

РОЗДІЛ II. ІНФОРМАЦІЙНО-КОМП'ЮТЕРНІ ТЕХНОЛОГІЇ

УДК 004.056.5

DOI: 10.25140/2411-5363-2020-1(19)-98-103

Інна Стеценко, Вікторія Савчук

МЕТОД АВТОМАТИЗАЦІЇ ТЕСТУВАННЯ НА ПРОНИКНЕННЯ ВЕБАТАК

Актуальність теми дослідження. Збільшення можливості для розширення мереж та вибір у використанні сервісів у сучасному світі, призводить до прогалин у безпеці. З цієї причини виконується моделювання системи, що дозволяє відтворити характеристики інформаційної атаки, а також провести оцінювання рівня її небезпеки.

Постановка проблеми. Нині немає програмного застосунку, що дозволяє автоматизовано моделювати та оцінювати захищеність власної інформаційної системи без втручання в саму систему.

Аналіз останніх досліджень і публікацій. Процес моделювання атак за допомогою мереж Петрі розглядався в декількох роботах закордонних авторів.

Виділення не досліджених частин загальної проблеми. Наявні моделі графових атак створювались лише на прикладі певної архітектури мережі, проблемою залишилось створення універсального автоматизованого застосування, на основі вхідних даних про зв'язки та архітектуру мережі.

Постановка завдання. Автоматизація виявлення зовнішніх вразливостей для аналізу захищеності інформаційної системи, шляхом розробки програмного забезпечення, що моделює за допомогою мереж Петрі розповсюдження атаки залежно від її архітектури.

Виклад основного матеріалу. Програмне забезпечення розроблено на основі мови програмування Java, включає графічну оболонку Java FX, для пошуку сервісів хосту використовується Shodan REST API, для ідентифікації вразливостей ресурси NVD і CVE details. Зібрані дані використовує окремий програмний модуль, що працює зі створенням моделі на основі мережі Петрі.

Висновки відповідно до статті. Запропонований метод дозволяє автоматизувати перевірку інформаційної системи на вразливість до хакерських атак.

Ключові слова: моделювання кібератак; тест на проникнення; уразливості; мережа Петрі.

Рис.: 4. Бібл.: 13.

Актуальність теми дослідження. Сучасні компанії мають великі можливості для розширення своїх інформаційних систем та широкий вибір у використанні сервісів, що надає платформу для розвитку та перспектив. Разом із позитивними можливостями такі зміни привносять складності, що стосуються підтримки великих мереж. Оскільки кількість хостів продовжує рости, оцінка їх вразливості до атак стає все більш важливою для автоматизації.

Велика мережа будується на декількох платформах і різних програмних пакетах і підтримує кілька режимів підключення. Неминуче така мережа буде містити дірки в безпеці, які вислизнули від уваги навіть старанного системного адміністратора.

Щоб оцінити вразливість мережі вузлів, необхідно виявити ефекти взаємодії локальних вразливостей і знаходити глобальні проблеми. Здебільшого ці роботи здійснюються засобами моделювання, що дозволяють відтворити необхідні властивості та характеристики інформаційної атаки, а також провести оцінювання рівня її небезпеки. Моделі дозволяють більш точно визначити ефективність існуючих засобів захисту за допомогою модельованих інформаційних атак.

Постановка проблеми. Нині немає програмних засобів, що дозволяють автоматизовано моделювати та оцінювати захищеність власної інформаційної системи без втручання в саму систему. Існують сценарії тестування системи на проникність – пентести, які виконуються спеціалістами з інформаційної безпеки, використовуючи безпосередньо саму систему.

Аналіз останніх досліджень і публікацій. Процес моделювання атак за допомогою мереж Петрі розглядався в декількох роботах закордонних авторів.

У роботі [1] J. P. McDermott запропонував моделювання атак мережами Петрі, в яких позиції – це важливі для безпеки інформаційної системи стани. Переходи – події, команди чи дані, які можуть бути важливими для зміни стану системи відносно безпе-

TECHNICAL SCIENCES AND TECHNOLOGIES

ки. Маркери в моделі переходять від позиції до позиції, показуючи розвиток атаки. Цей метод добре показує процес розвитку атаки. Але великим мінусом є те, що неможливо отримати вимоги до безпеки та яким чином впливає архітектура системи на безпеку.

Недавні дослідження були зосереджені на впливі мережевих топологій на поширення шкідливих програм. У роботі [2] авторами отримані результати для безрозмірної топології мережі з використанням математичної моделі. Топологія характеризувалася кількістю вузлів і параметрами степеневого показника. У дослідженні [3] структура соціальних і технологічних мереж досліджувалася при їх атаці комп'ютерним вірусом або хробаком. Автори використовували моделювання поширення атаки на основі моделі сприйнятливої інфікованої відновленої епідемії. Вони запропонували структурну модель ризику.

У роботі [4] розглянуто моделювання атак за допомогою кольорових мереж Петрі. Така модель є досить гнучкою для моделювання вторгнень з Інтернету, включаючи статичні і динамічні аспекти атаки. Для того щоб оцінити можливі втрати від вторгнення, у модель були введені вартісні оцінки. Також показано, як можливо її використовувати для моделювання методів захисту, але в цій роботі не було спроб зв'язати можливість проведення атаки з властивостями компонентів інформаційної системи та налаштуваннями систем захисту.

Виділення недосліджених частин загальної проблеми. Наявні моделі графових атак створювались лише на прикладі певної архітектури мережі, проблемою залишилось створення універсального автоматизованого застосування, на основі вхідних даних про зв'язки про архітектуру мережі.

Постановка завдання (мета статті). Головною метою є автоматизація виявлення зовнішніх вразливостей для аналізу захищеності інформаційної системи, шляхом розробки програмного забезпечення, що моделює за допомогою мереж Петрі розповсюдження атаки залежно від її архітектури.

Виклад основного матеріалу. Моделювання атак проводилось у спеціально розробленому середовищі об'єктно орієнтованою мовою програмування Java. Компоненти мережі Петрі (позиції, переходи, дуги) представлені у вигляді об'єктів і всі процеси відтворюються методами ООП. Застосування використовує графічну оболонку для спрощення створення та редагування моделі. Її особливістю є можливість зберігання моделі в кількох форматах, автоматизоване генерування програмного коду методу для створення мережі Петрі за її графічним зображенням у редакторі та можливість відновлення графічного зображення мережі Петрі виключно за програмним кодом відповідного методу [5].

На рис. 1 представлено фрагмент моделі проникнення хакера. Мережа Петрі містить два переходи на атаку відповідно до двох подій «вхід через Інтернет» та «використання вразливості». Перехід «Internet» має вхідну позицію «hacker», маркер в якій символізує вихід хакера в Інтернет. Перехід «service», оточений позиціями «vulnerability» і «target_host», спрацьовує за наявності відповідної вразливості в системі, що символізує маркер в позиції, з'єднаний із переходом інформаційним (пунктирним) зв'язком. Позиція «vulnerability» демонструє ініціацію атаки, використовуючи певну вразливість сервісу, перехід «service» показує експлуатацію сервісу, після якого хакер опиняється всередині цільового хосту. Наявність позицій з інформаційним зв'язком надає можливість відтворювати умови для подій, які після здійснення події не зникають. Дійсно, наявність вразливості в системі є необхідною умовою для події «service», проте запуск сервісу для конкретного користувача не означає, що вразливість зникла. Якщо відповідної вразливості в системі немає, то відсутність маркера у вхідній позиції не дозволить використання сервісу.

На рисунку 2 зображено фрагмент моделі, де представлено три можливі варіанти проникнення хакера в інформаційну систему, використовуючи одну з вразливостей (“vulnerability_1”, “vulnerability_2”, “vulnerability_3”) певного сервісу “service”. Це означає, що в інформаційній системі на певній IP-адресі є сервіс з версією, встановлення якого призводить до наявності трьох вразливостей.

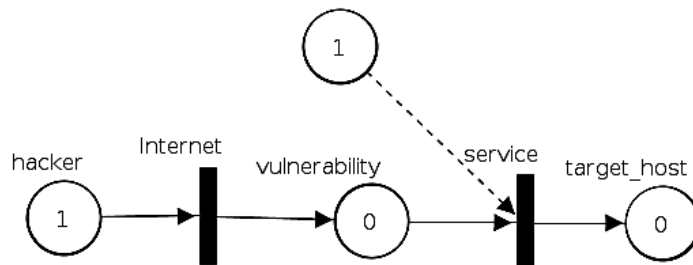


Рис. 1. Проста мережа Петрі, що моделює проникнення хакера всередину хосту, використовуючи вразливість сервісу

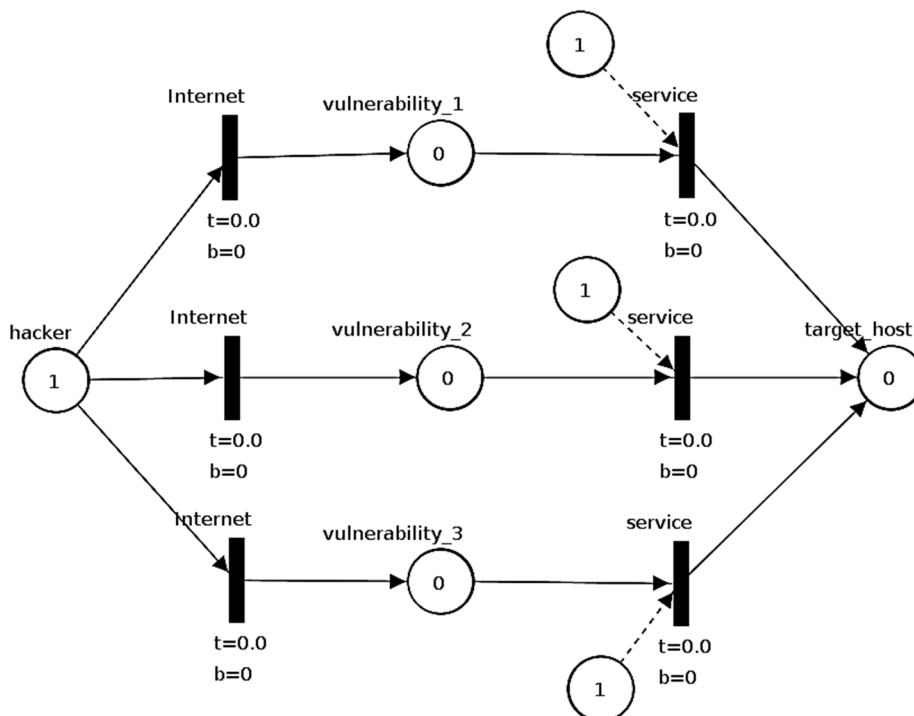


Рис. 2. Проста мережа Петрі, що моделює проникнення хакера всередину хосту, використовуючи вразливість сервісу

Найчастіше дані, що є цільовими, тобто важливими, не лежать на вузлах, що мають вихід у Всесвітню мережу. У такому випадку для досягнення своєї мети, хакер має пройти через низку мереж до тієї, де, наприклад, розгорнута база даних. На рис. 3 зображено побудову моделі мережі Петрі для розповсюдження атаки хакером по інформаційній системі, що має просту архітектуру і складається з вебсервера, файл-серверу та сервера з базою даних (БД). До сервера з БД можливо потрапити через вебсервер або файл-сервер. Вебсервер містить вебсервіс «service_1», який має дві вразливості «vulnerability_1» та «vulnerability_2». Файл-сервер, у свою чергу, містить файловий сервіс «service_2», який має вразливість «vulnerability_3». Експлуатація однієї з трьох представлених вразливостей призводить до отримання прав на проміжному хості «transitional_host», який має доступ до цільового хосту «target_host», базою даних у сервісі «service_3».

TECHNICAL SCIENCES AND TECHNOLOGIES

Пошукова система Shodan [6] надає можливість за хостом визначити відкриті сервіси, що розгорнуті на ньому. Система має відкрите безкоштовне API – Shodan REST API [7]. Запит повертає достатньо велику кількість інформації про хост та його сервіси, але в застосуванні використовуються такі дані: IP, порт та CPE (Common Platform Enumeration). CPE – це стандартизований метод опису та ідентифікації класів програм, операційних систем і апаратних пристроїв, присутніх серед обчислювальних активів підприємства [8].

Застосування, як було зазначено, включає в себе тестування на проникнення. З цієї причини першим завданням є виявлення вразливих місць, які можна використати для проникнення в комп’ютерну мережу, що тестується.

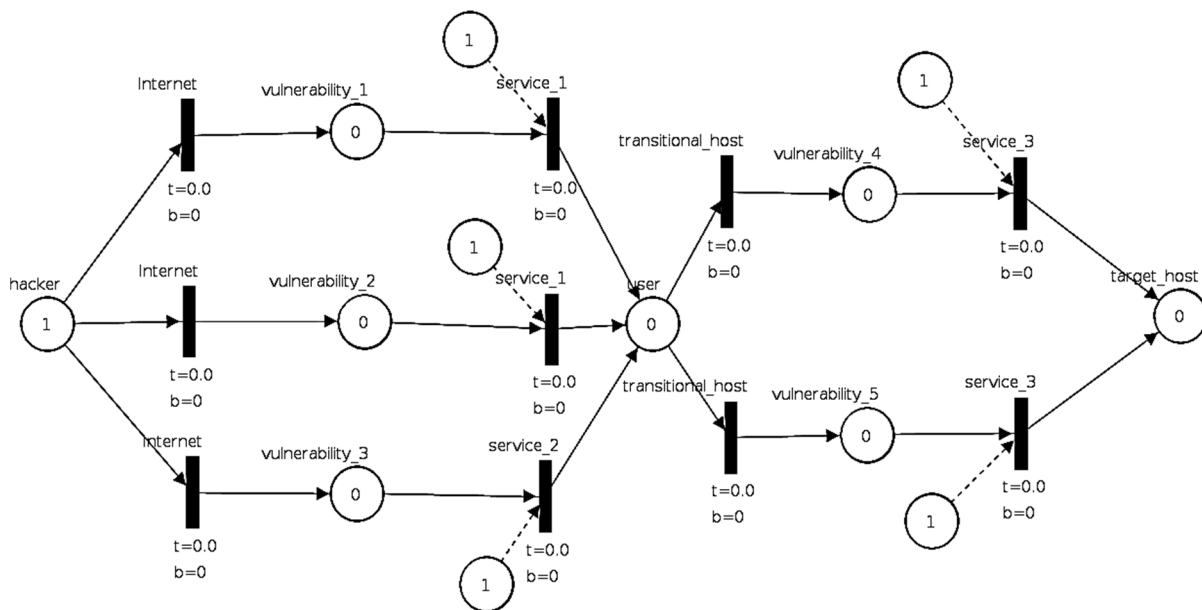


Рис. 3. Мережа Петрі, що моделює проникнення хакера всередину інформаційної системи, що має просту архітектуру

IP та порт – це дані, що в подальшому використовуються як вхідні дані мережі Петрі, а за допомогою CPE відбувається ідентифікація наявних вразливостей, використовуючи пошук у NVD (National Vulnerability Database). NVD – це державне сховище даних управління вразливостями на основі стандартів США [9]. Пошук у NVD за CPE повертає список ідентифікаторів вразливостей сервісу CVE (Common Vulnerabilities and Exposures), що є списком загроз і ризиків інформаційної безпеки. Ідентифікатор вразливостей має формат CVE-0000-0000 [10-12].

Для генерування моделі за допомогою мережі Петрі необхідні характеристики кожної вразливості. Така інформація представлена в зручному вигляді на ресурсі CVE Details [13]. Через відсутність бібліотеки для взаємодії з ресурсом, збір даних відбувається з використанням фреймворку Selenium для Java, який є платформою для управління браузерами за допомогою драйвера WebDriver.

Вхідними даними моделі є IP-адреси, CVE ID, тип доступу (локальний/віддалений) складність (проста, середня, висока), автентифікація (права необхідні для експлуатації сервісу) і набуті права доступу.

Рисунок 4 демонструє архітектуру програмного забезпечення, що розроблено.

Висновки відповідно до статті. Запропоновано спосіб проведення тестів на проникнення без експлуатації інформаційної системи. Проникність системи оцінюється за допомогою моделі, побудованої з використанням мережі Петрі. Представлено архітектуру програмного забезпечення, що за допомогою загальнодоступних способів пасивного сканування хостів організації визначає сервіси та їх вразливості. На основі зібраної та наданої інформації про інфраструктуру системи автоматично генерується модель системи, результатом імітації якої є час розповсюдження атаки. Таким чином, це дає можливість оцінити критичність наявності певної вразливості в системі та зробити висновки щодо її усунення або прийняття ризику, який існує через її наявність у системі.

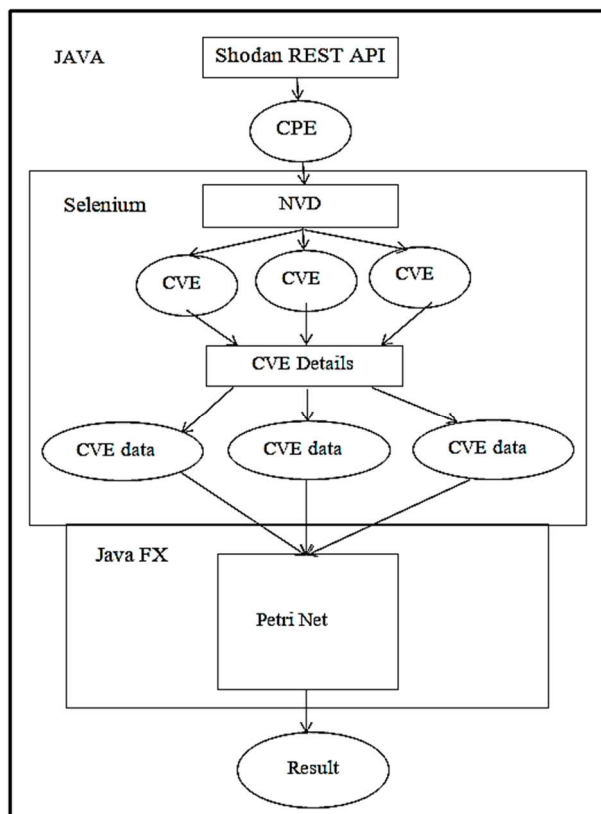


Рис. 4. Архітектура програмного забезпечення

Список використаних джерел

1. McDermott J. P. Attack Net Penetration Testing. *Proc. of the 2000 Workshop on New Security Paradigm*. 2000. URL: <https://www.nspw.org/papers/2000/nspw2000-mcdermott.pdf>.
2. Yang L., Yang X. The effect of network topology on the spread of computer viruses: A modeling study. *International Journal of Computer Mathematic*. 2017. Vol. 94, № 8. P. 1–19.
3. Guo H., Cheng H. K., Kelley K. Impact of Network Structure on Malware Propagation: A Growth Curve Perspective. *Journal of Management Information Systems*. 2016. Vol. 33, № 1. P. 296–325.
4. Xinlei Li, Di Li. A Network Attack Model based on Colored Petri Net. *Journal of networks*. 2014. Vol. 9, № 7. P. 1883–1891. URL: <https://pdfs.semanticscholar.org/003a/05bc845716439ed2c1ef494ac1e8ae7585bb.pdf>.
5. Stetsenko I. V., Dyfuchyn A., Leshchenko K., Davies J. Web application for visual modeling of discrete event systems. *Proceedings of the Seventh International Conference on Internet Technologies and Applications (ITA2017)* / Picking R., Cunningham S., Houlden N., Oram D., Grout V., Mayers J. (eds). Wrexham, UK, 2017. P. 86–91.
6. Shodan. URL: <https://www.shodan.io>.
7. Shodan REST API Documentation. URL: <https://developer.shodan.io/api/>.
8. Common Platform Enumeration (CPE). URL: <https://csrc.nist.gov/Projects/Security-Content-Automation-Protocol/Specifications/cpe>.
9. National Vulnerability Database. URL: <https://nvd.nist.gov>.
10. A Complete Guide to the Common Vulnerability Scoring System Common Vulnerabilities and Exposures. URL: <https://www.first.org/cvss/v2/guide>.
11. CVE official. URL: <https://cve.mitre.org>.
12. CVE format. URL: <https://nvd.nist.gov/vuln/detail/CVE>.
13. CVE Details – the ultimate security. URL: <https://www.cvedetails.com>.

References

1. McDermott, J. P. (2000). Attack Net Penetration Testing. *Proc. of the 2000 Workshop on New Security Paradigm*. Retrieved from <https://www.nspw.org/papers/2000/nspw2000-mcdermott.pdf>.
2. Yang, L., Yang, X. (2017). The effect of network topology on the spread of computer viruses: A modeling study. *International Journal of Computer Mathematic*, 94 (8), 1–19.

TECHNICAL SCIENCES AND TECHNOLOGIES

3. Guo, H., Cheng, H. K., Kelley, K. (2016). Impact of Network Structure on Malware Propagation: A Growth Curve Perspective. *Journal of Management Information Systems*, 33 (1), 296–325.
4. Xinlei, Li, Di, Li (2014). A Network Attack Model based on Colored Petri Net. *Journal of networks*, 9 (7), 1883–1891. Retrieved from <https://pdfs.semanticscholar.org/003a/05bc845716439ed2c1ef494ac1e8ae7585bb.pdf>.
5. Stetsenko, I. V., Dyfuchyn, A., Leshchenko, K., Davies, J. (2017). Web application for visual modeling of discrete event systems. In: *Picking R., Cunningham S., Houlden N., Oram D., Grout V., Mayers J. (eds) Proceedings of the Seventh International Conference on Internet Technologies and Applications (ITA2017)* (pp. 86-91). Wrexham, UK [in English].
6. Shodan. [shodan.io](https://www.shodan.io). Retrieved from <https://www.shodan.io>.
7. Shodan REST API Documentation. developer.shodan.io. Retrieved from <https://developer.shodan.io/api/>.
8. Common Platform Enumeration (CPE). [csrc.nist.gov](https://csrc.nist.gov/Projects/Security-Content-Automation-Protocol/Specifications/cpe). Retrieved from <https://csrc.nist.gov/Projects/Security-Content-Automation-Protocol/Specifications/cpe>.
9. National Vulnerability Database. nvd.nist.gov. Retrieved from <https://nvd.nist.gov>.
10. A Complete Guide to the Common Vulnerability Scoring System Common Vulnerabilities and Exposures. [first.org](https://www.first.org/cvss/v2/guide). Retrieved from <https://www.first.org/cvss/v2/guide>.
11. CVE official. cve.mitre.org. Retrieved from <https://cve.mitre.org>.
12. CVE format. [nvd.nist.gov](https://nvd.nist.gov/vuln/detail/CVE). Retrieved from <https://nvd.nist.gov/vuln/detail/CVE>.
13. CVE Details – the ultimate security. [cvedetails.com](https://www.cvedetails.com/). Retrieved from <https://www.cvedetails.com/>.

UDC 004.056.5

Inna Stetsenko, Viktoriia Savchuk

AUTOMATING WEB ATTACK PENETRATION TESTING METHOD

Urgency of the research. Increasing opportunities for network expansion and choice in the use of services in today's world leads to security gaps. For this reason, the simulation of the system allows reproducing the characteristics of an information attack, as well as assessing the level of its danger.

Target setting. No software application allows you to simulate and evaluate the security of an information system without interfering with the system.

Actual scientific researches and issues analysis. The process of modeling attacks using Petri nets has been discussed in several papers by foreign authors.

Uninvestigated parts of general matters defining. The existing models of graph attacks were created only for specific network architecture. The problem remained is the creation of a universal automated application based on the input data about the network architecture.

The research objective. Automation of detection of external vulnerabilities for the analysis of the security of information system, by the development of software application, models using Petri nets of distribution of attack depending on its architecture.

The statement of basic materials. The software application is developed based on the Java programming language, including the Java FX graphical shell, Shodan REST API is used to search for host services, NVD and CVE details resources are used to identify vulnerabilities. The collected data uses a separate software module that works with the creation of a model based on the Petri net.

Conclusions. The proposed method makes it possible to automate the verification of the information system for vulnerability to hacker attacks.

Keywords: cyber-attack simulation; penetration test; vulnerabilities; Petri net.

Fig.: 4. References: 13.

Стетенко Інна Вячеславівна – доктор технічних наук, професор кафедри автоматизованих систем обробки інформації та управління, НТУУ «КПІ ім. Ігоря Сікорського» (просп. Перемоги, 37, Київ, 03056, Україна).

Stetsenko Inna – Doctor of Science, Professor of Computer-Aided Management And Data Processing Systems Department, NTUU «Igor Sikorsky Kyiv Polytechnic Institute» (37 Peremogy Av., 03056 Kyiv, Ukraine).

E-mail: stiv.inna@gmail.com

SCOPUS Author ID: 55368781500

ORCID: <https://orcid.org/0000-0002-4601-0058>

ResearcherID: C-1512-2019

Савчук Вікторія Володимирівна – студентка, НТУУ «КПІ ім. Ігоря Сікорського» (просп. Перемоги, 37, Київ, 03056, Україна).

Savchuk Viktoriia – student, NTUU «Igor Sikorsky Kyiv Polytechnic Institute» (37 Peremogy Av., 03056 Kyiv, Ukraine).

E-mail: viktoria859@gmail.com