https://doi.org/10.31891/2219-9365-2024-78-39
УДК

СЕМЕНЕЦЬ Олександр
Національний аерокосмічний університет ім. М.Є. Жуковського «Харківський авіаційний інститут»
e-mail: o.y.semenets@csn.khai.edu
ТЕЦЬКИЙ Артем
Національний аерокосмічний університет ім. М.Є. Жуковського «Харківський авіаційний інститут»
e-mail: a.tetskiy@csn.khai.edu

АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ВИБОРУ ТА КОМПЛЕКСУВАННЯ СКАНЕРІВ ВРАЗЛИВОСТЕЙ ДЛЯ ТЕСТУВАННЯ НА ПРОНИКНЕННЯ ІНТЕРНЕТ СИСТЕМ

Наведено результати аналізу літературних джерел з питань комплексування сканерів вразливостей. Розглянуто існуючі підходи до використання сканерів типу "чорного ящику" для виявлення вразливостей. Проаналізовано метрики та набори даних, які використовуються при оцінюванні методів виявлення об'єктів в комп'ютерному зорі, описані тенденції розвитку цих методів. Проаналізовано метрики та набори даних, які використовуються при виборі та комплексуванні сканерів вразливостей для тестування на проникнення інтернет систем. Зроблено висновки щодо можливостей застосування тих чи інших сканерів для виявлення вразливостей з урахуванням різноманіття технологій побудови інтернет систем. Виділено перспективні напрямки досліджень.

Ключові слова: веб-додатки, IoT, сканери тестування, метрики оцінки веб-сканерів, дерево атак.

SEMENETS Oleksandr, TETSKYI Artem
National Aerospace University «Kharkiv Aviation Institute»

ANALYSIS OF METHODS AND MEANS FOR SELECTING AND INTEGRATING VULNERABILITY SCANNERS FOR PENETRATION TESTING OF INTERNET SYSTEMS

With the rapid growth in popularity and use of the Internet over the years, the number of web applications has increased significantly. We use web applications in almost all areas of our lives, including communication, banking, education, and more. Web applications are always available from anywhere there is an Internet connection, allowing us to communicate and collaborate. From the point of view of private and public organizations, the full or partial transfer of professional activities to cyberspace has increased their vulnerability to cyberattacks. Web applications are popular with hackers because the same features that make them attractive to users also attract hackers. Web applications store large amounts of data that hackers can use for their own purposes. The increase in the use of web applications was influenced by the COVID-19 pandemic, which caused many changes in many areas of people's lives, which directly contributed to the emergence of the phenomenon of cyber pandemic. Existing infrastructures are forced to cope with increased network traffic, making them vulnerable to various types of attacks. But not only the development of the use of web applications has influenced the growth of cyber attacks, nowadays we cannot forget about war.

This paper presents the results of a literature review on the integration of vulnerability scanners. Existing approaches to the use of black-box scanners for vulnerability detection are discussed. Metrics and datasets used in evaluating object detection methods in computer vision are analyzed, along with the trends in the development of these methods. The metrics and datasets used in the selection and integration of vulnerability scanners for penetration testing of Internet systems are examined. Conclusions are drawn regarding the applicability of various scanners for vulnerability detection, considering the diversity of technologies used in building Internet systems. Prospective research directions are identified.

Keywords: web applications, IoT, testing scanners, web scanner evaluation metrics, attack tree.

ПОСТАНОВКА ПРОБЛЕМИ У ЗАГАЛЬНОМУ ВИГЛЯДІ ТА ЇЇ ЗВ'ЯЗОК ІЗ ВАЖЛИВИМИ НАУКОВИМИ ЧИ ПРАКТИЧНИМИ ЗАВДАННЯМИ

Зі стрімким зростанням популярності та використання Інтернету протягом багатьох років кількість веб-додатків значно зросла. Ми використовуємо веб-програми майже в усіх сферах нашого життя, включаючи зв'язок, банківську справу, освіту, тощо. Веб-програми завжди доступні з будь-якого місця, де є підключення до Інтернету, що дозволяє нам спілкуватися та співпрацювати. З точки зору приватних і державних організацій, повне або часткове перенесення професійної діяльності в кіберпростір підвищило їх вразливість до кібератак. Веб-програми користуються популярністю серед хакерів, тому що ті самі функції, які роблять їх привабливими для користувачів, також приваблюють і хакерів. Веб-програми зберігають великі обсяги даних, які хакери можуть використовувати у власних цілях. На збільшення використання веб-додатків вплинула пандемія COVID-19, яка спричинила багато змін у багатьох сферах життя людей, що безпосередньо сприяло виникненню явища кібер-пандемії. Існуючі інфраструктури змушені справлятися зі збільшенням мережевого трафіку, що робить їх вразливими до різних типів атак. Але не лише розвиток використання веб-додатків вплинув на зріст кібератак, в наш час не можна також забувати про війну. З 2022 року Україна зазнала значного збільшення кібератак з боку Російської Федерації. Ці атаки стали невід'ємною частиною війни, яку Росія веде проти України, поряд з традиційними військовими діями. Кібератаки націлені на різні сфери, включаючи державні установи, енергетичну інфраструктуру, фінансовий

сектор та медіа. Аналітичний звіт Держспецзв'язку про рік повномасштабної кібервійни Росії проти України містить інформацію про діяльність російських хакерів в Україні протягом другої половини 2022 року порівнює його з їх діяльністю протягом першої половини, аналізує цілі та мотиви російських хакерів, а також інструменти, які вони використовують. Звіт показує як змінився фокус російських хакерів з телекомунікаційних галузей, які були серед основних цілей на початку війни, на енергетичну систему, яка з жовтня 2022 року стала однією з основних цілей ракетних атак Росії [1]. Аналітика від Держспецзв'язку за 2022-2023 роки приведена на рисунку 1.

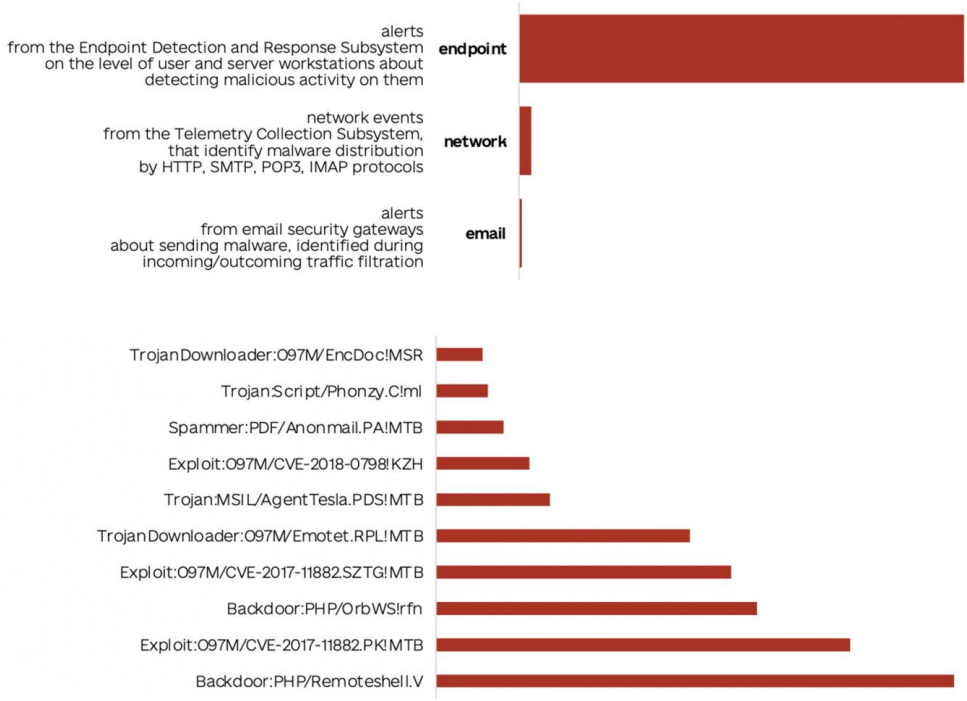

Рис. 1. Приведена аналітика Держспецзв'язку

ФОРМУЛЮВАННЯ ЦІЛЕЙ СТАТТІ

Метою даної роботи є привести оглядову інформацію про різні об'єкти сканування їх особливості і вразливості, різні вразливості і причини їх появи, різноманітні сканери веб вразливостей, плюси і мінуси їх використання, проведено дослідження використання методу дерева атак.

ВИКЛАД ОСНОВНОГО МАТЕРІАЛУ

Огляд джерел. За результатами огляду ключових джерел їх можна класифікувати у такий спосіб:

- Джерела, у яких висвітлено питання про тестування і проникнення [2-4]. Проведено систематичний огляд літератури стосовно тестування на проникнення. Для вирішення проблем використання відкритих портів рекомендовано використання глибокого підкріплювальний навчання. Дослідження спрямовані на проникнення з використанням розподіленого брандмауера. Та проектування шляхів тестування на проникнення.
- Джерела, присвяченні скануванню пристроїв в Інтернеті речей та методам виявлення вразливостей в IoT [5-10]. Також була запропонована і розглянута загальноприйнята архітектура в IoT з вразливостями котрим підтверджена той чи інший рівень в Інтернет речах та рекомендації по запобіганню та виправленню в роботі систем. Також було розглянуто використання опкодів в роботі смарт контрактів. Статті в котрих було розглянуто процес пріорітезації.
- Статті присвяченні архітектурі веб-сканерів [11-13], їх класифікації та досліджень, що було проведенні на сканерах з вільним доступом і не тільки.
- Джерела, котрі розглядають побудови моделей для аналізу вразливостей в багатошарових бездротових мережах та статті з створенням моделі для аналізу вразливостей [14-17]. Та дисертація в котрій, А.Г. Тецький проводить дослідження з використанням дерева-атак.

Тестування на проникнення.

Автори у своїй статті [2] проводять систематичний огляд літератури щодо тестування на проникнення в мережі. Однією з цілей роботи дослідження є підвищення усвідомленості організацій, щодо важливості даної проблеми, адже ці організації можуть піддати ризику особисту та фінансову інформацію своїх клієнтів. Автори визначили три підходи тестування на проникнення в залежності від підходів до тестування (тестування "чорної скриньки" (black-box), "білої скриньки" (white-box) та "сірої скриньки" (gray-

box)). Серед загальновідомих стандартів для запобігання атак автори визначили такі як: фреймворк для оцінювання безпеки інформаційних систем (ISAAF), спеціальна публікація Національного інституту стандартів і технологій 800-115 (NIST SP 800-115), ручний посібник з методології тестування безпеки з відкритим вихідним кодом (OSSTMM), стандарт виконання тестування на проникнення (PTES).

Було проаналізовано використання тих чи інших інструментів для тестування на проникнення.

Aircrack-ng - це повний набір інструментів для оцінки безпеки мереж Wi-Fi.

Nmap - інструмент для сканування мережі з метою виявлення портів, хостів, операційних систем та служб для виявлення вразливостей.

Metasploit - інструмент, який дозволяє перевіряти вразливості в операційних системах та програмах.

BeEF (Browser Exploitation Framework) - інструмент для експлуатації у контексті веб-браузерів.

Shadow - пошукова система, яка дозволяє знаходити конкретні пристрої та їх типи, аналіз працює основуючись на результатах банерів від про сканованих пристроїв.

Wireshark - інструмент, який ще відомий як мережевим сніфером. За допогмогою захоплених пакетів можна провести аналіз трафіку без виявлення вразливою системою.

Zed Attack Proxy (ZAP) - це інструмент що працює як проксі-сервер, перехоплюючи та аналізуючи трафік між веб-браузером і веб-додатком. Гарно підходить для початківців.

Netcat - це інструмент командного рядка, використовує протоколи TCP та UDP для зчитування та запису даних через мережеві з'єднання.

Мережі як об’єкт сканування були обрані через свою розповсюдженість та простоту використання. Головною вразливістю використання мереж є наявність в них відкритих портів. Найпоширенішою атакою було приведено DoS атаки, а найпопулярнішим інструментом Nmap. Рекомендована техніка для усунення вразливостей портів є використання глибокого підкріплювального навчання. Глибоке підкріплювальне навчання (deep reinforcement learning, DRL) - підхід, який дозволяє створювати алгоритми, здатні навчатися оптимальній поведінці і об’єднує в собі глибоке навчання (deep learning) і підкріплювальне навчання (reinforcement learning). Майбутні дослідження автори планують зосередити на автоматизованому тестуванні мережевої безпеки на основі цього підходу, для виявлення KRACK атак.

В статті “Дослідження вразливостей безпеки за допомогою тестування на проникнення в розподіленому брандмауері” [3] автори розглядають на вирішенні основних загальних проблем, виявлених у відкритому брандмауері. З’являються нові загрози в кібербезпеці тому проводи аудит безпеки потрібно регулярно. Для підтримання безпеки системи, автори пропонують використовувати розподілений брандмауер. Для проведення ручного тестування на проникнення автори рекомендують використовувати інструменти вбудовані в Kali Linux, а особливо Nmap. А автоматизоване тестування за допомогою інструмента Nessus. Ці тестування необхідно поєднати для досягнення найкращого результату. Дослідження якості роботи розподілених брандмауерів автори проводять на при кінці статті. Ці брандмауери використовують адресні простори IPv4 та IPv6 для контролю вхідного та вихідного трафіку. Автори використовують доступні інструменти для тестування на проникнення. Зазвичай кіберзлочинці використовують поширені інструменти з відкритим кодом, адже простота їх використання робить їх такими привабливими. Майбутні роботи автори зосередять на використанні інших засобів для виявлення вразливостей. А зараз вони пропонують протестувати кілька сценаріїв кібератак на їх розподілений брандмауер, щоб переконатися в його ефективності.

Автори у своїй статті [4] розглядають алгоритми навчання з підкріпленням та їх найбільш розповсюджену проблему планування шляхів проникнення. Для експерименту автори створили 3 набори сценаріїв складність планування шляхів атаки в яких поступово зростає. Алгоритми розглянуті в цій статті MDDQN, DQN, DDQN та DuelingDQN, їх робота порівнюється між собою. Автори прийшли до висновку що найкраще справився алгоритм MulVAL подвійної глибокої Q-мережі (MDDQN), незважаючи на те що він має певні обмеження, такі як нездатність автономно сканувати та отримувати інформацію про мережу. Цей алгоритм поєднує графи атак MulVAL з алгоритмом подвійної глибокої Q-мережі (DDQN). У майбутніх дослідженнях автори планують розглянути більш складні середовища тестування на проникнення.

IoT та виявлення вразливостей.

У своїй статті [5] автори наголошують на проблемі вразливості Інтернету речей. Серед яких можна вказати відсутність шифрування через енергетичні обмеження. Для пошуку вразливих пристроїв автори виділили 3 способи: пошук за допомогою Shodan, для пошуку вразливих маршрутизаторів Cayman DSL; з використанням Masscan для швидкого пошуку великого діапазону IP-адрес, для пошуку пристроїв вразливих до Heartbleed Bug; з використанням Nmap та PFT для пошуку та підключення до вразливих мережевих принтерів. Сканування вразливих пристроїв в IoT є надзвичайно важливим для забезпечення належної безпеки та конфіденційності.

У статті [6] автори провели систематичний огляд літератури, про сканування вразливостей в Інтернеті речей. Головна проблема в кібербезпеці IoT пов'язана з різноманітністю пристроїв від різних виробників, що часто не відповідають стандартам і мають різні функціональні можливості.

Автори пропонують реалізувати процес сканування в 6 ітеративних кроків: вибір, конфігурація, ініціювання, збір даних, валідація і аналіз.

Впроваджено термін сканувальний простір, що охоплює всі спостережувані процеси сканування, що допомагає візуалізувати покриття, складність та вимоги до часу таких сканувань.

Проведені дослідження, дотримувались запропонованому алгоритмові для сканування і використовували доступні інструменти для оцінки стану підключення та вразливостей. Результати показали значні вразливості в промислових протоколах IoT, таких як Modbus і DNP, а небезпечне впровадження протоколів FTP та SMB.

У своїй статті [7] автори проводять дослідження для визначення загальних фреймворків, які використовуються для оцінки та виявлення вразливостей IoT. В статті автори привели загальну архітектуру IoT систем, розбивши її на 3 рівні, для кожного рівня були описані вразливості котрим вони піддаються, загальну архітектуру можна побачити на рисунку 2.

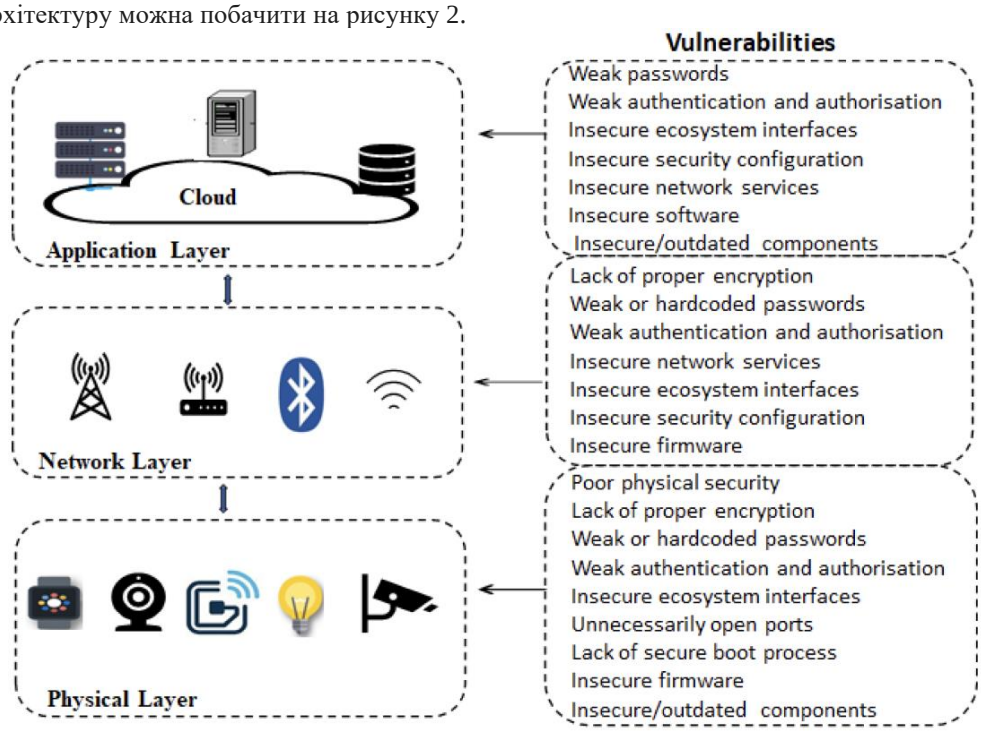

Рис. 2. Загальноприйнята архітектура IoT систем з вразливостями кожного рівня

Фізичний рівень є найнижчим рівнем архітектури IoT. Він включає всі фізичні пристрої, такі як IP-камери, Fitbits, датчики, RFID-мітки та зчитувачі RFID. Ці пристрої можуть виявляти різну інформацію, таку як рух і температура, і надсилати зібрану інформацію на мережевий рівень.

Найпоширеніші вразливості фізичного рівня включають відсутність фізичної безпеки, неправильне шифрування, слабкі або зашифровані паролі, слабку автентифікацію та авторизацію та надлишкові відкриті порти (наприклад, відкриті порти SSH або порти Telnet тощо), відсутність безпечного процесу завантаження, небезпечне програмне забезпечення та застарілі компоненти.

Відсутність надійного пароля є значною вразливістю безпеки на цьому рівні, і чим слабший пароль, тим легше та швидше його відновити за допомогою атак brute force.

Вразливості, пов’язані зі зловмисним програмним забезпеченням, можуть дозволити зловмиснику впровадити шкідливий код у пристрій, надаючи хакеру повний доступ до мережі.

Відсутність безпечного механізму завантаження створює ще одну вразливість в пристроях IoT. Безпечне завантаження необхідне, щоб зловмисники не могли зламати операційну систему або встановити шкідливі завантажувачі на пристрої IoT.

Інша вразливість IoT виникає через неправильне керування оновленнями пристрою. До них належать неможливість перевірки мікропрограми пристрою, неможливість безпечної передачі даних (вони передаються публічно), відсутність захисту від відкату та нездатність в попередженні користувачів про зміни безпеки, спричинені оновленнями.

Використовуючи вразливий механізм оновлення, зловмисники можуть здійснювати різноманітні атаки. Відкриті порти дозволяють отримати доступ до операційної системи пристрою IoT.

Мережевий рівень забезпечує зв'язок між пристроями, дозволяє пристроям IoT отримувати доступ до Інтернету та забезпечує ефективну передачу інформації. Цей рівень використовує різні технології передачі даних, такі як Wi-Fi, LTE, Ethernet, BLE і ZigBee.

Вразливості мережевого рівня включають відсутність належного шифрування, слабку автентифікацію та авторизацію, ненадійні інтерфейси екосистеми, ненадійні конфігурації безпеки та ненадійне програмне забезпечення.

Зловмисники часто отримують доступ до мережі, використовуючи вразливості в ненадійних інтерфейсах. Зловмисники можуть використати вразливості в протоколах зв'язку IoT, щоб перехопити конфіденційну інформацію та розпочати масштабні атаки або запобігти передачі інформації.

Багато пристроїв IoT не мають шифрування, оскільки 98% трафіку пристроїв IoT в Інтернеті не зашифровано. Зважаючи на це, зловмисник може використати атаки типу "людина посередині" (MITM) або інші засоби підслуховування.

Стандартні або зашифровані паролі - одна з найпоширеніших вразливостей. Зловмисники можуть скомпрометувати пристрої IoT, а потім використовувати скомпрометовані пристрої для запуску великомасштабних ботнетів або інші атаки.

Приблизно 70% пристроїв IoT налаштовано на використання заводських облікових даних за замовчуванням, які легко розшифровуються. Крім того, споживачі зазвичай не змінюють облікові дані за замовчуванням або використовують прості облікові дані для входу. Зловмисники часто використовують словники, що містять список загальних облікових даних (імена користувачів і паролі).

Рівень додатків, також відомий як рівень обслуговування, є верхнім рівнем архітектури IoT. Цей рівень містить інтерфейс користувача програми та надає кінцевим користувачам послуги, пов'язані з програмою, наприклад інтеграцію та аналіз даних із пристроїв IoT. Найпоширеніші вразливості на цьому рівні включають слабкі, вгадані або хешовані паролі, слабку автентифікацію та авторизацію, незахищені інтерфейси екосистеми, незахищені конфігурації безпеки та незахищене програмне забезпечення. Пристрої IoT використовуються в різних програмах IoT, таких як розумні будинки, розумні мережі, безпілотні автомобілі, розумні міста.

Вразливості, на які зловмисники націлені в додатках IoT, включають незахищені мережеві з'єднання, невідповідні та незахищені протоколи, незашифроване зберігання даних, застарілі компоненти додатків IoT, слабкі паролі, неправильні методи оновлення та вразливості, пов'язані з автентифікацією.

Зловмисники використовують вразливості, пов'язані з програмним забезпеченням, щоб зламати програми та викрасти чи змінити конфіденційні дані. Вразливості через незахищені інтерфейси екосистеми можуть призвести до компрометації пристроїв IoT. Протоколи на цьому рівні, такі як Message Queuing Telemetry Transport (MQTT), відіграють важливу роль. Вони формують основу для взаємодії між додатками та службами, що працюють на різних пристроях IoT і хмарних/граничних інфраструктурах. Тому безпека протоколу дуже важлива.

Проте кожен протокол має власні вразливості, що дає зловмисникам кілька способів перехоплювати сеанси та маніпулювати даними. Тому забезпечити такий рівень безпеки надзвичайно складно через вбудовані вразливості, такі як відсутність належних служб безпеки в протоколах, незахищені конфігурації безпеки та неправильні конфігурації продуктів і послуг, які наражають пристрої IoT на низку загроз безпеці. Наприклад, зловмисник може розпочати атаку на відмову в обслуговуванні (DoS) на цьому рівні, використовуючи вразливі місця в реалізації або дизайні протоколу. На відміну від широкомасштабних DoS-атак, вони характеризуються прихованими атаками, націленими на конкретні програми, які використовують жертви.

Для виправлення багатьох вразливостей потрібно багато ресурсів, але не всі з них мають однаковий ризик, тому можна не витрачати ресурси на виправлення тих вразливостей що мають не значний ризик. Оцінюючи бали вразливостей, особи, що приймають рішення, можуть порівнювати рейтинги вразливостей та швидко вирішувати, на яких критичних вразливостях слід зосередитися, оскільки їх експлуатація може мати серйозні наслідки. Для автоматичного виявлення та оцінювання вразливостей в IoT системах використовують Nessus та Shodan. Доцільно використовувати їх разом, Shodan буде збирати інформацію про відомі вразливості обладнання що досліджується, а Nessus буде оцінювати виявленні вразливості. Для кількісної оцінки виявлених вразливостей автори радять використовувати систему Common Weakness Enumeration (CWE), National Vulnerability Database (NVD) та систему CVSS. Автори приходять до висновку, що загально прийняті методи оцінки вразливостей не встигають за розвитком вразливостей що використовують хакери, тому майбутні дослідження планують зосередити на розробці фреймворків оцінки вразливостей.

VRM - це циклічна практика виявлення, класифікації, оцінки та усунення вразливості. Етап оцінювання VRM вимагає втручання експертів для оцінки ризику вразливостей для організацій і визначення пріоритетності виправлень, що займає багато часу та ресурсів. Тому автори в своїй статті пропонують автоматизований контекст-орієнтовний (ACVRM) щоб автоматизувати процедуру VRM та зменшити втручання експертів.

У цій статті [8] автори зосередилися на процесі класифікації та оцінки VRM. Було проведені інтерв'ю з експертами, щоб дізнатися, які критерії відіграють роль у визначенні пріоритетів виправлень. Автори виявили, що оцінка безпеки, вектор атаки, складність атаки, конфіденційність, цілісність і

доступність є важливими критеріями для пріорітезації виправлень. Тому було запропоновано базуватись на критеріях залежно від специфіки організацій.

Автори порівняли пріорітезацію виправлень ACVRM з плагіном CVE Rudder. Отриманий результат показав, що ACVRM міг налаштувати пріоритети виправлень для кожної організації з меншими залученням експертів із безпеки.

Використовуючи ACVRM організації можуть налаштувати пріоритет виправлень зваживши критерії. Цілю майбутніх досліджень автори планують провести дослідження стосовно вирішення проблеми в управлінні виправленнями, включаючи автоматичну перевірку розгортання виправлень, перевірку побічних ефектів виправлення вразливостей і можливість узагальненого процесу перевірки, а також дослідити часову ефективність рішення та порівняти пріорітезацію виправлень порівняно з найсучаснішими підходами.

Автори [9] вивели такий термін як кіберпандемія, на спричинення якої вплинула пандемія COVID-19, зростання мережевого трафіку та хакерських атак. В роботі було розглянуто використання системи оцінки загальних вразливостей (CVSS) версії 3.x. Рейтинг вразливості впливає на три категорії оцінок - Основну (BS), Часову (TS) та Екологічну (ES). Оцінка TS та ES не обов'язкові. Різницю між CVSS 2.0 та CVSS 3.x можна побачити на рисунку 3.

CVSS 2.0 Standard	CVSS 3.x Standard
Access Vector (AV)/L, A, N	Attack Vector (AV)/N, A, L, P
Access Complexity (AC)/H, M, L	Attack Complexity (AC)/L, H
Authentication (Au)/M, S, N	Privileges Required (PRs)/N, L, H
	User Interaction (UI)/N, R
	Scope (S)/U, C
Confidentiality Impact (C)/N, P, C	Confidentiality Impact (C)/H, L, N
Integrity Impact (I)/N, P, C	Integrity Impact (I)/H, L, N
Availability Impact (A)/N, P, C	Availability Impact (A)/H, L, N

Рис. 3. Різниця між версіями стандарту CVSS 2.0 та CVSS 3.x

В стандарті 2 версії оцінка описується 6-ма компонентами, котрі на вхід приймають одне з 3-х значень. В 3 версії на 2 компоненти більше, також вектор атаки та складність атаки відрізняться за вхідними параметрами. Ці деталі впливають на роботу систем пріорітезації. 3 версія задіє деякі алгоритми для автоматичної оцінки базового рейтингу, котрі поліпшують швидкість публікації рейтингів у Національній базі даних про вразливості (NVD). Автори використали алгоритм машинного навчання, для перетворення рейтингу зі стандарту 2-ї версії у 3. Метод цього перетворення описаний авторами у 4 етапи (отримання даних, вибір навчального набору, вибір алгоритмів машинного навчання та їх тестування). Використання запропонованого алгоритму машинного навчання поліпшує процес класифікації параметрів метрик базової оцінки CVSS 3.x. Запропоновані алгоритми мали змогу переоцінювати результати, це допомогло позбутися наслідків недооцінки. Використання нового стандарту з запропонованими алгоритмами дозволяє поліпшити оцінку рівня безпеки. Майбутні дослідження будуть спрямованні на удосконалення алгоритмів машинного навчання.

Автори у даній [10] статті розглянули архітектуру розумних контрактів. Головна проблема полягає в тому, що для розробки розумних контрактів використовується мова Solidity яка підтверджена численним вразливостям безпеки. Проблеми безпеки викликані використанням доступного розумного контракту, до якого мають доступ не лише програмісти а й хакери. Наразі програмісти намагаються вирішити цю проблему. Серед вразливостей, котрі можуть використовувати хакери можна виділити такі як: вразливість типу Reentrant, вразливості переповнення цілочисельного типу, вразливість коротких адрес, вразливості залежності від часових міток та вразливості залежності від послідовності транзакцій. Для виявлення вразливостей смарт-контрактів автори запропонували п'ять типів методів виявлення, таких як: символьне виконання, формальна верифікація, статичний аналіз, динамічний аналіз та виявлення за моделлю. Автори у статті запропонували метод для покращення ефективності виявлення вразливостей, спрямований на поліпшення внутрішнього модуля інструменту Mythril. Знаючи специфікацію вихідного коду можна автоматично виявляти вразливість розумного контракту. Його ефективність було доведено через експерименти.

Веб-сканери

Сканер веб-додатків - це програма, яка автоматично виявляє потенційні вразливості у веб-додатках. Сканує веб-програми на наявність різноманітних вразливостей безпеки, зокрема: Приклади включають міжсайтовий сценарій (XSS), впровадження SQL, недостатню автентифікацію та авторизацію, вразливості керування сеансами тощо.

Сканери веб-додатків можуть виявляти вразливості, аналізуючи HTTP-запити та відповіді та взаємодіючи з веб-сайтами для тестування різних типів атак.

У своїй статті [11] автор представив оцінку одинадцяти сканерів веб-вразливостей чорного ящика, як комерційних, так і з відкритим вихідним кодом. Результат показав, що важливо не лише виявляти вразливості, але й обходити веб-додаток і досягати "глибинних" ресурсів додатка. Існують цілі класи вразливостей, які не можуть бути виявлені найсучаснішими сканерами. Автори виявили, що вісім з шістнадцяти вразливостей не були виявлені жодним із сканерів. Необхідно покращити реверс-інжиніринг для відстеження стану додатка, для автоматизації виявлення складних вразливостей.

Автор прийшов до висновку, що немає сильної кореляції між вартістю сканера та його функціональністю, оскільки деякі безкоштовні або дуже економічно ефективні сканери працювали так само добре, як і сканери, які коштують тисячі доларів.

У своїй статті [12] автори проаналізували та оцінили точність виявлення сканерів типу "чорного ящика" на виявлення вразливостей впровадження SQL-запитів. SQL-ін’єкція - це тип кібератаки, під час якої зловмисник вводить шкідливі SQL-запити в поля введення веб-додатку або іншого інтерфейсу, який взаємодіє з базою даних. Ця атака використовує вразливість в обробці запитів SQL, яка може призвести до неконтрольованого доступу до бази даних і дозволяє зловмисникам змінювати дані бази даних. Основна ідея полягає в тому, що зловмисник отримує або надсилає на сервер спеціально створений вхідний рядок, що містить код SQL. Інші поширені ін'єкції включають SQLi, NoSQL, ін'єкції команд операційної системи, ін'єкції легкого протоколу доступу до каталогів (LDAP), ін'єкції мов виразів, SSTi та ін'єкції бібліотеки навігації графів об'єктів (OGNL).

Автори надали список функціональних вимог, які повинні задовольняти всі сканери веб-вразливостей:

- Можуть ідентифікувати конкретний набір вразливостей безпеки у веб-додатку;
- Можуть генерувати текстовий звіт, що описує атаку для кожної виявленої вразливості;
- Мають низький рівень помилкових спрацьовувань.

Також автори привели архітектуру сканерів, які складаються з трьох модулів: сканування, атаки та аналізу.

Модуль сканування використовує сканер для навігації по веб-додатку, щоб ідентифікувати та відновлювати веб-сторінки, вхідні вектори (наприклад, поля вводу у формах HTML), параметри запитів GET/POST і кукі. Потім сканер генерує індексований список усіх доступних уніфікованих локаторів ресурсів (URL). Виявлення веб-вразливостей залежить від якості сканера.

Модуль атаки (фаззинг) використовує фаззер для аналізу URL і вхідних векторів, ідентифікованих сканером, після чого надсилає потенційні шаблони атак до точок входу. Фаззер створює список потенційно вразливих значень, щоб викликати вразливість для кожного введення і типу. Наприклад, компонент фаззера намагається ввести шкідливий код JavaScript, щоб перевірити наявність вразливості XSS.

Модуль аналізу аналізує результати, отримані на попередньому етапі, щоб виявити існуючі вразливості та надати іншим модулям коментарі. Наприклад, якщо повернена сторінка містить повідомлення про помилку бази даних у відповідь на тести введення для ін'єкції SQL. У цьому випадку модуль аналізу прогнозує потенційну вразливість SQLi на цій сторінці.

Автори виділили метрики для оцінки роботи сканерів.

Істинно-позитивні результати (TP) - це вразливості, виявлені сканером, які дійсно існують у коді.

Хибно-позитивні результати (FP) - це вразливості, виявлені сканером, які насправді не існують. Хибно-позитивні результати становлять значну проблему для користувачів. Якщо кількість хибно-позитивних результатів висока, користувач змушений вручну перевіряти кожну виявлену вразливість, щоб оцінити її достовірність.

Хибно-негативні результати (FN) - це вразливості, які насправді існують у коді, але не були виявлені сканером.

Точність (Precision) - це відношення правильно виявлених вразливостей до загальної кількості виявлених вразливостей.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Відкликання (Recall) - це відношення правильно виявлених вразливостей до загальної кількості існуючих вразливостей.

$$\text{Recall} = \frac{TP}{TP + FN}$$

F-міра (F-measure) - це гармонічне середнє точності та відкликання.

$$\text{F-Measure} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Для оцінювання були обрані доступні вразливі веб-додатки та чорні сканери. У цьому дослідженні автори мали на меті оцінити як комерційні, так і відкриті сканери. Вони обрали п'ять сканерів чорного ящика (один комерційний і чотири з відкритим кодом): Burp Suite Professional, OWASP ZAP, Vega, Skipfish і Wapiti. Сканери були відібрані на основі їхньої доступності та здатності виявляти вразливості ін'єкції.

ZAP - безкоштовний інструмент для тестування на проникнення з відкритим кодом для виявлення вразливостей у веб-додатках. Має функцію проксі для перехоплення та перевірки повідомлень, що надсилаються між клієнтом і веб-додатком.

Burp Suite Professional - комерційний інструмент для забезпечення безпеки. Здатний проводити пасивний і активний аналіз. Його проксі/історія дозволяє змінювати всі захищені HTTPS-комунікації, що проходять через браузер.

Vega - безкоштовний інструмент з відкритим кодом для тестування веб-додатків і виявлення вразливостей. Надає автоматичне сканування для швидких тестів і має компонент перехоплювального проксі.

Skipfish - безкоштовний сканер вразливостей з відкритим кодом, який може створювати інтерактивну карту сайту для сканованого веб-додатка. Може запускати повторювані сканування та сканування на основі словників.

Wapiti - безкоштовний інструмент з відкритим кодом, командний додаток для сканування веб-додатків. Може виявляти форми на сторінках веб-додатку, в які можна ввести дані.

Авторами було проаналізовано сповіщення та звіти, що згенерував кожен сканер у кожному режимі, та вручну перевели їх у значення FN, TP та FP.

Було з'ясовано, що ZAP і Burp Suite Professional відпрацювали найкраще. Оцінені сканери чорного ящика пропустили більшість існуючих вразливостей, а деякі з них взагалі не змогли нічого виявити. Це було спричинено двома причинами. Перша, сканери мають обмежену здатність в скануванні веб-додатків. Друга, сканери не можуть виявити всі типи ін'єкційних дефектів. Автори радять забезпечувати сканери проксі-компонентом, щоб виявляти більше існуючих вразливостей.

Автори пропонують при виборі сканера враховувати кілька аспектів [13].

Сканер повинен:

- Підтримувати протоколи та алгоритми автентифікації.
- Підтримувати основні типи методів введення даних.
- Задовольняти технічні можливості людини, яка буде ним користуватися.
- Бути стабільним і регулярно оновлюватися, щоб виявляти нові вразливості.
- Задовольняти бюджет.

Автори визначили три характеристики:

Якість індексації. Процес визначення сторінок веб-додатка, що вразливі до певної атаки. Кількість індексованих сторінок - визначає його якість індексації. Індексація не залежить від вразливості, яка аналізується.

Якість фазингу. Процес введення вхідних даних для виявлення вразливості. Якість залежить від вхідних даних, які вводить фазер для пошуку певної вразливості.

Якість аналізу. Полягає в аналізі результатів, які генерує фазер. Повинен виявляти вразливості без генерації помилкових сигналів.

Автори оцінили шість сканерів з відкритим кодом: ZAP, Nikto, W3af, Wapiti, Arachni і BurpSuite з однаковими параметрами. У цьому дослідженні автори зосередилися на розрахунку хибно позитивних результатів (FP) і привели обмеження інструментів для тестування на проникнення:

- не можуть виявити всі вразливості такі як недоліки в шифруванні.
- складніше виявляти помилки управління доступом

Автоматизовані інструменти генерують помилкові позитиви та негативи (від 10 до 70%, в залежності від методології та продукту), що не відповідають достатнім вимогам безпеки. Тому автоматизовані інструменти не підходять для пенетраційного тестування логічних вразливостей системи.

У своїй статті автори проаналізували одинадцять інструментів оцінки веб-застосунків та оцінили їх з різними можливостями шляхом навмисного сканування веб-застосунків, таких як DVWA. Обрані інструменти оцінки веб-застосунків були топовими в списках сканерів за 2019 рік. Інструменти сканування веб-застосунків було оцінено за допомогою таких показників, як точність виявлення, точність, здатність їх виявляти різні вразливості. Згідно з оцінкою авторів, OWASP-ZAP має вищий рівень виявлення вразливостей в категорії відкритих джерел. Крім того, це дослідження надає порівняні результати виявлених різних вразливостей. Порівняльний аналіз буде проведено шляхом розробки плагінів та алгоритмів для відкритих інструментів оцінки вебу. Цей підхід допоможе збільшити покриття сканування за допомогою кращих функцій аутентифікації та допоможе зменшити кількість помилкових позитивів.

Моделювання та методи забезпечення кібербезпеки

У цій статті [14] автори досліджували поширення шкідливого програмного забезпечення (malware) в багатошарових мережах. Автори провели моделювання шкідливого програмного забезпечення в двошарових топологіях, які найбільш наближені до реальних кіберфізичних систем, адже вони поєднують різні типи складних мереж, що використовуються у нашому повсякденному житті. Автори використовували стохастичну модельну структуру, засновану на Марковських випадкових полях, для аналізу динаміки поширення шкідливого ПЗ у таких мережах. Поєднавши відбір Гіббса з імітаційним відпалом, була отримана довгострокову статистику поширення такого шкідливого ПЗ в розглянутих топологіях через очікувану кількість інфікованих вузлів у кожному сценарії. Автори прийшли до висновку, що чим щільніша мережа, тим більше гнучкості вона надає для зрештою пом'якшення шкідливого ПЗ. Щодо вразливості, найбільш надійні та найменш уразливі мережі спостерігалися в комбінаціях топологій REG-RG і RG-RG, тобто поєднання фізичної топології типу сітки або топології, яка могла б відповідати мережі однорангового обміну файлами, у поєднанні з топологією, яка відповідає випадковим користувачам. У майбутніх дослідженнях автори планують зосередитися на досліді з моделюванням “преференційного” шкідливе ПЗ і аналізі системи з більшою кількістю мережевих шарів.

Автори у своїй статті [15] вивчають проблему кібербезпеки промисловості, котра пов’язана з використання технології 5G. Авторами зосередили своє дослідження на розробці нової моделі для аналізу похибок основаної на графі залежностей (EDG). Ця модель цілісно може оцінювати помилки протягом усього життєвого циклу промислових компонентів. Модель була побудована в 5 кроків. Далі за допомогою набору метрик, система може покращити процес розробки. Модель надає інформацію про вразливості, котра має інформацію про місцезнаходження вразливості та інформацію про час де вона була виявлена, на якому етапі роботи системи. Також модель може визначати нові вимоги і генерувати тестові випадки для аналізу. В майбутніх дослідженнях автори планують додати математичні моделі для генерування значень метрик CVSS. Будуть внесені покращення в пріорітезацію виправлення з урахуванням контексту та функціональності система під тестуванням (System Under Test, SUT).

Багато середовищ моделювання ігнорують фактор соціальної інженерії. Автори [16] пропонують використовувати вдосконалену модель графа мережі для тестування на проникнення (NMPT), ця модель може краще описувати процес тестування на проникнення. Також у своїй роботі автори використовують SE-AIPT, для симуляції інтелектуального тестування на проникнення на основі соціальних інженерних факторів. Цей метод ефективно підвищує реалістичність середовища моделювання та має універсальність та розширюваність. На основі цих технологій автори побудували своє середовище симуляції котре непогано наближене до реальних атак.

У цій дисертації [17] автором було проаналізовано методи і засоби пошуку проблем кібербезпеки Web-застосунків. Було розроблено методи оцінювання та забезпечення кібербезпеки систем керування вмістом. Було одержано метод забезпечення кібербезпеки систем керування вмістом, який базується на виборі контрзаходів з врахуванням їх характеристик і сумісності. Автор вдосконалив метод оцінювання кібербезпеки систем керування вмістом шляхом використання ТА-АБО дерев аналізу атак.

Основні наукові положення дисертації було реалізовано у вигляді алгоритмів і програмних засобів.

Підсумовуючи, щоб забезпечити безпеку веб-додатків, IoT систем і т.д. необхідно розглядати різні види технологій (множина технологій), за допомогою яких створюють веб-додатки та IoT системи. Кожна система має свої вразливості (множину вразливостей), яким підтвердженні ті чи інші версії технологій. І підібрати найбільш оптимальні веб-сканери із множину веб-сканерів котрі з різним успіхом виявляють вразливості.

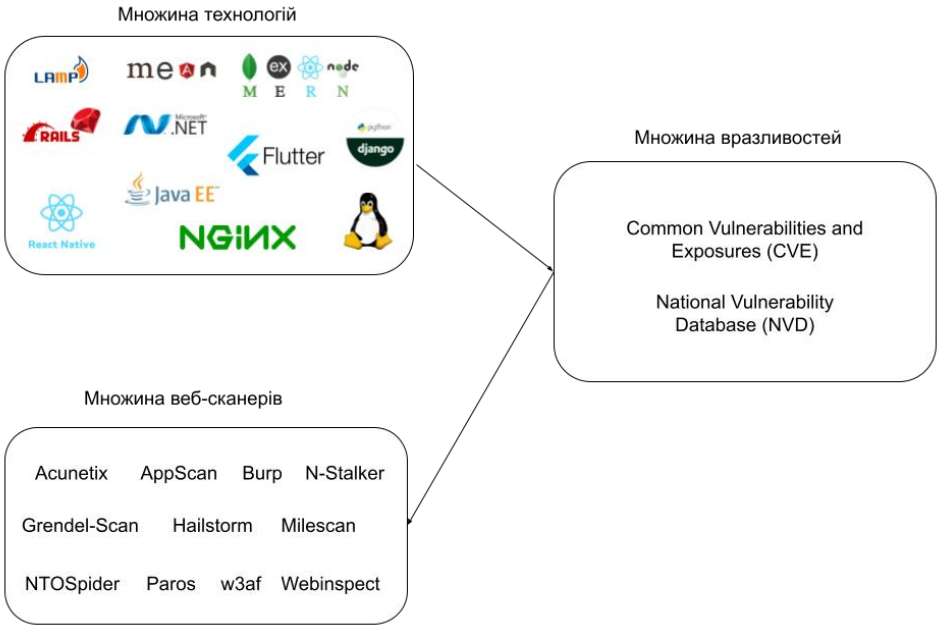

Рис. 4. Умовний взаємозв'язок між технологіями, вразливостями та веб-сканерами

Для створення алгоритму підбору сканерів, можна взяти аналітичний метод аналіз дерева атак і модифікувати його до поставлених цілей, а саме для підбору сканерів виявлення вразливостей. Метод дерев буде залежати від вхідних параметрів використаних технологій, а саме від обраного стеку технологій при створенні веб-додатку, від обладнання на якому буде розміщено веб-додаток (операційна система, технологій веб-сервера, стратегічного керівника відповідального за розгортання інфраструктури, тощо). Метою аналітичного метода дерева буде виступати оцінка, що буде базуватися на вірогідності виявлення веб-сканером тієї чи іншої веб-вразливості. В залежності від рівня до якого буде відноситись використана технологія (ОС, серверна частина, тощо), буде активуватися відповідна гілка дерева.

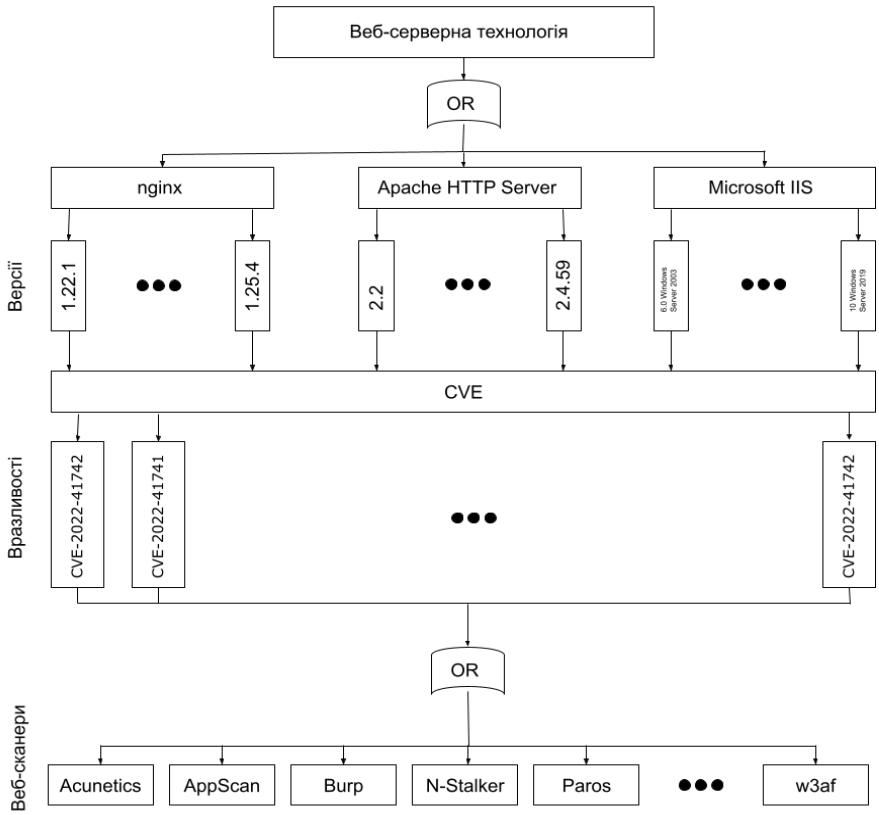

Рис. 5. Умовна активація гілки для технологій веб серверів

ВИСНОВКИ З ДАНОГО ДОСЛІДЖЕННЯ І ПЕРСПЕКТИВИ ПОДАЛЬШИХ РОЗВІДОК У ДАНОМУ НАПРЯМІ

Для майбутніх досліджень можна виділити такі теми як: модифікація дерева, доцільність використання штучного інтелекту.

Література

1. Russia’s Cyber Tactics: Lessons Learned 2022 – аналітичний звіт Держспецзв'язку про рік повномасштабної кібервійни росії проти України [reports] / State Service of Special Communications and Information Protection of Ukraine // 2022 . URL: https://cip.gov.ua/en/news/russia-s-cyber-tactics-lessons-learned-in-2022-ssscip-analytical-report-on-the-year-of-russia-s-full-scale-cyberwar-against-ukraine (date of access: 10.05.2024).
2. A Systematic Literature Review on Penetration Testing in Networks: Future Research Directions [Text] / Mariam Alhamed and M. M. Hafizur Rahman // Appl. Sci. 2023, 13(12), 6986. DOI: https://doi.org/10.3390/app13126986
3. Research on Security Weakness Using Penetration Testing in a Distributed Firewall [Text] / Andrei-Daniel Tudosi, Adrian Graur, Doru Gabriel Balan and Alin Dan Potorac // Sensors 2023, 23(5), 2683. DOI: https://doi.org/10.3390/s23052683
4. Deep Reinforcement Learning for Intelligent Penetration Testing Path Design [Text] / Junkai Yi and Xiaoyan Liu // Appl. Sci. 2023, 13(16), 9467. DOI: https://doi.org/10.3390/app13169467
5. Scanning for Vulnerable Devices in the Internet of Things [Text] / Linda Markowsky and George Markowsky // 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems 2015. Warsaw Poland Volume: 1. DOI: https://doi.org/10.1109/IDAACS.2015.7340779
6. IoT Vulnerability Scanning: A State of the Art [Text] / Ahmed Amro // Computer Security, 84-99. DOI: https://doi.org/10.1007/978-3-030-64330-0_6
7. Analysis of Consumer IoT Device Vulnerability Quantification Frameworks [Text] / Samira A. Baho and Jemal Abawajy // Electronics 2023, 12(5), 1176. DOI: https://doi.org/10.3390/electronics12051176
8. Automated Context-Aware Vulnerability Risk Management for Patch Prioritization [Text] / Vida Ahmadi Mehri, Patrik Arlos and Emiliano Casalicchio // Electronics 2022, 11(21), 3580. DOI: https://doi.org/10.3390/electronics11213580
9. Support for the Vulnerability Management Process Using Conversion CVSS Base Score 2.0 to 3.x [Text] / Maciej Roman Nowak, Michał Walkowski and Sławomir Sujecki // Sensors 2023, 23(4), 1802. DOI: https://doi.org/10.3390/s23041802
10. An Opcode-Based Vulnerability Detection of Smart Contracts [Text] / Jia Sui, Lili Chu and Han Bao // Appl. Sci. 2023, 13(13), 7721. DOI: https://doi.org/10.3390/app13137721
11. Why Johnny Can’t Pentest: An Analysis of Black-box Web Vulnerability Scanners [Text] / Adam Doupe, Marco Cova, and Giovanni Vigna // University of California, Santa Barbara DBLP - 2010. DOI: https://doi.org/10.3390/s23041802
12. Evaluation of Black-Box Web Application Security Scanners in Detecting Injection Vulnerabilities [Text] / Muzun Althunayyan, Neetesh Saxena, Shancang Li and Prosanta Gope // Electronics 2022, 11(13). DOI: https://doi.org/10.3390/electronics11132049
13. A Comparative Study of Web Application Security Parameters: Current Trends and Future Directions [Text] / Jahanzeb Shahid, Muhammad Khurram Hameed, Ibrahim Tariq Javed, Kashif Naseer Qureshi, Moazam Ali and Noel Crespi // Appl. Sci. 2022, 12(8), 4077. DOI: https://doi.org/10.3390/app12084077
14. Markov-Based Malware Propagation Modeling and Analysis in Multi-Layer Networks [Text] / Stavros Karageorgiou and Vasileios Karyotis // Network 2022, 2(3), 456-478. DOI: https://doi.org/10.3390/network2030028
15. A Novel Model for Vulnerability Analysis through Enhanced Directed Graphs and Quantitative Metrics [Text] / Ángel Longueira-Romero, Rosa Iglesias, Jose Luis Flores and Iñaki Garitano // Sensors 2022, 22(6), 2126. DOI: https://doi.org/10.3390/s22062126
16. An Intelligent Penetration Test Simulation Environment Construction Method Incorporating Social Engineering Factors [Text] / Yang Li, Yongjie Wang, Xinli Xiong, Jingye Zhang and Qian Yao // Appl. Sci. 2022, 12(12), 6186. DOI: https://doi.org/10.3390/app12126186
17. Методи інформаційної технології забезпечення кібербезпеки систем керування вмістом при створенні web-застосунків [Дисертація] / Тецький А. Г. // Харків 2019 - 187с.

References

1. Russia’s Cyber Tactics: Lessons Learned 2022 – analytical report by the State Service of Special Communications and Information Protection of Ukraine on a year of full-scale cyberwar by Russia against Ukraine [reports] / State Service of Special Communications and Information Protection of Ukraine // 2022. URL: https://cip.gov.ua/en/news/russia-s-cyber-tactics-lessons-learned-in-2022-ssscip-analytical-report-on-the-year-of-russia-s-full-scale-cyberwar-against-ukraine (date of access: 10.05.2024).

2. A Systematic Literature Review on Penetration Testing in Networks: Future Research Directions [Text] / Mariam Alhamed and M. M. Hafizur Rahman // Appl. Sci. 2023, 13(12), 6986. DOI: https://doi.org/10.3390/app13126986
3. Research on Security Weakness Using Penetration Testing in a Distributed Firewall [Text] / Andrei-Daniel Tudosi, Adrian Graur, Doru Gabriel Balan and Alin Dan Potorac // Sensors 2023, 23(5), 2683. DOI: https://doi.org/10.3390/s23052683
4. Deep Reinforcement Learning for Intelligent Penetration Testing Path Design [Text] / Junkai Yi and Xiaoyan Liu // Appl. Sci. 2023, 13(16), 9467. DOI: https://doi.org/10.3390/app13169467
5. Scanning for Vulnerable Devices in the Internet of Things [Text] / Linda Markowsky and George Markowsky // 8th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems 2015. Warsaw Poland Volume: 1. DOI: https://doi.org/10.1109/IDAACS.2015.7340779
6. IoT Vulnerability Scanning: A State of the Art [Text] / Ahmed Amro // Computer Security, 84-99. DOI: https://doi.org/10.1007/978-3-030-64330-0_6
7. Analysis of Consumer IoT Device Vulnerability Quantification Frameworks [Text] / Samira A. Baho and Jemal Abawajy // Electronics 2023, 12(5), 1176. DOI: https://doi.org/10.3390/electronics12051176
8. Automated Context-Aware Vulnerability Risk Management for Patch Prioritization [Text] / Vida Ahmadi Mehri, Patrik Arlos and Emiliano Casalicchio // Electronics 2022, 11(21), 3580. DOI: https://doi.org/10.3390/electronics11213580
9. Support for the Vulnerability Management Process Using Conversion CVSS Base Score 2.0 to 3.x [Text] / Maciej Roman Nowak, Michał Walkowski and Sławomir Sujecki // Sensors 2023, 23(4), 1802. DOI: https://doi.org/10.3390/s23041802
10. An Opcode-Based Vulnerability Detection of Smart Contracts [Text] / Jia Sui, Lili Chu and Han Bao // Appl. Sci. 2023, 13(13), 7721. DOI: https://doi.org/10.3390/app13137721
11. Why Johnny Can't Pentest: An Analysis of Black-box Web Vulnerability Scanners [Text] / Adam Doupe, Marco Cova, and Giovanni Vigna // University of California, Santa Barbara DBLP - 2010. DOI: https://doi.org/10.3390/s23041802
12. Evaluation of Black-Box Web Application Security Scanners in Detecting Injection Vulnerabilities [Text] / Muzun Althunayyan, Neetesh Saxena, Shancang Li and Prosanta Gope // Electronics 2022, 11(13). DOI: https://doi.org/10.3390/electronics11132049
13. A Comparative Study of Web Application Security Parameters: Current Trends and Future Directions [Text] / Jahanzeb Shahid, Muhammad Khurram Hameed, Ibrahim Tariq Javed, Kashif Naseer Qureshi, Moazam Ali and Noel Crespi // Appl. Sci. 2022, 12(8), 4077. DOI: https://doi.org/10.3390/app12084077
14. Markov-Based Malware Propagation Modeling and Analysis in Multi-Layer Networks [Text] / Stavros Karageorgiou and Vasileios Karyotis // Network 2022, 2(3), 456-478. DOI: https://doi.org/10.3390/network2030028
15. A Novel Model for Vulnerability Analysis through Enhanced Directed Graphs and Quantitative Metrics [Text] / Ángel Longueira-Romero, Rosa Iglesias, Jose Luis Flores and Iñaki Garitano // Sensors 2022, 22(6), 2126. DOI: https://doi.org/10.3390/s22062126
16. An Intelligent Penetration Test Simulation Environment Construction Method Incorporating Social Engineering Factors [Text] / Yang Li, Yongjie Wang, Xinli Xiong, Jingye Zhang and Qian Yao // Appl. Sci. 2022, 12(12), 6186. DOI: https://doi.org/10.3390/app12126186
17. Methods of Information Technology for Ensuring Cybersecurity of Content Management Systems in the Creation of Web Applications [Dissertation] / Tetskyi A. H. // Kharkiv 2019 - 187 pages.