https://doi.org/10.31891/2219-9365-2023-73-1-8
УДК 004.9: 004.05

Дмитро МЕДЗАТИЙ
Хмельницький національний університет
https://orcid.org/0000-0002-1879-2945
e-mail: medza@ukr.net
Юрій ВОЙЧУР
Хмельницький національний університет
https://orcid.org/0000-0003-3085-7315
e-mail: voichury@khmnu.edu.ua
Олег ВОЙЧУР
Хмельницький національний університет
https://orcid.org/0000-0001-8503-6464
e-mail: o.voichur@gmail.com

ТЕХНОЛОГІЯ ІДЕНТИФІКАЦІЇ ТА КЛАСИФІКАЦІЇ ВІДМОВ І ВРАЗЛИВОСТЕЙ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Всі основні підходи щодо забезпечення безпеки ПЗ спрямовані на запобігання повної відмови ПЗ, але не на ідентифікацію відмов та вразливостей ПЗ. Успішність підходів до забезпечення безпеки ПЗ можливе лише за рахунок ідентифікації та скорочення кількості помилок, тому актуальною задачею наразі є ідентифікація відмов та вразливостей ПЗ.

Проведений огляд літератури щодо відомих методів і технологій виявлення відмов і вразливостей програмного забезпечення показав, що, хоча проаналізовані методи та технології й мають величезний потенціал для галузі інженерії програмного забезпечення, проте жодне з відомих рішень не призначене для ідентифікації та класифікації відмов і вразливостей програмного забезпечення згідно із правилами класифікації відмов та правилами класифікації вразливостей. Отже, необхідно спроектувати та реалізувати технологію ідентифікації та класифікації відмов і вразливостей ПЗ на основі правил класифікації відмов та вразливостей ПЗ, що і є метою даного дослідження.

У статті розроблено опитувальники для збору інформації про відмову(и) та про вразливість(і), а також розроблено правила класифікації відмов на основі аналізу відповідей на питання опитувальника для збору інформації про відмову(и) та класифікації вразливостей на основі аналізу відповідей на питання опитувальника для збору інформації про вразливість(і). Розроблені правила дають можливість ідентифікувати та класифікувати відмову(и) та про вразливість(і), які мали місце в процесі функціювання ПЗ.

У статті розроблено технологію ідентифікації та класифікації відмов і вразливостей, яка надає висновок щодо наявності чи відсутності відмов(и) ПЗ; висновок щодо наявності чи відсутності вразливості(ей) ПЗ; висновок про тип відмови та тип вразливості в разі їх наявності, завдяки чому пропонована технологія є корисною для користувачів ПЗ за рахунок ідентифікації та класифікації відмов і вразливостей.

Ключові слова: відмова ПЗ, вразливість ПЗ, ідентифікація вімов і вразливостей, класифікація відмов і вразливостей.

Dmytro MEDZATYI, Yurii VOICHUR, Oleg VOICHUR
Khmelnytskyi National University

TECHNOLOGY OF IDENTIFICATION AND CLASSIFICATION OF SOFTWARE FAILURES AND VULNERABILITIES

All major software security approaches are aimed at preventing total software failure, but not at identifying software failures and vulnerabilities. The success of software security approaches is only possible due to the identification and reduction of the number of errors, therefore, the identification of software failures and vulnerabilities is an urgent task at the moment.

A review of the literature on known methods and technologies for detecting software failures and vulnerabilities showed that, although the analyzed methods and technologies have great potential for the field of software engineering, none of the known solutions are designed to identify and classify software failures and vulnerabilities according to with failure classification rules and vulnerability classification rules. Therefore, it is necessary to design and implement the technology of identification and classification of software failures and vulnerabilities based on the rules of classification of software failures and vulnerabilities, which is the purpose of this study.

The article develops questionnaires for collecting information about failure(s) and vulnerability(s), as well as developed rules for classification of failures based on the analysis of answers to questionnaire questions for collecting information about failure(s) and classification of vulnerabilities based on the analysis of answers to questionnaire questions to collect information about the vulnerability(s). The developed rules make it possible to identify and classify failure(s) and vulnerability(s) that occurred during the software's operation.

The article develops a technology of identification and classification of software failures and vulnerabilities, which provides a conclusion on the presence or absence of software failure(s); conclusion on the presence or absence of software vulnerability(s); conclusion about the type of failure and the type of vulnerability in case of their presence, thanks to which the proposed technology is useful for software users due to the identification and classification of failures and vulnerabilities.

Keywords: software failure, software vulnerability, identification of failures and vulnerabilities, classification of failures and vulnerabilities.

Постановка проблеми у загальному вигляді
та її зв'язок із важливими науковими чи практичними завданнями

Сучасне програмне забезпечення (ПЗ) є складним багатофункціональним виробом, при створенні якого неминуче мають місце помилки, ненавмисні програмні дефекти, незахищені функції. У сучасну цифрову епоху ПЗ широко адаптоване та стало невід'ємною складовою людського суспільства. Таке широке використання ПЗ пов'язане з використанням великих і критичних даних, які неминуче потребують захисту. Вкрай важливо переконатися, що це ПЗ не тільки задовольняє потреби користувачів або функціональні вимоги, але не менш важливо забезпечити безпеку цього ПЗ Створення безпечного програмного забезпечення є складним процесом. Це процес, неформально керований загальними знаннями, передовою практикою та незадокументованими експертними знаннями. Загалом безпеку ПЗ можна розглядати як одну з найважливіших проблем у галузі розробки програмного забезпечення, оскільки вона може впливати на ефективність програмного продукту через різноманітні технологічні вразливості та загрози.

Безпека програмного забезпечення є властивістю певного програмного забезпечення функціонувати без різних негативних наслідків для конкретної комп'ютерної системи. Причини, що призводять до порушення безпеки, можуть бути різними: відмови та збої ПЗ, вразливості ПЗ через помилки програмістів та дефекти в програмах.

Відмова ПЗ (failure) – це подія, що характеризується порушенням роботоздатності ПЗ, внаслідок якого ПЗ припиняє виконувати свої функції (цілком або частково) [1].

Вразливість ПЗ (vulnerability) – це недолік ПЗ (недолік проектування ПЗ, помилка програмування, застосування шкідливого ПЗ), при використанні якого можна навмисне порушити цілісність ПЗ та викликати його некоректну роботу; це нездатність ПЗ протистояти реалізації певної загрози або сукупності загроз [2].

Щороку виявляються тисячі нових вразливостей, що вимагають від компаній виправлення операційних систем і додатків, а також переналаштування параметрів безпеки у всьому мережевому середовищі. Для попереджувального усунення вразливостей до того, як вони будуть використані для кібератаки, організації, які серйозно ставляться до безпеки свого мережевого середовища, проводять управління вразливостями, щоб забезпечити максимально можливий рівень безпеки.

Виявлення вразливості програмного коду є важливим методом забезпечення безпеки ПЗ. Сьогодні, коли розмір і складність програмного забезпечення швидко зростають, вразливості стають різноманітними, і їх стає все важче ідентифікувати.

Основними причинами появи вразливостей є: спільне використання ресурсів і спрощення обміну інформацією між вузлами мережі; суттєве ускладнення ПЗ; відсутність повної інформації про об'єкт і використання механізмів пошуку; ненадійні джерела даних і величезна кількість зловмисників; низька кваліфікація користувачів ПЗ, особливо в питаннях захисту інформації – ПЗ нездатне протистояти загрозам з боку зловмисників, якщо користувачі під їх впливом несвідомо виконують руйнівні дії; складність нових технологій; тенденція об'єднання даних і програмного коду, вбудовування програмного коду (макросів, сценаріїв) у документи; відставання у розвитку нормативно-правової бази, стандартів від змін методів та технологій обробки інформації; відсутність безпечних процесів у життєвому циклі розробки ПЗ.

Стрімке зростання обчислювальної потужності комп'ютерів та обсягів оброблюваних даних, розширення кола задач, які вирішуються програмним забезпеченням, ускладнюють проведення повного і детального аналізу можливих вразливостей і виключення умов їх появи.

Наразі чимало провідних вчених провели ряд досліджень щодо підвищення безпеки ПЗ, проте вразливості та відмови ПЗ все ще створюють серйозні проблеми для користувачів ПЗ, проявляючись витоками інформації, втратою інформації, призводячи до фінансових та репутаційних втрат. Так, наприклад, через уразливості ПЗ стався витік інформації у вигляді доступу до 500 млн. записів користувачів Yahoo [3]; компанія Equifax втратила інформацію про 140 млн осіб, що призвело до фінансових втрат у 575 млн дол. США [4]; зловмисники отримали доступ до 50 млн. профілів користувачів Facebook [5]; викрадена інформація про 600 тис. водіїв і 57 млн облікових записів користувачів сервісу Uber, що призвело до фінансових втрат у 148,1 млн дол. США [6]; хакерська атака на урядові сайти України 14 січня 2022 року, спричинена вразливістю системи керування вмістом веб-сайтів October CMS [7].

Всі основні підходи щодо забезпечення безпеки ПЗ спрямовані на запобігання повної відмови ПЗ, але не на ідентифікацію відмов та вразливостей ПЗ. Успішність підходів до забезпечення безпеки ПЗ можливе лише за рахунок ідентифікації та скорочення кількості помилок (наразі щільність помилок у ПЗ коливається від 2 до 100 помилок на 1000 рядків коду [8]), тому *актуальною задачею* наразі є ідентифікація відмов та вразливостей ПЗ.

Проведений у [9] огляд літератури щодо відомих методів і технологій виявлення відмов і вразливостей програмного забезпечення показав, що, хоча проаналізовані методи та технології й мають величезний потенціал для галузі інженерії програмного забезпечення, проте жодне з відомих рішень не призначене для ідентифікації та класифікації відмов і вразливостей програмного забезпечення згідно із правилами класифікації відмов та правилами класифікації вразливостей. Отже, необхідно спроектувати та

реалізувати технологію ідентифікації та класифікації відмов і вразливостей ПЗ на основі правил класифікації відмов та вразливостей ПЗ, що і є *метою даного дослідження*.

Технологія ідентифікації та класифікації відмов і вразливостей програмного забезпечення

Враховуючи правила класифікації відмов та вразливостей ПЗ, розроблені авторами у [9], розробимо опитувальники для збору інформації про відмову(и) та про вразливість(і), які мали місце в процесі функціювання ПЗ.

Опитувальник для збору інформації про відмову(и):

1. Чи є стан ПЗ після припинення його функціювання роботоздатним?
2. Чи за час припинення функціювання ПЗ відбулась втрата даних?

На кожне із запитань опитувальника для збору інформації про відмову(и) передбачено відповідь «так» або «ні».

Правила класифікації відмов на основі аналізу відповідей на питання опитувальника для збору інформації про відмову(и):

1. Якщо обрано відповідь «так» на перше запитання опитувальника для збору інформації про відмову(и) та обрано відповідь «ні» на друге запитання опитувальника збору інформації про відмову, то змінна *sf = 1*

2. Якщо обрано відповідь «так» на перше запитання опитувальника для збору інформації про відмову(и) та обрано відповідь «так» на друге запитання опитувальника збору інформації про відмову, то змінна *sf = 2*

3. Якщо обрано відповідь «ні» на перше запитання опитувальника для збору інформації про відмову(и), то змінна *sf = 3*

Опитувальник для збору інформації про вразливість(і):

1. Чи під час виконання певної функційної можливості ПЗ припинило функціювання на час, що перевищує заданий пороговий час?
2. Чи після виконання певної функційної можливості відбулась втрата повноти даних?
3. Чи після виконання певної функційної можливості відбувся витік даних?
4. Чи після виконання певної функційної можливості виникла неможливість одержання дозволеної користувачу інформації?

На кожне із запитань опитувальника для збору інформації про вразливість(і) передбачено відповідь «так» або «ні».

Правила класифікації вразливостей на основі аналізу відповідей на питання опитувальника для збору інформації про вразливість(і):

1. Якщо обрано відповідь «так» на перше запитання опитувальника для збору інформації про вразливість(і), то елемент матриці *sv[1,1] = 1*

2. Якщо обрано відповідь «так» на друге запитання опитувальника для збору інформації про вразливість(і), то елемент матриці *sv[1,2] = 1*

3. Якщо обрано відповідь «так» на третє запитання опитувальника для збору інформації про вразливість(і), то елемент матриці *sv[1,3] = 1*

4. Якщо обрано відповідь «так» на четверте запитання опитувальника для збору інформації про вразливість(і), то елемент матриці *sv[1,4] = 1*

Отже, розроблено опитувальники для збору інформації про відмову(и) та про вразливість(і), а також розроблено правила класифікації відмов на основі аналізу відповідей на питання опитувальника для збору інформації про відмову(и) та класифікації вразливостей на основі аналізу відповідей на питання опитувальника для збору інформації про вразливість(і). Розроблені правила дають можливість ідентифікувати та класифікувати відмову(и) та про вразливість(і), які мали місце в процесі функціювання ПЗ.

Розроблені опитувальники та правила є підґрунтям для проектування технології ідентифікації та класифікації відмов і вразливостей – Рис. 1.

Ця технологія надає висновок щодо наявності чи відсутності відмов(и) ПЗ; висновок щодо наявності чи відсутності вразливості(ей) ПЗ; висновок про тип відмови та тип вразливості в разі їх наявності, завдяки чому пропонована технологія забезпечення безпеки програмного забезпечення шляхом ідентифікації та класифікації відмов і вразливостей є корисною для користувачів ПЗ за рахунок ідентифікації та класифікації відмов і вразливостей.

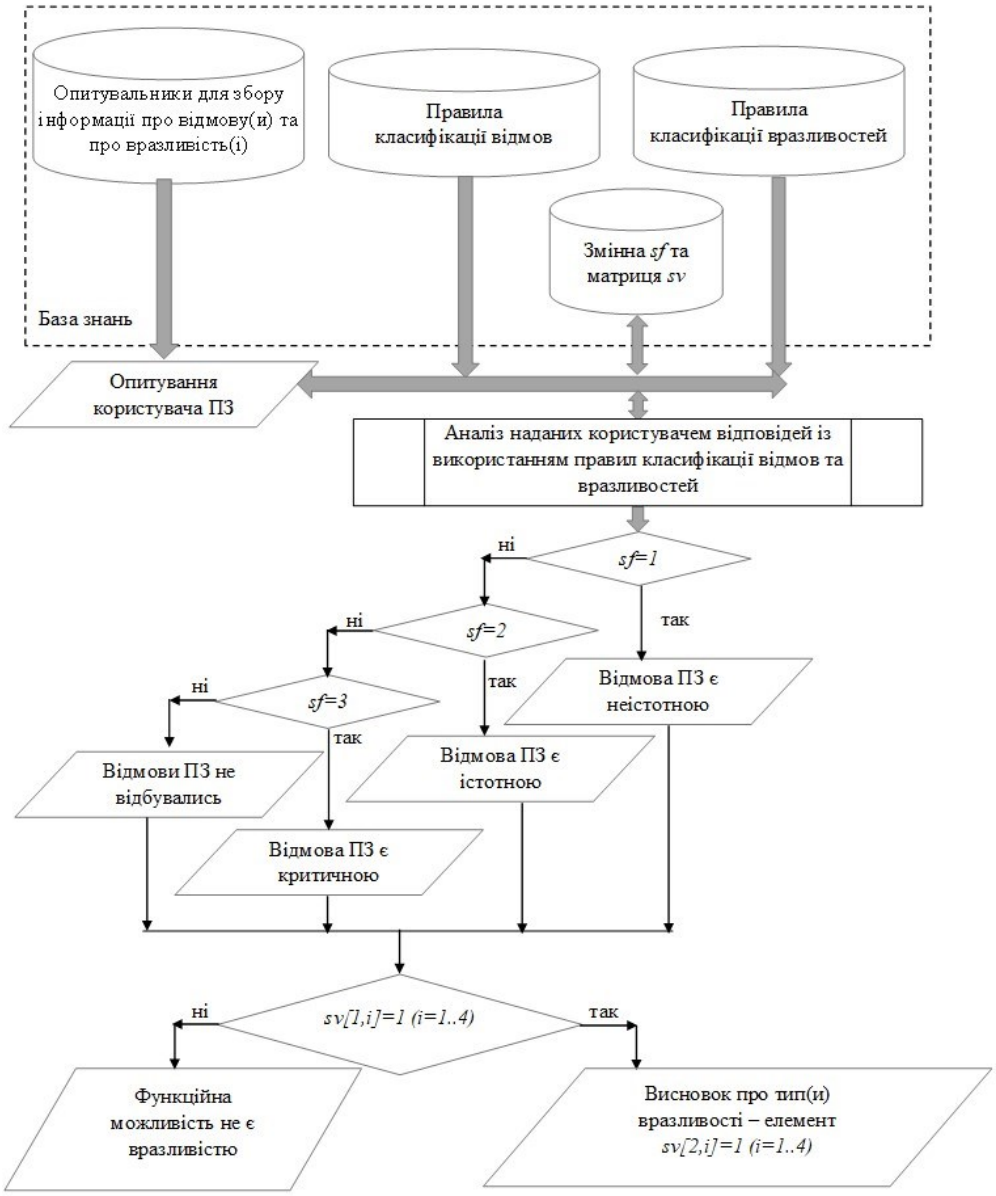

Рис. 1. Технологія ідентифікації та класифікації відмов і вразливостей

Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Всі основні підходи щодо забезпечення безпеки ПЗ спрямовані на запобігання повної відмови ПЗ, але не на ідентифікацію відмов та вразливостей ПЗ. Успішність підходів до забезпечення безпеки ПЗ можливе лише за рахунок ідентифікації та скорочення кількості помилок, тому актуальною задачею наразі є ідентифікація відмов та вразливостей ПЗ.

Проведений огляд літератури щодо відомих методів і технологій виявлення відмов і вразливостей програмного забезпечення показав, що, хоча проаналізовані методи та технології й мають величезний потенціал для галузі інженерії програмного забезпечення, проте жодне з відомих рішень не призначене для ідентифікації та класифікації відмов і вразливостей програмного забезпечення згідно із правилами класифікації відмов та правилами класифікації вразливостей. Отже, необхідно спроектувати та реалізувати технологію ідентифікації та класифікації відмов і вразливостей ПЗ на основі правил класифікації відмов та вразливостей ПЗ, що і є метою даного дослідження.

У статті розроблено опитувальники для збору інформації про відмову(и) та про вразливість(і), а також розроблено правила класифікації відмов на основі аналізу відповідей на питання опитувальника для збору інформації про відмову(и) та класифікації вразливостей на основі аналізу відповідей на питання опитувальника для збору інформації про вразливість(і). Розроблені правила дають можливість ідентифікувати та класифікувати відмову(и) та про вразливість(і), які мали місце в процесі функціювання ПЗ.

У статті розроблено технологію ідентифікації та класифікації відмов і вразливостей, яка надає висновок щодо наявності чи відсутності відмов(и) ПЗ; висновок щодо наявності чи відсутності вразливості(ей) ПЗ; висновок про тип відмови та тип вразливості в разі їх наявності, завдяки чому пропонована технологія є корисною для користувачів ПЗ за рахунок ідентифікації та класифікації відмов і вразливостей.

Література

1. What is Software Failure. URL: https://www.igi-global.com/dictionary/investigation-of-software-reliability-prediction-using-statistical-and-machine-learning-methods/59093.
2. Howard M. 24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them / M. Howard, D. LeBlanc, J. Viega. – Redmond: McGraw-Hill Education, 2010. – 432 p.
3. Yahoo says 500 million accounts stolen. URL: https://money.cnn.com/2016/09/22/technology/yahoo-data-breach.
4. Equifax Made Major Errors That Led to Hack, Ex-CEO Concedes. URL: https://www.bloomberg.com/news/articles/2017-10-02/ex-equifax-ceo-says-human-tech-failures-allowed-breach-to-occur.
5. Facebook Says Breach Affected About 50 Million Accounts. URL: https://www.bloomberg.com/news/articles/2018-09-28/facebook-says-security-breach-affected-about-50-million-accounts.
6. A.G. Underwood Announces Record $148 Million Settlement With Uber Over 2016 Data Breach. URL: https://ag.ny.gov/press-release/2018/ag-underwood-announces-record-148-million-settlement-uber-over-2016-data-breach.
7. It could have been prevented: it became known why government websites "went down". URL: https://www.epravda.com.ua/news/2022/01/14/681448/.
8. Ostrand T. Predicting bugs in large industrial software systems / T. Ostrand, E. Weyuker // Lecture Notes in Computer Science. 2013. – Vol. 7171. – Pp. 71-93.
9. Hovorushchenko T. Criteria and Rules for Classification of Software Failures and Vulnerabilities / T. Hovorushchenko // CEUR-WS. 2021. – Vol. 3039. – Pp. 217-224.

References

1. What is Software Failure. URL: https://www.igi-global.com/dictionary/investigation-of-software-reliability-prediction-using-statistical-and-machine-learning-methods/59093.
2. M. Howard, D. LeBlanc, J. Viega. 24 Deadly Sins of Software Security: Programming Flaws and How to Fix Them. – Redmond: McGraw-Hill Education, 2010. – 432 p.
3. Yahoo says 500 million accounts stolen. URL: https://money.cnn.com/2016/09/22/technology/yahoo-data-breach.
4. Equifax Made Major Errors That Led to Hack, Ex-CEO Concedes. URL: https://www.bloomberg.com/news/articles/2017-10-02/ex-equifax-ceo-says-human-tech-failures-allowed-breach-to-occur.
5. Facebook Says Breach Affected About 50 Million Accounts. URL: https://www.bloomberg.com/news/articles/2018-09-28/facebook-says-security-breach-affected-about-50-million-accounts.
6. A.G. Underwood Announces Record $148 Million Settlement With Uber Over 2016 Data Breach. URL: https://ag.ny.gov/press-release/2018/ag-underwood-announces-record-148-million-settlement-uber-over-2016-data-breach.
7. It could have been prevented: it became known why government websites "went down". URL: https://www.epravda.com.ua/news/2022/01/14/681448/.
8. T. Ostrand, E. Weyuker. Predicting bugs in large industrial software systems // Lecture Notes in Computer Science. 2013. – Vol. 7171. – Pp. 71-93.
9. T. Hovorushchenko. Criteria and Rules for Classification of Software Failures and Vulnerabilities // CEUR-WS. 2021. – Vol. 3039. – Pp. 217-224.