмування Python.

Література

[1] Про поняття “чистий код” у розробці ПЗ [Електронний ресурс]. – Режим доступу до ресурсу: https://foxminded.ua/shcho-take-chystyi-kod/

[2] Mariano Anaya Clean Code in Python: Develop maintainable and efficient code, 2nd Edition 2nd ed. Edition. Packt>, Birmingham - Mumbai, 2021. 422 p. ISBN: 978-1800560215

[3] PEP 8 – Style Guide for Python Code [Електронний ресурс]. – Режим доступу до ресурсу: https://peps.python.org/pep-0008/

ТЕХНОЛОГІЯ РОЗПІЗНАВАННЯ МОВЛЕННЯ ДЛЯ ПОКРАЩЕННЯ КОРИСТУВАЦЬКОГО ДОСВІДУ В ОСВІТНЬОМУ ДОДАТКУ

Пироженко М.Ю.
Керівник: Вишняк М.Ю.
E-mail: mykhailo.pyrozhenko@nure.ua
Харків, Харківський національний університет радіоелектроніки

Розпізнаванням мовлення — це процес, що дозволяє комп'ютеру ідентифікувати та реагувати на звуки людської мови. Завдання розпізнавання мови вирішується вже давно, але якісного рівня вдалося досягти лише в останні роки. Загалом, з появою віртуальних помічників та технологічним розвитком смартфонів, міні-комп'ютерів, інших мобільних пристроїв, користувачам стало зручніше взаємодіяти з технікою за допомогою голосових команд ніж за допомогою традиційних інтерфейсів.

На сьогодні існує два основних підходи до розпізнавання мови: гібридний та end2end. За принципом роботи end2end переводить послідовність звуків у послідовність літер. Особливістю гібридного підходу є окреме використання акустичної та лінгвістичної моделей, які часто працюють незалежно. Не залежно від обраного підходу, застосування технологій розпізнавання мовлення та голосового керування може бути надзвичайно корисна в багатьох сферах діяльності. Однією з інновацій, яка має великий потенціал, є інтеграція таких технологій в освітніх цілях. Метою проекту була розробка освітнього додатка для вивчення іноземної мови.

Для виконання мети, було передбачено створити форк до гри Unlucky[1]. Unlucky — це рольова гра, з відкритим кодом, що заснована на генераторі випадкових чисел з покроковою бойовою системою. В ході роботу нами було перероблено її дизайн, щоб зробити її придатною для викладання іноземної мови. Для цій грі гравці отримали можливість керувати своїми персонажами та взаємодіяти з іншими ігровими об'єктами за допомогою голосових команд. Вивчення забезпечується тестування на пригадування та повторенням з застосуванням методів з когнітивної науки. Хоча голосове керування передбачає виконання дій, що вже могли бути виконані за допомогою миші та клавіатури, така реалізація значно покращила користувацький досвід.

Для виконання роботи з реалізації голосового керування була використана бібліотека Vosk[2]. Важливою перевагою Vosk є невеликий розмір моделі, що дозволяє застосовувати це рішення в прикладних додатках. Vosk базується на поширеній архітектурі DNN-HMM. Глибока нейронна мережа використовується для акустичного скорингу, фреймворки HMM і WFST - для лінгвістичних моделей. Вхідні дані, що надходять із зовнішнього середовища, обробляються та перетворюються на наявність ігрових команд. Обробник команд відповідно оновлює стан компонентів. Оновлення екрану програми відбувається на основі різниці в часі та значеннях компонентів.

Нами досліджено можливості використання голосового управління для покращення користувацького досвіду в освітньому додатку. Розробка голосового управління є значним кроком в ігрових та освітніх технологіях. Хоча робота була обмежена конкретною областю, цей досвід може бути корисним при розробці освітніх додатків з інших предметів, дозволяючи студентам проводити віртуальні експерименти за допомогою голосових команд. Результат дослідження показує, що голосове управління може відігравати більш помітну роль в освітніх додатках.

Литература

[1] Li M. Unlucky [Електронний ресурс] / Ming Li. – 2019. – Режим доступу до ресурсу: https://github.com/mingli1/Unlucky.

[2] Shmyrev N. V. Vosk Speech Recognition Toolkit [Електронний ресурс] / Nickolay V. Shmyrev. – 2020. – Режим доступу до ресурсу: https://github.com/alphacep/vosk-api.

ЗАСТОСУВАННЯ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У ПРОТИДІЇ АТАКАМ СОЦІАЛЬНОЇ ІНЖЕНЕРІЇ

Писар В.О.
Керівник: Солодовник Г.В.
E-mail: ganna.solodovnyk@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Актуальність. В міру того, як технології розвиваються, зростає й кількість кібератак, серед яких атаки соціальної інженерії становлять значну частку. Такі атаки використовують психологічний маніпулятивний вплив для отримання конфіденційної інформації, доступу до систем або введення користувачів в оману. Незважаючи на значні інвестиції в технічні засоби захисту, людський фактор залишається найбільш вразливим компонентом у кібербезпеці. Атаки соціальної інженерії цілеспрямовано використовують цю вразливість, експлуатуючи довіру, страх, невідання або зайнятість жертв.

Особливістю атак соціальної інженерії є складність їх ідентифікації традиційними інструментами кібербезпеки, оскільки вони не завжди включають використання шкідливого коду або вразливостей в програмному забезпеченні. Тому дослідження в цей сфері та обізнаність користувачів з ключовими аспектами протидії є актуальними питаннями на сьогодня.

Наслідки атак соціальної інженерії можуть бути руйнівними, включаючи фінансові втрати, втрату даних, пошкодження репутації та навіть фізичну шкоду, ці факти підкреслюють необхідність активних заходів щодо протидії та обізнаності.

Метою роботи є аналіз можливостей безплатного програмного забезпечення та програмного забезпечення з відкритим кодом у формуванні заходів протидії атакам з боку соціальної інженерії.

Через те, що здебільшого методи соціальної інженерії не потребують наявність спеціальних технічних знань у зловмисників, використовувати ці методи може будь-хто як дрібні злодії, так і досвідчені кіберзлочинці. Найвідомішими методиками соціальної інженерії є спам та фішинг. Сьогодні все більше компаній усвідомлюють, що вони є потенційними цілями кіберзлочинців: 44% компаній з числельністю працівників 251-500 з стикалися з випадками втрати даних протягом 2020 року; 88% представників малого бізнесу є потенційною ціллю кіберзлочинців.

Застосування вільного програмного забезпечення у протидії атакам соціальної інженерії може бути реалізоване через різноманітні інструменти та платформи. Прикладом такого застосування є Gophish – відкрите програмне забезпечення, призначене для проведення тренінгів з фішингу, яке дозволяє організаціям створювати симуляції фішингових атак для навчання співробітників розпізнавати такі спроби та правильно на них реагувати. Іншим прикладом може слугувати комплексний набір інструментів для моніторингу мережі та безпеки Security Onion, який дозволяє виявляти та аналізувати підозрілі активності. Security Onion включає функціонал для глибокого аналізу трафіку та журналів, що може допомогти ідентифікувати спроби соціальної інженерії. Ще однією можливістю застосування вільного програмного забезпечення для протидії атакам соціальної інженерії є Wireshark – мережевий аналізатор, який дозволяє здійснювати глибокий аналіз мережевого трафіку в реальному часі або з записаних сесій. Використання Wireshark може допомогти виявити неавторизовані спроби доступу до мережі, які можуть бути частиною атаки соціальної інженерії.

Висновки. Регулярне застосування вільного програмного забезпечення для навчання з кібербезпеки та протидії атакам з боку соціальної інженерії дозволяє значно посилити захищеність інформаційних систем. Заходи з підвищення обізнаності в сфері кібербезпеки повинні демонструвати та моделювати ситуації з реального життя, оскільки методи соціальної інженерії, як правило розраховані на користувачів з незначним досвідом та низьким рівнем обізнаності.

АНАЛІЗ БЕЗПЕКИ ТЕХНОЛОГІЇ VOIP

Рибальченко Д. А.
Керівник: Лимаренко В.В.
E-mail: danil333rybalchenk@gmail.com
Харків. Харківський національний економічний університет імені Семена Кузнеця

VoIP (англ. voice over IP — голос через IP) — технологія передачі медіа-даних у реальному часі за допомогою сімейства протоколів TCP/IP. IP-телефонія — система зв'язку, в якій аналоговий звуковий сигнал абонента дискретизується (кодується в цифрову форму), компресується й пересилається цифровими каналами зв'язку до іншого абонента, де проводиться зворотня операція — декомпресія, декодування й відтворення аналогового сигналу.[1]

VoIP був розроблений з урахуванням основних пріоритетів надійності, сумісності та якості обслуговування, і тому турбота про безпеку, як правило, була другорядною по відношенню до них у ранніх впровадженнях VoIP. На жаль, VoIP страждає від низки загальних проблем безпеки, зокрема:

1) Передача через IP/Інтернет – оскільки VoIP використовує ту саму інфраструктуру, що й служба передачі даних, VoIP страждає від основних проблем безпеки даних, а також проблем, властивих лише VoIP.

2) VoIP не має стандартизованого протоколу для надсилання та отримання інформації. Існують різні протоколи (наприклад, SIP і H.323), хоча багато пристроїв підтримують більше одного. Це збільшує ймовірність зловмисного використання погано написаних програм/реалізацій.

3) Безпека може знизити якість обслуговування (QoS) – заходи безпеки можуть збільшити дані, що передаються під час сеансу VoIP, таким чином збільшуючи ризик зниження якості обслуговування через перевантаження мережі.

Обсяг безпеки VOIP ґрунтується на трьох основоположних поняттях:

1) Конфіденційність – забезпечення захисту конфіденційних даних від сторонніх вух і забезпечення конфіденційності розмов.

2) Цілісність – виявлення того, чи була інформація змінена (зловмисно чи випадково), і оцінка того, чи можна довіряти даним голосового повідомлення та вважати їх автентичними.

3) Доступність – забезпечення надійності та своєчасного доступу до голосових даних і ресурсів.[2]

Шифрування VoIP перетворює аудіодані в закодовану форму, що робить їх нерозбірливими для неавторизованих користувачів. Це критично важливо для конфіденційності, цілісності даних і дотримання нормативних вимог.

Ось деякі ключові технології, які використовуються для захисту VoIP:

1) Безпека транспортного рівня TLS (Transport Layer Security)— це протокол, який забезпечує конфіденційність між програмами, що спілкуються, і користувачами Інтернету. Він шифрує пакети під час фази сигналізації процесу зв’язку, гарантуючи, що інформація про встановлення виклику залишається в безпеці.

2) Захищений транспортний протокол реального часу SRTP (Secure Real-Time Transport Protocol) — на відміну від TLS, SRTP розроблений спеціально для зв’язку за протоколом реального часу (RTP), який включає VoIP і відеоконференції. SRTP шифрує голосові дані, захищаючи дзвінок від прослуховування та перехоплення.

3) Прикордонні контролери сеансів (SBC) — SBC додають додатковий рівень безпеки для мереж VoIP, контролюючи сигналізацію, залучену до здійснення телефонних дзвінків.

4) Віртуальні приватні мережі (VPN) — VPN створюють приватну мережу з загальнодоступного підключення до Інтернету. Маршрутизуючи трафік VoIP через VPN, компанії додають додатковий рівень безпеки та шифрування, що ускладнює перехоплення дзвінків хакерам.[3]

Література

[1] Вікіпедія. VoIP [Електроний ресурс] — Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/VoIP

[2] VoIP Security/[Секретаріат Азіатсько-Тихоокеанського економічного співробітництва]. — **APEC Secretariat**, Сінгапур : 2008.

[3] The Daily Egg. How To Make VoIP Security and Encryption Issues Disappear [Електроний ресурс] — Режим доступу до ресурсу: https://www.crazyegg.com/blog/voip-security-and-encryption/

ОГЛЯД ДАТАСЕТІВ ДЛЯ АНАЛІЗУ МЕРЕЖЕВОГО ТРАФІКУ

Рихва В.І.
Керівник: Солодовник Г.В.
E-mail: ascentman91@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Завдання машинного навчання полягає в його здатності отримувати потрібну інформацію з даних, саме тому продуктивність машинного навчання напряму залежить від якісних вхідних даних. Для систем виявлення вторгнень IDS (Intrusion Detection Systems) вибрані дані мають бути легкими для отримання та відображати поведінку хостів або мереж. Основні типи вхідних даних для IDS - це пакети, потоки, сеанси, log-файли. Створення датасетів для цих систем є складним і трудомістким процесом. Однак, коли такий набір даних створено, він приносить користь численним дослідникам у різних дослідженнях. По-перше, контрольні набори даних визнаються надійними джерелами, що надає більшої обґрунтованості експериментальним результатам. По-друге, широке впровадження цих наборів даних полегшує порівняння результатів нових досліджень із результатами попередніх досліджень, збагачуючи колективне розуміння та прогрес у цій галузі.

DARPA1998 [1] є одним із перших і найвідоміших датасетів, створених для дослідження в області виявлення вторгнень IDS. Він був розроблений в рамках проекту DARPA (Defense Advanced Research Projects Agency) в 1998 році лабораторією Лінкольна Массачусетського технологічного інституту і мав на меті надати дослідникам дані, що імітують реальний трафік мережі, змішаний з різноманітними атаками, щоб тестувати та вдосконалювати системи IDS. Щоб скласти його, дослідники збирали Інтернет-трафік протягом дев’яти тижнів; перші сім тижнів складають навчальний набір, а останні два тижні складають тестовий набір. Набір даних містить необроблені пакети та мітки. Є п’ять типів міток: звичайний трафік, denial of service (DOS), Probe, User to Root (U2R) і Remote to Local (R2L). Оскільки необроблені пакети не можна безпосередньо застосувати до традиційних моделей машинного навчання, набір даних KDD99 було створено, щоб подолати цей недолік.

Набір даних KDD99 [2] є найпоширенішим набором даних IDS на даний час. Його було скомпільовано з даних DARPA1998 і він має такі ж самі мітки, як у DARPA1998. В KDD99 є чотири типи функціональних особливостей, тобто основних функцій: функції вмісту, статистичні функції на основі хоста та статистичні функції на основі часу. На жаль, набір даних KDD99 містить багато дефектів. По-перше, дані сильно незбалансовані, що робить результати класифікації неточними. Крім того, існує багато дублікатів записів і існують зайві записи. Багатьом дослідникам доводиться ретельно фільтрувати набір даних, перш ніж вони зможуть його використовувати. У результаті експериментальні результати різних дослідників важко порівнювати. Ще одним важливим недоліком є неможливість використання цього датасету в сучасних умовах через застарілість типів атак.

NSL-KDD [3] був створений, щоб усунути недоліки у KDD99. Записи різних класів збалансовані в NSL-KDD, що дозволяє уникнути проблеми зміщення класифікації. NSL-KDD також видалив дублікати та зайві записи. Таким чином, на основі повного датасету можливе проведення експериментів, що забезпечує узгодженість і можливість порівнювати

результати, отримані з різних джерел. Датасет NSL-KDD в деякій мірі вирішує проблеми, пов'язані зі зміщенням класифікації. Проте, NSL-KDD не включає нові дані, через що продовжує бути нерелевантним з погляду сучасних загроз.

Набір даних UNSW-NB15 [4] був складений університетом південного Уельсу (University of South Wales) у 2015 році, в якому дослідники налаштували три віртуальні сервери для захоплення мережевого трафіку та витягнули 49 характеристик за допомогою інструменту під назвою Bro. Набір даних містить більше типів атак, ніж набір даних KDD99 (9 видів атак), а також має більше функцій: функції потоку, основні функції, функції вмісту, функції часу та функції з мітками. UNSW-NB15 є новішим, ніж KDD99, проте його поширеність значно менша.

Набір даних CICIDS2017 [5] підготовлений Канадським інститутом кібербезпеки за результатами мережевого аналізу трафіку в ізольованому середовищі, в якому моделювалися дії 25 легальних користувачів, а також шкідливі дії порушників. Набір об'єднує понад 50Гб «сирих» даних у форматі PCAP і включає 8 попередньо оброблених файлів у форматі CSV, що містять розмічені сесії з виділеними ознаками в різні дні спостереження. Період збору даних – 5 днів: з понеділка 3 липня 2017 року до п’ятниці 7 липня 2017 р. Понеділок включає лише нормальний трафік. В інші дні здійснювалися такі атаки: Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Infiltration, Botnet and DDoS. Основними перевагами датасету є його актуальність та різноманіття сценаріїв атак, що робить його цінним ресурсом для розробки та тестування IDS. Проте, серед недоліків можна відзначити великий обсяг даних, що вимагає значних обчислювальних ресурсів для обробки та потенційну недостатність представлення деяких специфічних типів атак. Ще більше симульованих атак, а саме DDoS+PortScan, Web attacks (DVWA, XSS), Botnet attacks, включають наступні датасети від Канадського інституту кібербезпеки: CICIDS2018 і CSE-CIC-IDS2018 [6].

Датасет CIDDS-001, створений Кобурзьким університетом прикладних наук [7], має на меті вдосконалення виявлення вторгнень у хмарних сервісах, використовуючи для цього симуляцію реального трафіку хмарних сервісів та різноманітні атаки. Він включає в себе різноманітні типи атак, такі як сканування портів, DoS, підбір паролів та інші зловмисні дії, що забезпечує реалістичне тестове середовище для алгоритмів виявлення вторгнень. Основними перевагами датасету є висока репрезентативність реального мережевого трафіку та широкий спектр симульованих атак, що робить його корисним для розробки та тестування IDS. Робота над проектом ще продовжується, щоб генерувати більш реалістичний мережевий трафік, а також використовувати складніші сценарії атак (експлойти браузера та трояни).

Література

[1] DARPA1998 Dataset. 1998 [Електронний ресурс]. – Режим доступу до ресурсу: http://www.ll.mit.edu/r-d/datasets/1998-darpa-intrusion-detection-evaluation-dataset.

[2] KDD99 Dataset. 1999 [Електронний ресурс]. – Режим доступу до ресурсу: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[3] NSL-KDD99 Dataset. 2009 [Електронний ресурс]. – Режим доступу до ресурсу: https://www.unb.ca/cic/datasets/nsl.html

[4] Moustafa, N.; Slay, J. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In Proceedings of the 2015 Military Communications And Information Systems Conference (MilCIS), Canberra, Australia, 10–12 November 2015; pp.1–6. [Електронний ресурс]. – Режим доступу до ресурсу: https://ieeexplore.ieee.org/abstract/document/7348942

[5] Intrusion Detection Evaluation Dataset (CICIDS2017) [Електронний ресурс]. – Режим доступу до ресурсу: https://www.unb.ca/cic/datasets/ids-2017.html

[6] CSE-CIC-IDS2018 on AWS [Електронний ресурс]. – Режим доступу до ресурсу: https://www.unb.ca/cic/datasets/ids-2018.html

[7] The CIDDS concept [Електронний ресурс]. – Режим доступу до ресурсу: https://www.hs-coburg.de/forschung/forschungsprojekte-oeffentlich/informationstechnologie/cidds-coburg-intrusion-detection-data-sets.html

НАЙБІЛЬШ ПОПУЛЯРНІ ВРАЗЛИВОСТІ МЕРЕЖЕВИХ ПРОТОКОЛІВ У КОРПОРАТИВНІЙ МЕРЕЖІ

Романов Д.В.
Керівник: Муржа Д.Ю.
E-mail: danil200380067@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Мережеві протоколи представляють собою комплекс норм і правил, які організовують та регулюють обмін інформацією, використовуючи безпечні, надійні та прості методи. Ці норми і правила здійснюються для різноманітних застосувань. Наприклад, провідні мережі (такі як Ethernet), бездротові мережі (наприклад, WLAN) та інтернет-комунікації використовують відомі приклади протоколів. Величезний набір інтернет-протоколів, який використовується для передачі даних через мережу, включає десятки конкретних протоколів.

Протокол адресації дозволу (ARP), який належить до канального рівня, використовується для визначення MAC-адреси за IP-адресою. В умовах однорангової мережі хост не може визначити джерело мережевого пакету, що створює вразливість і сприяє підміні ARP. Зловмисник може використовувати цю вразливість, знаходячись в тій самій локальній мережі, що й цільовий об'єкт, або використовуючи скомпрометовану машину в одній мережі.

Система доменних імен (DNS) вирішує проблему числового формату IP-адрес, роблячи їх більш зрозумілими для людини за допомогою ієрархічної системи. Однак DNS також має вразливість у вигляді отруєння кешу, коли зловмисник підмінює легітимну IP-адресу, спрямовуючи аудиторію на шкідливі ресурси. Посилення DNS може також використовуватися на DNS-сервері для рекурсивного пошуку, підвищуючи масштаб атак.

Протокол передавання файлів/безпечний (FTP/S) є мережевим протоколом для обміну файлами між клієнтом і сервером. Атаки на FTP часто використовують міжсайтовий скриптинг, де зловмисник впроваджує шкідливий код у вигляді скрипта на стороні браузера або cookies. Віддалений протокол передачі файлів (FTP) не контролює з'єднання та не шифрує дані.

Протокол доступу до інтернет-повідомлень (IMAP) - це протокол електронної пошти, що дозволяє зберігати повідомлення на сервері і одночасно надає користувачеві можливість управління ними. Протокол може стикатися з загрозами, такими як перехоплення інформації на незахищених каналах зв'язку та атаки "відмова в обслуговуванні" на поштовому сервері.

Протокол ініціювання сеансів (SIP) є сигнальним протоколом для реального часу, який використовується для управління комунікаційними сеансами. Цей протокол може бути підданий різним загрозам, таким як переповнення буфера, ін'єкційні атаки та флуд-атаки, що можуть легко виправлятися з мінімальними витратами для зловмисника.

Отже, використання протоколів у корпоративних мережах вимагає ключового уваги до захисту. Зазначені протоколи виявляють численні вразливості, які можуть бути активно використані для порушення безпеки мережі.

Література:

[1] Типи мережевих протоколів і їх призначення [Электроний ресурс]. – Режим доступа до ресурсу: https://deltahost.ua/ua/tipi-merezhevix-protokoliv-i-ih-priznachennya-http-ip-ssh-ftp-pop3-mac

[2] Мережеві протоколи [Электроний ресурс]. – Режим доступа до ресурсу: https://ukrayinska.libretexts.org\

ПРОГРАМНА РЕАЛІЗАЦІЯ СУБ'ЄКТИВНИХ МЕТОДІВ ОЦІНЮВАННЯ РИЗИКІВ

Стеценко М.Т.
Керівник: Солодовник Г.В.
E-mail: ganna.solodovnyk@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Актуальність. Динамічність та невизначеність умов функціонування та поведінки об'єктів управління, а також відкладеність у часі реалізації управлінських рішень призводить до необхідності оцінювання ризиків та вираховування невизначеності у процесі підготовки та ухвалення будь-яких рішень, пов'язаних з роботою та розвитком соціально-економічних систем. Вираховування ризику та неповноти інформації передбачає обробку великих обсягів даних, тому потребує впровадження засобів автоматизації в управлінський процес. Наведені факти визначають актуальність розробки програмної реалізації оцінювання ризиків за допомогою суб'єктивних методів.

Метою роботи є створення програмної реалізації оцінювання ризиків за допомогою суб'єктивних методів засобами вільного програмного забезпечення.

Вибір методу оцінювання ризиків залежить від форми загального оцінювання та його результату, які мають бути узгоджені з критеріями ризику, сформованими під час аналізу умов функціонування об'єкта управління. Обраний метод повинен відповідати наступним вимогам: бути обґрунтованим та доречним у наявній ситуації для конкретного об'єкта управління; забезпечувати отримання результатів у формі, яка найкращим чином сприяє розумінню характеру ризику та способу його врахування; надавати можливість моніторингу, відтворення та перевірки оцінювання [1]. Використання суб'єктивних методів оцінювання ризиків доцільно у викладках, коли відсутність достовірної інформації унеможливлює використання економіко-математичних методів. Суб'єктивні методи базуються на використанні особистісних якостей певних людей. В результаті порівняльного аналізу методів було обрано метод Делфі. Цей метод забезпечує найкраще поєднання врахування думок всіх експертів, які беруть участь в оцінюванні ризику та зниження впливу авторитетів на учасників опитування.

Програмна реалізація оцінювання ризиків методом Делфі створена мовою Python , яка є високо адаптованою та широко використовується в різних середовищах. Python дозволяє легко розробляти та підтримувати проекти різного рівня складності. До переваг Python належать: гнучкість, швидкий розвиток, масштабованість та продуктивність. Python – багатоплатформена, інтерактивна, інтерпретована мова програмування, яка працює майже на всіх відомих операційних системах (Linux/UNIX, Window, Macintosh, Solaris, macOS, iPhone OS, Palm OS, Windows Mobile, Symbian, Android) Дозволяє взаємодіяти з інтерпретатором в режимі реального часу та не вимагає компіляцій для виконання коду. Python є п'ятою серед найпоширеніших мов програмування та використовується NASA [2, 3].

Література.

[1] Еколого-економічний ризик-менеджмент: методи оцінювання ризиків : [Електронний ресурс] : навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки та інформаційні технології», спеціалізації «Інформаційні технології моніторингу довкілля» / Н. В. Караєва. Київ: КПІ імені Ігоря Сікорського, 2019. [Електронний ресурс] / Режим доступу: https://apeps.kpi.ua/downloads/%D0%9A%D0%B0%D1%80%D0%B0%D1%94%D0%B2%D0%B0_%D0%B5%D0%BA%D0%BE%D0%BB%D0%BE%D0%B3_%D0%B5%D0%BA%D0%BE%D0%BD%D0%BE%D0%BC_%D1%80%D0%B8%D0%B7%D0%B8%D0%BA.pdf

[2] Офіційний сайт Python // Python 3.12.1 documentation: [Електронний ресурс] / Режим доступу: https://docs.python.org/3/

[3] Сайт NewITSchool // Цікаві факти про Python [Електронний ресурс]. – Режим доступу до ресурсу: http://www.itschool.vn.ua/interesting-python/

ДОСЛІДЖЕННЯ МЕТОДІВ ЗБЕРІГАННЯ ПАРОЛІВ І НАДАННЯ РЕКОМЕНДАЦІЙ ДЛЯ ПІДВИЩЕННЯ БЕЗПЕКИ ОСОБИСТИХ ДАНИХ

Тищенко А.А.
Керівник: Муржа Д.Ю.
E-mail: skulbuster03@gmail.com
Харків, Харківський національний економічний університет імені Семена Кузнеця

Аналіз і порівняння різних методів зберігання паролів у системах інформаційної безпеки є важливою складовою в розробці стратегій забезпечення безпеки даних користувачів. Особливо у контексті загроз кібербезпеки, які стають все більш розповсюдженими і складними. Різноманітні методи зберігання паролів можуть включати хешування, шифрування, використання солей, а також варіанти, що базуються на біометричних даних або токенах.

Хешування паролів є одним із найпоширеніших методів зберігання, де пароль перетворюється на випадковий рядок фіксованої довжини за допомогою хеш-функції. Цей процес є необоротнім, тобто неможливо відновити вихідне значення пароля з хешу, що робить його відносно безпечним для зберігання. Однак, атаки методом перебору можуть бути успішними, якщо використовуються слабкі паролі або неефективні алгоритми хешування.

Шифрування паролів зазвичай полягає в застосуванні алгоритму шифрування до пароля перед збереженням його у базі даних. Цей метод передбачає наявність ключа шифрування, який необхідно захистити від несанкціонованого доступу. Шифрування може забезпечити високий рівень безпеки, але воно також може бути вразливим до атак, якщо ключі шифрування не захищені належним чином або якщо використовуються слабкі алгоритми шифрування.

Використання солей полягає в додаванні унікального, пароля перед хешуванням. Це ускладнює атаки методом рейнджування або використання рейнбоу-таблиць, оскільки навіть однакові паролі будуть мати різні хеш-значення через використання різних солей. Використання солей сприяє безпеці, особливо коли паролі користувачів витікають через атаку на базу даних, оскільки навіть однакові паролі не можуть бути віднайдені шляхом зіставлення хешів.

Біометричні методи зберігання паролів використовують біометричні дані, такі як відбитки пальців, розпізнавання обличчя або голосу, для ідентифікації користувачів. Ці дані можуть бути використані для автентифікації безпосередньо, або для відновлення втраченого пароля. Цей підхід може забезпечити високий рівень безпеки, оскільки біометричні дані важко підробити, але він може бути вразливим до атак, що використовують підроблені біометричні дані або методи соціальної інженерії.

Токени є іншим методом зберігання паролів, де автентифікація користувача здійснюється за допомогою унікального токену або ключа, який генерується за допомогою спеціального пристрою або програмного забезпечення. Токени можуть бути фізичними пристроями, такими як карти доступу або ключі USB, або можуть бути вбудовані в мобільні додатки або програмне забезпечення. Використання токенів може забезпечити додатковий шар безпеки, оскільки навіть якщо пароль скомпрометований, необхідно також мати доступ до токену для автентифікації.

В результаті, аналіз і порівняння різних методів зберігання паролів у системах інформаційної безпеки дозволяє виявити їх переваги та недоліки з точки зору безпеки даних користувачів. Розуміння цих методів дозволяє розробити оптимальні стратегії зберігання та управління паролями з метою максимального захисту особистих даних користувачів від кіберзагроз.

ПРОФІЛЮВАННЯ СТУДЕНТІВ: ПРОЦЕС, ЗНАЧЕННЯ І ЗНАЧИМІСТЬ ДЛЯ КАР'ЄРНОГО ЗРОСТАННЯ

Ткаченко О.М.
E-mail: tkachenko.ck@gmail.com
Ужгород, Ужгородський національний університет

Сучасний світ надзвичайно швидко змінюється, впливаючи на всі сфери життя, в тому числі і на систему освіти. З'являється все більше професій, які раніше не існували, а деякі традиційні сфери діяльності стають менш вимогливими. Це акцентує важливість належної підготовки молоді до вибору свого майбутнього шляху.Важливість даної теми полягає в тому, як оптимально налаштувати освітній процес, щоб він максимально відповідав потребам студента та сприяв його професійному та особистісному зростанню, використовуючи веб-додаток, розроблений компанією EPAM Systems – «Кар'єрний тест» (https://training.epam.ua/ua/career-test/guidance).

Тест для визначення кар'єрних перспектив представляє собою точний, оперативний та науково обґрунтований інструмент для оцінювання ключових характеристик вашої персональності та визначення найбільш відповідного технічного напряму кар'єри, заснований на отриманих результатах. Основу цього тесту покладено на багаточисленні психологічні та нейробіологічні наукові розробки, включаючи теорію домінуючих тенденцій [1] та дослідження стосовно вроджених та набутих особливостей структури та роботи мозку [2] та використовується класичний MMPI тест (550 запитань), адаптований та скомпонований у такий спосіб, що робить його коротшим (75 пар запитань із орієнтовним часом проходження до 15 хвилин), зрозумілішим і дає змогу отримати результати одразу після проходження. Для складання профілю особистості алгоритм тесту використовує 8 шкал психометрії: тривожність (уважність, обережність, виваженість рішень, контроль), спонтанність (гнучкість, рухливість, адаптивність, самостійність, незалежність, нестандартність мислення, дивергентні рішення), агресивність (самореалізація, завзятість, наявність стійких принципів та інтересів; практичне сприйняття дійсності, схильність до активної діяльності), ригідність (прагнення до відстоювання своїх інтересів і принципів, критичне ставлення до позиції оточуючих; завзятість; схильність до точних наук, технічних видів діяльності), сензитивність (уважність, спостережливість, емпатичність, здатність слухати, чути і розуміти інших людей, уміння пояснювати і прогнозувати думки, відчуття і поведінку іншої людини або соціальних груп людей), екстраверсія (людина пізнає зовнішній світ через взаємодію із зовнішніми джерелами, комунікація та соціалізація заряджають енергією), інтроверсія (людина пізнає зовнішній світ через внутрішнє осмислення, заряджається енергією на самоті).

Особливість тесту в тому, що він не обирає за людину професію, а лише пропонує тестуючому різні професії і зображає, на який відсоток вона їй підійде. Ці рекомендації не звужують вибір, а скоріше розкривають набір галузей, у якій людина має переваги завдяки своїм провідним тенденціям та особистісним особливостям, оскільки більшість сьогоднішніх IT-професій знаходиться на стику дисциплін. Це полегшує вхід у професію та знижує шанси розчаруватися у виборі. Використання даного інструменту для профілювання кар'єрного шляху має позитивний вплив на розуміння та взаємодію зі студентами в освітніх закладах. Воно дозволяє краще розуміти індивідуальні особливості студентів, їхні потреби та інтереси, сприяє прийняттю обґрунтованих рішень щодо кар'єри, а також подальшому їхньому професійному розвитку і зростанню.

Література

[1] Ackoff R. L. On Purposeful Systems: An Interdisciplinary Analysis of Individual and Social Behavior as a System of Purposeful Events / R. L. Ackoff, F. E. Emery. – Routledge, 2005.

[2] Schutz, W.C. (1958). FIRO: A Three-Dimensional Theory of Interpersonal Behavior. New York, NY: Holt, Rinehart, & Winston.

ДОСЛІДЖЕННЯ ІНФОРМАЦІЙНИХ РИЗИКІВ ЗАСОБАМИ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Фатєєв О.Д.
Керівник: Солодовник Г.В.
E-mail: ganna.solodovnyk@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Актуальність. Насьогодні масштаби інформаційних ресурсів і даних, якими доводиться оперувати сучасним організаціям та підприємствам для успішного функціонування є вражаючими. Більша кількість цих даних є критично важливою для виживання цих підприємств. Тому все більшої актуальності набувають питання захисту даних. Однією з ключових складових сучасних систем інформаційної безпеки є системи управління інформаційною безпекою.

Фундаментальним принципами захисту інформації є збереження цілісності, конфіденційності та доступності даних. Разом з цим захисту потребують усі авторизовані користувачі. Сучасні інформаційні системи є вразливими до таких загроз, як несанкціонований доступ та втручання. Захист інформаційних ресурсів та сервісів від можливих загроз потребує впровадження комплексу заходів з управління ризиками та безпеки.

Як правило об’єктом аналізу інформаційних ризиків та вирішення завдань інформаційної безпеки є компоненти (засоби та інформаційні технології) автоматизованих інформаційних систем, автоматизованих систем управління, телекомунікаційних систем, а також інформаційні ресурси, які накопичуються та обробляються такими системами. З іншого боку об’єктом управління інформаційними ризиками є певна установа.

Метою роботи є огляд можливостей, які надає вільного програмного забезпечення для аналізу риків з боку соціальної інженерії.

Найчастіше несанкціонований доступ до інформаційних систем та конфіденційної інформації, трапляється через людський фактор та атаки соціальної інженерії. Основні несанкціоновані втручання виконуються за допомогою фішингових листів, або посилань.

Фішинг як тип кібератаки, призначений для виманювання у неуважних користувачів мережі персональних даних, є досить розповсюдженим в наш час. Проявами фішингу можуть бути як електронні листи від вашої компанії із проханням підтвердити особисту або конфіденційну інформації, так і листи, що інформують про «напружену» ситуацію, щоб прибрати з голови жертви усі роздуми та ввести у паніку. З метою запобігання злому у компаніях, керівництву потрібно подбати про тренінги з кібербезпеки та попередження атак соціальної інженерії для своїх співробітників та мати у штаті не тільки системного адміністратора, але й фахівця у галузі кібербезпеки.

Використання засобів вільного програмного забезпечення підвищує рівень безпеки у інформаційному просторі. Ці засоби є цілком безкоштовними та ліцензійними, а тому не сприяють додатковим загрозами для інформації. Прикладом вільних програмних інструментів, за допомогою яких можна досліджувати інформаційні ризики та попереджати про фішингові атаки є OWASP ZAP (Zed Attack Proxy). Призначенням OWASP ZAP є виявлення та усунення вразливостей веб-додатків шляхом активного та пасивного сканування та інтерактивного аудиту безпеки веб-додатків. Іншим прикладом є ClamAV – це антивірус з відкритим кодом, який призначений для виявлення різноманітних загроз, включаючи фішингові електронні листи та шкідливе програмне забезпечення: віруси, трояни та інши загрози. За допомогою PhishTank – відкритої бази фішингових атак можна перевірити веб-сайт на належність до списку фішингових. Вільний безкоштовний браузер з відкритим кодом Mozila Firefox з розширенням PhishGuard допомагає виявляти фішингові сайти, перевіряти URL на фішингові атаки та блокування доступу до підозрілих сайтів. Аналізатор мережевого трафіку Wireshark використовується для виявлення підозрілих мережевих з’єднань.

СЕРВІС БРОНЮВАННЯ ГОТЕЛІВ НА ОСНОВІ ТЕХНОЛОГІЙ REACT ТА MONGODB

Хоменко В.М.
Керівник: Журавська І.М.
E-mail: youngodforcsq1@gmail.com
Миколаїв, Чорноморський національний університет імені Петра Могили

У наш час використання інтернет-технологій та смартфонів стало звичайною практикою, сприяючи росту запиту на різноманітні онлайн-сервіси. Зокрема, це стосується онлайн-резервування готелів, яке вже стало ключовим елементом сучасних подорожей. Ці інтернет-платформи перетворились на важливий інструмент для планування поїздок, дозволяючи користувачам легко обирати та бронювати номери у готелях, незалежно від їхнього місцезнаходження, що зробило процес більш зручним та доступним.

Відмова від онлайн-бронювання може призвести до втрати конкурентних переваг і обмеження ринкового охоплення. Більшість сучасних клієнтів використовують інтернет для пошуку та бронювання готелів, тому відсутність онлайн-бронювання може обмежити можливості реклами, просування та зменшити дохід готелів.

За мету було поставлено підвищення ефективності та зручності процесу бронювання готелів онлайн за допомогою розробки сучасного, інтуїтивно зрозумілого та надійного спеціалізованого вебсервісу для бронювання готелів.

Для застосунку було вибрано модель взаємодії клієнт-сервер, яка дуже популярна для створення вебзастосунків (рис. 1). Згідно з цією моделлю, клієнтська та серверна частини працюють разом через мережу, обмінюючись даними та обробляючи запити й відповіді [1].

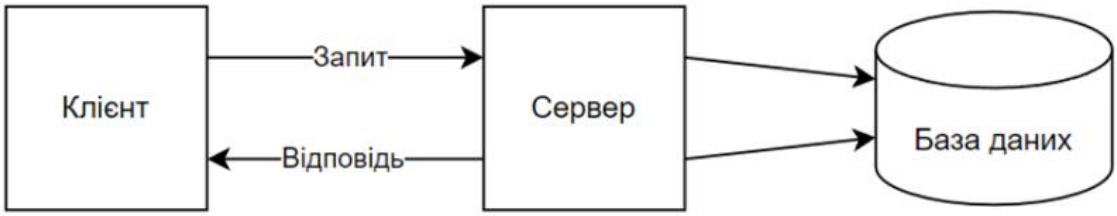


Рисунок 1 – Схема клієнт-серверної архітектури

Для розробки ефективної системи важливо розробити критерії ефективності:

– Час відгуку системи (T): час, необхідний для опрацювання запиту користувача, починаючи від моменту його надсилання і до отримання відповіді. Цей параметр критично важливий для зручності користувача і його загального враження від роботи з сервісом.

– Відсоток успішних бронювань (S, %): відношення числа успішно завершених бронювань до загальної кількості спроб бронювання. Цей показник відображає, наскільки надійно працює система.

– Навантажувальна здатність (N): максимальна кількість запитів, які система може обробити за одиницю часу без зниження продуктивності. Цей критерій показує масштабованість і стабільність сервісу.

– Задоволеність користувача (U): оцінка, отримана від користувачів після використання сервісу. Може вимірюватися через опитування, аналіз відгуків або систему рейтингу.

– Вартість обслуговування на одного користувача (C): загальна вартість підтримки працездатності системи, поділена на кількість користувачів. Цей показник допомагає оцінити економічну ефективність сервісу.

Отже маємо формулу:

$$E = \left(W_1 * \left(\frac{1}{T}\right)\right) + \left(W_2 * S\%\right) + \left(W_3 * N\right) + \left(W_4 * U\right) - \left(W_5 * C\right)$$

Ваги критеріїв будуть такими:
Час відгуку системи (T) або W1 = 0,1.
Відсоток успішних бронювань (S%) або W2 = 0,3.
Навантажувальна здатність (N) або W3 = 0,3.
Задоволеність користувача (U) або W4 = 0,2.
Вартість обслуговування на одного користувача (C) або W5 = 0,1.

У результаті цього дослідження, було виявлено, що використання комбінації React та MongoDB дозволяє створити високопродуктивний, масштабований та легко адаптований вебсервіс бронювання готелів [2]. Потенційні напрямки подальшого розвитку сервісу:

– інтеграція зі штучним інтелектом та машинним навчанням: Використання алгоритмів машинного навчання для персоналізації пропозицій користувачам на основі їхніх попередніх пошуків, переваг і поведінки. Це може включати рекомендації готелів, спеціальні пропозиції та оптимізацію ціноутворення;

– розширення мобільної присутності: Розробка мобільного додатку для iOS та Android, щоб забезпечити користувачам ще більш зручний доступ до сервісу з будь-якого пристрою. Мобільний додаток може включати додаткові функції, такі як мобільні сповіщення про зміни в бронюванні.

Література

[1] Kadam Y., Akhil Goplani A., Mattoo S., et al. Introduction to MERN stack & comparison with previous technologies. *European Chemical Bulletin*. June 2023. Vol. 12. Is. 4. P. 14382–14386. DOI: 10.48047/ecb/2023.12.si4.1300.

[2] Rawal B. S., Karne R. K., Wijesinha A. L. Split protocol client/server architecture. Proc. of the IEEE Internat. Symp. on Computers and Communic. July 2012. P. 348–353. DOI:10.1109/ISCC.2012.6249320.

ВИКОРИСТАННЯ ВІЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ У ПРОТИДІЇ ЗАГРОЗАМ СОЦІАЛЬНОЇ ІНЖЕНЕРІЇ

Хохлов В.А.
Керівник: Солодовник Г.В.
E-mail: ganna.solodovnyk@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Актуальність. досліджень обумовлена стрімким поширенням та зростанням небезпеки загроз соціальної інженерії в сучасному цифровому світі. Протидія загрозам соціальної інженерії є важливою як з точки зору забезпечення безпеки інформації, особистих даних, так і забезпечення стабільності та надійності суспільства в цілому. Швидкий темп розвитку технологій та постійне вдосконалення методів атак вимагають постійного удосконалення заходів захисту та адаптації до нових викликів.

Метою роботи є вивчення різних методів та стратегій протидії соціально-інженерним атакам, та аналізі використання засобів вільного програмного забезпечення рівня цифрової безпеки.

Аналіз основних методів соціальної інженерії та їхніх впливових факторів на користувачів показує, що методи соціальної інженерії використовують різноманітні психологічні та соціальні техніки: фішинг, фармінг, видача себе за авторитетні джерела тощо, які призводять до отримання конфіденційної інформації або впливу на цільових осіб [1].

Потенційні наслідки вдалих та невдалих атак соціальної інженерії для індивідів, компаній та суспільства в цілому можуть призвести до фінансових втрат, порушення конфіденційності, руйнування репутації та інших серйозних наслідків для жертв [2]. Ефективні стратегії захисту від соціально-інженерних атак на різних рівнях найбільшим чином залежать від навичок користувачів розпізнавати шахраїв, встановлювати механізми перевірки ідентичності, та передбачають регулярні нагадування про правила кібербезпеки. Підвищення рівня освіти та свідомості користувачів, зокрема, стосовно методів соціальної інженерії може значно зменшити ризики вдалих атак [1].

За використання технологій штучного інтелекту, машинного навчання та аналізу даних можна ефективно виявляти та блокувати потенційні загрози соціальної інженерії шляхом аналізу великих обсягів даних.

Згідно з аналізом дослідження [2], швидкість розвитку технологій вимагає постійного удосконалення стратегій та методів протидії соціально-інженерним атакам з метою забезпечення цифрової безпеки в майбутньому. Під час трансформації суспільства у цифрове, використання вільного та відкритого програмного забезпечення може стати важливим елементом забезпечення безпеки та захисту від соціально-інженерних атак.

Висновок. Проведений огляд підтверджує необхідність постійного удосконалення заходів протидії соціально-інженерним атакам у сучасному світі. Використання різноманітних методів та стратегій захисту, підвищення рівня свідомості користувачів та постійне вдосконалення технологій є ключовими для забезпечення безпеки інформації та особистих даних. Тільки шляхом поєднання зусиль усіх зацікавлених сторін, а також сучасних засобів автоматизації можна забезпечити надійний та стійкий захист від соціально-інженерних загроз у цифровому суспільстві.

Література

[1] ESKA // Fishing protection [Електронний ресурс]. – Режим доступу до ресурсу: https://eska.global/blog/antifishing-kak-zashititsya-v-sovremennyh-realiyah

[2] Як вберегтися від фішингу - рекомендації кіберполіції [Електронний ресурс]. – Режим доступу до ресурсу: https://cyberpolice.gov.ua/article/yak-vberegtysya-vid-fishyngu---rekomendacziyi-kiberpolicziyi-3030/

[3] Фішинг: методи та приклади атак [Електронний ресурс]. – Режим доступу до ресурсу: https://gridinsoft.ua/phishing

ОГЛЯД СУЧАСНИХ ІНСТРУМЕНТІВ ТА ТЕХНОЛОГІЙ ВИЯВЛЕННЯ КІБЕРЗАГРОЗ

Цеба К.Я.
Керівник : Міхєєв І.А.
E-mail: ivan.mikhieiev@hneu.net
Харків, Харківський національний економічний університет імені Семена Кузнеця

Сучасні інструменти та технології виявлення кіберзагроз включають в себе різноманітні програмні та апаратні рішення, спрямовані на виявлення, аналіз та захист від кібератак. Деякі з них включають:

1. Системи виявлення вторгнень (IDS) та системи захисту від вторгнень (IPS). IDS спеціалізуються на моніторингу та виявленні аномалій у мережі, виявляючи потенційні загрози. IPS виходять за межі простого виявлення, вони активно реагують на потенційні загрози, застосовуючи блокування шкідливого трафіку та інші захисні заходи в реальному часі. Узгоджений робочий процес між IDS та IPS забезпечує повний цикл виявлення та реагування на потенційні вторгнення, гарантуючи високий рівень безпеки для мережі та системи.

2. Машинне навчання та штучний інтелект. Впровадження технологій штучного інтелекту та машинного навчання для автоматизації виявлення нових, раніше невідомих загроз.

3. Аналізатори вразливостей. Використання аналізу великих обсягів даних для виявлення патернів та аномалій, що можуть бути індикаторами кіберзагроз

4. Системи реагування на інциденти (IR). Системи реагування на інциденти (IR) грають ключову роль у виявленні, аналізі та відповіді на кіберінциденти. Ці системи охоплюють моніторинг, класифікацію, реагування, збір доказів, комунікацію, відновлення та навчання, забезпечуючи організоване управління та підвищуючи стійкість до кіберзагроз.

Ці інструменти та технології узгоджено працюють, створюючи комплексний підхід до виявлення кіберзагроз і забезпечення кібербезпеки.

Література

[1] Вікіпедія. Система виявлення вторгнень. [Електронний ресурс]. – Режим доступу до ресурсу: https://uk.m.wikipedia.org/wiki/IDS

[2] Кібербезпека: актуальні загрози та методи захисту. [Електронний ресурс]. – Режим доступу до ресурсу: https://lemon.school/blog/kiberbezpeka-aktualni-zagrozy-ta-metody-zahystu

ПРОГРАМНО-АПАРАТНІ ЗАСОБИ ІНТЕРАКТИВНОГО ОТОЧЕННЯ НА БАЗІ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ

Черянвський Р.А.
Керівник: Крайник Я.М.
E-mail: chernyavskyj@gmail.com
Миколаїв, Чорноморський національний університет імені Петра Могили

В сучасному світі, де технології постійно розвиваються та перетворюють наше повсякденне життя, програмно-апаратні засоби інтерактивного оточення на основі великих мовних моделей набувають все більшого значення. Ця еволюція відбувається в контексті стрімкого розвитку мовних технологій та штучного інтелекту, де великі мовні моделі стають краєвидним елементом інновацій та трансформацій. Спостерігаючи за актуальністю даної теми, важливо відзначити, що великі мовні моделі стають справжнім каталізатором для трансформації способу, яким ми взаємодіємо з цифровими системами. Їх можливості в обробці природної мови та аналізу контексту надають нові перспективи для створення ефективних та інтелектуальних інтерфейсів.

Великі мовні моделі вражають своєю складною архітектурою та принципами функціонування. Вони представляють собою потужні та складні системи, здатні до розуміння та генерації людської мови. Основною їхньою характеристикою є масштабність, яка визначається великою кількістю параметрів і нейронів. Наприклад, моделі, такі як GPT-3, BERT, BLOOM, NeMo можуть мати мільярди параметрів, що дозволяє їм ефективно узагальнювати інформацію з великого обсягу текстових даних[1]. Архітектура великих мовних моделей базується на глибоких нейронних мережах, зокрема трансформерних архітектурах. Ці архітектури дозволяють моделям ефективно опрацьовувати послідовності слів та розуміти їхній контекст за допомогою механізмів уваги.

Ключовим елементом успішного функціонування великих мовних моделей є доступ до обширних обсягів текстових даних. Ці дані допомагають моделі виробляти глибокі репрезентації мови, що включають семантичні, синтаксичні та структурні аспекти. Великий обсяг даних дозволяє моделям "розуміти" різноманіття мовленнєвих відтінків та контекстів, що робить їх універсальними та потужними інструментами для обробки природної мови.

Великі мовні моделі ідеально підходять для класифікації текстової інформації, надаючи високу точність інтерпретації різноманітних текстів. У контексті класифікації текстів вони допомагають визначати теми, категорії та стилі, що спрощує аналіз великих обсягів

інформації. Великі мовні моделі забезпечують надійний інструмент для автоматизованого визначення тематичного спрямування та важливих характеристик текстів[2].

Дана технологія знайшла широке застосування в інтерактивних системах, чат-ботах та віртуальних асистентах, реформуючи спосіб взаємодії людей з технологією. Їх інтеграція дозволяє створювати інтуїтивні та ефективні інтерфейси, здатні розуміти та відповідати на запитання користувачів, вирішувати завдання та надавати корисні рекомендації. За допомогою великих мовних моделей створюються інтерактивні інтерфейси, які взаємодіють з користувачем на природній мові. Це полегшує спілкування з технікою, зменшує необхідність вводити точні команди та робить взаємодію із системою більш інтуїтивно зрозумілою. Великі мовні моделі оптимізують виконання завдань, таких як пошук інформації, складання розкладів, надання порад чи проведення консультацій.

Апаратні засоби відіграють ключову роль у забезпеченні ефективності та швидкості роботи великих мовних моделей. З урахуванням зростаючих вимог до обчислювальної потужності, важливо розглядати оптимальні стратегії використання апаратного забезпечення для максимізації продуктивності. Вивчення впливу програмно-апаратних підходів стає актуальним у контексті оптимізації роботи великих мовних моделей. Оптимальне використання апаратного забезпечення дозволяє досягати високої ефективності обчислень та забезпечує швидке виконання завдань, пов'язаних із обробкою природної мови та аналізом контексту.

Використання великих мовних моделей не лише приносить великі переваги, але й викликає певні складнощі та обмеження. Слід зосередити ретельну увагу на ідентифікації труднощів, які можуть виникнути при застосуванні цих моделей, такі як високі вимоги до обчислювальних ресурсів, необхідність великої кількості даних для навчання, та можливі проблеми з етикою та конфіденційністю. Не менш важливо докладно розглянути потенційні шляхи подолання викликів, що стоять перед використанням великих мовних моделей. Для того, щоб подолати технічні перешкоди та обмеження потрібно розглянути технологічні та методологічні підходи, спрямовані на оптимізацію використання обчислювальних ресурсів, покращення ефективності навчання та вирішення питань етики та конфіденційності [3].

У світі стрімкого технологічного розвитку ключовою завданням є створення високоефективного програмно-апаратного забезпечення, яке забезпечить безперебійну взаємодію з апаратурою та надасть користувачам інтуїтивний та зручний інтерфейс для взаємодії з системою. Створення такого інтерактивного оточення базується на використанні великих мовних моделей, що дозволяють системі більш повно та точно розуміти та обробляти природну мову користувача.

Не менш важливою є розробка ефективних алгоритмів обробки отриманих даних з великих мовних моделей. Це є фундаментом для швидкого та точного реагування системи на запитання та команди користувачів. Вдосконалення цих аспектів є критичним для покращення якості взаємодії людини із сучасними технічними системами, забезпечуючи найвищий рівень зручності та продуктивності.

Література

[1] AiMultiple. Top Large Language Model Examples in 2024 [Електронний ресурс]. – Режим доступу до ресурсу: https://research.aimultiple.com/large-language-models-examples/

[2] Unveiling the Power of Large Language Models (LLMs) [Електронний ресурс]. – Режим доступу до ресурсу: https://www.unite.ai/large-language-models/

[3] Advantages and Disadvantages of Large Language Models [Електронний ресурс]. – Режим доступу до ресурсу: https://www.profolus.com/topics/advantages-disadvantages-of-large-language-models/

Матеріали XV-ої Міжнародної науково-практичної конференції
«FREE AND OPEN SOURCE SOFTWARE»

Харківський національний економічний університет імені Семена Кузнеця

Відповідальний за випуск: Старкова О.В.

Редактор: Міхєєв І.А.

Затверджено засіданням кафедри кібербезпеки та інформаційних технологій
ХНЕУ імені С. Кузнеця
протокол № 11 від «16» лютого 2024 р.

Видавець і виготовлювач – ХНЕУ імені С. Кузнеця, 61166, м. Харків, просп. Науки, 9-А
Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру ДК № 4853 від 20.02.2015 р.