Державного університету інформаційно-комунікаційних технологій.

Maksym Fesenko, Ph.D., Associate Professor, Associate Professor of the Department of artificial intelligence of the State University of Information and Communication Technologies.
E-mail: fesenkomaksim81@gmail.com.
Orcid ID: 0000-0001-8218-4154.

Вишнівський Віктор Вікторович, доктор технічних наук, професор, завідувач кафедри комп'ютерних наук Державного університету інформаційно-комунікаційних технологій.

Viktor Vyshnivskyi, Dc.S, Professor, Head of the Department of Computer Science of the State University of Information and Communication Technologies.
E-mail: vyshnivskyi.viktor@gmail.com.
Orcid ID: 0000-0003-1923-4344.

DOI: 10.18372/2410-7840.26.18833
УДК 004.056.5

РОЗРОБКА ЕФЕКТИВНИХ ЗАХОДІВ ВЕБ-БЕЗПЕКИ ДЛЯ МЕРЕЖІ ШЛЯХОМ ПРОВЕДЕННЯ ТЕСТУВАННЯ НА ПРОНИКНЕННЯ З ВИКОРИСТАННЯМ ФРЕЙМВОРКУ OWASP

Марія Козловська, Андріян Піскозуб

З кожним кроком у розвитку технологій, веб-безпека стає більш актуальною складовою для забезпечення надійності та захисту мережевих систем. Зростання кількості кіберзагроз та потенційних порушень безпеки підкреслює необхідність удосконалення заходів захисту мережевих систем. Щоб допомогти розробникам та адміністраторам у цьому процесі, існує важливий інструмент – фреймворк OWASP (Open Web Application Security Project). Він надає широкий спектр інструментів, рекомендацій та ресурсів для забезпечення безпеки веб-додатків. Цей фреймворк допомагає розробникам перевірити веб-додатки на наявність потенційних вразливостей та знайти способи їх виправлення. Для кращого розуміння, можна уявити, що мережа – це будинок, а веб-додатки - це його двері та вікна. Якщо ці двері та вікна не міцно закриті, зловмисники можуть легко проникнути всередину та завдати шкоди. Отож для порівняння, можна сказати, що так само, як ви перевіряєте чи всі двері та вікна в мережі надійно захищені, OWASP надає засоби для перевірки веб-додатків на вразливості, що можуть бути використані зловмисниками для атак. Отже, використання фреймворку OWASP є важливим кроком у розробці ефективних заходів веб-безпеки для мережі, допомагає забезпечити надійність та захист системи від можливих кібератак та зловмисних дій.

***Ключові слова**: фреймворк OWASP, веб-безпека, тестування на проникнення, захист даних, кіберзахист.*

ВСТУП

У сучасну цифрову епоху, коли інформація є джерелом життя бізнесу, забезпечення безпеки корпоративних мереж стало першочерговим завданням. Організації значною мірою покладаються на використання продуктів і послуг інформаційних технологій у своїй повсякденній діяльності, тому захист цих активів має вирішальне значення для їхнього успіху та загального виживання в умовах жорсткої конкуренції. Інформаційна безпека - це захист конфіденційних даних від несанкціонованого доступу, використання, розкриття, порушення, модифікації або знищення. Зважаючи на зростаючу складність і частоту кібератак, організаціям вкрай необхідно впроваджувати надійні заходи веб-безпеки для захисту своїх мереж і даних.

Оскільки Інтернет продовжує ставати повсюдним засобом комунікації та комерції, компанії все частіше ведуть бізнес онлайн. Однак зручність та ефективність цифрового ринку може бути затьмарена зростаючими ризиками безпеки, які супроводжують онлайн-транзакції та зберігання даних. Численні переваги ведення бізнесу в Інтернеті супроводжуються складним завданням навігації в середовищі, насиченому потенційними загрозами, такими як системна корупція, шахрайство, крадіжки та віруси. Кіберзлочини призводять не лише до прямих фінансових втрат, вони також підривають довіру клієнтів і шкодять репутації бізнесу, що потенційно може призвести до довгострокових збитків.

Тому для організацій дуже важливо оцінювати та посилювати безпеку своїх мережевих систем. У цьому ландшафті цифрових загроз фреймворк OWASP стає основою безпеки вебдодатків, забезпечуючи важливий фундамент для розробки безпечних систем. Після ретельного тестування на проникнення ці стратегії утворюють надійний захист від різноманітних кіберзагроз, що дозволяє забезпечити безпеку системи. Конвергенція надійних фреймворків з комплекс-

ними методами тестування дозволяє компаніям розробляти і підтримувати безпечне середовище, яке може адаптуватися до постійно мінливого характеру кіберзагроз [1].

АНАЛІЗ ОСТАННІХ ДОСЛІДЖЕНЬ ТА ПУБЛІКАЦІЙ

Нещодавні дослідження та публікації в галузі веб-безпеки підкреслили зростаючу складність кіберзагроз і необхідність для організацій впроваджувати надійні заходи безпеки. Одна з важливих публікацій, , підготовлена групою експертів з безпеки, містить всебічний аналіз останніх тенденцій щодо вразливостей веб-додатків і пропонує рекомендації щодо зниження цих ризиків за допомогою фреймворку OWASP [2]. Важливим аспектом посилення веб-безпеки є проведення тестування на проникнення, яке дозволяє виявити потенційні вразливості та слабкі місця в мережі. Такий проактивний підхід дозволяє організаціям усунути прогалини в безпеці до того, як ними скористаються зловмисники. Використовуючи фреймворк OWASP у поєднанні з тестуванням на проникнення, організації можуть отримати повне уявлення про стан безпеки своїх веб-додатків і визначити пріоритети для усунення недоліків. Крім того, інтеграція найкращих практик і засобів контролю безпеки, рекомендованих OWASP, може значно підвищити стійкість мережі до нових кіберзагроз. Концепція OWASP є широко визнаним і прийнятим галузевим стандартом, який надає рекомендації щодо безпеки веб-додатків.

Крім того, дослідження, опубліковане в статті [3], заглиблюється в ефективність тестування на проникнення у виявленні та усуненні прогалин у безпеці. У дослідженні наведено приклади організацій, які успішно використовували тестування на проникнення для виявлення критичних вразливостей та проактивного зміцнення своєї веб-безпеки. Висновки цього дослідження підкреслюють важливість регулярного проведення тестування на проникнення як частини комплексної стратегії веб-безпеки.

В статті [4] розглядаються нові найкращі практики захисту веб-додатків в умовах еволюції кіберзагроз. У документі наводяться конкретні приклади успішних впроваджень безпеки на основі фреймворку OWASP та методологій тестування на проникнення, що дає цінну інформацію для організацій, які прагнуть посилити свою позицію у сфері веб-безпеки.

В статті [5] розглядаються аспекти веб-безпеки та піднімається питання сучасних тенденцій в області пентестування з аналізом переваг і недоліків ручного та автоматизованого тестування на проникнення. Сучасна тенденція, відзначена авторами, полягає у поєднанні обох методів - ручного та автоматизованого тестування. Такий підхід дозволяє скористатися перевагами обох методів, компенсуючи їхні недоліки. Наприклад, ручне тестування може бути застосоване для виявлення складних вразливостей, в той час як автоматизовані засоби можуть використовуватися для швидкого сканування та виявлення типових проблем.

Результати останніх досліджень і публікацій підкреслюють важливість того факту, що слід бути в курсі останніх подій у сфері веб-безпеки і використовувати перевірені фреймворки і методології, щоб захиститися від нових кіберзагроз.

ПОСТАНОВКА ЗАВДАННЯ

Основною проблемою, з якою сьогодні стикається сфера кібербезпеки, є невідповідність між швидкозмінним ландшафтом загроз і статичністю багатьох існуючих заходів безпеки. Аналіз останніх досліджень і публікацій показує, що, незважаючи на усвідомлення критичної ролі кібербезпеки, багато організацій залишаються недостатньо підготовленими до того, щоб впоратися з витонченістю і частотою нових кіберзагроз. Оскільки кіберзагрози стають все більш витонченими і поширеними, організації повинні визначити веб-безпеку як основний компонент своєї загальної стратегії управління ризиками.

Атаки вже не є поодинокими інцидентами, а є частиною постійної і динамічної стратегії кібервійни, яка безперервно націлена на вразливі місця в інформаційних системах. Це призводить до важливого питання: як організації можуть вийти за рамки традиційних, реактивних підходів і розробити проактивні, динамічні стратегії кібербезпеки, що безперервно розвиваються?

Щоб вирішити критичну проблему швидкої зміни ландшафту загроз і статичності існуючих заходів безпеки, організаціям необхідно надати пріоритет проактивному і динамічному підходу до кібербезпеки. Вихід за рамки традиційних, реактивних підходів передбачає прийняття підходу до безпеки, заснованого на оцінці ризиків, що вимагає постійної оцінки систем і даних, виявлення вразливостей і впровадження заходів для зниження ризиків.

Інвестиції в сучасні засоби виявлення загроз і реагування на них, такі як системи безпеки на основі машинного навчання, мають важливе значення для того, щоб не відставати від кіберзагроз,

які постійно змінюються. Ці системи здатні адаптуватися до нових і нових загроз у режимі реального часу, забезпечуючи організаціям більш проактивну стратегію захисту.

Крім того, співпраця та обмін інформацією між організаціями та фахівцями з кібербезпеки відіграють вирішальну роль у подоланні динамічного характеру кіберзагроз. Обмінюючись найкращими практиками, організації можуть спільно випереджати нові загрози та зміцнювати свою загальну систему кібербезпеки.

Впроваджуючи ці проактивні заходи та постійно розвиваючи свої стратегії кібербезпеки, організації можуть краще захиститися від швидкозмінного ландшафту кіберзагроз та мінімізувати наслідки кібератак.

Метою даної статті є дослідження та акцентування важливості впровадження комплексних заходів веб-безпеки в корпоративних мережах для захисту від зростаючої складності та частоти кібератак. Завдяки такому комплексному підходу є можливість забезпечити ефективні проактивні та адаптивні заходи безпеки, вирішуючи виклик індустрії кібербезпеки, пов'язаний з еволюцією загроз. У статті детально розглядається фреймворк OWASP як наріжний камінь для розробки безпечних веб-додатків у поєднанні з основними методами тестування на проникнення для виявлення та усунення вразливостей. Це має допомогти організаціям вийти за рамки традиційних, реактивних заходів безпеки і перейти до динамічних стратегій кібербезпеки, що постійно розвиваються, підкреслюючи важливість досліджень для зміцнення цифрової інфраструктури і забезпечення безперервності та надійності сучасних бізнес-операцій.

ОСНОВНА ЧАСТИНА

Тестування на проникнення у веб-безпеці

Тестування на проникнення, також відоме як етичний хакінг, відіграє життєво важливу роль у виявленні та виправленні вразливостей у веб-додатках. Тестування на проникнення передбачає імітацію реальної атаки на мережу або додаток для виявлення потенційних слабких місць і вразливостей.

Існує кілька типів тестування на проникнення, включаючи тестування методом "чорного ящика", "білого ящика" і "сірого ящика" [8]:

1. Тестування методом чорного ящика (Black Box Testing):

• опис: виконується без попереднього знання про внутрішню структуру або реалізацію системи;

• підходи: тестувальник зосереджується на зовнішньому аналізі, використовуючи доступні публічні дані;

• мета: виявлення вразливостей, які можуть бути використані зовнішнім зловмисником;

2. Тестування методом білого ящика (White Box Testing):

• опис: проводиться з повним доступом до інформації про систему, включаючи вихідний код, архітектуру та іншу внутрішню інформацію;

• підходи: тестувальник має можливість проводити детальний аналіз внутрішніх механізмів системи;

• мета: виявлення вразливостей на рівні реалізації, оптимізація або розробка безпечних практик програмування;

3. Тестування методом сірого ящика (Grey Box Testing):

• опис: комбінує елементи обох підходів, де тестувальник має часткові знання про систему;

• підходи: тестувальник має обмежений доступ до внутрішніх деталей системи;

• мета: поєднання переваг кількох підходів для максимально ефективного виявлення вразливостей.

Під час процесу тестування на проникнення тестувальники використовують різні методи та інструменти для виявлення вразливостей. Ці методи можуть включати сканування мережі, сканування вразливостей, злам паролів, соціальну інженерію та використання відомих вразливостей. Після виявлення вразливостей тестувальник тісно співпрацює з організацією, щоб виправити ці проблеми та посилити загальний рівень безпеки мережі.

Після того, як тестувальник виявив вразливості і повідомив про них, дуже важливо, щоб організації вжили відповідних заходів для вирішення цих проблем.

Кроки для успішного тестування на проникнення, згідно зі Стандартом проведення тестування на проникнення [9], є наступними:

1. Взаємодія перед початком роботи (Pre-engagement interactions): на цьому етапі створюється підґрунтя для успішного проведення тесту на проникнення. Зацікавлені сторони узгоджують обсяг, цілі, терміни і протоколи комунікації. Цей етап планування часто включає огляд попередніх тестів на проникнення і збір будь-якої відповідної документації, яка може підтримати процес;

2. Збір розвідувальної інформації (Intelligence gathering): тестувальники беруть участь в активній і пасивній розвідці для збору інформації про ці-

льову систему. Вони використовують інструменти для сканування мережі і досліджують публічні записи або аномалії в цифрових слідах, які можуть виявити вразливості. Для виявлення методів обробки даних, які можуть бути використані, також застосовуються офлайн-методи розвідки;

Рис. 1. Кроки для успішного тестування на проникнення

3. Моделювання загроз (Threat modeling): маючи чітке розуміння активів системи та профілів потенційних зловмисників, формулюється модель загроз. Вона вказує на ймовірні цілі, вектори атаки та пріоритети безпеки. Документація тут має вирішальне значення для забезпечення прозорості та повторюваності основи оцінки;

4. Аналіз вразливостей (Vulnerability analysis): зазвичай це передбачає використання автоматизованих інструментів, таких як сканери вразливостей, для виявлення та каталогізації потенційних слабких місць у захисті мережі або системи. Для виявлення вразливостей сканери можуть використовувати комбінацію динамічного аналізу (тестування програмного забезпечення шляхом виконання та моніторингу його поведінки) або статичного аналізу (вивчення коду без виконання програми). Пентестери аналізують архітектурні, процедурні та реалізаційні аспекти з метою виявлення прогалин у безпеці. Вони оцінюють серйозність кожної вразливості та підтверджують ризик, пов'язаний з нею. Як результат, надається докладна оцінка вразливостей, підтверджена глибоким дослідженням;

5. Експлоітація (Exploitation): на даному етапі тестувальники намагаються використати виявлені вразливості для отримання несанкціонованого доступу до систем, підвищення привілеїв або проведення несанкціонованих дій в мережі.

Якщо на попередньому етапі аналіз вразливості було виконано належним чином, цей етап має бути добре спланованим на основі сформованого на попередньому етапі цільового списку сервісів, що дає можливість визначити основну точку входу в організацію та цінні цільові активи.

Мета цього етапу – продемонструвати, чого може досягти зловмисник, якщо скористається вразливими місцями, виявленими під час оцінки вразливостей. Результати спроб експлоітації мають вирішальне значення, оскільки вони надають чіткі докази потенційних порушень безпеки та допомагають визначити пріоритетність зусиль з усунення недоліків;

6. Пост-експлоітація (Post-exploitation): пост-експлоітація є важливим етапом тестування на проникнення, який спрямований на ретельну оцінку можливостей і вразливостей цільової системи. Основна мета - отримати необмежений доступ до всіх аспектів цільової системи, не викликаючи жодних тривог і не будучи виявленим заходами безпеки. Виявлення зловмисника захисниками системи-мішені зробить зусилля зловмисника неефективними і зведе нанівець будь-який потенційний вплив.

На етапі пост-експлоітації пентестер використовує вразливості цільової системи, не вимагаючи автентифікації, що дозволяє йому проаналізувати цінність даних, присутніх в системі жертви. Якщо ці дані будуть визнані цінними, пентестер може заглибитися глибше, щоб отримати додаткову інформацію про цільову систему. Крім того, він вивчає різні аспекти конфігурації системи, протоколи зв'язку, налаштування реєстру та методи мережевого підключення, щоб зрозуміти архітектуру системи та її потенційні слабкі місця.

Важливо зазначити, що методи та вимоги до пост-експлоітації можуть відрізнятися залежно від конкретних обставин. Дотримання етичних принципів і правових міркувань має першорядне значення в процесі тестування на проникнення, щоб гарантувати, що тест буде проведений відповідально і етично;

7. Звітність (Reporting): на фінальному етапі складається комплексний звіт, в якому детально описується обсяг, використані методи, знайдені вразливості та рекомендації щодо їх усунення. Звіт є важливим документом для розуміння зацікавленими сторонами стану захищеності їхньої системи і слугує дорожньою картою для вирішення проблем безпеки.

Зі стратегічної точки зору, роль тестування на проникнення у веб-безпеці полягає у забезпе-

ченні проактивної позиції щодо загроз. Виявляючи та усуваючи вразливості до того, як вони можуть бути використані, організації можуть запобігти потенційним інцидентам безпеки. Це також допомагає підтримувати довіру з користувачами та зацікавленими сторонами, демонструючи належну ретельність у захисті конфіденційних даних.

Використання фреймворку OWASP для розробки заходів веб-безпеки

Фреймворк OWASP (Open Web Application Security Project) – це широка і всесвітньо визнана платформа, призначена для посилення безпеки веб-додатків. Вона слугує централізованим центром для безлічі ресурсів, інструментів, найкращих практик та спільних ініціатив, спрямованих на вирішення багатогранних проблем, пов'язаних із загрозами кібербезпеки у сфері розробки та розгортання веб-додатків.

В основі OWASP лежать принципи відкритості, прозорості та співпраці, керованої спільнотою. Даний фреймворк використовує колективний досвід і внесок фахівців з кібербезпеки, розробників, дослідників та ентузіастів з усього світу для створення всеосяжного сховища знань і ресурсів.

OWASP проводить різні проекти, спрямовані на покращення різних аспектів безпеки. Ці проекти часто включають інструменти та ресурси, які можна використовувати для підвищення мережевої безпеки:

• OWASP Top 10 [10] надає пріоритетний список найбільш критичних ризиків безпеки, з якими стикаються веб-додатки. Хоча він в першу чергу зосереджений на безпеці веб-додатків, багато з перерахованих вразливостей також можуть вплинути на мережеву безпеку. Розуміючи і усуваючи ці вразливості, організації можуть підвищити загальний рівень мережевої безпеки;

• OWASP ZAP (Zed Attack Proxy) [11] можна використовувати для сканування і тестування безпеки мережевих додатків, API і сервісів для виявлення вразливостей і неправильних конфігурацій безпеки;

• OWASP ModSecurity Core Rule Set (CRS)[12] надає набір правил для брандмауера веб-додатків ModSecurity (WAF) для захисту веб-додатків та API від поширених атак;

• OWASP Cheat Sheets [13] містять стислі практичні рекомендації щодо впровадження безпечних методів кодування, усунення поширених вразливостей безпеки та дотримання найкращих практик у сфері безпеки веб-додатків. Вони слугують безцінним ресурсом для розробників, пропонуючи короткі довідники для безпечного проектування, розробки та розгортання веб-додатків;

• OWASP Documentation: OWASP пропонує обширну документацію [14], що охоплює різні аспекти безпеки веб-додатків, включаючи безпечну мережеву архітектуру, моделювання загроз і методології тестування безпеки. Використовуючи ці ресурси, організації можуть отримати уявлення про найкращі практики проектування та впровадження безпечних мережевих архітектур і протоколів;

• OWASP Community: спільнота OWASP відіграє ключову роль у просуванні місії організації. Вона складається з волонтерів, учасників, локальних відділень та робочих груп, що спрямовані на розвиток та адаптацію OWASP до сучасних викликів у галузі кібербезпеки, завдяки активному обміну знаннями та спільним зусиллям.

Загалом, фреймворк OWASP слугує незамінним ресурсом для організацій та приватних осіб, які прагнуть створювати, розгортати та підтримувати безпечні веб-додатки. Використовуючи багатство ресурсів і досвіду, пропонованих OWASP, зацікавлені сторони можуть бути в курсі нових загроз, переймати кращі практики і зміцнювати свій захист від кібератак [15].

Практичне застосування інноваційних підходів до безпеки додатків OWASP

Зі збільшенням частоти кібератак і витоків даних організаціям вкрай важливо впроваджувати інноваційні підходи до безпеки додатків, щоб захистити свою конфіденційну інформацію і зберегти довіру користувачів. Ось кілька прикладів того, як ці інноваційні підходи можуть бути застосовані на практиці:

• використання машинного навчання та штучного інтелекту: практичне застосування машинного навчання та штучного інтелекту включає реалізацію систем аналізу поведінки користувачів та трафіку додатків з метою виявлення аномальних дій, які можуть свідчити про потенційні загрози [16]. Наприклад, можна використовувати алгоритми аналізу журналів доступу, щоб виявити незвичні підключення або запити, що можуть свідчити про атаки. Додатково, можна реалізувати системи детекції аномалій, які аналізують зміни у поведінці користувачів та системи для виявлення вразливостей та потенційних атак. Тестування на проникнення традиційно протистоїть автоматизації через великий досвід, необхідний професіоналам. Великі мовні моделі (Large Language Models (LLM)) показали значний прогрес у різних

сферах, і їхні нові можливості свідчать про їхній потенціал автоматизувати галузь пентесту [17]. У цьому дослідженні автори оцінили продуктивність LLM у реальних завданнях тестування на проникнення, зазначивши, що хоча LLM демонструють майстерність у конкретних підзавданнях у процесі тестування на проникнення, таких як використання інструментів тестування, інтерпретація результатів і пропонування наступних дій, вони також стикаються з труднощами підтримання інтегрованого розуміння загального сценарію тестування. Автори запропонували PentestGPT – інструмент автоматичного тестування на проникнення на базі LLM, довів свою ефективність у вирішенні завдань тестування на проникнення в реальному світі;

• застосування блокчейну для захисту даних: практичне застосування технології блокчейну для захисту даних може включати створення розподіленої системи зберігання даних, де інформація зберігається у вигляді блоків, які підтверджуються мережею вузлів [18]. Це забезпечує надійність даних, оскільки будь-які спроби маніпулювання або зміни даних будуть виявлені мережею. Наприклад, можна застосувати технологію блокчейну для зберігання основних даних аутентифікації користувачів або даних про транзакції в фінансових додатках;

• реалізація багаторівневого автентифікаційного підходу: практичне впровадження багаторівневого підходу до автентифікації може включати використання різних методів ідентифікації та автентифікації користувачів на різних рівнях доступу до системи [19]. Наприклад, використання двофакторної автентифікації для доступу до важливих ресурсів або використання біометричних даних для автентифікації в мобільних додатках. Кожен рівень автентифікації може мати свої власні механізми захисту, що зробить систему стійкішою до атак;

• використання контейнеризації та мікросервісів: практичне застосування контейнеризації та мікросервісів може включати розгортання окремих компонентів додатку в контейнерах з мінімальною кількістю привілеїв доступу [20]. Це дозволяє ізолювати окремі частини додатку та обмежувати їх вплив на інші компоненти системи. Наприклад, можна використовувати контейнери для розгортання мікросервісів, які відповідають за різні функціональні аспекти додатку (наприклад, автентифікація, авторизація, зберігання даних), що дозволяє ефективно управляти ізоляцією та безпекою кожного сервісу;

• використання систем Continuous Integration/Continuous Deployment (CI/CD): практичне застосування CI/CD включає автоматизацію процесів тестування та розгортання додатків для швидкого виявлення та усунення вразливостей. Наприклад, можна використовувати системи автоматичного тестування, які перевіряють код на наявність вразливостей та автоматично створюють звіти з результатами тестів [21]. Після цього, процес CI/CD автоматично розгортає безпечні версії додатків у виробниче середовище, що дозволяє швидко реагувати на потенційні загрози та мінімізує час простою системи.

Ці інноваційні підходи допомагають організаціям забезпечити ефективний захист своїх вебдодатків від сучасних кіберзагроз та зберегти довіру користувачів. Використання цих методів дозволяє підвищити безпеку та стійкість додатків до потенційних атак і зберегти конфіденційність даних користувачів.

Ручне та автоматизоване тестування на проникнення: порівняльний аналіз

Ручне та автоматизоване тестування на проникнення – це дві основні стратегії в області тестування безпеки програмного забезпечення. Ручне тестування на проникнення включає в себе активну участь людських експертів, які використовують свої знання та інтуїцію для виявлення потенційних вразливостей у програмному забезпеченні. Цей підхід дозволяє отримати глибше розуміння контексту програми та виявити складні вразливості, які можуть залишитися непоміченими автоматизованими інструментами. Однак ручне тестування може бути витратним за часом і засобами, особливо при великих проектах.

Автоматизоване тестування на проникнення використовує спеціальні програмні засоби для автоматизації процесу виявлення вразливостей у програмному забезпеченні. Цей підхід зазвичай швидший та більш ефективний для виявлення широкого спектру вразливостей та виконання повторюваних тестів. Однак він може не виявляти складні вразливості, які потребують аналізу експертів, та може давати помилкові сигнали [22].

Ось практичні приклади автоматизованих інструментів для тестування на проникнення для кожного типу тестування:

1. Unit Testing (Модульне тестування) [23]:

• інструмент: JUnit або NUnit для Java та .NET відповідно;

• приклад: написання модульних тестів для окремих функцій або методів, щоб переконатися, що вони працюють належним чином;

2. Smoke Testing (Димове тестування) [24]:

• інструмент: Selenium для тестування веб-додатків;

• приклад: автоматизація процесу входу в систему та перевірка базової функціональності, наприклад, навігації по різних сторінках;

3. Integration Testing (Інтеграційне тестування) [25]:

• інструмент: Postman для тестування API;

• приклад: тестування інтеграції різних мікросервісів шляхом надсилання запитів до API та перевірки відповідей;

4. Regression Testing (Регресійне тестування) [26]:

• інструмент: Selenium або Appium для тестування інтерфейсу користувача;

• приклад: автоматизація тестування критично важливих робочих процесів після кожної нової зміни коду, щоб переконатися, що існуюча функціональність залишається недоторканою;

5. API Testing (Тестування API) [27]:

• інструмент: OWASP ZAP (Zed Attack Proxy) для тестування безпеки API;

• приклад: запуск сканування безпеки API для виявлення вразливостей, таких як недоліки ін'єкцій або небезпечні методи автентифікації;

6. Security Testing (Тестування безпеки) [28]:

• інструмент: Nessus або OpenVAS для сканування вразливостей;

• приклад: сканування мережі або веб-додатку на наявність відомих вразливостей і неправильних конфігурацій;

7. Performance Testing (Тестування продуктивності) [29]:

• інструмент: Apache JMeter для тестування навантаження;

• приклад: імітація тисяч одночасних користувачів для вимірювання продуктивності веб-додатку під великим навантаженням;

8. Acceptance Testing (Приймальне тестування) [30]:

• інструмент: Cucumber або Behave для поведінково-орієнтованої розробки;

• приклад: написання критеріїв прийнятності в синтаксисі Gherkin і автоматизація виконання цих тестів для перевірки відповідності програмного забезпечення заданим вимогам;

9. Dynamic Application Security Testing (DAST) (Динамічне тестування безпеки додатків) [31]:

• інструмент: OWASP ZAP або Burp Suite для динамічного аналізу веб-додатків;

• приклад: сканування веб-додатку, виявлення всіх доступних кінцевих точок і виконання тестів безпеки для пошуку вразливостей, таких як XSS або SQL-ін'єкції.

Наведено порівняльний аналіз ручного та автоматизованого тестування на проникнення (табл. 1).

Таблиця 1

Порівняльний аналіз ручного та автоматизованого тестування на проникнення

Характеристика	**Ручне тестування на проникнення**	**Автоматизоване тестування на проникнення**
Залучення людини	Потребує висококваліфікованих спеціалістів з безпеки для ручного аналізу систем, ідентифікації вразливостей та їх експлуатації.	В основному ґрунтується на автоматизованих інструментах та скриптах для сканування вразливостей та здійснення атак з мінімальною участю людини.
Глибина аналізу	Пропонує глибший аналіз, оскільки пентестери можуть зрозуміти контекст, бізнес-логіку та виконувати складні атаки, які автоматизовані інструменти можуть пропустити.	Обмежена глибина, оскільки автоматизовані інструменти дотримуються попередньо визначених алгоритмів та можуть пропустити деякі вразливості або неправильно тлумачити результати.
Налаштування	Високий рівень налаштованості для адаптації до конкретних середовищ, програм та потреб у безпеці. Пентестери можуть розробляти унікальні сценарії атак.	Обмежена налаштованість, оскільки автоматизовані інструменти дотримуються попередньо визначених шаблонів сканування, які можуть не підходити для всіх сценаріїв.
Часові затрати	Зазвичай вимагає більше часу через ручну роботу, пов'язану з розвідкою, аналізом та експлуатацією.	Швидше, оскільки автоматизовані інструменти можуть сканувати великі мережі або програми за відносно короткий час.
Вартість	Часто більш дороге через потребу у висококваліфікованих фахівцях та часоємність ручного тестування.	Загалом більш ефективно за рахунок зменшеної потреби в людських ресурсах та швидкості тестування.

Точність виявлення	Висока точність, оскільки пентестери можуть перевіряти результати, усувати помилкові виявлення та надавати детальні звіти.	Схильний до помилкових та неправильних виявлень, оскільки автоматизовані інструменти можуть неправильно тлумачити результати.
Адаптованість до нових загроз	Більш пристосований до нових загроз та вразливостей «нульового дня» за рахунок людського інтелекту та креативності.	Обмежена адаптованість до нових загроз, якщо база даних автоматизованого інструменту не регулярно оновлюється або не налаштовується для виявлення невідомих вразливостей.
Оцінка ризиків	Забезпечує комплексне розуміння ризиків, включаючи потенційний вплив та ймовірність, на основі людського волевиявлення та експертизи.	Оцінка ризиків може бути менш нюансованою, зосереджуючись більше на наявності або відсутності вразливостей, а не на їх потенційному впливі.
Придатність для складних систем	Добре підходить для складних систем, програм або середовищ, де автоматизовані інструменти можуть мати проблеми через унікальні конфігурації або поведінку.	Може мати проблеми з ефективним тестуванням складних систем, особливо тих, що мають нестандартні архітектури або корпоративні програми.

Отже, оптимальний підхід полягає у поєднанні обох стратегій – використанні як ручного, так і автоматизованого тестування на проникнення. Такий комплексний підхід дозволяє забезпечити більш повне покриття виявлення вразливостей та мінімізувати ймовірність пропуску потенційних загроз безпеці.

ВИСНОВКИ

У цифровій сфері, що швидко розвивається, де кіберзагрози підстерігають на кожному кроці, розробка надійних заходів веб-безпеки є наріжним каменем сучасних оборонних стратегій. Завдяки ретельному дослідженню та аналізу розкрито важливість нагальної потреби в ефективних протоколах веб-безпеки, зокрема, через призму тестування на проникнення з використанням потужної платформи OWASP.

Використовуючи можливості OWASP, організації можуть розпочати активний шлях до зміцнення своїх цифрових фортець. Цей фреймворк слугує не просто інструментом, а вартовим, який невтомно прочісує глибини веб-додатків та інфраструктури, щоб виявити вразливості до того, як зловмисники зможуть ними скористатися. Це щит, який захищає від постійно зростаючого спектру кіберзагроз, пропонуючи стійкий захист в дедалі небезпечнішому онлайн-ландшафті.

Хоча OWASP є потужною зброєю в арсеналі кібербезпеки, вона є лише одним з аспектів багатогранної оборонної стратегії. Справжня стійкість проти цифрових супротивників вимагає цілісного підходу, що включає такі елементи, як моніторинг загроз в режимі реального часу, адаптивні механізми реагування на інциденти і ініціативи з безперервної освіти в галузі безпеки.

Коли ми вдивляємося в горизонт кіберзахисту, стає очевидним, що цей шлях нескінченний. Противники, з якими ми стикаємось, динамічні, а їхня тактика постійно розвивається. Таким чином, завершення цього дослідження знаменує собою не кінець, а новий початок – заклик до подальших інновацій, досліджень і співпраці у сфері веб-безпеки.

У вирі постійних інновацій лежить перспектива більш безпечного цифрового майбутнього. Однак ця перспектива може бути реалізована лише за умови, якщо ми приділимо належну увагу кібербезпеці на всіх рівнях: від індивідуальних користувачів до великих корпорацій та урядів. Розробка та впровадження стратегій превентивних заходів, посилення культури кібербезпеки, постійне оновлення і підвищення кваліфікації спеціалістів у цій сфері – всі ці кроки відіграють важливу роль у забезпеченні нашої цифрової безпеки. Крім того, необхідно продовжувати інвестувати в дослідження та розробки нових технологій у галузі кібербезпеки, щоб відповідати зростаючим вимогам та загрозам. Активна співпраця між сектором приватного та громадського секторів також відіграє важливу роль у виявленні та запобіганні кіберзагрозам.

Отже, лише шляхом поєднання зусиль, інновацій та постійного вдосконалення ми зможемо побудувати більш безпечне та надійне цифрове майбутнє для нашого суспільства.

ЛІТЕРАТУРА

[1]. J. Song, E. Jo and J. Kim, "DTA: Run TrustZone TAs Outside the Secure World for Security Testing," in IEEE Access, vol. 12, pp. 16715-16727, 2024, doi: 10.1109/ACCESS.2024.3358612.

[2]. Altulaihan, E.A.; Alismail, A.; Frikha, M. A Survey on Web Application Penetration Testing. Electronics 2023, 12, 1229. doi: 10.3390/electronics12051229.

[3]. Diogenes, Y., & Ozkaya, E. Cybersecurity, Attack and Defense Strategies: Counter Modern Threats and Employ State-of-the-Art Tools and Techniques to Protect Your Organization Against Cybercriminals, 2nd Edition. Packt Publishing, 2019. ISBN 9781838822217.

[4]. Riadi, I., Fadlil, A., & Mu'min, M. A. OWASP Framework-based Network Forensics to Analyze the SQLi Attacks on Web Servers. Matrik: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer, 2023. doi: 10.30812/matrik.v22i3.3018.

[5]. Y.Stefinko, A.Piskozub and R. Banakh, "Manual and automated penetration testing. Benefits and drawbacks. Modern tendency," 2016 13th International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET), Lviv, Ukraine, 2016, pp. 488-491, doi: 10.1109/TCSET.2016.7452095.

[6]. Barakovic, S., & Baraković Husić, J. "The Importance of Security Matters for Quality of Experience in Mobile Web Context". International Journal of Human–Computer Interaction 39, no.8(2023): 1712–22. doi: 10.1080/10447318.2022.2072454.

[7]. Sterle, L., & Bhunia, S. "On SolarWinds Orion Platform Security Breach," 2021 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Internet of People and Smart City Innovation (SmartWorld / SCALCOM / UIC /ATC/ IOP/SCI), Atlanta, GA, USA, 2021, pp. 636-641, doi: 10.1109/SWC50871.2021.00094.

[8]. Types of Testing Techniques: Black, White and Grey Box. URL: https: // kratikal.com/blog/types-of-testing-techniques-black-white-and-grey-box/.

[9]. The Penetration Testing Execution Standard. URL: https://www.pentest-standard.org/.

[10]. The OWASP Top Ten. URL: https://www.owasp-topten.org/.

[11]. Ashari, I. F. A., Affandi, M., Putra, H. T., & Nur, M. T. Security Audit for Vulnerability Detection and Mitigation of UPT Integrated Laboratory (ILab) ITERA Website Based on OWASP Zed Attack Proxy (ZAP). Jurnal JTIK (Jurnal Teknologi Informasi Dan Komunikasi), 2023, 7(1), pp. 24-34. doi: 10.35870 / jtik.v7i1.657.

[12]. Oliveira, M. P. de, & Silva, C. M. R. D. Uma Arquitetura de Firewall derivada do OWASP ModSecurity Core Rule Set baseada em ganchos de APIs I/O. Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais, 2020, pp. 147-160. doi: 10.5753/sbseg.2020.19234.

[13]. Gopalakrishnan, R., Mercado, I. T., da Silva Santos, J. C., & Matos, L. Detecting OWASP Cheat Sheets in the Source Code. Rochester Institute of Technology & University of Notre Dame, 2015. doi: 10. 13140/RG.2.1.4120.2963.

[14]. Glemser, T. OWASP Top 10. Datenschutz Datensich 46, 695-698 (2022). doi: 10.1007 / s11623-022-1685-5.

[15]. M. Idris, I. Syarif and I. Winarno, "Development of Vulnerable Web Application Based on OWASP API Security Risks," 2021 International Electronics Symposium (IES), Surabaya, Indonesia, 2021, pp. 190-194. doi: 10.1109/IES53407.2021.9593934.

[16]. Wang, P., Lyu, X., Yu, X., Zhang, C. Instant Messaging Application Traffic Recognition. In: Sun, X., Zhang, X., Xia, Z., Bertino, E. (eds) Advances in Artificial Intelligence and Security. ICAIS 2021. Communications in Computer and Information Science, vol 1423. Springer, Cham. doi: 10.1007/978-3-030-78618-2_60.

[17]. Deng, G., Liu, Y., Mayoral-Vilches, V., Liu, P., Li, Y., Xu, Y., Zhang, T., Liu, Y., Pinzger, M., & Rass, S. PentestGPT: An LLM-empowered Automatic Penetration Testing Tool. arXiv preprint arXiv:2308. 06782, 2023. doi: 10.48550/arXiv.2308.06782.

[18]. Yu, K., Tan, L., Yang, C., Choo, K., Bashir, A., Rodrigues, J., & Sato, T., "A Blockchain-Based Shamir's Threshold Cryptography Scheme for Data Protection in Industrial Internet of Things Settings," in IEEE Internet of Things Journal, vol. 9, no. 11, pp. 8154-8167, 1 June1, 2022. doi: 10.1109/JIOT. 2021.3125190.

[19]. Singh, K., & Singh, N., "Multi-level authentication model with Political Dingo Optimizer-enabled ZFNet," 2023 International Conference on Artificial Intelligence and Smart Communication (AISC), Greater Noida, India, 2023, pp. 1022-1026. doi: 10. 1109/AISC56616.2023.10085133.

[20]. Hampau, R.-M., Kaptein, M., Emden, R. V., Rost, T., & Malavolta, I. An empirical study on the Performance and Energy Consumption of AI Containerization Strategies for Computer-Vision Tasks on the Edge. EASE '22 Proceedings of the 26th International Conference on Evaluation and Assessment in Software Engineering, June 2022, pp.50–59. doi: 10.1145/3530019.3530025.

[21]. T. Rangnau, R. v. Buijtenen, F. Fransen and F. Turkmen, "Continuous Security Testing: A Case Study on Integrating Dynamic Security Testing Tools in CI/CD Pipelines," 2020 IEEE 24th International Enterprise Distributed Object Computing Conference (EDOC), Eindhoven, Netherlands, 2020, pp. 145-154, doi: 10.1109/EDOC49727.2020. 00026.

[22]. Sanchez-Gordón, M.-L., Rijal, L., & Palacios, R. C. Beyond Technical Skills in Software Testing: Automated versus Manual Testing. ICSEW'20 Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops, June 2020, pp.161-164. doi: 10.1145/3387940.3392238.

[23]. Devroey, X., Panichella, S., & Gambi, A. (2020). Java Unit Testing Tool Competition: Eighth Round. ICSEW'20 Proceedings of the IEEE/ACM 42nd In-

ternational Conference on Software Engineering Workshops, June 2020, pp. 545-548. doi: 10.1145/3387940.3392265.
[24]. Cannavacciuolo, C., Mariani, L., "Smoke Testing of Cloud Systems," 2022 IEEE Conference on Software Testing, Verification and Validation (ICST), Valencia, Spain, 2022, pp. 47-57. doi: 10.1109 / ICST53961.2022.00016.
[25]. Chen, Q. Z., Schnabel, T., Nushi, B., & Amershi, S. HINT: Integration Testing for AI-based features with Humans in the Loop. IUI '22 Proceedings of the 27th International Conference on Intelligent User Interfaces, March 2022, pp. 549-565. doi: 10.1145/3490099.3511141.
[26]. Yoo, S., & Harman, M. Regression testing minimization, selection and prioritization: a survey. John Wiley & Sons, Ltd., 2012/3, Vol.22, Issue 2, pp. 67-120. doi: 10.1002/stvr.430.
[27]. Liu, Y., Li, Y., Deng, G., Liu, Y., Wan, R., Wu, R., Ji, D., Xu, S., & Bao, M. Morest: Model-based RESTful API Testing with Execution Feedback. arXiv:2204.12148, 2022. doi: 10.48550/arXiv.2204.12148.
[28]. Qasaimeh M, Hammour RA, Yassein MB, Al-Qassas RS, Torralbo JAL, Lizcano D. Advanced security testing using a cyber-attack forecasting model: A case study of financial institutions. J Softw Evol Proc. 2022; 34(11): e2489. doi: 10.1002/smr.2489.
[29]. Chen, S., Haque, M., Liu, C., & Yang, W. DeepPerform: An Efficient Approach for Performance Testing of Resource-Constrained Neural Networks. ASE '22 Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering, October 2022, Article No.:31, pp. 1-13. doi: 10.1145/3551349.3561158.
[30]. Wickström, O., & O'Connor, L. Quickstrom: property-based acceptance testing with LTL specifications. PLDI 2022 Proceedings of the 43rd ACM SIGPLAN International Conference on Programming Language Design and Implementation, June 2022, pp. 1025-1038. doi: 10.1145/3519939.3523728.
[31]. Abdillah, E. J., Khoriyah, R., Abqariy, A. N., & Susilo, P. H. (2022). Pengembangan Keamanan Website Menggunakan Teknik Penetration Testing dan DAST (Dynamic Application Security Testing). Media Jurnal Informatika, Vol 14, No 2 (2022), ISSN: 2477-2542. doi: 10.35194/mji.v14i2.2546.

DEVELOPMENT OF EFFECTIVE WEB SECURITY MEASURES FOR THE NETWORK BY CONDUCTING PENETRATION TESTING USING THE OWASP FRAMEWORK

With each step in the development of technology, web security is becoming a more relevant component for ensuring the reliability and protection of network systems. The growing number of cyber threats and potential security breaches emphasizes the need to improve the protection of network systems. To help developers and administrators in this process, there is an important tool – the OWASP (Open Web Application Security Project) framework. It provides a wide range of tools, guidelines, and resources for securing web applications. This framework helps developers check web applications for potential vulnerabilities and find ways to fix them. To better understand, you can imagine that the network is a house and web applications are its doors and windows. If these doors and windows are not tightly closed, attackers can easily get in and cause damage. So, to put it in comparison, just as you check if all the doors and windows in your network are secure, OWASP provides a means to check web applications for vulnerabilities that can be exploited by attackers. Therefore, using the OWASP framework is an important step in developing effective web security measures for your network, helping to ensure that your system is reliable and protected from possible cyberattacks and malicious actions.

Keywords: OWASP framework, web security, penetration testing, data protection, cyber security.

Піскозуб Андріян Збігнєвич, к.т.н., доцент, доцент кафедри Захисту інформації Національного університету «Львівська політехніка».

Piskozub Andrian, Ph.D., Associate Professor at the Department of Information Security, Lviv Polytechnic National University.
E-mail: andriian.z.piskozub@lpnu.ua.
Orcid ID: 0000-0002-3582-2835.

Козловська Марія Іванівна, студентка кафедри Захисту інформації Національного університету «Львівська політехніка».

Kozlovska Mariia, Student at the Department of Information Security, Lviv Polytechnic National University.
E-mail: mariia.kozlovska.kb.2021@lpnu.ua.
Orcid ID: 0009-0003-4959-0312.

DOI: 10.18372/2410-7840.26.18834
УДК 336.71:004.056

АЛГОРИТМ ЗАСТОСУВАННЯ ТЕОРЕМИ БАЙЄСА ДЛЯ ВИЯВЛЕННЯ НЕПРАВДИВОЇ ІНФОРМАЦІЇ У СИСТЕМАХ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

Сергій Глухов, Ігор Половінкін, Максим Кузьменко, Віталій Пономаренко

Захист інформації стає більш актуальним у сучасному світі. Це пов'язано зі зростанням технічного прогресу та перетворенням світу у інформаційних світ. Особливо це стало помітним після всесвітнього карантину від короно вірусу, людство перейшло загалом у інформаційне спілкування. Набули подальший розвиток соціальні мережи та загалом інформаційне спілкування через всесвітню мережу інтернет- кіберпростір. В зв'язку з чим виникає наукове завдання по розробки нових та удосконаленню існуючих методів захисту інформації.