МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

КОВАЛЕНКО ОЛЕКСАНДР ВОЛОДИМИРОВИЧ

УДК 004.05

МОДЕЛІ ТА МЕТОДИ РОЗРОБЛЕННЯ БЕЗПЕЧНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНИХ СИСТЕМ

Спеціальність 05.13.05 – Комп'ютерні системи та компоненти

Автореферат
дисертації на здобуття наукового ступеня
доктора технічних наук

Черкаси – 2020

Дисертацією є рукопис.

Робота виконана на кафедрі кібербезпеки та програмного забезпечення Центральноукраїнського національного технічного університету Міністерства освіти і науки України.

Науковий консультант: доктор технічних наук, професор
Смірнов Олексій Анатолійович,
Центральноукраїнський національний технічний університет, завідувач кафедри кібербезпеки та програмного забезпечення.

Офіційні опоненти: доктор технічних наук, доцент,
Федоров Євген Євгенович,
Черкаський державний технологічний університет, професор кафедри робототехніки та спеціалізованих комп'ютерних систем;

доктор технічних наук, старший науковий співробітник,
Чемерис Олександр Анатолійович,
Національна академія наук України, заступник директора з наукової роботи Інституту проблем моделювання в енергетиці ім. Г.Є. Пухова;

доктор технічних наук, професор,
Мухін Вадим Євгенович,
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», професор кафедри математичних методів системного аналізу Інституту прикладного системного аналізу.

Захист дисертації відбудеться « 27 » листопада 2020 р. о 14:00 год. на засіданні спеціалізованої вченої ради Д 73.052.04 у Черкаському державному технологічному університеті за адресою: 18006, м. Черкаси, бул. Шевченка, 460.

З дисертацією можна ознайомитися в бібліотеці Черкаського державного технологічного університету за адресою: 18006, м. Черкаси, бул. Шевченка, 460.

Автореферат розіслано « 24 » жовтня 2020 р.

Вчений секретар
спеціалізованої вченої ради Ю.Ю. Бондаренко

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність теми. Практичний досвід останніх років свідчить, що сучасні засоби захисту програмного забезпечення комп’ютерних систем (ПЗ КС) не можуть забезпечити необхідний рівень безпеки. Почастішали випадки кібератак на комп'ютерні системи державних об'єктів критичної інфраструктури, на банківські рахунки, на інформаційні ресурси оборонного відомства та інших державних служб.

В першу чергу це пов'язано з тим, що з одного боку масове поширення комп'ютерних інформаційних мережевих технологій істотно розширило можливості та арсенал кіберзлочинців, а з іншого боку компанії-розробники найчастіше нехтують питаннями безпеки ПЗ. Крім того, рівень розвитку методологій розроблення ПЗ не дозволяє акцентовано забезпечити ІТ-компанії необхідним методологічним і практичним контентом, що підвищує рівень безпеки.

В теорії забезпечення кібербезпеки та захисту інформації накопичено значний теоретичний і практичний матеріал та досвід. Найбільш суттєвими роботами в цій області є дослідження іноземних та вітчизняних учених, серед яких: І. Д. Горбенко, В. І. Долгов, О. О. Кузнєцов, О. Г. Корченко, М. Брантон-Сполл, Д. Деннінг, Е. Спаффорд, Р. Смит, В. Столлінгс, Б. Шнайєр, Р. Сіакорд та ін.

Але, враховуючи динамічний розвиток інтелектуалізації комп'ютеризованих керуючих рішень, сучасні інформаційні технології, різноманітність технологічних та апаратно-інформаційних рішень сучасного підходу програмування, сприяють тому, що постановка завдань підвищення безпеки ПЗ КС істотно видозмінюється, це пов'язано з тим, що необхідно враховувати дії нових факторів.

Таким чином, одночасно зі збільшенням кількості ПЗ КС, підвищенням вимог до його безпеки відбувається збільшення кількісного і підвищення якісного рівня зловмисних дій, що призводить до протиріччя між зниженням показників безпеки ПЗ КС, викликаним зовнішніми зловмисними діями з одного боку, і жорсткими вимогами до гарантованого рівня безпеки даних з іншого боку.

Усе вищезазначене визначає **актуальність** нових моделей та методів розроблення безпечного ПЗ КС.

Зв'язок роботи з науковими планами, програмами, темами. Дисертаційну роботу виконано на кафедрі кібербезпеки та програмного забезпечення Центральноукраїнського національного технічного університету. Здобувач, як співвиконавець окремих етапів, проводив дослідження у рамках держбюджетних НДР МОН України: №36Б113 «Розробка методів підвищення оперативності передачі і захисту інформації в телекомунікаційних системах» (ДР №0113U003086), №36Б115 «Розробка методів синтезу тестових моделей поведінки програмних об'єктів, підвищення оперативності передачі і захисту інформації в телекомунікаційних системах» (ДР №0115U003103), науково-дослідних робіт: №36.Д117 «Розробка методів якісного аналізу та кількісної

оцінки ризиків розроблення програмного забезпечення» (ДР №0117U006991), №36.Д217 «Розробка методу управління ризиками розроблення програмного забезпечення» (ДР №0117U006992), №36.Д317 «Розробка імітаційної моделі технології тестування безпеки» (ДР №0117U006993), №36.Д417 «Розробка комплексу математичних моделей технології тестування Web-програм» (ДР №0117U006994).

Мета і завдання дослідження. Метою дисертаційної роботи є підвищення рівня безпеки програмних компонентів комп'ютерних систем шляхом створення нових та удосконалення існуючих моделей та методів розроблення безпечного програмного забезпечення. Відповідно до мети роботи необхідно вирішити ***науково-технічну проблему, що полягає в синтезі моделей та методів розроблення безпечного ПЗ КС.***

Аналіз існуючого стану вимог безпеки даних і якості ПЗ КС, основних моделей, методів і методологій розроблення безпечного програмного забезпечення і факторів, що впливають на безпеку, показників і критеріїв оптимізації виявив необхідність вирішення наступних проблемних ***задач***:

1. Удосконалення методу якісного аналізу вразливостей розроблення ПЗ, що враховує фактори експлуатаційних вразливостей, особливо вразливості невиявлення загроз безпеки ПЗ КС.
2. Удосконалення методу кількісної оцінки вразливостей розроблення ПЗ, що враховує негативні фактори можливого невиявлення загроз безпеки ПЗ КС.
3. Удосконалення методу оптимізації розподілу ресурсів розроблення ПЗ з використанням напівмарківської моделі прийняття рішень для керованого марківського процесу у безперервному часі.
4. Розробка математичної моделі технології тестування комплексу *DOM XSS* вразливостей, що враховує специфіки комплексного аналізу різних типів *XSS* вразливості.
5. Розробка математичної моделі технології тестування вразливості до *SQL*-ін'єкцій на основі критерію Джаро-Вінклера.
6. Розробка методу математичного моделювання процесу тестування *DOM XSS* вразливості та вразливості до *SQL*-ін'єкцій на основі підходу мережевого *GERT* моделювання.
7. Удосконалення імітаційної моделі технології тестування безпеки на основі положень теорії масштабування імітаційних моделей з урахуванням адаптації вибору вхідних операторів управління і даних до підвищення вимог оперативності розроблення.
8. Розробка методу передтестової компіляції і розподілу доступу, що враховує профілі користувачів при розроблені застосунку, а також використання ресурсів «хмарних сховищ» в процесі отримання інсталяційних версій.

Об'єктом дослідження є процеси розроблення і експлуатації безпечних програмних компонентів комп'ютерних систем.

Предметом дослідження є моделі та методи розроблення безпечного програмного забезпечення комп'ютерних систем.

Методи досліджень. При вирішенні науково-технічної проблеми було використано широкий спектр методів. Так, при розробці математичних моделей процесу тестування *DOM XSS* вразливості та вразливості до *SQL*-ін'єкцій використовувалися методи теорії графів і мережевого *GERT* моделювання. При розробці методу якісного аналізу використовувався інструмент аналізу причинно-наслідкових зв'язків між різними факторами і вразливостями, а також звуження Парето за допомогою «кванта» інформації. Для кількісної оцінки вразливостей використовувалася графічна модель *FTA*. В основу методу оптимізації розподілу ресурсів розроблення ПЗ була покладена напівмарківська модель прийняття рішень для керованого марківського процесу у безперервному часі. Імітаційне моделювання було проведене на основі положень теорії масштабування імітаційних моделей. Оцінка експериментальних даних, отриманих в ході роботи, проводилася на основі методів математичної статистики.

Наукова новизна отриманих результатів.

1. *Удосконалено* метод якісного аналізу вразливостей розроблення програмного забезпечення, що відрізняється від відомих врахуванням факторів експлуатаційних вразливостей, особливо вразливості невиявлення загроз безпеки ПЗ КС, і оцінкою довільного несуперечливого кінцевого набору «квантів інформації», це дозволяє звузити множину важливих вразливостей і знизити можливі фінансові та іміджеві втрати організацій-розробників ПЗ.

2. *Удосконалено* метод кількісної оцінки вразливостей розроблення ПЗ, що відрізняється від відомих комплексним використанням методики «Аналізу дерева відмов» і способу оцінки показника чистої приведеної вартості проекту розроблення безпечного ПЗ з урахуванням негативних факторів можливого невиявлення загроз безпеки ПЗ КС, це дозволило підвищити точність кількісної оцінки вразливостей розроблення ПЗ.

3. *Удосконалено* метод оптимізації розподілу ресурсів розроблення ПЗ на основі напівмарківської моделі прийняття рішень для керованого марківського процесу у безперервному часі. Відмінною особливістю запропонованого методу є використання псевдобулевих методів бівалентного програмування з нелінійною цільовою функцією і лінійними обмеженнями для визначення оптимальної стратегії усунення експлуатаційних помилок, це дозволяє оптимізувати процес проектування стратегії розподілу ресурсів розроблення ПЗ.

4. *Вперше розроблено* математичну модель технології тестування комплексу *DOM XSS* вразливостей, яка за рахунок урахування специфіки комплексного аналізу різних типів *XSS* вразливості («*stored XSS*», «*reflected XSS*» і *DOM Based XSS*), а також включенням в алгоритм процедур автоматичного аудиту *DOM Based XSS* окремо, дозволяє провести аналітичну оцінку часових витрат тестування вказаних вразливостей в умовах реалізації стратегії розроблення безпечного програмного забезпечення.

5. *Вперше розроблено* математичну модель технології тестування вразливості до *SQL*-ін'єкцій, яка за рахунок використання критерію

Джаро-Вінклера, для порівняння результатів ін'єкції *SQL*-коду і введення порогового значення дозволяє підвищити точність результатів тестування безпеки програмного забезпечення.

6. *Вперше розроблено* метод математичного моделювання технологій тестування *DOM XSS* вразливості та вразливості до *SQL*-ін'єкцій, в основу якої покладений підхід мережевого *GERT* моделювання, це дозволило досліджувати процеси в комп'ютеризованих системах при розробці нових засобів і протоколів захисту даних, а також зменшити час тестування безпеки програмного забезпечення.

7. *Отримано подальший розвиток* імітаційної моделі технології тестування безпеки на основі положень теорії масштабування імітаційних моделей. Відмінною особливістю розробленої імітаційної моделі є адаптація вибору вхідних операторів управління і даних до підвищення вимог оперативності розроблення і реалізації моделі, виражена в реалізації процедури взаємодії з реальним браузером, з використанням засобів автоматизації браузеру і формуванні даних для атаки на декількох діалектах, це дозволило понизити обчислювальну складність алгоритмів, що реалізовуються.

8. *Вперше розроблено* метод передтестової компіляції і розподілу доступу, який за рахунок врахування профілів користувача при розроблені застосунку, а також використання ресурсів «хмарних сховищ» в процесі отримання інсталяційних версій дозволяє підвищити рівень безпеки застосунків, що розробляються.

Практичне значення отриманих результатів в області розроблення ПЗ полягає в тому, що запропоновані в дисертаційній роботі моделі та методи є науково-методичною основою для розроблення відповідних компонентів програмних продуктів, алгоритмів, системних утиліт та протоколів. Практична значущість отриманих результатів полягає в наступному:

– метод якісного аналізу вразливостей розроблення ПЗ дозволив на 55% звузити сукупність множин Парето та більш точно обирати пріоритетність напрямків фінансування профілактичних заходів;

– метод кількісної оцінки вразливостей розроблення ПЗ дозволив за рахунок використання удосконаленої методики "Аналізу дерева відмов" підвищити точність кількісної оцінки вразливостей розроблення ПЗ до 20% та спроектувати програмну систему оцінки чистої приведеної вартості проекту розроблення безпечного ПЗ;

– метод математичного моделювання технологій тестування *DOM XSS* вразливості та вразливості до *SQL*-ін'єкцій дозволив розробити автоматизований програмний засіб виявлення вразливості ПЗ, що дає можливість зменшити час тестування безпеки від 1,05 до 1,5 разів.

Практична значущість отриманих результатів підтверджується їх застосуванням: при розробленні автоматизованих систем виявлення вразливостей ПЗ в Інтернет сервіс провайдері ТОВ «ІМПЕРІАЛ-НЕТ»; при удосконаленні гнучкої методології розроблення ПЗ у компанії-розробнику програмного забезпечення ТОВ «МІФ ПРОДЖЕКТС» (м. Кропивницький); при розробленні

систем захисту інформації ТОВ «САЙФЕР ІТ» (м. Київ); у навчальному процесі Центральноукраїнського національного технічного університету.

Крім того, практична значущість дисертаційної роботи визначається можливістю застосування запропонованих моделей та методів не лише при розробленні безпечного ПЗ КС, але і в комп'ютерних і інформаційних системах загального призначення.

Особистий внесок здобувача. Усі основні наукові положення, результати, висновки і рекомендації, приведені в дисертаційній роботі, отримані автором самостійно. Вони викладені як в роботах, які опубліковані без співавторів [6, 16-31], так і у співавторстві. У наукових роботах, які опубліковані у співавторстві, внесок автора полягає в наступному: у [1] розроблено метод оцінки рівнів зрілості програмних систем та підходу розрахунку тривалості необхідного прогнозованого періоду; у [2] розроблено комплекс самоподібних генераторів трафіку за допомогою ланцюгів Маркова, які відрізняються від аналогів меншими вимогами до обчислювальної потужності для імітаційних систем технологій тестування безпеки; у [3] проаналізовано показники якості вразливостей розроблення програмного забезпечення; у [4] розроблено математичну модель технології тестування на вразливості *DOM XSS*; у [5] розроблено технологію тестування *DOM XSS* вразливості; у [7] проведено аналіз підходів розроблення безпечного програмного забезпечення з використанням запропонованих методів якісного аналізу та кількісної оцінки вразливостей; у [8] розроблено метод оптимізації розподілу ресурсів розроблення безпечного програмного забезпечення; у [9] розроблено комплекс математичних моделей технології тестування *web*-застосунків; у [10] сформульовано задачі розпізнавання ситуацій у *ERP*-системах; у [11] розроблено методи якісного аналізу та кількісної оцінки вразливостей розроблення програмного забезпечення; у [12] визначено проблеми аналізу та оцінки вразливостей інформаційної діяльності; у [13] розроблено метод якісного аналізу вразливостей розроблення програмного забезпечення; у [14] розроблено метод кількісної оцінки вразливостей розроблення програмного забезпечення; у [15] обґрунтовано використання псевдобулевих методів бівалентного програмування для оптимізації розподілу ресурсів розроблення безпечного програмного забезпечення. Зазначений особистий внесок здобувача в роботах, які виконані у співавторстві, відповідає темі та змісту дисертації.

Апробація результатів дисертаційних досліджень. Основні положення дисертаційної роботи були представлені на наступних міжнародних науково-технічних конференціях, симпозіумах та конгресах: «Актуальні питання забезпечення кібернетичної безпеки та захисту інформації» (Київ, 2016-2018 рр.); «Securitea informationala» (Chisinau, Moldova, 2016-2018 рр.); «Інформатика та системні науки» (Полтава, 2016 р.); «Проблеми кібербезпеки інформаційно-телекомунікаційних систем» (Київ, 2016-2017 рр.); «Інформаційна безпека та комп'ютерні технології» (Кропивницький, 2016-2018 рр.); «Проблеми науково-технічного та правового забезпечення кібербезпеки у сучасному світі» (Харків, 2016-2017 рр.); «Комбінаторні конфігурації та їх застосування» (Кропивницький,

2016-2017 рр.); «Проблеми і перспективи розвитку ІТ-індустрії» (Харків, 2016-2018 рр.); «Інформаційна та економічна безпека» (Харків, 2016 р.); «Стратегия качества в промышленности и образовании» (Варна, Болгарія, 2016 р., 2018 р.); «Кібербезпека в Україні: правові та організаційні питання» (Одеса, 2016 р.); «Актуальні задачі та досягнення у галузі кібербезпеки» (Кропивницький, 2016 р.); «Information technologies, systems and networks» (Chisinau, Moldova, 2017 р.); «Автоматика та комп'ютерно-інтегровані технології у промисловості, телекомунікаціях, енергетиці та транспорті» (Кропивницький, 2017 р.); «Комп'ютерні інтелектуальні системи та мережі» (Кривий Ріг, 2018 р.); «Комп'ютерна інженерія і кібербезпека: досягнення та інновації» (Кропивницький, 2018 р.).

Публікації. Основні положення і результати дисертації опубліковано у 59 наукових працях, у тому числі: 1 монографія; 3 колективні монографії; 2 наукові статті у міжнародних рецензованих виданнях, що входять до бази даних Scopus; 3 наукові статті у закордонних фахових наукових журналах, та 22 статті у наукових журналах та збірниках наукових праць, що входять до переліку фахових видань України (з них без співавторів – опубліковано 16), а також 28 матеріалів і тез доповідей на всеукраїнських та міжнародних конференціях.

Структура та обсяг дисертації. Дисертація складається з анотації, змісту, вступу, шести розділів, висновків, списку використаних джерел і додатків. Загальний обсяг роботи становить 317 сторінок, з них обсяг основного тексту – 250 сторінок, 58 рисунків, 17 таблиця, список використаних джерел складає 269 найменувань і займає 28 сторінок, а також 3 додатки на 29 сторінках.

ОСНОВНИЙ ЗМІСТ РОБОТИ

У вступі обґрунтовано актуальність теми, сформульовано мету та задачі дослідження, наукова новизна, практичне значення отриманих результатів дослідження, наведено результати впровадження основних положень роботи, наведено інформацію щодо апробації та публікації результатів дисертації, визначено особистий внесок здобувача, зв'язок дослідження з науковими програмами та планами.

У першому розділі проведено аналіз сучасних тенденцій розвитку моделей та методів розроблення безпечного ПЗ КС і вимог до програмних засобів, основних моделей, методів і методологій розроблення безпечного ПЗ КС і факторів, що впливають на безпеку, сучасних підходів математичного моделювання процесу розроблення безпечного ПЗ КС і факторів, що впливають на безпеку, формулюється завдання розроблення.

Результати аналізу і проведених досліджень основних тенденцій і показників безпеки ПЗ дозволили сформувати загальну схему характеристик і показників, що відносяться до якості програмного забезпечення, а також виділити серед них характеристики безпеки.

Проведені дослідження і аналіз показників якості ПЗ у вигляді ієрархічної векторної системи, а також показників безпеки ПЗ зокрема, дозволили

представити комплексний показник безпеки ПЗ КС $Y_i^{(ПЗ)}$ у вигляді добутку матриць:

$$Y_i^{(ПЗ)} = (X_{ik} \cdot Y_k) \cdot A, \quad (1)$$

де $X_{ik} = \left[x_{\psi}^{(\xi)}\right]$ – матриця усереднених коефіцієнтів впливу зовнішніх процесів і факторів впливу на окремі показники безпеки ПЗ, i – кількість зовнішніх факторів, що впливають на функціонування системи, k – кількість програмних засобів ПЗ КС, $x_{\psi}^{(\xi)} = 1/N \sum_{j=1}^{N} x_{\ell_j}^{(\psi)}$ – усереднений коефіцієнт впливу зовнішніх процесів (Ψ) на показники безпеки окремих програмних засобів КС (ξ), l – найменування окремого показника безпеки ПЗ, A – матриця усереднених коефіцієнтів взаємовпливу різних характеристик якості ПЗ, $Y_k = \left[\overline{Y_{mat}^{(ПЗ)}}, \overline{Y_{fft}^{(ПЗ)}}, \overline{Y_{rec}^{(ПЗ)}}, \overline{Y_{conft}^{(ПЗ)}}, \overline{Y_{intt}^{(ПЗ)}}, \overline{Y_{autht}^{(ПЗ)}}, \overline{Y_{avb}^{(ПЗ)}}, \overline{Y_{fir}^{(ПЗ)}}\right]$ – матриця показників безпеки ПЗ, $Y_{mat}^{(ПЗ)}$, $Y_{fft}^{(ПЗ)}$, $Y_{rec}^{(ПЗ)}$, $Y_{conft}^{(ПЗ)}$, $Y_{intt}^{(ПЗ)}$, $Y_{autht}^{(ПЗ)}$, $Y_{avb}^{(ПЗ)}$, $Y_{fir}^{(ПЗ)}$ – векторні показники безпеки ПЗ. В результаті перемноження буде сформована матриця, що являє собою комплексний показник безпеки $Y_i^{(ПЗ)}$ ПЗ КС.

Результати аналізу літератури, а також методичних рекомендацій по розробленню ПЗ у ряді джерел, дають підстави стверджувати про те, що при розробленні безпечного ПЗ КС нині переважно використовують послідовні (прогнозовані) методології. Серед них перевага віддається водоспадній моделі. Проте сучасні тенденції в розробленні, а також безліч об'єктивно-суб'єктивних факторів (дивергенція обов'язків і ролей при розробленні ПЗ, можлива територіальна і культурна віддаленість у складі команди міні і мульти економічні кризи, що почастішали та ін.) вимагають від розробників гнучкіших підходів і швидкого реагування на зовнішні фактори. У цій ситуації виникає протиріччя між підвищеними вимогами до безпеки ПЗ (врахуванням усіх вразливостей) і необхідністю адаптації до існуючих об'єктивно-суб'єктивних факторів, властивих сучасному світу IT-індустрії.

Дослідження основних підходів моделювання показало, що більшість моделей, пов'язаних з реалізацією технології розроблення ПЗ не враховують факторів апріорної невизначеності в параметрах безпеки.

Крім того, відсутність врахування в моделях динамічних змін в ході розроблення безпечного ПЗ (особливостей *Agile*) вимагає відповідних досліджень і розробок.

На рис. 1 представлена порівняльна характеристика найбільш відомих підходів математичної формалізації процесів управління розробленням зі вказанням їх переваг і недоліків. Результати, отримані при моделюванні таких процесів, показали можливість використання *GERT*-моделей для вирішення окремих завдань моделювання процесу розроблення безпечного ПЗ.

Як результат, в дисертаційній роботі сформульовано завдання дисертаційного дослідження, в якому визначено, що основним завданням розроблення моделей та методів розроблення безпечного ПЗ є удосконалення і вибір моделей, методів і засобів, що підвищують рівень захисту інформації.

МОДЕЛІ УПРАВЛІННЯ РОЗРОБЛЕННЯМ БЕЗПЕЧНОГО ПЗ

Нейронні мережі	**Апарат випадкових процесів**	**Автоматні**	**Графові**
Переваги: – можливість моделювання адаптивних систем; – можливість врахування фактору апріорної невизначеності вхідних сигналів; – можливість урахування специфіки зовнішніх впливів.	**Переваги:** – можливість реалізації в системах у вигляді монітора посилань і системи аудиту.	**Переваги:** – різноманіття політик управління, що визначають порядок взаємодії суб'єктів і об'єктів управління між собою.	**Переваги:** – можливість визначення довільних функцій розподілу випадкових величин; – простота реалізації.
Недоліки: – необхідність розбиття моделі на ряд простих моделей; – складність опису аналітичного представлення.	**Недоліки:** – відсутність урахування специфіки методології SCRUM і вимог безпеки.	**Недоліки:** – складність практичної реалізації.	**Недоліки:** – не враховуються параметри, що змінюються і підлаштовуються в процесі функціонування.
Загальний недолік: Складність врахування фактору апріорної невизначеності в параметрах безпеки, відсутність врахування в моделях динамічних змін.			

Рис. 1. Порівняльний аналіз існуючих моделей управління розробленням безпечного ПЗ

У другому розділі розроблено методи якісного аналізу і кількісної оцінки вразливостей розроблення ПЗ КС, що дозволило вирішити протиріччя, що виникають при розробленні ПЗ, і які полягають у зневазі компаніями-розробниками ПЗ факторами вразливості безпеки ПЗ.

В якості вирішення вказаної проблеми запропоновано використання розроблених методів якісного аналізу і кількісної оцінки вразливостей розроблення ПЗ КС.

В дисертаційній роботі ідентифіковано і класифіковано вразливості розроблення ПЗ КС. Результат представлений у вигляді структурної схеми на рис. 2.

Як видно з рис. 2, основні вразливості розроблення ПЗ КС можна представити у вигляді сукупності множин організаційних $Z = \{Id\,1,\ldots,Id\,5\}$, управлінських $U = \{Id\,6,\ldots,Id\,9\}$, операційних $Y = \{Id\,10,\ldots,Id\,15\}$, технологічних $T = \{Id\,16,\ldots,Id\,20\}$, експлуатаційних $E = \{Id\,21,\ldots,Id\,24\}$, соціальних $C = \{Id\,25,\ldots,Id\,27\}$ та правових $W = \{Id\,28, Id\,29\}$ вразливостей.

Відмінною особливістю представленої класифікації є врахування експлуатаційних вразливостей. Особливої важливості ці вразливості набувають в умовах підвищеного рівня кіберзлочинності, коли зневага вразливостями програмного забезпечення може призвести до експлуатаційних проблем, а часто і неможливості експлуатації ("краху") ПЗ.

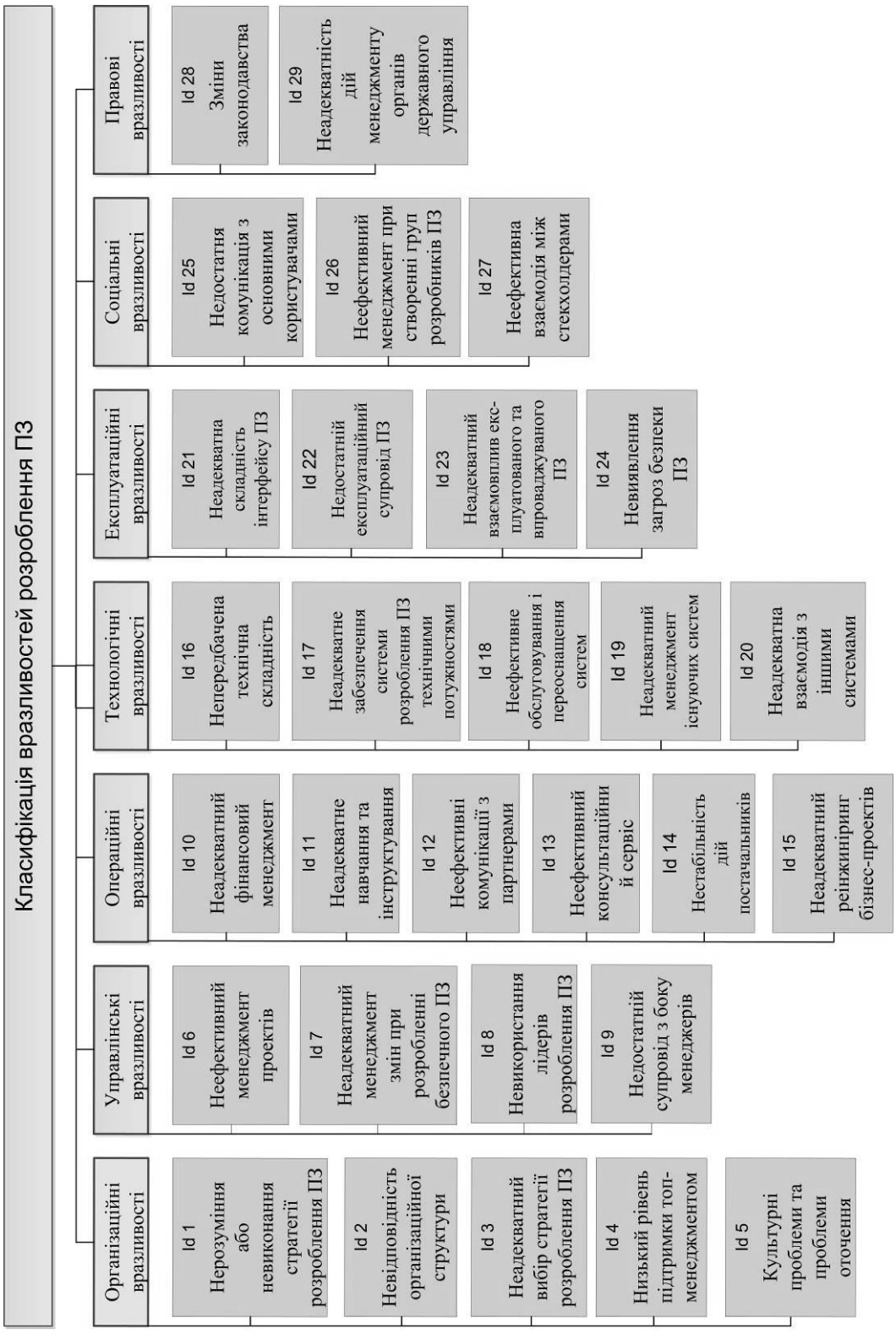

Рис. 2. Класифікація вразливостей розроблення ПЗ

В цілому можна виділити множину вразливостей, що безпосередньо впливають на процес розроблення безпечного ПЗ КС: $MR = \{Z, U, Y, C, T, W\}$, і множину вразливостей, що безпосередньо впливають на процес експлуатації безпечного ПЗ КС: $ME = \{Z, U, Y, C, T, W, E\}$, $(Id\,9, Id\,10,\ Id\,15, Id\,29) \notin ME$.

В дисертаційній роботі показано, що для вирішення задачі визначення взаємовпливу вразливостей доцільно використовувати інструмент аналізу

причинно-наслідкових зв'язків між різними факторами і вразливостями, розроблений Каору Ішикава. Використовуючи запропонований алгоритм, діаграму Ішикави можна представити у вигляді рис. 3.

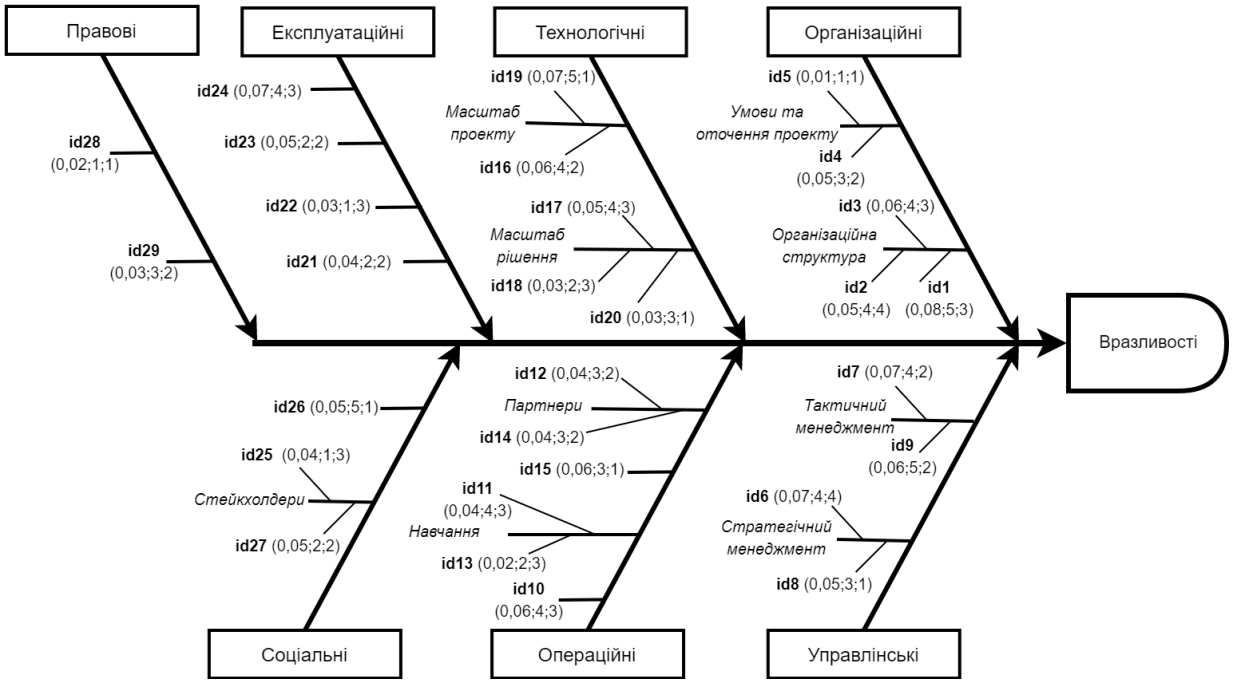

Рис. 3. Дерево рішень діаграми вразливостей розроблення ПЗ КС

У результаті застосування запропонованого алгоритму приклад обраних параметрів оцінки вразливостей та ймовірність виникнення вразливості наведено у табл. 1.

Таблиця 1

Обрані параметри оцінки вразливостей розроблення ПЗ

№	Ймовірність виникнення вразливості	Обрані параметри оцінки вразливостей ітерації		№	Ймовірність виникнення вразливості	Обрані параметри оцінки вразливостей ітерації	
		Збитки економічні (1-15)	Збитки репутації компанії (1-4)			Збитки економічні (1-15)	Збитки репутації компанії (1-4)
Id1	0,08	5	3	*Id16*	0,06	4	2
Id2	0,05	4	4	*Id17*	0,05	4	3
Id3	0,06	4	3	*Id18*	0,03	2	3
Id4	0,05	3	2	*Id19*	0,07	5	1
Id5	0,01	1	1	*Id20*	0,03	3	1
Id6	0,07	4	4	*Id21*	0,04	2	2
Id7	0,07	4	2	*Id22*	0,03	1	3
Id8	0,05	3	1	*Id23*	0,05	2	2
Id9	0,06	5	2	*Id24*	0,07	4	3
Id10	0,06	4	3	*Id25*	0,04	1	3
Id11	0,04	4	3	*Id26*	0,05	5	1
Id12	0,04	3	2	*Id27*	0,05	2	2
Id13	0,02	2	3	*Id28*	0,02	1	1
Id14	0,04	3	2	*Id29*	0,03	3	2
Id15	0,06	3	1				

Для вирішення завдання вибору найбільш пріоритетних вразливостей пропонується використати математичний апарат багатокритеріальної оптимізації, заснованої на локальній геометрії множини Парето з послідуючим звуженням початкової множини Парето на основі отриманих даних «квантів інформації».

Для цього розглянемо довільні оцінки вразливостей розроблення ПЗ $y'=(y_1',\ldots,y_m')$ та $y''=(y_1'',\ldots,y_m'')$, що належать до множини парето-оптимальних векторів $f(P_f(X))$.

За визначенням множини Парето, повинні знайтися такі дві непорожні підмножини номерів критеріїв $A,B\subset I=\{1,2,\ldots,m\}$, що

$$y_i'>y_i'', \quad y_i'-y_i''=w_i>0, \quad \forall i\in A, \tag{2}$$

$$y_j''>y_j', \quad y_j''-y_j'=w_j>0, \quad \forall j\in B \tag{3}$$

$$y_s''=y_s', \quad \forall s\in I\setminus(A\cup B) \tag{4}$$

Згідно з умовами (2-4), перший вектор перевершує другий за компонентами групи критеріїв *A*, тоді як другий перевершує перший за компонентами групи критеріїв *B*. За іншими компонентами (якщо такі є) два вказані вектори співпадають. Звуження множини Парето, тобто видалення деяких парето-оптимальних векторів, зазвичай відбувається на основі порівняння. Людині найпростіше порівнювати пари.

Якщо при порівнянні фіксованої пари парето-оптимальних векторів *y′* і *y″* виду (2-4) менеджер, який приймає рішення (МПР) "вибраковує" один з цих векторів (наприклад, другий), то це означає, що для нього перший вектор прийнятніший за другий, тобто $y'\succ y''$, де $\succ$ − відношення переваги, визначене на усьому критеріальному просторі $\Re^m$ і співпадаюче на множині *Y* з відношенням $\succ y$.

Співвідношення $y'\succ y''$, задає "квант інформації" про відношення строгої переваги, який свідчить про готовність МПР до компромісу − вона згодна піти на втрати за усіма критеріями групи *B* в розмірі w_j заради того, щоб отримати надбавки в розмірі w_i за критеріями групи *A*, зберігши при цьому значення усіх інших критеріїв. Наявність вказаного "кванта інформації" дозволяє скоротити множину Парето на один вектор y″.

Для того, щоб добитися більшого скорочення, можна прийняти, що $y'\succ y''$, має місце не лише для даної пари векторів, але і для всіх тих векторів, які задовольняють умовам (2-4) при незмінних значеннях w_i і w_j. В цьому випадку пропонується говорити, що група критеріїв *A* важливіша за групу *B*.

При вказаному розширенні дії "кванта інформації" можна розраховувати на помітніше звуження множини Парето, хоча нерідко і воно виявляється недостатнім для остаточного вибору. У таких випадках має сенс накласти додаткові вимоги на відношення переваги так, щоб дія "кванта інформації" в звуженні множини Парето виявилася ефективнішою. У дисертаційній роботі наведено ряд аксіом, що доводять можливість і доцільність використання цього математичного апарату. Як результат розробленої методики в табл. 2. приведено

результати якісної оцінки рангу вразливостей розроблення ПЗ. Як видно з табл. 2, основна частина організаційних, операційних, управлінських і експлуатаційних вразливостей складає найбільш вагомі при реалізації. Це говорить про важливість врахування цих вразливостей (особливо в сучасних умовах застосування гнучких методологій розроблення ПЗ).

Таблиця 2

Результати якісної оцінки рангу вразливостей розроблення ПЗ

Сукупності множин Парето по важливості врахування вразливості																					
Множина Парето	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
	Id 1	*Id 6*	*Id 24*	*Id 19*	*Id 9*	*Id 2*	*Id 7*	*Id 10*	*Id 26*	*Id 16*	*Id 17*	*Id 15*	*Id 11*	*Id 23*	*Id 12*	*Id 25*	*Id 18*	*Id 22*	*Id 13*	*Id 28*	*Id 5*
								Id 3					*Id 4*	*Id 27*	*Id 14*	*Id 21*	*Id 29*	*Id 20*			
														Id 8							

В результаті градація по важливості врахування вразливостей від 10 сукупності множин Парето було здійснено перехід до 21 множини.

Це дозволило на 55% звузити сукупність множин Парето та більш точно обирати пріоритетність напрямків першочергових рішень управління та фінансування профілактичних заходів.

Наступним етапом дослідження є рішення задачі кількісного аналізу вразливостей. При цьому одним з перших підетапів в рішенні цієї задачі є побудування дерева вразливостей.

Проведені дослідження показали, що при цьому адекватним інструментом являється "Аналіз дерева відмов" (*Fault Tree Analysis, FTA*). Аналіз цього підходу кількісної оцінки вразливостей показав доцільність використання графічної моделі *FTA* в термінах математичної логіки. Це допоможе формалізувати умови впливу факторів вразливостей в різних їх комбінаціях на кінцеві показники проекту розроблення ПЗ. Приклад побудованого дерева вразливостей розроблення ПЗ КС наведений на рис. 4. Очевидно, що для цієї схеми загальний коефіцієнт вразливості розроблення ПЗ можна розрахувати по формулі:

$$P(Un\,13)=1-((1-P(Un\,2))\cdot(1-P(Un\,12))\cdot(1-P(Un\,6))\cdot(1-P(Un\,7))\cdot \\ \cdot(1-P(Un\,8))\cdot(1-P(Un\,9))\cdot(1-P(Un\,10))\cdot(1-P(Un\,11))), \quad (5)$$

де $P(Un\,1)=P(Id\,1)\cdot P(Id\,3)$ – коефіцієнт вразливості вибору неправильної стратегії розроблення ПЗ; $P(Un\,2)=P(Un\,1)\cdot P(Id\,2)$ – коефіцієнт вразливості вибору неправильної методики розроблення ПЗ; $P(Un\,3)=P(Id\,4)\cdot P(Id\,9)$ – коефіцієнт вразливості зневаги топ-менеджментом розроблення ПЗ; $P(Un\,4)=P(Id\,5)\cdot P(Id\,6)\cdot P(Id\,7)\cdot P(Id\,8)$ – коефіцієнт вразливості неадекватного менеджменту активної стадії розроблення ПЗ; $P(Un\,5)=1-((1-P(Id\,10))\cdot \cdot(1-P(Id\,26)))$ – коефіцієнт вразливості неадекватного операційного та соціального менеджменту; $P(Un\,6)=1-((1-P(Id\,11))\cdot(1-P(Id\,13))\cdot(1-P(Id\,14))\cdot (1-P(Id\,15)))$ – коефіцієнт вразливості невідповідності професійного рівня учасників проекту; $P(Un\,7)=1-((1-P(Id\,16))\cdot(1-P(Id\,17))\cdot(1-P(Id\,18))\cdot \cdot(1-P(Id\,19))\cdot(1-P(Id\,20)))$ – коефіцієнт технологічних вразливостей;

$P(Un\,8) = P(Id\,21) \cdot P(Id\,22)$ – коефіцієнт вразливості невідповідності складності ПЗ рівню підготовки експлуатанта; $P(Un\,9) = 1 - ((1 - P(Id\,23)) \cdot (1 - P(Id\,24)))$ – коефіцієнт вразливості експлуатації ПЗ; $P(Un10) = P(Id\,12) \cdot \cdot P(Id\,25) \cdot P(Id\,27)$ – коефіцієнт вразливості неадекватної комунікації учасників проекту; $P(Un11) = 1 - ((1 - P(Id\,28)) \cdot (1 - P(Id\,29)))$ – коефіцієнт настання правових вразливостей; $P(Un\,12) = 1 - ((1 - P(Un\,3)) \cdot (1 - P(Un\,4)) \cdot (1 - P(Un\,5)))$ – коефіцієнт вразливості неадекватного менеджменту проекту.

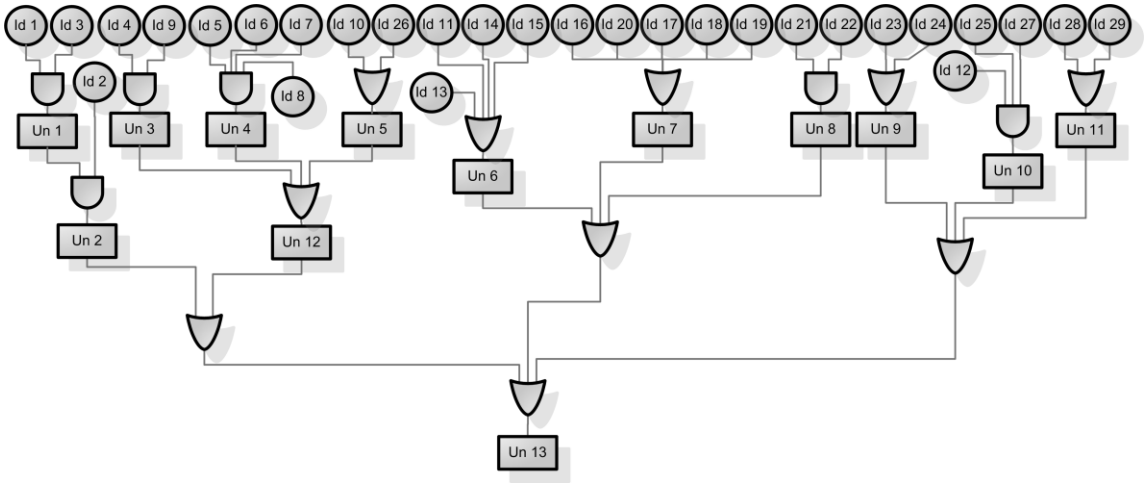

Рис. 4. Приклад дерева вразливостей розроблення ПЗ КС

Значення наведених вище коефіцієнтів, отриманих в результаті якісного аналізу вразливостей розроблення ПЗ КС на основі прикладу, наведені в табл. 3.

Таблиця 3

Значення коефіцієнтів дерева вразливостей розроблення ПЗ КС

Коефіцієнт вразливості	Ймовірність виникнення вразливості	Коефіцієнт вразливості	Ймовірність виникнення вразливості
*Un*1	0,5%	*Un*8	0,1%
*Un*2	0%	*Un*9	19,9%
*Un*3	0,3%	*Un*10	0%
*Un*4	0%	*Un*11	4,9%
*Un*5	10,7%	*Un*12	10,9%
*Un*6	13,4%	***Un*13**	**54,1%**
*Un*7	21,9%		

Проведені дослідження дозволили знайти такий підхід ймовірнісної оцінки вразливостей, який дозволив використати основні положення нечітко-множинної теорії при оцінці ключових показників результативності проекту. Можливість його використання оцінимо на прикладі показника чистої приведеної вартості C_{NPV} (Net Present Value). Проведені дослідження показали, що з урахуванням додаткових факторів вразливостей для розрахунку C_{NPV} доцільно використати наступний вираз:

$$C_{NVP} = \sum_{i=1}^{n} \frac{(B - AC_i)}{(1+r)^i} - \sum_{i=1}^{n} \frac{\left(\sum_{k=1}^{\ell} C_i^{(k)}\right)}{(1+r)^i} - \sum_{i=1}^{n} \frac{\left(\sum_{k=1}^{\ell} e^{-\lambda \cdot C_i^{(k)}}\right)}{(1+r)^i}, \quad (6)$$

де B – фінансові інвестиції, що поступають в процесі розроблення безпечного ПЗ КС в період i; AC_{i} – поточні витрати на підтримку і розвиток системи

розроблення в період i; l – кількість додаткових видів витрат на придбання, налаштування і модернізацію технічної, технологічної, програмної та інших складових в процесі розроблення безпечного ПЗ; $C_{\text{i}}^{(\text{k})}$ – витрати на врахування безпеки та тестування вразливостей ПЗ; λ – параметр впливу факторів безпеки та тестування вразливостей ПЗ на чисту приведену вартість проекту.

Відмінною особливістю математичного виразу (6) є введення додаткових складових $\sum_{i=1}^{n}\left(\sum_{k=1}^{\ell} C_i^{(k)}\right)\Big/(1+r)^i$; $\sum_{i=1}^{n}\left(\sum_{k=1}^{\ell} e^{-\lambda\cdot C_i^{(k)}}\right)\Big/(1+r_1)^i$, які характеризують врахування додаткових витрат на апаратну та програмну модернізацію компанії, удосконалення системи її управління, а також врахування безпеки та тестування вразливостей ПЗ. Оцінка ефективності проекту полягає у порівнянні C_{NVP} з деяким значенням $C_{\text{NVPreference}}$, яке визначає мінімально допустимий рівень приведеної вартості проекту розроблення безпечного ПЗ.

На рис. 5 представлено графік залежності [C_{NVP1}, C_{NVP2}] від значення фінансових інвестицій B, що поступають в процесі розроблення безпечного ПЗ в період i (графіки CN3 і CN4 ілюструють залежність за відсутності додаткових витрат на врахування безпеки і тестування вразливості ПЗ). Як видно з цього рисунка, введення додаткових витрат в процесі розроблення безпечного ПЗ до 1,5 разів підвищує чисту приведену вартість проекту.

На рис. 6 представлено графік залежності C_{NVP} від значення C_{i} показана залежність вартості проекту в залежності від вкладених коштів в засоби запобіганню вразливостям. Очевидно, при фінансуванні заходів уникнення вразливостей до певного моменту збільшують вартість проекту частина кривої AC за рахунок зменшення витрат на вирішення проблем, які є результатом вразливостей, які здійснилися. Однак, згодом ймовірність виникнення вразливих ситуацій зменшується частина кривої CB не так значно, тому фінансові вливання починають переважати отримані прибутки, це виражається в поступовому переході графіку знову до від'ємних величин. Таким чином маємо наявну оптимальну точку фінансування профілактичних заходів O_{BS}, при якій вартість проекту є максимальною. Для максимального значення C_{H}, мінімального C_{L} та моди C_{M} нечіткого значення ця точка оптимуму відрізняється на незначну величину (нечітке число).

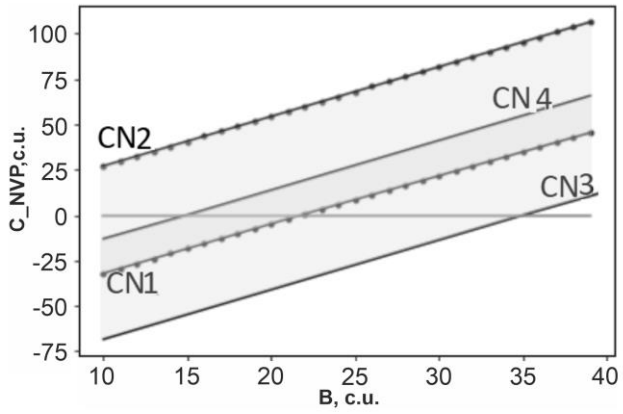

Рис. 5. Графіки залежностей [C_{NVP1}, C_{NVP2}] від значення фінансових інвестицій B

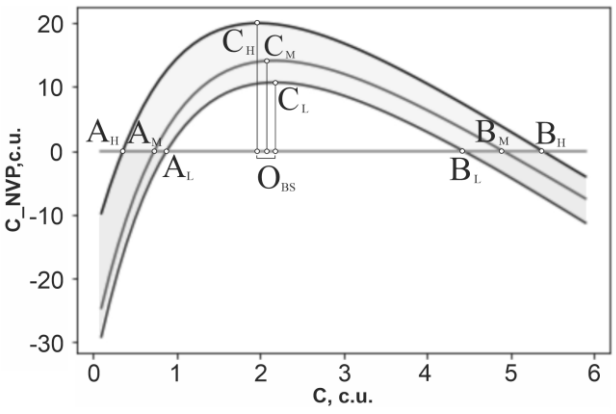

Рис. 6. Графік залежності C_{NVP} значення фінансових витрат C

Таким чином, використання удосконаленої методики "Аналізу дерева відмов" дозволить на 20% підвищити точність кількісної оцінки вразливостей розроблення ПЗ. В той же час, використання методики оцінки показника чистої приведеної вартості проекту розроблення безпечного ПЗ дозволяє оцінити ступінь вразливості як ймовірність отримання від'ємної вартості ітерації.

Для наведеного прикладу проекту розроблення ПЗ, за результатом чисельного інтегрування маємо нечітку ймовірність рівня вразливості, що вартість ітерації (спринту) буде від'ємною: $p_{вр}$=(0,034;0,219;0,421).

Це дозволяє розглядати проект комплексно, з урахуванням необхідності врахування безпеки і тестування вразливості ПЗ, із залученням інструментів, які дозволяють здолати складність, невизначеність і довгостроковість проектів.

У третьому розділі удосконалено метод оптимізації розподілу ресурсів розроблення безпечного ПЗ КС. При цьому завдання оптимізації розподілу ресурсів розроблення ПЗ за умови обмеженості засобів, виділених на усунення помилок безпеки, розглядається у вигляді напівмарківської моделі прийняття рішень для керованого процесу у безперервному часі з критерієм мінімуму витрат на усунення аномалій.

Ймовірності переходів розглянутого для системи розроблення безпечного ПЗ напівмарківського процесу прийняття рішень в моменти стрибків із стану i в стан j при прийнятті рішення $r \in R_i$ визначається стохастичною $(N+1)\times(N+1)$ матрицею $P^{(r)} = \left\{p_{ij}^{(r)}\right\}$, яка задає вкладений ланцюг Маркова.

Елементи $p_{ij}^{(r)}$ при будь-яких $i, j \in S$ і $r \in R_i$ дозволяють визначати спільну ймовірність $Q_{ij}^{(r)}(t)$ того, що тривалість перебування в стані i не перевершує час t із стану i при $r \in R_i$ процес переходить в стан j зі ймовірністю $p_{ij}^{(r)}$.

Функції $Q_{ij}^{(r)}(t)$ задовольняють умовам:

$$Q_{ij}^{(r)}(0) = 0,\ i,\ j \in S, r \in R_i; \tag{7}$$

$$\sum_{j\in S} Q_{ij}^{(r)}(\infty) = \sum_{j\in S} p_{ij}^{(r)} = 1,\ i, \in S, r \in R_i\,. \tag{8}$$

За допомогою матриці $Q^{(r)}(t) = \left\{Q_{ij}^{(r)}(t)\right\}$ перехідних розподілів визначимо функцію

$$H_i^{(r)}(t) = \sum_{j\in S} Q_{ij}^{(r)}(t), \quad i, \in S, r \in R_i\,, \tag{9}$$

що є функцією розподілу часу перебування процесу у стані i при прийнятті рішення $r \in R_i$.

Випадковий процес $(Z_t), t \ge 0$ зі значеннями $Z_t = i$, якщо в момент t система знаходиться в стані i, є напівмарківським, і задається величинами $N, y, Q_{ij}^{(r)}(t), i,\ j \in S, r \in R_i$.

Напівмарківський процес називається регулярним, якщо за кінцевий проміжок часу він із одиничною ймовірністю перейде у будь-який стан скінчену кількість разів.

Таким чином, регулярний напівмарківський процес за кінцевий проміжок часу завжди здійснює лише кінцеве число переходів. Далі розглядатимемо лише регулярні напівмарківські процеси.

У разі одноелементної множини рішень R_i в результаті стандартних для теорії відновлення суджень отримуємо наступне рівняння відновлення

$$\upsilon_i(t)=(1-H_i(t))\frac{k_i}{\alpha}(1-e^{-\alpha t})+\sum_{j\in S}\int_0^t\left(\frac{k_i}{\alpha}(1-e^{-\alpha t})+e^{-\alpha t}\upsilon_j(t-\tau)\right)dQ_{ij}(\tau),\ i\in S,$$

де $\upsilon_i(t)$ – короткий запис сумарної середньої витрати $\upsilon_i(t,\beta)$ за час *t*.

В разі скінченних множин R_i рівняння відновлення з урахуванням ймовірностей $d_i^{(r)}$ прийняття рішень r у стані *i* запишемо у вигляді

$$\begin{aligned}\upsilon_i(t)&=\sum_{r\in R_i}d_i^r(1-H_i^{(r)}(t))\frac{k_i^{(r)}}{\alpha}(1-e^{-\alpha t})+\\&+\sum_{j\in S}\sum_{r\in R_i}\int_0^t d_i^r\left(\frac{k_i^{(r)}}{\alpha}(1-e^{-\alpha t})+e^{-\alpha t}\upsilon_j(t-\tau)\right)dQ_{ij}^{(r)}(\tau),\ i\in S,\end{aligned}\qquad(10)$$

де $k_i^{(r)}$ – витрата системи за одиницю часу перебування у стані *i* при рішенні $r\in R_i$; $\upsilon_j(t)$ – сумарна середня витрата з урахуванням переоцінки, за умови, що процес починається в момент $t=0$ зі стану *j*.

Величини $\upsilon_i(\beta)$ з виразу можна записати у вигляді $\upsilon_i(\alpha)$, і для цього рівняння скористатися основними положеннями рівняння (інтеграла) Лапласа-Стілтьєса.

Відповідно до робіт, для будь-якої функції $F(t)$, похідна $F'(t)$ якої є функцією-оригіналом, що задовольняє нерівності $F'(t)<Ce^{\alpha t}$ для всіх $t<0$, при всіх комплексних *s*, коли $\operatorname{Re} s>\alpha$ існує функція

$$F^*(s)=L_s^*\langle F(t)\rangle=\int_0^\infty e^{-st}dF(t),\qquad(11)$$

тобто функція e^{-st} при $\operatorname{Re} s>\alpha$ інтегрована за функцією *F*(*t*). Функцію $F^*(s)$ називають перетворенням Лапласа-Стілтьєса функції *F*(*t*).

З виразів (8), (9) слідує, що $H_i^{(r)}(\infty)=1,\ \ i\in S, r\in R_i$, тому перша сума у виразі (10) при $t\to\infty$ обертається на нуль. Інтегруючи по частинах вираз (11) для $L_s^*\langle F(t)\rangle$, отримуємо

$$sL_s^*\langle F(t)\rangle=L_s^*\langle F(t)\rangle-F(0),\qquad(12)$$

де $F(s)=L_s\langle F(t)\rangle=\int_0^\infty e^{-st}F(t)\,dt$.

З перетворення Лапласа функції *F*(*t*). З (12) при s≠0 знаходимо

$$L_s\langle F(t)\rangle=\frac{1}{s}\left(L_s^*\langle F(t)\rangle-F(0)\right).\qquad(13)$$

Інтегруємо по частинах з урахуванням виразу (9), знаходимо

$$\sum_j\int_0^t(1-e^{-\alpha t})\,dQ_{ij}^{(r)}(\tau)=(1-e^{-\alpha t})\sum_j dQ_{ij}^{(r)}(\tau)\Big|_0^t-\sum_j\alpha\int_0^t e^{-\alpha t}H_i(\tau)\,d\tau.\qquad(14)$$

Переходячи у виразі (14) до границі $t \to \infty$ і застосовуючи формулу (13) для $s = \alpha, (\alpha > 0)$, з урахуванням співвідношень (7,8) отримаємо

$$\sum_j \int_0^t \left(1 - e^{-\alpha t}\right) dQ_{ij}^{(r)}(\tau) = (1-\alpha) L_{s=\alpha}\left\langle H_i^{(r)}(\tau)\right\rangle = 1 - \alpha \frac{1}{\alpha} L_{s=\alpha}^*\left\langle H_i^{(r)}(\tau)\right\rangle = 1 - h_i^{(r)}(\alpha), \qquad (15)$$

де $h_i^{(r)}(\alpha) = L_{s=\alpha}^*\left\langle H_i^{(r)}(t)\right\rangle$.

Застосовуючи до функції $\Phi_i^{(r)}(t) = \int_0^r e^{-\alpha t} \upsilon_j(t-\tau)\, dQ_{ij}^{(r)}(t)$ теорему про граничний перехід в інтегралі по параметру, від якого залежать межі інтегрування і підінтегральна функція, при $t \to \infty$ отримуємо

$$\Phi_i^{(r)}(\infty) = \int_0^\infty e^{-\alpha t} \upsilon_j(\alpha)\, dQ_{ij}^{(r)}(\tau) = \nu_j(\alpha)\, q_{ij}^{(r)}(\alpha), \qquad (16)$$

де $q_{ij}^{(r)}(\alpha) = L_{s=\alpha}^*\left\langle Q_i^{(r)}(\alpha)\right\rangle$. Переходячи у вираженні (10) до границі при $t \to \infty$, з урахуванням (15,16) отримуємо наступний аналітичний вираз

$$\upsilon_i(t) = \sum_{r \in R_i} d_i^{(r)}\left(\zeta_i^{(r)}(\alpha)\right) + \sum_{j \in S} q_{ij}^{(r)}(\alpha)\, \upsilon_j(\alpha), \qquad (17)$$

де

$$\zeta_i^{(r)}(\alpha) = \frac{k_i^{(r)}}{\alpha}\left(1 - h_i^{(r)}(\alpha)\right). \qquad (18)$$

Нехай $\zeta_i(\alpha) = \sum_{r \in R_i} d_i^r\left(\rho_i^{(r)}(\alpha)\right)$ і $\Im(\alpha) = \left(\zeta_0(\alpha), \ldots, \zeta_N(\alpha)\right)^T$, $\wp(\alpha) = \left(\upsilon_0(\alpha), \ldots, \upsilon_N(\alpha)\right)^T$, ($T$ – символ транспонування матриці). Тоді отримуємо

$$\wp(\alpha) = \Im_0(\alpha) + q(\alpha)\wp(\alpha), \qquad (19)$$

де $q(\alpha) = \left\{q_{ij}(\alpha)\right\}$, $q_{ij}(\alpha) = \sum_{r \in R_i} d_i^{(r)}\left(q_{ij}^{(r)}(\alpha)\right)$.

З виразу (19) знайдемо

$$\wp(\alpha) = \left\{I - q(\alpha)\right\}^{-1} \Im_0(\alpha). \qquad (20)$$

Даний вираз справедливий, оскільки при $\alpha > 0$ матриця $\left\{I - q(\alpha)\right\}$ – невироджена, I – одинична матриця розміру $(N \times 1) \times (N \times 1)$.

Помноживши обидві частини рівності (19) ліворуч на вектор y, отримаємо наступне

$$\begin{gathered} y\upsilon(\alpha) = \sum_{i \in S} \sum_{j \in \tilde{S}} \sum_{r \in R_i} y_i \mu_{ij}(\alpha) \zeta_j^{(r)}(\alpha)\, d_i^{(r)}, \\ \left\{I - q(\alpha)\right\}^{-1} = \left\{\mu_{ij}(\alpha)\right\}. \end{gathered} \qquad (21)$$

Величини $\mu_{ij}(\alpha)$ залежать від $d_i^{(r)}$, $r \in R_i$, $i \in S$, так як елементи матриці $\left\{I - q(\alpha)\right\}$ можна виразити через $d_i^{(r)}$, $r \in R_i$, $i \in S$.

Нехай $\left\{d_i^{(r)}\right\}$ $\left(r \in R_i\right)$ – нерандомізована марківська стаціонарна стратегія системи розроблення ПЗ у стані j $d_j^{(r)} \in \{0,1\}$, $\sum_{j \in S} d_j^{(r)} = 1$, при цьому заданої x як $x_{00} = 1$, $x_{rj} = d_j^{(r)}$, $r \in R_i$, $j \in \tilde{S}$.

Мінімізація витрат приводить до наступної задачі оптимізації для булевих змінних $\mathrm{X}=\{x_{rj}\},\ \ r\in R_i,\ \ j\in\widetilde{S}$.

$$f(\alpha,\mathrm{X})=\sum_{i\in S}\sum_{j\in\widetilde{S}}\sum_{r\in R_i}y_i\mu_{ij}(\alpha,\mathrm{X})\zeta_j^{(r)}x_{rj}\to\min \tag{22}$$

$$\sum_{r\in R_i}x_{rj}=1,\ j\in\widetilde{S} \tag{23}$$

$$\sum_{j\in\widetilde{S}}c_{rj}x_{rj}\le b_r,\ r\in R_i,\ \ j\in\widetilde{S} \tag{24}$$

$$x_{rj}\in\{0,1\},\ j\in\widetilde{S},\ r\in R_i \tag{25}$$

Наведене вище завдання оптимізації для булевих змінних $\mathrm{X}=\{x_{rj}\}$, (22), (25) включає до себе множину псевдобулевих нерівностей (24), (25).

Додаткова умова (23) розширює межі системи, при цьому можна записати, що $\mathrm{X}_r^{(k)}=\{x_{r1}^{(k)},\ldots,x_{rN}^{(k)}\},\ k=1,\ldots k_r$ – множина допустимих рішень r-ої нерівності запропонованої системи.

Для побудови рішень удосконаленої системи при відомих допустимих рішеннях (24), можна зменшити кількість дій використавши наступні обмеження. Представимо рішення удосконаленої системи як $Z=\{s_j\},\ \ j=1,\ldots N$, де s_j – множина номерів r, для яких допустимо обмеження $x_{rj}=1$. Для рішення цієї системи потрібно m кроків, де m – число обмежень (24).

У початковому стані кожна з $s_j^{(0)}$ вектору $Z^{(o)}$ включає усі можливі значення $r\in R_i$. На r-му кроці відбувається перетин вектору $Z^{(r-1)}$ з одним з рішень множини r-ої нерівності.

Допускаючи, що умови r-ій нерівності відповідає $r=r_1$, а також, що α_j є j-м елементом множини допустимих рішень r-ої нерівності, $\alpha_j\in\{0,1,\varphi\}$, де φ – невизначений параметр з множини $\{0,1\}$, сформулюємо обмеження для r-го кроку алгоритму побудови рішень удосконаленої системи.

1. Якщо значення α_j не фіксоване, то $s_j^{(r)}=s_j^{(r-1)}$.

2. Якщо $\alpha_j=1$, то при $r_1\in s^{(r-1)}$ приймаємо $s^{(r)}=\{r_1\}$, а при $r_1\overline{\in}s^{(r-1)}$ приймаємо $s^{(r)}$ дорівнює пустій множині.

3. Якщо $\alpha_j=0$, то $s^{(r)}=s^{(r-1)}/\{r_1\}$.

Пошук рішень розподілу ресурсів розроблення ПЗ здійснюється з урахуванням (23).

На m кроці алгоритму реалізується вектор $Z^{(m)}=\{\alpha_1^{(m)},\ldots,\alpha_N^{(m)}\}$, кожна компонента $\alpha_j^{(m)}$, якого є найпростішою множиною $\{r\},\ \ r\in R,\ \ R=\{1,\ldots,m\}$, і отже, $Z^{(m)}$ є рішенням удосконаленої системи.

Якщо з допомогою удосконаленої системи існує можливість отримання декількох рішень розподілу ресурсів розроблення безпечного ПЗ необхідно знайти сукупність усіх рішень удосконаленої системи, та обрати з них оптимальне рішення, що доставляє мінімум цільової функції $f(\alpha,\mathrm{X})$.

Це рішення може знаходитися різними відомими методами лінійного програмування, або просто шляхом безпосереднього порівняння значень $f(\alpha, X)$ при визначенні удосконаленої системи.

У дисертаційній роботі представлена чисельна реалізація викладеного методу оптимізації розподілу ресурсів розроблення безпечного ПЗ КС для напівмарківської моделі прийняття рішень при аномальних ситуаціях безпеки. Як приклад, розглянуто ситуації виникнення помилок безпеки ПЗ, і визначено оптимальну стратегію оптимізації для усунення вказаної аномальної ситуації.

У четвертому розділі представлено результати дослідження і алгоритми тестування на вразливість до одних з найбільш поширених видів атак на *Web*-застосунки – *DOM XSS* і *SQL*-ін'єкції.

Розроблено комплекс математичних моделей процесу тестування *Web*-застосунків. В основу математичного моделювання покладено підхід мережевого *GERT* моделювання. В результаті розроблено математичні моделі технології тестування комплексу *DOM XSS* вразливостей і технології тестування вразливості до *SQL*-ін'єкцій.

Відповідно до представленого опису розроблено мережеву *GERT*-модель технології тестування комплексу *DOM XSS* вразливостей. Графічне зображення *GERT*-моделі представлене на рис. 7.

У представленій мережі вузли графа інтерпретуються станами комп'ютерної системи в процесі функціонування *DOM* структури, а гілки графа – ймовірнісно-часовими характеристиками переходів між станами. Зокрема, гілка (1,2) описує процес отримання і аналізу вмісту тега. Гілка (2,3) відображає процес виконання атаки у разі наявності "*source*" структури. Гілка (2,4) визначається процедурами звернення до вмісту віддаленого файлу (пошук "*sink*"). Гілка (4,2) характеризує повернення на виконання атаки. Гілка (3,5) описує продовження атаки, зокрема перевірку вмісту *DOM*. Гілка (5,6) характеризує одну з основних особливостей аналізу алгоритму *XSS* вразливостей різних типів – автоматичний аудит коду (при необхідності віддалений). Гілка (6,1) відображає процес переходу до нового тега. Далі гілка (5,7) характеризує завершальну стадію прийняття рішення про вразливість.

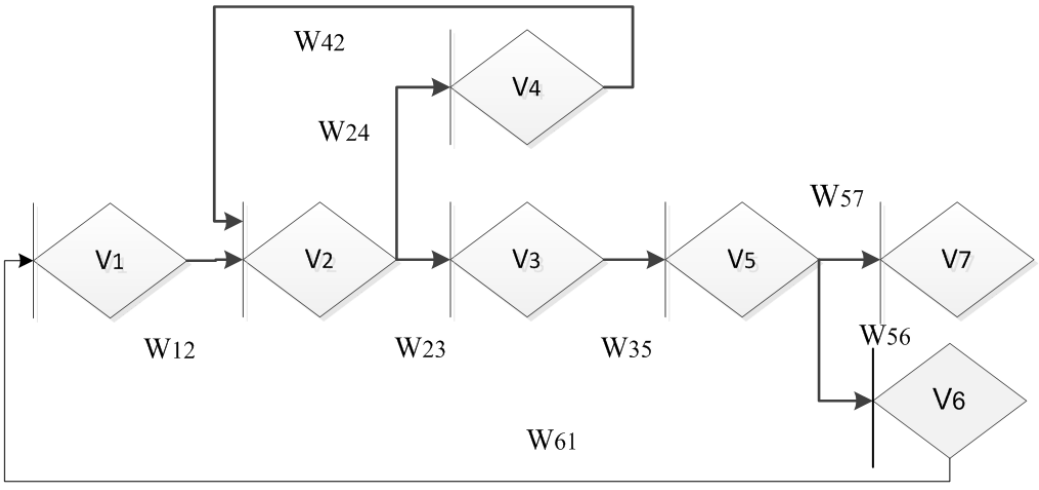

Рис. 7. *GERT*-модель технології тестування комплексу *DOM XSS* вразливостей

Характеристики гілок моделі представлені в табл. 4. Еквівалентна *W*-функція часу виконання алгоритму тестування комплексу *DOM XSS* різних типів (у тому числі *DOM Based XSS*) вразливостей дорівнює:

$$W_E(s) = \frac{W_{12}W_{23}W_{35}W_{56} + W_{12}W_{24}W_{42}W_{23}W_{35}W_{57}}{1 - W_{12}W_{23}W_{35}W_{56}W_{61} - W_{12}W_{24}W_{42}W_{23}W_{35}W_{56}W_{61}} =$$

$$= \frac{p_1 p_2^2 \lambda_1 \lambda_2^2 \left(p_4 \lambda_4 (\lambda_3 - s)^2 (\lambda_5 - s) + p_3^2 q_1 \lambda_3^2 \lambda_5 (\lambda_4 - s)\right)}{(\lambda_4 - s)\begin{pmatrix}(\lambda_1 - s)(\lambda_2 - s)^2 (\lambda_3 - s)^2 (\lambda_5 - s) - \\ - p_1 \lambda_1 p_2^2 \lambda_2^2 q_1 \lambda_5 (\lambda_3 - s)^2 - p_1 p_2^2 p_3^2 p_4 \lambda_1 \lambda_2^2 q_1 \lambda_3^2 \lambda_4 \lambda_5\end{pmatrix}} \qquad (26)$$

де $1 - p_4 = q_1$.

Таблиця 4

Характеристики гілок моделі технології тестування комплексу *DOM XSS* вразливостей

№ з/п	Гілка	*W*-функція	Ймовірність	Твірна функція моментів
1	(1,2)	W_{12}	$p1$	$\lambda_1/(\lambda_1 - s)$
2	(2,3)	W_{23}	$p2$	$\lambda_2/(\lambda_2 - s)$
3	(2,4)	W_{24}	$p3$	$\lambda_3/(\lambda_3 - s)$
4	(3,5)	W_{35}	$p2$	$\lambda_2/(\lambda_2 - s)$
5	(5,6)	W_{56}	$p4$	$\lambda_4/(\lambda_4 - s)$
6	(6,1)	W_{61}	$1 - p4$	$\lambda_5/(\lambda_5 - s)$
7	(4,2)	W_{42}	$p3$	$\lambda_3/(\lambda_3 - s)$
8	(5,7)	W_{57}	$p4$	$\lambda_4/(\lambda_4 - s)$

Виконуючи комплексне перетворення $z = -is$, отримаємо:

$$\Phi(z) = \frac{uz^3 + vz^2 + bz + k}{(\lambda_4 + z)(z^6 + cz^5 + dz^4 + gz^3 + hz^2 + wz + m)}, \qquad (27)$$

де $u = -p_1 p_2^2 p_4 \lambda_1 \lambda_2^2 \lambda_4$,

$v = p_1 p_2^2 p_4 \lambda_1 \lambda_2^2 \lambda_4 (\lambda_5 + 2\lambda_3)$,

$b = -p_1 p_2^2 p_4 \lambda_1 \lambda_2^2 \lambda_4 \lambda_3 (2\lambda_5 - \lambda_3)$,

$k = -p_1 p_2^2 \lambda_1 \lambda_2^2 \lambda_3^2 \lambda_4 \lambda_5 (p_4 + p_3^2 q_1)$,

$c = \lambda_1 + 2\lambda_2 + 2\lambda_3 + \lambda_4 + \lambda_5$,

$d = -(2\lambda_3\lambda_5\lambda_4 + \lambda_1\lambda_5\lambda_4 + 2\lambda_2\lambda_5\lambda_4 + \lambda_3^2 + 2\lambda_1\lambda_3 + 4\lambda_2\lambda_3 + 2\lambda_1\lambda_2 + \lambda_2^2)$,

$g = \begin{pmatrix}\lambda_3^2\lambda_4\lambda_5 + 4\lambda_1\lambda_2\lambda_4\lambda_5 + 4\lambda_2\lambda_3\lambda_4\lambda_5 + \lambda_2^2 + \lambda_3^2\lambda_1 + 2\lambda_3^2\lambda_2 + 4\lambda_1\lambda_2\lambda_3\lambda_4 + \\ + 2\lambda_2^2\lambda_3 + \lambda_2^2\lambda_1 + \lambda_3^2\lambda_4 + \lambda_2^2\lambda_4\end{pmatrix}$,

$h = -\begin{pmatrix}\lambda_1\lambda_3^2\lambda_4\lambda_5 + 2\lambda_2\lambda_3^2\lambda_4\lambda_5 + 4\lambda_1\lambda_2\lambda_3\lambda_4\lambda_5 + 2\lambda_2^2\lambda_3\lambda_4\lambda_5 + \lambda_2^2\lambda_3^2\lambda_4 + \\ + 2\lambda_1\lambda_2^2\lambda_3\lambda_4 - p_1 p_2^2 p\ q_1\lambda_1\lambda_2\lambda_4\lambda_5\end{pmatrix}$,

$w = \lambda_1\lambda_2\lambda_3^2\lambda_4\lambda_5 + \lambda_2^2\lambda_3^2\lambda_4\lambda_5 + 2\lambda_1\lambda_2^2\lambda_3\lambda_4\lambda_5 + \lambda_1\lambda_2^2\lambda_4\lambda_3 - 2p_1p_2^2p_4q_1\lambda_1\lambda_2\lambda_3\lambda_4\lambda_5$,

$m = p_1 p_2^2 p_4 q_1 \lambda_1 \lambda_2 \lambda_3^2 \lambda_4 \lambda_5 + p_1 p_2^2 p_3 p_4 q_1 \lambda_1 \lambda_2^2 \lambda_3^2 \lambda_4 \lambda_5 - \lambda_1 \lambda_2^2 \lambda_3 \lambda_4 \lambda_5$.

Щільність розподілу ймовірностей часу виконання алгоритму аналізу *DOM XSS* вразливості:

$$\varphi(t)=\frac{1}{2\pi i}\int_{-i\infty}^{i\infty} e^{zt}\frac{uz^3+vz^2+bz+j}{(\lambda_4+z)(z^6+cz^5+dz^4+gz^3+hz^2+wz+m)}dz. \qquad (28)$$

На рис. 8 представлено криву щільності розподілу $\varphi(t)$ ймовірності часу виконання алгоритму виявлення комплексу *DOM XSS* вразливостей для приведених вище умов.

Зовнішній вигляд графіку дає підстави припустити, що випадкова величина часу виконання алгоритму аналізу *DOM XSS* вразливості має гамма-розподіл. На рис. 9 представлена крива розподілу ймовірності загального часу тестування в годинах комплексу *DOM XSS* вразливостей в концепції ітерації гнучкої методологій розроблення ПЗ.

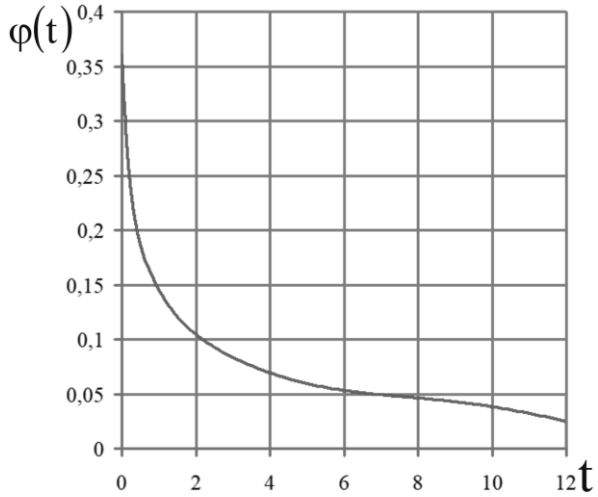

Рис. 8. Щільність ймовірності загального часу тестування в годинах комплексу *DOM XSS* вразливостей

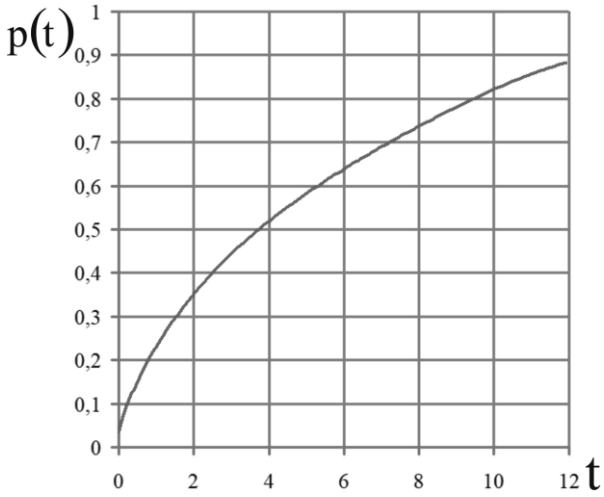

Рис. 9. Розподіл ймовірності загального часу тестування в годинах комплексу *DOM XSS* вразливостей

В дисертаційній роботі, відповідно до представленого опису, побудовано мережеву *GERT*-модель технології тестування вразливості до *SQL*-ін'єкцій. Графічне зображення *GERT*-моделі представлене на рис. 10

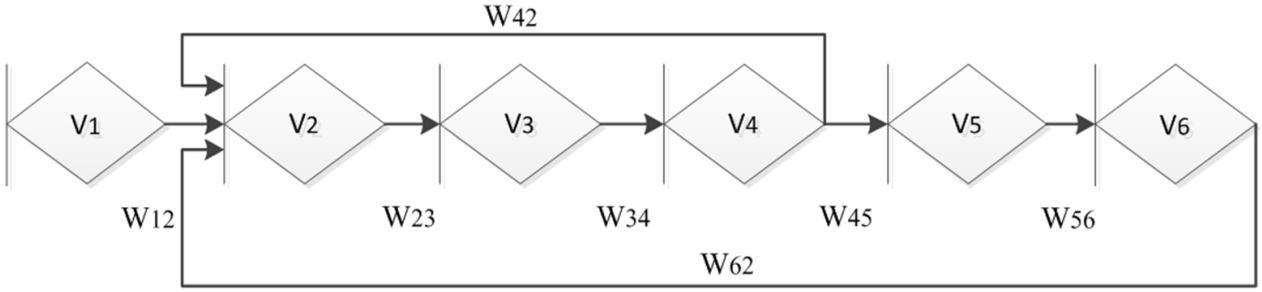

Рис. 10. *GERT*-модель технології тестування вразливості до *SQL*-ін'єкцій

У представленій мережі вузли графа інтерпретуються станами комп'ютерної системи в процесі тестування вразливості до *SQL*-ін'єкцій, а гілки графа – ймовірнісно-часовими характеристиками переходів між станами.

Зокрема, гілка (1,2) характеризує час отримання і аналізу *GET*-параметрів з введеного *URL*-посилання. Гілка (2,3) відображає час відправлення первинних і вторинних запитів до *Web*-сторінки. Гілка (3,4) задає випадковий час порівняння сторінок (час обчислення відстані між змістом *HTML*-коду сторінки за допомогою критерію Джаро-Вінклера). Гілка (4,5) характеризує час, за який виконується ін'єкція *SQL*-коду, який не змінює результат запиту до бази даних, а також який змінює результат запиту до бази даних відповідно. Далі гілка (5,6) характеризує час порівняння результатів ін'єкції *SQL*-коду. Гілка (4,2) характеризує часові характеристики повернення системи в первинний стан, коли значення критерію Джаро-Вінклера менше певного порогового значення, в той же час гілка (6,2) відображає часові характеристики переходу до нової перевірки у випадку, якщо значення критерію Джаро-Вінклера більше певного порогового значення. Характеристики гілок моделі представлені в табл. 5.

Таблиця 5

Характеристики гілок моделі технології тестування вразливості до *SQL*-ін'єкцій

№ з/п	Гілка	*W*-функція	Ймовірність	Твірна функція моментів
1	(1,2)	W_{12}	p_1	$\lambda_1/(\lambda_1 - s)$
2	(2,3)	W_{23}	p_2	$\lambda_2/(\lambda_2 - s)$
3	(3,4)	W_{34}	p_3	$\lambda_3/(\lambda_3 - s)$
4	(4,5)	W_{45}	p_4	$\lambda_4/(\lambda_4 - s)$
5	(5,6)	W_{56}	p_5	$\lambda_3/(\lambda_3 - s)$
6	(4,2)	W_{42}	$1 - p_4$	$\lambda_5/(\lambda_5 - s)$
7	(6,2)	W_{62}	p_6	$\lambda_6/(\lambda_6 - s)$

Еквівалентна *W*-функція часу виконання технології тестування вразливості до *SQL*-ін'єкцій дорівнює:

$$W_E(s) = \frac{W_{12}W_{23}W_{34}W_{45}W_{56}}{1 - W_{12}W_{23}W_{34}W_{42} - W_{12}W_{23}W_{34}W_{45}W_{56}W_{62}} =$$

$$= \frac{p_1 p_2 p_3 p_4 p_5 \lambda_1 \lambda_2 \lambda_3^2 \lambda_4 (\lambda_3 - s)(\lambda_5 - s)(\lambda_6 - s)}{\left[\begin{array}{l}(\lambda_1 - s)(\lambda_2 - s)(\lambda_3 - s)^2(\lambda_4 - s)(\lambda_5 - s)(\lambda_6 - s) - \\ - \left(\begin{array}{l} p_1 p_2 p_3 \lambda_1 \lambda_2 \lambda_3 \cdot \\ \cdot \left(q_1 \lambda_5 (\lambda_3 \lambda_4 - \lambda_4 s - \lambda_3 s - s^2)(\lambda_6 - s) - p_4 p_5 p_6 \lambda_3 \lambda_4 \lambda_6 (\lambda_5 - s)\right) \end{array}\right) \end{array}\right]} \quad (29)$$

де $1 - p_4 = q_1$.

Виконуючи комплексне перетворення $z = -is$, отримаємо:

$$\Phi(z) = \frac{vz^2 + bz + k}{\left(z^7 + rz^6 + cz^5 + dz^4 + gz^3 + hz^2 + wz + m\right)} \quad (30)$$

де $v = -p_1 p_2 p_3 p_4 p_5 \lambda_1 \lambda_2 \lambda_3^2 \lambda_4$,

$$b = p_1 p_2 p_3 p_4 p_5 \lambda_1 \lambda_2 \lambda_3^2 \lambda_4 (\lambda_5 + \lambda_6),$$

$$k = -p_1 p_2 p_3 p_4 p_5 \lambda_1 \lambda_2 \lambda_3^2 \lambda_4 \lambda_5 \lambda_6,\ r = \lambda_1 + \lambda_2 + \lambda_4 + \lambda_5 - 2\lambda_3 - \lambda_6,$$

$$c = \begin{pmatrix} \lambda_1\lambda_4 + \lambda_2\lambda_4 + \lambda_1\lambda_5 + \lambda_2\lambda_5 + \lambda_3^2 + 2\lambda_3\lambda_6 - \lambda_4\lambda_6 - \lambda_5\lambda_6 - \lambda_1\lambda_6 - \lambda_4\lambda_5 - \\ -2\lambda_3\lambda_4 - 2\lambda_3\lambda_5 - \lambda_1\lambda_2 - 2\lambda_1\lambda_3 - 2\lambda_2\lambda_3 \end{pmatrix},$$

$$d = -\lambda_1\lambda_2\lambda_3\lambda_4\lambda_5\lambda_6 \left(\begin{array}{l} \frac{1}{\lambda_2\lambda_3\lambda_5} + \frac{1}{\lambda_1\lambda_3\lambda_5} + \frac{1}{\lambda_2\lambda_3\lambda_4} + \frac{1}{\lambda_1\lambda_3\lambda_4} + \frac{\lambda_3}{\lambda_1\lambda_2\lambda_4\lambda_5} + \\ + \frac{1}{\lambda_2\lambda_3\lambda_6} + \frac{1}{\lambda_1\lambda_3\lambda_6} + \frac{1}{\lambda_3\lambda_5\lambda_6} + \frac{2}{\lambda_2\lambda_5\lambda_6} + \frac{2}{\lambda_1\lambda_5\lambda_6} + \frac{1}{\lambda_3\lambda_4\lambda_6} + \\ + \frac{2}{\lambda_2\lambda_4\lambda_6} + \frac{2}{\lambda_1\lambda_4\lambda_6} - \frac{1}{\lambda_1\lambda_2\lambda_3} - \frac{2}{\lambda_1\lambda_2\lambda_5} - \frac{2}{\lambda_1\lambda_2\lambda_4} - \frac{1}{\lambda_3\lambda_4\lambda_5} - \\ - \frac{2}{\lambda_2\lambda_4\lambda_5} - \frac{2}{\lambda_1\lambda_2\lambda_6} - \frac{\lambda_3}{\lambda_1\lambda_2\lambda_5\lambda_6} - \frac{\lambda_3}{\lambda_1\lambda_2\lambda_4\lambda_6} - \frac{2}{\lambda_4\lambda_5\lambda_6} - \\ - \frac{\lambda_3}{\lambda_2\lambda_4\lambda_5\lambda_6} - \frac{\lambda_3}{\lambda_1\lambda_4\lambda_5\lambda_6} \end{array} \right),$$

$$g = \left(\lambda_1\lambda_2\lambda_3\lambda_4\lambda_5\lambda_6 \cdot \left(\begin{array}{l} \frac{2}{\lambda_1\lambda_2} + \frac{\lambda_3}{\lambda_1\lambda_2\lambda_5} + \frac{\lambda_3}{\lambda_1\lambda_2\lambda_4} + \frac{2}{\lambda_4\lambda_5} + \frac{\lambda_3}{\lambda_2\lambda_4\lambda_5} + \frac{\lambda_3}{\lambda_1\lambda_4\lambda_5} + \\ + \frac{\lambda_3}{\lambda_1\lambda_2\lambda_6} - \frac{1}{\lambda_2\lambda_3} - \frac{1}{\lambda_1\lambda_3} - \frac{1}{\lambda_3\lambda_5} - \frac{2}{\lambda_2\lambda_5} - \frac{2}{\lambda_1\lambda_5} - \frac{1}{\lambda_3\lambda_4} - \\ - \frac{2}{\lambda_2\lambda_4} - \frac{2}{\lambda_1\lambda_4} - \frac{1}{\lambda_3\lambda_6} - \frac{2}{\lambda_2\lambda_6} - \frac{2}{\lambda_1\lambda_6} - \frac{2}{\lambda_5\lambda_6} - \frac{\lambda_3}{\lambda_2\lambda_5\lambda_6} - \\ - \frac{\lambda_3}{\lambda_1\lambda_5\lambda_6} - \frac{2}{\lambda_4\lambda_6} - \frac{\lambda_3}{\lambda_2\lambda_4\lambda_6} - \frac{\lambda_3}{\lambda_1\lambda_4\lambda_6} - \frac{\lambda_3}{\lambda_4\lambda_5\lambda_6} \end{array} \right) \right) + p_1 p_2 p_3 q_1 \lambda_1\lambda_2\lambda_3\lambda_5,$$

$$h = -\left(\begin{array}{l} \lambda_1\lambda_2\lambda_3\lambda_4\lambda_5\lambda_6 \cdot \\ \cdot \left(\begin{array}{l} \frac{\lambda_3}{\lambda_1\lambda_2} + \frac{\lambda_3}{\lambda_4\lambda_5} - \frac{1}{\lambda_3} - \frac{2}{\lambda_2} - \frac{2}{\lambda_1} - \frac{2}{\lambda_5} - \\ - \frac{\lambda_3}{\lambda_2\lambda_5} - \frac{\lambda_3}{\lambda_1\lambda_5} - \frac{2}{\lambda_4} - \frac{\lambda_3}{\lambda_2\lambda_4} - \frac{\lambda_3}{\lambda_1\lambda_4} - \\ - \frac{2}{\lambda_6} - \frac{\lambda_3}{\lambda_2\lambda_6} - \frac{\lambda_3}{\lambda_1\lambda_6} - \frac{\lambda_3}{\lambda_5\lambda_6} - \frac{\lambda_3}{\lambda_4\lambda_6} \end{array} \right) \end{array} \right) - \left(\begin{array}{l} p_1 p_2 p_3 q_1 \lambda_1\lambda_2\lambda_3\lambda_5\lambda_6 \cdot \\ \cdot \left(\frac{1}{\lambda_5\lambda_6} + \frac{\lambda_3}{\lambda_4\lambda_5\lambda_6} - \frac{1}{\lambda_4\lambda_5} \right) \end{array} \right),$$

$$w = \left(\begin{array}{l} \lambda_1\lambda_2\lambda_3\lambda_4\lambda_5\lambda_6 \cdot \\ \cdot \left(\begin{array}{l} 2 + \frac{\lambda_3}{\lambda_2} + \frac{\lambda_3}{\lambda_1} + \\ + \frac{\lambda_3}{\lambda_5} + \frac{\lambda_3}{\lambda_4} + \frac{\lambda_3}{\lambda_6} \end{array} \right) \end{array} \right) - \left(\begin{array}{l} p_1 p_2 p_3 q_1 \lambda_1\lambda_2\lambda_3\lambda_5\lambda_6 \cdot \\ \cdot \left(-1 - \frac{\lambda_3}{\lambda_4} - \frac{\lambda_3}{\lambda_6} \right) \end{array} \right) + p_4 p_5 p_6 \lambda_3\lambda_4\lambda_6,$$

$$m = -\lambda_1\lambda_2\lambda_3^2\lambda_5\lambda_6\left(\lambda_4 + p_1p_2p_3q_1 - \frac{p_4p_5p_6}{\lambda_1\lambda_2}\right).$$

Щільність розподілу ймовірності часу виконання технології тестування вразливості до *SQL*-ін'єкцій дорівнює:

$$\varphi(t) = \frac{1}{2\pi i}\int_{-i\infty}^{i\infty} e^{zt}\frac{vz^2 + bz + k}{\left(z^7 + rz^6 + cz^5 + dz^4 + gz^3 + hz^2 + wz + m\right)}dz. \qquad (31)$$

Тоді вираз $e^{zx}\Phi(z)$ можна представити у вигляді:

$$e^{zx}\Phi(z) = \frac{e^{zx}\left(vz^2 + bz + k\right)}{z^7 + rz^6 + cz^5 + dz^4 + gz^3 + hz^2 + wz + m} = \frac{\mu(z)}{\psi(z)}. \qquad (32)$$

Тоді щільність розподілу часу виконання алгоритму тестування вразливості до *SQL*-ін’єкцій дорівнює:

$$\varphi(x) = \sum_{k=1}^{7}\mathrm{Res}\left[e^{zx}\Phi(z)\right] = \sum_{k=1}^{7}\frac{\mu(z_k)}{\psi(z_k)} = \\ = \sum_{k=1}^{7}\frac{e^{zx}\left(vz^2 + bz + k\right)}{7z_k^6 + 6rz_k^5 + 5cz_k^4 + 4dz_k^3 + 3gz_k^2 + 2hz_k + w}. \qquad (33)$$

На рис. 10 представлено крива щільності розподілу $\varphi(t)$ ймовірності часу виконання технології тестування вразливості до *SQL*-ін'єкцій для приведених вище умов.

Зовнішній вигляд графіку дає підстави припустити, що випадкова величина часу виконання технології тестування вразливості до *SQL*-ін'єкцій відповідає гамма-розподілу (близьке до експоненційного).

На рис. 11 представлена крива ймовірності загального часу тестування в годинах вразливості до *SQL*-ін’єкцій в концепції ітерації гнучкої методологій розроблення ПЗ.

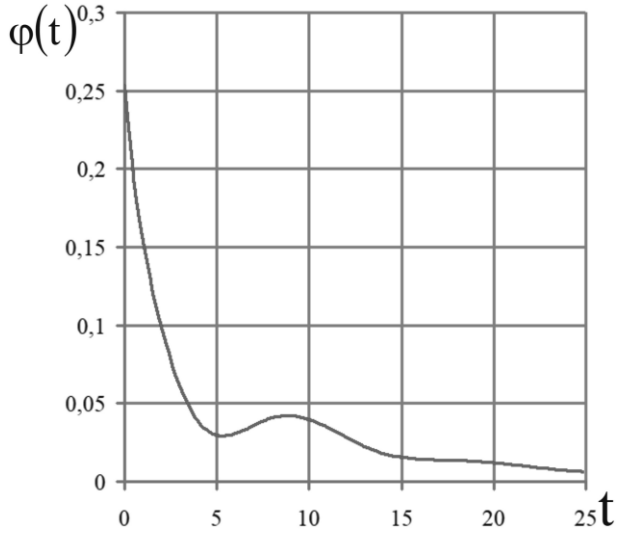

Рис. 10. Щільність ймовірності загального часу тестування в годинах вразливості до *SQL*-ін’єкцій

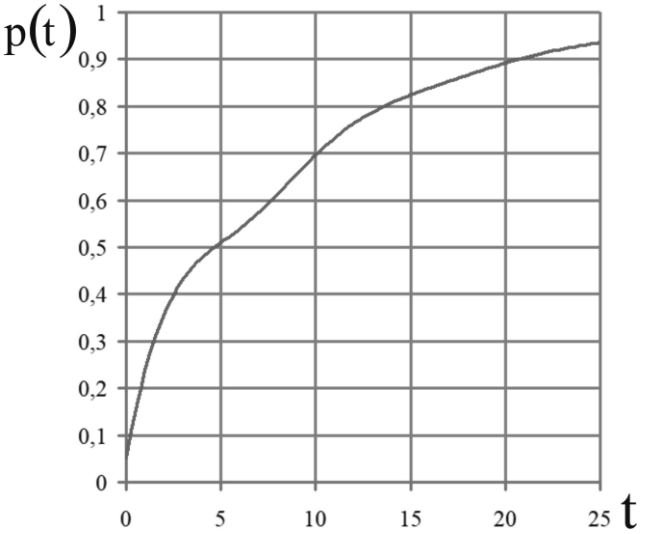

Рис. 11. Розподіл ймовірності загального часу тестування в годинах вразливості до *SQL*-ін’єкцій

У п'ятому розділі удосконалено імітаційну модель технології тестування безпеки на основі положень теорії масштабування імітаційних моделей, що відрізняється від відомих адаптацією вибору вхідних операторів управління і даних до підвищення вимог оперативності розроблення і реалізації моделі, яке виразилося в реалізації процедури взаємодії з реальним браузером з використанням засобів автоматизації браузеру і формуванні даних для атаки на декількох діалектах.

При розробці алгоритму масштабування імітаційної моделі технології тестування вразливостей були використані наступні постулати і визначення.

Визначення 1. Вершина n є батьком деякої вершини m (нащадка) в графі $G = (N, E, n_0)$, якщо $(n, m) \in G.E$.

Множину усіх нащадків вершини n в графі G будемо позначати як $desc(n, G)$.

Визначення 2. Шляхом «*way*» з $n_i \in G.N$ до $n_k \in G.N$ називається послідовність вершин n_i, n_{i+1} ,..., n_k така, що будь-які дві сусідні вершини в ній пов'язані дугою в графі:

$$(n_j, n_{j+1}) \in G.E,\ j = i, k-1.$$

Запис $n \in$ «*way*» означає, що вершина n зустрічається у шляху «*way*».

Визначення 3. Шлях «*way*» називається простим, якщо він складається з однієї вершини.

Визначення 4. Максимальним називається шлях, який нескінченний, або закінчується у вершині, що не має нащадків.

Визначення 5. У графі управління G послідовного процесу p оператор $n_j \in G.N$ прямо залежить за управлінням чутливо до зациклення від оператора $n_j \in G.N$ тоді і лише тоді, коли у n є два нащадки n_k і n_z такі, що:

1) в усіх максимальних шляхах, що починаються з n_k, зустрічається n_j, і або $n_i = n_j$, або n_j строго передує будь-якому входженню n_i;

2) існує максимальний шлях, що починається з n_z, такий, що або в ньому не зустрічається n_j, або n_i строго передує будь-якому входженню n_j.

Проведені дослідження показали, що для даного завдання імітаційного моделювання технології пошуку вразливостей не потрібно знаходження прямої залежності за управлінням.

Для коректного моделювання даного процесу досить використати слабкіше поняття транзитивної залежності за управлінням. Це істотно знизить обчислювальну складність алгоритму масштабування.

У шостому розділі проведено дослідження ефективності розроблених моделей та методів тестування безпеки застосунків, оцінку достовірності отриманих результатів математичного моделювання, а також обґрунтування практичних рекомендацій з використання методів та засобів управління безпекою.

У рамках практичних рекомендацій був розроблений метод передтестової компіляції і розподілу доступу.

Структурна схема методу представлена на рис. 12. Слід зауважити, що представлені на рис. 12. етапи і додаткові програмні інструменти передбачені з урахуванням використовуваної технології *LLVM*.

У дисертаційній роботі в якості основи для оцінки ефективності розробленої технології тестування безпеки застосунків використаний один з методів статистичного аналізу – зведення і угрупування статистичних даних, що отримав теоретичне обґрунтування в роботах.

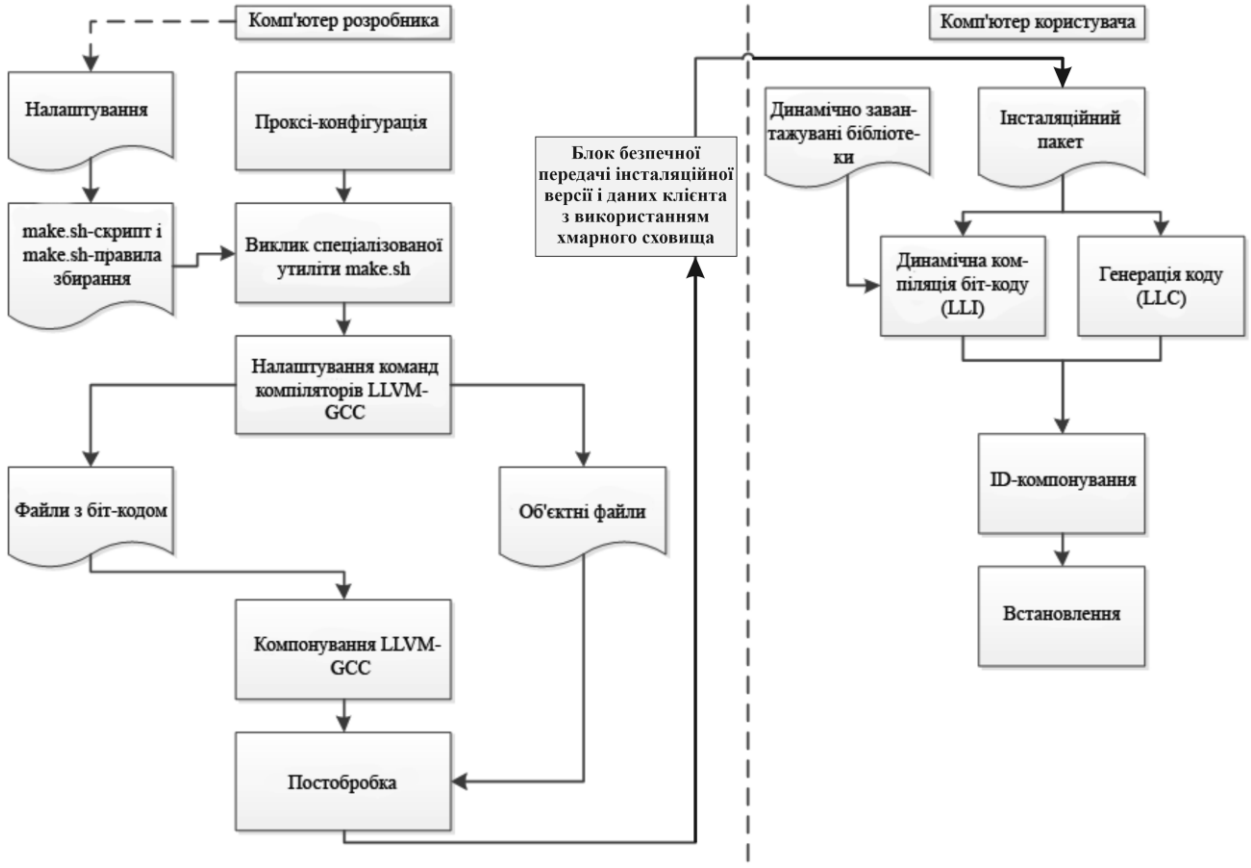

Рис. 12. Структурна схема методу передтестової компіляції і розподілу доступу

При проведенні дослідження було взято 20 *Web*-застосунків з різною кількістю (від 31 до 77) елементів, що тестуються.

В результаті експериментів отримано значення часу тестування безпеки застосунків для способів що використовують алгоритм Пурдома, і розроблену технологію тестування безпеки. Результати тестування представлені в табл. 6.

Побудуємо інтервальний ряд розподілу часу тестування безпеки, для чого оберемо оптимальний інтервал k і встановимо розмах інтервалу h. Оптимальне число груп оберемо так, щоб в достатній мірі відбилася різноманітність значень ознаки в сукупності, і в той же час закономірність розподілу, його форма не спотворювалася випадковими коливаннями частот, при цьому скористаємося формулою Стерждесса.

У нашому випадку оптимальний інтервал: $k = 1 + 3{,}322 \lg 20 = 5{,}32$. Оскільки число груп не може бути дробовим, то округлюємо $k = 5{,}32$ до найближчого цілого числа за правилами округлень – 5.

Враховуючи усі описані вище етапи, загальну структуру документування результатів запропонованих методів можна представити у вигляді схеми рис. 13.

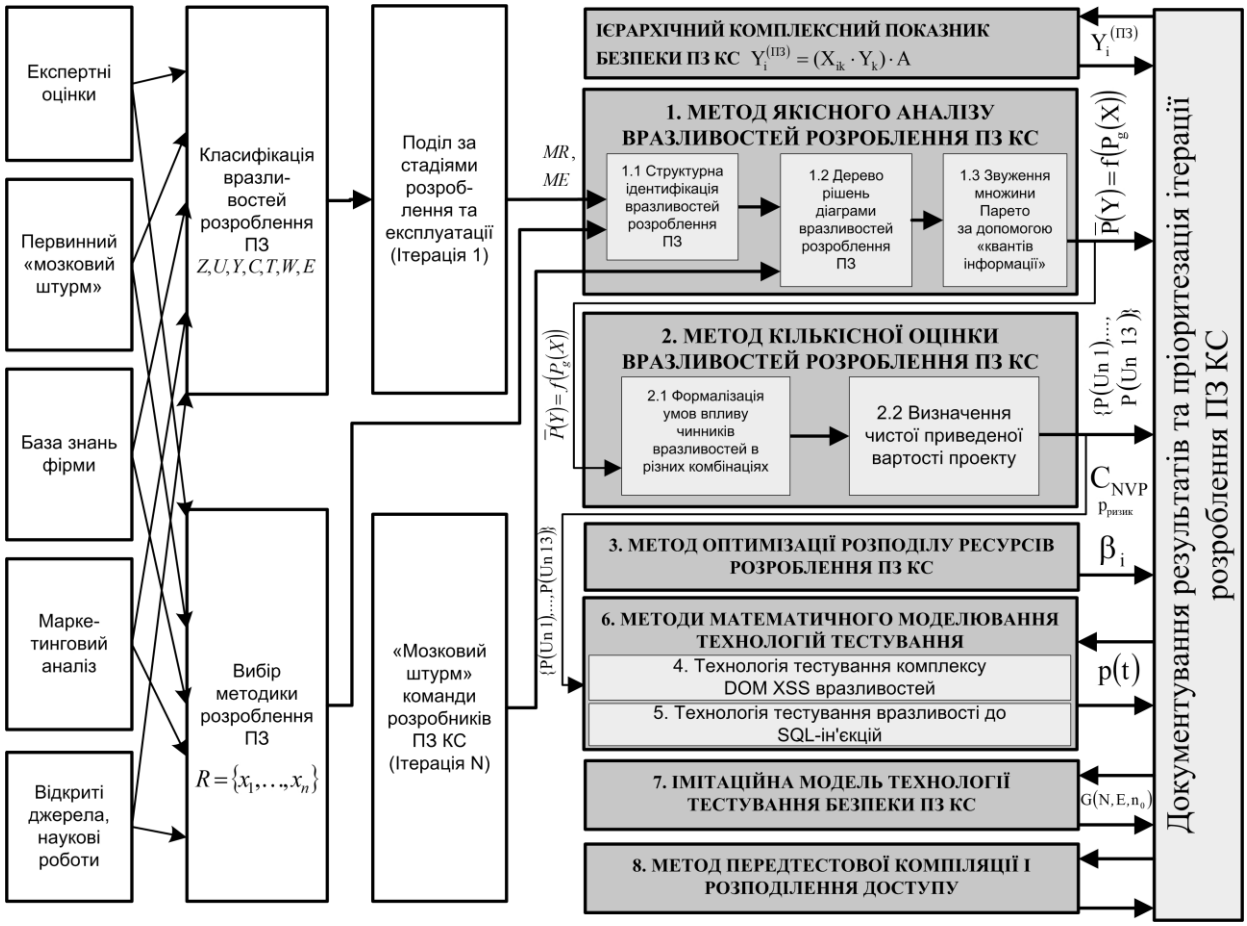

Рис. 13. Загальна структура документування результатів запропонованих методів

Таблиця 6

Результати тестування безпеки *Web*-застосунків

N	Час тестування (технологія тестування на основі алг. Пурдома) (с.)	Час тестування (розроблена технологія тестування) (с.)	N	Час тестування (технологія тестування на основі алг. Пурдома) (с.)	Час тестування (розроблена технологія тестування) (с.)
1	23	22,4	11	14,7	14,5
2	15,3	14,5	12	29,2	28,2
3	44	41,1	13	45,6	44,1
4	23,5	22,1	14	11,1	11
5	43,7	41,0	15	19,3	18,6
6	24,1	22,7	16	12,4	12
7	33	31,6	17	31,6	30,2
8	52	48,7	18	20,1	19.6
9	17,8	17,4	19	33,1	31,4
10	20,1	19,3	20	24	22,9

Знаючи число груп, розрахуємо довжину (розмах) інтервалу за формулою:

$$h = (X_{\max} - X_{\min})/k$$

Виходячи з даних, визначених вище:

$$h1 = (52 - 11{,}1)/5 = 8{,}18,\ h2 = (48{,}7 - 11)/5 = 7{,}54$$

Таким чином, інтервальний ряд розподілу часу тестування безпеки розіб'ємо на 5 груп з інтервалом по 8,18 *с*. і 7,68 *с*. Представимо інтервальний ряд розподілу тестування безпеки застосунків у вигляді табл. 7. Як видно з представленої таблиці, навіть така невелика вибірка застосунків, що тестується, показала переваги розробленої технології тестування безпеки.

Так, максимальне значення інтервального ряду зменшилося при використанні розроблення в 1,07 рази, зменшилося число попадань в максимальний часовий інтервал, а також сумарний час тестування зменшився в 1,05 рази.

Таблиця 7

Інтервальний ряд розподілу тестування безпеки *Web*-застосунків

Час (технологія тестування на основі алг. Пурдома) (с.)	Число потраплянь в інтервал	Час (розроблена технологія тестування) (с.)	Число потраплянь в інтервал
11,1-19,28	5	11-18,54	5
19,28-27,46	7	18,54-26,08	7
27,46-35,64	4	26,08-33,62	4
35,64-43,82	1	33,62-41,16	2
43,82-52	3	41,16-48,7	2

Слід зауважити, що на практиці найчастіше не використовується теоретично обґрунтований алгоритм Пурдома. При цьому тестування безпеки проводиться виходячи з досвіду експертів. В цьому випадку розроблена технологія тестування має істотну перевагу (до 1,5 рази). Для оцінки ефективності запропонованих моделей та методів використаємо наведений показник безпеки ПЗ $Y_i^{(ПЗ)}$ та його векторні складові $Y_{mat}^{(ПЗ)}$, $Y_{fft}^{(ПЗ)}$, $Y_{rec}^{(ПЗ)}$, $Y_{conft}^{(ПЗ)}$, $Y_{intt}^{(ПЗ)}$, $Y_{autht}^{(ПЗ)}$, $Y_{avb}^{(ПЗ)}$, $Y_{fir}^{(ПЗ)}$.

У табл. 8. наведено результати порівняльних досліджень показників, що мають вплив на векторні показників безпеки ПЗ зокрема, та загальний стан безпеки ПЗ КС в цілому.

Як видно з табл. 8. використання запропонованих моделей та методів розроблення безпечного ПЗ дозволить покращити показники глибини тестового контролю безпеки ПЗ $K_{test\,depth}$ до 5-7%, ймовірності помилки $P_{softwareerror}$ до 5%, ймовірності блокування доступу до ресурсів $P_{blocking}$ до 30%.

Покращення досліджуваних показників, в свою чергу, дозволить покращити вказані в табл. 8. векторні показники.

Використання виразу 1 дозволило визначити показник безпеки ПЗ $Y_i^{(ПЗ)}$ та розрахувати його для наведеного прикладу КС. Проведені дослідження показали, що показник $Y_i^{(ПЗ)}$ збільшився до 15%.

Таблиця 8

Результати порівняльних досліджень показників, що мають вплив на векторні показників безпеки ПЗ

№ з.п	Досліджуваний показник	Значення при використанні розроблених моделей та методів	Значення при використанні існуючих систем	Векторні складові показники безпеки ПЗ
1	Глибина тестового контролю безпеки ПЗ $K_{test\,depth}$	90-97%	85-90%	$Y_{mat}^{(ПЗ)}$, $Y_{fft}^{(ПЗ)}$, $Y_{rec}^{(ПЗ)}$, $Y_{conft}^{(ПЗ)}$, $Y_{intt}^{(ПЗ)}$, $Y_{autht}^{(ПЗ)}$, $Y_{avb}^{(ПЗ)}$
2	Ймовірність помилки $P_{softwareerror}$	0,9-0,95	0,8-0,9	$Y_{fft}^{(ПЗ)}$, $Y_{avb}^{(ПЗ)}$, $Y_{rec}^{(ПЗ)}$, $Y_{conft}^{(ПЗ)}$, $Y_{intt}^{(ПЗ)}$, $Y_{autht}^{(ПЗ)}$
3	Ймовірність блокування доступу до ресурсів $P_{blocking}$	0,8-0,95	0,6-0,85	$Y_{fft}^{(ПЗ)}$, $Y_{rec}^{(ПЗ)}$, $Y_{intt}^{(ПЗ)}$, $Y_{avb}^{(ПЗ)}$

Таким чином, проведені дослідження та експертна оцінка дозволили зробити висновок, що використання запропонованих моделей та методів розроблення безпечного ПЗ дозволить підвищити рівень захисту інформації в існуючих КС.

ВИСНОВКИ

У дисертації розв'язана науково-технічна проблема, яка є актуальною та полягає в синтезі моделей та методів розроблення безпечного ПЗ КС. Проведені в дисертаційній роботі дослідження, результати вирішення науково-технічної проблеми і окремих наукових завдань, а також результати розрахунків і порівняльного аналізу, дали можливість отримати наступні наукові та практичні результати.

1. Удосконалено метод якісного аналізу вразливостей розроблення програмного забезпечення, що відрізняється від відомих врахуванням факторів експлуатаційних вразливостей, особливо вразливості невиявлення загроз безпеки ПЗ КС і оцінкою довільного несуперечливого кінцевого набору «квантів інформації». В основу розробленого методу покладена структурна ідентифікація вразливостей розроблення ПЗ, що відрізняється від відомих побудовою оцінки вразливостей розроблення ПЗ «зверху» у вигляді множини, за наявності довільного несуперечливого кінцевого набору «квантів інформації». Це дозволило на 55% звузити сукупність множин Парето та більш точно обирати пріоритетність напрямків фінансування профілактичних заходів.

2. Удосконалено метод кількісної оцінки вразливостей розроблення ПЗ. Його відмінною особливістю є комплексне використання методики "Аналізу дерева відмов" і способу оцінки показника чистої приведеної вартості проекту розроблення ПЗ з урахуванням негативних факторів можливого невиявлення загроз безпеки ПЗ. Використання удосконаленої методики "Аналізу дерева відмов" дозволило до 20% підвищити точність кількісної оцінки вразливостей розроблення ПЗ. В той же час, використання способу оцінки показника чистої приведеної вартості проекту розроблення ПЗ дозволяє розглядати проект комплексно, з урахуванням необхідності врахування безпеки і тестування вразливості ПЗ, із залученням інструментів, які дозволяють здолати складність, невизначеність і довгостроковість проектів.

3. Удосконалено метод оптимізації розподілу ресурсів розроблення безпечного ПЗ. В основу цього методу було покладено напівмарківську модель прийняття рішень для керованого марківського процесу у безперервному часі. Відмінною особливістю запропонованого методу є використання псевдобулевих методів бівалентного програмування з нелінійною цільовою функцією і лінійними обмеженнями для визначення оптимальної стратегії усунення експлуатаційних помилок. Це дозволило оптимізувати процес проектування стратегії управління вразливостями.

4. Розроблено математичну модель технології тестування комплексу *DOM XSS* вразливостей, яка відрізняється від відомих урахуванням специфіки комплексного аналізу різних типів *XSS* вразливості ("*stored XSS*", "*reflected XSS*" і *DOM Based XSS*), а також включенням в алгоритм процедур автоматичного аудиту *DOM Based XSS* окремо. Це надало можливість провести аналітичну оцінку часових витрат тестування вказаних вразливостей в умовах реалізації стратегії розроблення безпечного програмного забезпечення.

5. Розроблено математичну модель технології тестування вразливості до *SQL*-ін'єкцій, яка відрізняється від відомих удосконаленим способом визначення відстані між результатами ін'єкції. Використання в запропонованому методі критерію Джаро-Вінклера для порівняння результатів ін'єкції *SQL*-коду і введення порогового значення дозволило підвищити точність результатів тестування безпеки ПЗ.

6. Розроблено метод математичного моделювання технологій тестування *DOM XSS* вразливості і вразливості до *SQL*-ін'єкцій, в основу якого покладено підхід мережевого *GERT* моделювання. Це дозволило досліджувати процеси в комп'ютеризованих системах при розробці нових засобів і протоколів захисту даних, а також від 1,05 до 1,5 разів зменшити час тестування безпеки.

7. Отримано подальший розвиток імітаційної моделі технології тестування безпеки на основі положень теорії масштабування імітаційних моделей. Відмінною особливістю розробленої імітаційної моделі є адаптація вибору вхідних операторів управління і даних до підвищення вимог оперативності розроблення і реалізації моделі, виражена в реалізації процедури взаємодії з реальним браузером з використанням засобів автоматизації браузеру

і формуванні даних для атаки на декількох діалектах. Це дозволило знизити обчислювальну складність реалізованих алгоритмів до 1,5 рази.

8. Розроблено метод передтестової компіляції і розподілу доступу, що відрізняється від відомих врахуванням профілів користувача при розроблені застосунку, а також використанням ресурсів "хмарних сховищ" в процесі отримання інсталяційних версій ПЗ. Це дозволило підвищити рівень безпеки розроблених застосунків.

9. Проведено порівняльну оцінку ефективності застосування розроблених моделей та методів, а також достовірності отриманих результатів. В цілому проведені дослідження показали, що показник безпеки ПЗ КС збільшився до 15%, що дозволяє зробити висновок про підвищення рівня захисту інформації за допомогою розроблених моделей та методів розроблення безпечного ПЗ.

Після проведених експериментів результат показав, що статистична величина довірчої ймовірність відхилення для видів даних, що розглядаються, складає $P \approx 0{,}92$, тобто значення статистичної величини від математичного сподівання «не відхилиться» більше, ніж на одиницю.

Практична значущість отриманих результатів полягає в наступному. Отримані результати підтверджуються їх застосуванням: при розробці автоматизованих систем виявлення вразливостей ПЗ в Інтернет сервіс провайдері ТОВ «ІМПЕРІАЛ-НЕТ» (м. Кропивницький); при удосконаленні гнучкої методології розроблення ПЗ у компанії-розробнику програмного забезпечення ТОВ «МІФ ПРОДЖЕКТС» (м. Кропивницький); при розробленні систем захисту інформації ТОВ «САЙФЕР ІТ» (м. Київ); у навчальному процесі Центральноукраїнського національного технічного університету. Використання результатів дисертаційної роботи підтверджене відповідними актами впровадження.

Наукове використання результатів, отриманих в дисертаційній роботі, можливе у рамках подальшого розвитку наукового напряму, який пов'язаний з розробкою і удосконаленням ефективних методологій розроблення ПЗ.

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

1. Kovalenko O., Popereshnyak S., Grinenko S., Grinenko O., Radivilova T. «Methods for Assessing the Maturity Levels of Software Ecosystems». *CEUR Workshop Proceedings* Volume 2654, 2019, Pages 251-261. Режим доступу: https://www.scopus.com/record/display.uri?eid=2-s2.0-85091278920&origin=resultslist **(Scopus)**.

2. Kovalenko O., Drieieva H., Simakhin V., Bondar S., Drieiev O., Zhumadilova M. «Multifractal Properties of Traffic Generator Based on Markov Chains ». *CEUR Workshop Proceedings* Volume 2588, 2019, Pages 567-579. Режим доступу: https://www.scopus.com/record/display.uri?eid=2-s2.0-85083214331&origin=resultslist **(Scopus)**.

3. Kovalenko Oleksandr Qualitative risk analysis of software development / Oleksandr Kovalenko, Jamil Al-Azzeh, Oleksii Smirnov, Anna Kovalenko, Serhii Smirnov // Asian Journal of Information Technology. – Volume 17 Issue 3. – Medwell Journals. – 2018. – P. 218-230. ISSN: 1682-3915. URL: http://medwelljournals.com/abstract/?doi=ajit.2018.218.230

4. Kovalenko Oleksandr, The mathematical model of the testing technology for DOM XSS vulnerabilities / O. Kovalenko, O. Smirnov, A. Kovalenko, S. Smirnov, V. Vialkova // Scientific & practical cyber security journal (SPCSJ) Volume 2 Issue 1, P. 22-28. Georgia. Tbilisi. Scientific Cyber Security Association (SCSA), 2018 ISSN: 2587-4667. URL: https://journal.scsa.ge/wp-content/uploads/ 2018/12/04-3-o.kovalenko-a.kovalenko-o.smirnov-s.smirnov-v.vialkova.pdf

5. Коваленко А.В. Технология тестирования DOM XSS уязвимости / А.В. Коваленко, А.С. Коваленко, А.А. Смирнов, С.А. Смирнов // Scientific & practical cyber security journal (SPCSJ) Volume 1. Issue 1. P. 38-45 Georgia. Tbilisi. Scientific Cyber Security Association (SCSA), 2017 ISSN: 2587-4667. URL: https://journal.scsa.ge/wp-content/uploads/2018/12/8-dom-xss-testing-technology-vulnerabilities.pdf

6. Коваленко О.В. Моделі та методи розроблення програмного забезпечення комп'ютерних систем для підвищення безпеки даних: монографія / О.В. Коваленко // К.: Вид. «КОД» – 2019. – 295 с.

7. Коваленко А.В. Методы качественного анализа и количественной оценки рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Информационные технологии в управлении, образовании, науке и промышленности: монография / Под редакцией профессора В.С. Пономаренко. – Х.: Видавець Рожко С.Г., 2016. – 566 с.

8. Коваленко А.В. Разработка метода управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Інформаційні технології: проблеми та перспективи: монографія / За загальною редакцією В.С. Пономаренка. – Х.: Видавець Рожко С.Г., 2017. – 447 с.

9. Коваленко А.В. Комплекс математических моделей технологии тестирования web-приложений / А.В. Коваленко, А.А. Смирнов // Інформаційні технології: сучасний стан та перспективи: монографія / За редакцією В.С. Пономаренка. – Х.: ТОВ «ДІСА ПЛЮС», 2018. – 461 с.

10. Коваленко А.В. Задачи распознавания ситуаций в ERP системах / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник наукових праць "Системи обробки інформації". – Випуск 4(120). – Х.: ХУПС – 2014. – С. 161-164.

11. Коваленко А.В. Методы качественного анализа и количественной оценки рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник наукових праць "Системи обробки інформації". – Випуск 5(142). – Х.: ХУПС – 2016. – С. 153-157.

12. Коваленко А.В. Проблемы анализа и оценки рисков информационной деятельности / А.В. Коваленко, А.А. Смирнов, Н.Н. Якименко,А.П. Доренский

// Збірник наукових праць "Системи обробки інформації". – Випуск 3(140). – Х.: ХУПС – 2016. – С. 40-42.

13. Коваленко А.В. Метод качественного анализа рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, Н.Н. Якименко, А.П. Доренский // Наука і техніка Повітряних Сил Збройних Сил України. – Випуск 2(23). – Харків: ХУПС. – 2016. – С. 150-158.

14. Коваленко А.В. Метод количественной оценки рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, Н.Н. Якименко, А.П. Доренский // Збірник наукових праць Харківського університету Повітряних Сил. Випуск 2 (47). – Харків: ХУПС. – 2016. – С. 128-133.

15. Коваленко А.В. Использование псевдобулевых методов бивалентного программирования для управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Системи управління, навігації та зв'язку. – Випуск 1 (37). – Полтава: ПолтНТУ. – 2016. – С. 98-103.

16. Коваленко А.В. Метод управления рисками разработки программного обеспечения / А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 2 (38). – Полтава: ПолтНТУ. – 2016. – С. 93-100.

17. Коваленко А.В. Технология тестирования уязвимости к SQL инъекциям / А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 5 (45). – Полтава: ПолтНТУ. – 2017. – С. 66-71.

18. Коваленко А.В. Масштабирование имитационной модели технологии тестирования безопасности / А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 6 (46). – Полтава: ПолтНТУ. – 2017. – С. 181-184.

19. Коваленко А.В. Имитационная модель технологии тестирования безопасности Web-приложений / А.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 1 (47). – Полтава: ПолтНТУ. – 2018. – С. 114-123.

20. Коваленко О.В. Методи якісного аналізу та кількісної оцінки ризиків розроблення програмного забезпечення/ О.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 3 (49). – Полтава: ПолтНТУ. – 2018. – С. 116-125.

21. Коваленко О.В. Управління ризиками розроблення програмного забезпечення за умови обмеженості коштів виділених на усунення помилок безпеки/ О.В. Коваленко // Техніка в сільськогосподарському виробництві, галузеве машинобудування, автоматизація. – Випуск 31. – Кропивницький: ЦНТУ. – 2018. – С. 128-140.

22. Коваленко О.В. GERT-мережевий синтез технології тестування на вразливість WEB-додатків/ О.В. Коваленко // Захист інформації. – Випуск 20(2) – К.: НАУ. – 2018. – С. 89-94.

23. Коваленко О.В. Імітаційна модель технології тестування безпеки на основі положень теорії масштабування / О.В. Коваленко // Безпека інформації. – Випуск 24 (2). – К.: НАУ. – 2018. – С. 110-117.

24. Коваленко О.В. Оцінка ефективності технології тестування безпеки / О.В. Коваленко // Вчені записки Таврійського національного

університету імені В.І. Вернадського. Серія: Технічні науки. Том 29 (68) № 2, 2018. – С. 137-141

25. Коваленко О.В. Методи та засоби управління безпекою додатків / О.В. Коваленко // Інформаційно-керуючі системи на залізничному транспорті. №4, 2018. – С. 41-44.

26. Коваленко О.В. Розробка інформаційної технології передтестової компіляції та розподілу доступу / О.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 4 (50). – Полтава: ПолтНТУ. – 2018. – С. 115-119.

27. Коваленко О.В. Аналіз та дослідження інформаційних технологій розробки програмного забезпечення / О.В. Коваленко // Вчені записки Таврійського національного університету імені В.І. Вернадського. Серія: Технічні науки. Том 29 (68) № 5, 2018. – С. 131-137.

28. Коваленко О.В. Удосконалений метод управління ризиками розроблення програмного забезпечення на основі напівмарковської моделі прийняття рішень / О.В. Коваленко // Сучасні інформаційні системи. – Випуск 2(3). – Харків. – 2018. – С. 41-48.

29. Коваленко О.В. Математичні моделі технології тестування DOM XSS вразливості та вразливості до SQL ін'єкцій / О.В. Коваленко // Вісник Черкаського державного технологічного університету. Серія : Технічні науки №4, 2018. – С. 29-36.

30. Коваленко О.В. Математична модель технології тестування вразливості до SQL ін'єкцій / О.В. Коваленко // Системи управління, навігації та зв'язку. – Випуск 6 (58). – Полтава: ПолтНТУ. – 2019. – С. 43-47.

31. Коваленко О.В. Математична модель технології тестування комплексу DOM XSS вразливостей для аналітичної оцінки часових витрат / О.В. Коваленко // Центральноукраїнський науковий вісник. Технічні науки. № 2(33). с. 173-180, 2019.

32. Коваленко А.В. Проблемы анализа и оценки рисков информационной деятельности / А.В. Коваленко, А.А. Смирнов // Збірник наукових праць II міжнародної науково-практичної конференції «Актуальні питання забезпечення кібернетичної безпеки та захисту інформації». м. Київ. 24-27 лютого 2016 р. – Київ: Європейський університет. – 2016. – С. 138-139.

33. Коваленко А.В. Анализ и оценка рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез «Securitea informationala 2015-2016». Conferenta internationala (editia a XII-a). Chisinau. Moldova. 3 martie 2016. – Chisinau: ADSEM. – 2016. – P. 96-102.

34. Коваленко А.В. Исследование источников и причин риска разработки программного обеспечения, этапов и работ, при выполнении которых возникает риск / А.В. Коваленко, А.А. Смирнов // Збірник тез VII всеукраїнської науково-практичної конференції "Інформатика та системні науки (ІСН-2016)". м. Полтава. 10-12 березня 2016 р. – Полтава.: ПУЕТ – 2016. – С. 264-266.

35. Коваленко А.В. Оценка показателя чистой приведенной стоимости для количественной оценки рисков проекта разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез науково-практичної конференції “Проблеми кібербезпеки інформаційно-телекомунікаційних систем”. м. Київ. 10-11 березня 2016 р. – Київ: КНУ ім. Тараса Шевченко – 2016. – С. 81-82.

36. Коваленко А.В. Методика структурной идентификации рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез Міжнародної науково-практичної конференції «Інформаційна безпека та комп'ютерні технології» (IS&CT). м. Кіровоград. 24-25 березня 2016 р. – Кіровоград: КНТУ. – 2016. – С. 71-72.

37. Коваленко А.В. Методы качественного анализа рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез першої міжнародної науково-практичної конференції «Проблеми науково-технічного та правового забезпечення кібербезпеки у сучасному світі» (ПНПЗК-2016). м. Харків. 30 березня – 1 квітня 2016 р. – Харків: НТУ «ХПІ». – 2016. – С. 6-7.

38. Коваленко А.В. Структурная идентификация рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез XVIII міжнародного науково-практичного семінару «Комбінаторні конфігурації та їх застосування». м. Кіровоград. 15-16 квітня 2016 р. – Кіровоград: КНТУ. – 2016. – С. 175-182.

39. Коваленко А.В. Исследование разработанной методики структурной идентификации рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез VIII міжнародної науково-практичної конференції “Проблеми і перспективи розвитку ІТ-індустрії”. м. Харків. 28-29 квітня 2016 р. – Харків: ХНЕУ. – 2016. – С. 49.

40. Коваленко А.В. Исследование дерева рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез ІІІ міжнародної науково-практичної конференції «Інформаційна та економічна безпека» (INFECO-2016)». м. Харків. 28-30 квітня 2016 р. – Харків: ХНМІ ДВНЗ «УБС». – 2016. – С. 174-178.

41. Коваленко А.В. Методы качественного анализа и количественной оценки рисков разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Сборник тезисов XII международной конференции "Стратегия качества в промышленности и образовании". г. Варна. Болгария. 30 мая – 02 июня 2016 г – С. 585-589.

42. Коваленко А.В. Разработка метода управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Матеріали Всеукраїнської науково-практичної конференції «Кібербезпека в Україні: правові та організаційні питання». м. Одеса, 21 жовтня 2016 р. – Одеса : ОДУВС, 2016. – С.146-148.

43. Коваленко А.В. Метод управления рисками разработки программного обеспечения с использованием псевдобулевых методов

бивалентного программирования / А.В. Коваленко, А.А. Смирнов // Матеріали Всеукраїнської науково-практичної конференції «Актуальні задачі та досягнення у галузі кібербезпеки». м. Кропивницький, 23-25 листопада 2016 року – Кропивницький: ЦНТУ, 2016. – С. 162.

44. Коваленко А.В. Псевдобулевые методы бивалентного программирования для управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко, С.А. Смирнов // Збірник наукових праць III міжнародної науково-практичної конференції «Актуальні питання забезпечення кібернетичної безпеки та захисту інформації». м. Київ. 22-25 лютого 2017 р. – Київ: Європейський університет. – 2017. – С. 158-162.

45. Коваленко А.В. Метод управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов // Збірник тез II науково-практичної конференції "Проблеми кібербезпеки інформаційно-телекомунікаційних систем". м. Київ. 23-24 березня 2017 р. – Київ: КНУ ім. Тараса Шевченко – 2017. – С. 203-205.

46. Коваленко А.В. Алгоритмы анализа уязвимостей при управлении рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Conferenta internationala (editia a XIII-a). «Securitea informationala 2017». Chisinau. Republic of Moldova. 4-5 aprilie 2017. – Chisinau: ADSEM. – 2017. – P. 19-22.

47. Коваленко А.В. Алгоритм анализа DOM XSS уязвимости при управлении рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник тез дев'ятнадцятого міжнародного науково-практичного семінару «Комбінаторні конфігурації та їх застосування». м. Кропивницький 7-8 квітня 2017 р. – Кропивницький: ГЛА НАУ. – 2017. – С. 125-127.

48. Коваленко А.В. Алгоритм анализа уязвимости SQL Injection для управления рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник тез другої міжнародної науково-технічної конференції «Проблеми науково-технічного та правового забезпечення кібербезпеки у сучасному світі» (ПНПЗК-2017). м. Харків.10-12 квітня 2017 р. – Харків: НТУ «ХПІ». – 2017. – С. 27.

49. Коваленко А.В. Метод управления рисками разработки программного обеспечения на основе алгоритмов анализа уязвимостей / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник тез Міжнародної науково-практичної конференції «Інформаційна безпека та комп'ютерні технології» (IS&CT). м. Кіровоград. 20-22 квітня 2017 р. Кіровоград: КНТУ. – 2017. – С. 92.

50. Коваленко А.В. Алгоритмы анализа DOM XSS уязвимости и уязвимости SQL Injection при управлении рисками разработки программного обеспечения / А.В. Коваленко, А.А. Смирнов, А.С. Коваленко // Збірник тез IX міжнародної науково-практичної конференції "Проблеми і перспективи

розвитку ІТ-індустрії”. м. Харків. 20-21 квітня 2017 р. – Харків: ХНЕУ. – 2017. – С. 61.

51. Kovalenko O.V. Method of testing the dom xss vulnerability / Kovalenko Oleksandr, Kovalenko Anna, Smirnov Oleksii, Smirnov Serhii // International Conference «information technologies, systems and networks ITSN-2017». Chisinau, Republic of Moldova. 17 – 18 October 2017. – Chisinau: Academy of Sciences of Moldova, Military Academy of Armed Forces “Alexandru cel Bun”. – 2017. – P. 7.

52. Коваленко О.В. Метод тестування DOM XSS уразливості / О.В. Коваленко, О.А. Смірнов, А.С. Коваленко, С.А. Смірнов // Збірник тез всеукраїнської науково-практичної інтернет-конференції «Автоматика та комп’ютерно-інтегровані технології у промисловості, телекомунікаціях, енергетиці та транспорті». м. Кропивницький. 16-17 листопада 2017 р. – Кропивницький: ЦНТУ. – 2017. – С. 198-199.

53. Коваленко О.В. GERT-модель технології тестування DOM XSS уразливості / О.В. Коваленко, А.С. Коваленко, О.А. Смірнов, С.А. Смірнов // Збірник наукових праць IV міжнародної науково-практичної конференції «Актуальні питання забезпечення кібернетичної безпеки та захисту інформації». м. Київ. 21-24 лютого 2018 р. – Київ: Європейський університет. – 2018. – С. 65-70.

54. Коваленко О.В. Технології тестування уразливостей Web-застосунків з використанням GERT-моделі / О.В. Коваленко, А.С. Коваленко, О.А. Смірнов, С.А. Смірнов // Збірник тез всеукраїнської науково-практичної конференції "Комп'ютерні інтелектуальні системи та мережі (КІСМ-2018)". м. Кривий Ріг. 21-23 березня 2018 р. – Кривий Ріг.: ДВНЗ КНУ – 2018. – С. 227-230.

55. Коваленко А.В. Тестирование уязвимости Web-приложений к атаке вида межсайтовый скриптинг / А.В. Коваленко, А.С. Коваленко, А.А. Смирнов, С.А. Смирнов // Збірник тез «Securitea informationala 2018». Conferenta internationala (editia a XIV-a). Chisinau. Moldova. 20-21 martie 2018. – Chisinau: ADSEM. – 2018. – P. 54-56.

56. Коваленко А.В. Комплекс математических моделей технологии тестирования web-приложений / А.В. Коваленко, А.С. Коваленко, А.А. Смирнов, С.А. Смирнов // Збірник тез X міжнародної науково-практичної конференції “Проблеми і перспективи розвитку ІТ-індустрії”. м. Харків. 19-20 квітня 2018 р. – Харків: ХНЕУ. – 2018. – С. 38.

57. Коваленко О.В. Розробка методу передтестової компіляції й розподілу доступу / О.В. Коваленко, А.С. Коваленко, О.А. Смірнов, С.А. Смірнов // Збірник наукових праць III міжнародної науково-практичної конференції “Інформаційна безпека та комп’ютерні технології”, м. Кропивницькй. 19-20 квітня 2018 р. – Кропивницький: ЦНТУ. – 2018. – С. 214-215.

58. Коваленко О.В. Оцінка ефективності технологій тестування безпеки уразливостей DOM XSS й SQL-ін'єкцій / О.В. Коваленко, А.С. Коваленко,

О.А. Смірнов, С.А. Смірнов // Сборник тезисов XIV международной конференции "Стратегия качества в промышленности и образовании", Варна, Болгария. 04-07 июня 2018 г – Варна. ТУВ. – 2018. – С. 271-274.

59. Коваленко О.В. Аналіз основних підходів математичного моделювання та методологій для забезпечення максимальних показників безпеки програмного забезпечення / О.В. Коваленко, А.С. Коваленко // Збірник наукових праць всеукр. наук.-практ. конф. здобувачів вищої освіти й молодих учених «Комп'ютерна інженерія і кібербезпека : досягнення та інновації, м. Кропивницькй. 27-29 листопада 2018 р. – Кропивницький: ЦНТУ. – 2018. – С. 74.

АНОТАЦІЯ

Коваленко О.В. Моделі та методи розроблення безпечного програмного забезпечення комп'ютерних систем. – На правах рукопису.

Дисертація на здобуття наукового ступеня доктора технічних наук за спеціальністю 05.13.05 – комп'ютерні системи та компоненти. – Черкаський державний технологічний університет. – Черкаси, 2020.

Дисертаційна робота присвячена розробці моделей та методів розроблення безпечного програмного забезпечення комп'ютерних систем. Для вирішення поставлених завдань пропонується: удосконалити методи якісного аналізу та кількісної оцінки вразливостей розроблення ПЗ; удосконалити метод оптимізації розподілу ресурсів розроблення безпечного ПЗ; розробити математичну модель технології тестування комплексу *DOM XSS* вразливостей; розробити математичну модель технології тестування вразливості до *SQL*-ін'єкцій; розробити комплекс математичних моделей процесу тестування *DOM XSS* вразливості і вразливості до *SQL*-ін'єкцій; удосконалити імітаційну модель технології тестування безпеки; розробити метод передтестової компіляції і розподілу доступу. Результати дисертаційної роботи впроваджено в діяльність комерційних підприємств та навчальних закладах України.

Ключові слова: безпечне програмного забезпечення, ідентифікація вразливостей, якісний та кількісний аналіз вразливостей, безпека даних, оптимізація розподілу ресурсів розроблення програмного забезпечення, алгоритми тестування безпеки, масштабування, імітаційна модель, GERT-мережі.

АННОТАЦИЯ

Коваленко А.В. Модели и методы разработки безопасного программного обеспечения компьютерных систем. – На правах рукописи.

Диссертация на соискание ученой степени доктора технических наук по специальности 05.13.05 – компьютерные системы и компоненты. – Черкасский государственный технологический университет. – Черкассы, 2020.

Диссертационная работа посвящена разработке моделей и методов разработки безопасного программного обеспечения компьютерных систем. Для решения поставленных задач предлагается: усовершенствовать методы качественного анализа и количественной оценки уязвимостей разработки ПО;

усовершенствовать метод оптимизации распределения ресурсов разработки безопасного ПО; разработать математическую модель технологии тестирования комплекса *DOM XSS* уязвимостей; разработать математическую модель технологии тестирования уязвимости к *SQL*-инъекциям; разработать комплекс математических моделей процесса тестирования *DOM XSS* уязвимости и уязвимости к *SQL* инъекциям; усовершенствовать имитационную модель технологии тестирования безопасности; разработать метод предтестовой компиляции и распределения доступа. Результаты диссертационной работы внедрены в деятельность коммерческих предприятий и учебных заведений Украины.

Ключевые слова: безопасное программного обеспечения, идентификация уязвимостей, качественный и количественный анализ уязвимостей, безопасность данных, оптимизация распределения ресурсов разработки программного обеспечения, алгоритмы тестирования безопасности, масштабирование, имитационная модель, GERT-сети.

ANNOTATION

Kovalenko O.V. Models and methods of developing secure software for computer systems. – On the rights of the manuscript.

Dissertation for the degree of Doctor of Technical Sciences in specialty 05.13.05 – Computer systems and components. – Cherkasy State Technological University. – Cherkasy, 2020.

This dissertation work is devoted to solving a relevant scientific and technical problem consisting in synthesis of models and methods of development of secure software for computer systems.

The work has carried out analysis of modern trends in software development methodologies and software requirements, indicators and optimization criteria, as well as approaches of mathematical formalization of information processes which showed that with the introduction of computer technology in critical application systems, increase in the information which is stored, processed and circulating in them, as well as the increased vulnerability of unauthorized access to software by attackers, currently used models and methods of software development of computer systems do not provide the required level of data security. Based on the analysis, international and national standards, a general scheme of characteristics and indicators related to software quality has been formed. The analysis of software development methodologies and factors influencing security allowed to identify contradictions between increased software security requirements (taking into account all security vulnerabilities) and the need to adapt to existing objective and subjective factors inherent in the modern world of the IT industry. The conducted comparative research of the basic mathematical formalization approaches allowed us to define the basic directions of the dissertation research and to formulate the optimization task of synthesis of software development models and methods.

The work has improved the method of qualitative analysis of software development vulnerabilities, which differs from the known by considering factors of operational vulnerabilities, especially the vulnerability of non-

detection of security threats to the software of computer systems, and the assessment of an arbitrary consistent finite set of "information quanta".

The method of quantitative assessment of software development vulnerabilities has been improved. Its distinctive feature is the integrated use of "Fault Tree Analysis" and the method of estimating the net present value of the software development project, taking into account the negative factors of possible non-detection of software security threats.

The method of optimizing the allocation of resources for secure software development has been improved. This method was based on the semi-Markov model of decision-making for a controlled Markov process in continuous time.

A mathematical model of testing technology for DOM XSS vulnerabilities complex has been developed, which differs from the known ones by taking into account the specifics of complex analysis of different types of XSS vulnerabilities ("stored XSS", "reflected XSS" and DOM Based XSS), as well as inclusion of DOM Based XSS automatic audit procedures separately.

The work has improved the mathematical model of the technology for testing vulnerability to SQL injections, which differs from the known models by using an improved method of determining the distance between the injection results. The use of the Jaro-Winkler test in the proposed method to compare the results of the injection of SQL code and the introduction of a threshold value will increase software security testing accuracy.

A method of mathematical modeling of technologies for testing DOM XSS vulnerabilities and vulnerabilities to SQL -injections has been developed, which is based on the network GERT modeling approach.

Further development of the simulation model of security testing technology based on the provisions of the theory of simulation models scaling is obtained. A distinctive feature of the developed simulation model is the adaptation of the choice of input control operators and data to the increase in requirements of development efficiency and model implementation expressed in the implementation of the procedure of interaction with a real browser using browser automation means and data generation to attack in multiple dialects.

Further development of the pre-test compilation method and distribution of access, which differs from the known ones by taking into account user profiles in the synthesis of the application, as well as the use of "cloud storage" resources in the process of obtaining software installation versions has been obtained.

The work has conducted a comparative assessment of the efficiency of developed models and methods, as well as reliability of the results. In general, studies have shown that the security index of the software of the computer systems has increased to 15%, which allows us to conclude that the level of information protection has been increased with the help of synthesized models and methods of developing secure software. The results of the dissertation are implemented in the activities of commercial enterprises and educational institutions of Ukraine.

Keywords: secure software development, vulnerability identification, qualitative and quantitative analysis of vulnerabilities, data security, optimization of software development resource allocation, security testing algorithms, scaling, simulation model, GERT-networks.