DOI 10.31891/2307-5732-2022-305-1-31-42
УДК 004.75:004.49

СТЕЦЮК М. В.
https://orcid.org/0000-0003-3875-0416
e-mail: mikstt777@gmail.com
КАШТАЛЬЯН А. С.
https://orcid.org/0000-0002-4925-9713
e-mail: yantonina@ukr.net
Хмельницький національний університет

АБСТРАКТНА МОДЕЛЬ ВПЛИВІВ ЗЛОВМИСНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА МЕТОД ЗАБЕЗПЕЧЕННЯ ВІДМОВОСТІЙКОСТІ СПЕЦІАЛІЗОВАНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

В роботі запропонована абстрактна модель впливів зловмисного програмного забезпечення (ЗПЗ) дає змогу розглядати об'єкти комп'ютерної системи, на які можуть впливати ЗПЗ та комп'ютерні атаки. І, тому, вона була використана як основа для розробленого нового методу забезпечення відмовостійкості спеціалізованої ІТ в умовах впливів ЗПЗ та комп'ютерних атак.

В результаті, застосування розробленого методу здійснюється в системі, яка має механізми для перебудови та використовує надмірності. Особливістю основних кроків розробленого методу згідно параметричного контролю цілісності програмних файлів є можливість його застосування до групи програмних файлів, які не мають сталої контрольної суми і, цим самим він розширює можливості відомого методу виявлення ЗПЗ, а саме методу контролю цілісності програм, основаного на підрахунку контрольних сум. Порівняно з відомим застосування цього методу, то в попередніх роботах цей метод не міг бути використаний для контролю цілісності певної групи виконуваних файлів, які мають неоднорідну внутрішню структуру. До цієї групи відносяться і файли типу mde, які отримуються при компіляції програм, написаних в середовищі MS Access. Їх особливістю є та обставина, що вони мають складну внутрішню структуру, яка включає в себе, окрім програмного коду, структури, що є елементами бази даних, такі як таблиці, індекси, схему реляційної бази даних та інші. Цей крок методу розроблений для його застосування в ІС з підвищеною відмовостійкістю та посиленим захистом від ЗПЗ та комп'ютерних атак, а саме у другому, локальному контурі захисту, інтегрованому в програмне забезпечення спеціалізованої ІТ. Це диктується тією обставиною, що для його реалізації необхідна специфічна інформація про параметри реалізації програмного файла, яка є невідомою на загальносистемному рівні, але відома на локальному, оскільки є інформацією, отриманою в процесі проектування та реалізації цієї частини спеціалізованої ІТ як єдиного цілого. Тому, розширено сферу застосування методу виявлення ЗПЗ на основі підрахунку контрольної суми на файли з несталими контрольними сумами, як кроку методу забезпечення відмовостійкості ІТ.

В результаті застосування розробленого методу здійснюється в системі, яка має механізми для пербудови та використовує надмірності. Для дослідження розробленого методу розроблено методику оцінювання його ефективності в частині надмірногстей та резервування. Проведені експериментальні дослідження та оціночні розрахунки підтверджують ефективність розробленого методу забезпечення відмовостійкості ІТ в умовах впливів ЗПЗ та комп'ютерних атак.

Ключові слова: метод забезпечення відмовостійкості, зловмисне програмне забезпечення, спеціалізовані інформаційні технології, надмірності

MYKOLA STETSYUK, ANTONINA KASHTALIAN
Khmelnytskyi National University

ABSTRACT MODEL OF INFLUENCE OF MALICIOUS OF SOFTWARE AND METHOD OF ENSURING THE FAILURE RESISTANCE OF SPECIALIZED INFORMATION TECHNOLOGIES

The proposed abstract model of the effects of malicious software (SDR) allows us to consider the objects of the computer system that may be affected by SDR and computer attacks. Therefore, it was used as a basis for a new method of ensuring the resilience of specialized IT in the face of SDR and computer attacks.

As a result, the application of the developed method is carried out in a system that has mechanisms for restructuring and uses redundancies. A feature of the main steps of the developed method according to the parametric control of program file integrity is the possibility of its application to a group of program files that do not have a fixed checksum and thus it expands the possibilities of the known method of detecting. Compared with the known application of this method, in previous work, this method could not be used to control the integrity of a certain group of executable files that have a heterogeneous internal structure. This group includes files of type mde, which are obtained when compiling programs written in MS Access. Their feature is the fact that they have a complex internal structure, which includes, in addition to program code, structures that are elements of the database, such as tables, indexes, relational database schema and others. This step of the method is designed for application in IP with increased fault tolerance and enhanced protection against RAM and computer attacks, namely in the second, local security loop integrated into specialized IT software. This is dictated by the fact that its implementation requires specific information about the parameters of the program file, which is unknown at the system level, but known locally, as it is information obtained in the design and implementation of this part of specialized IT as a whole. Therefore, the scope of the SCR detection method based on checksum calculation for files with volatile checksums as a step of the IT fault tolerance method has been expanded.

As a result, the application of the developed method is carried out in a system that has mechanisms for rebuilding and uses redundancies. To study the developed method, a method of evaluating its effectiveness in terms of redundancy and redundancy has been developed. Experimental studies and evaluation calculations confirm the effectiveness of the developed method of ensuring the resilience of IT in the face of SDR and computer attacks.

Keywords: fault tolerance method, malicious software, specialized information technologies, redundancies

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

Проведений аналіз розвитку і поширення зловмисного програмного забезпечення та різноманітності в проведенні комп'ютерних атак підтверджує, що проблема протидії зловмисному програмному забезпеченню та комп'ютерним атакам буде залишатись актуальною і, її актуальність, буде тільки зростати по мірі становлення інформаційного суспільства, базованого на використані комп'ютерних інформаційних систем. Вирішування цієї проблеми є безперервним процесом, успішність якого в уникненні загроз, які створює зловмисне ПЗ та комп'ютерні атаки, можлива за умови використання наукових підходів, в основі яких використовуються моделі загроз лежать та методи, розроблені на їх основі. Вони, як правило, є локальними і, не охоплюють весь спектр загроз, які створює зловмисне ПЗ функціонуванню комп'ютерної системи. Крім великого спектру напрямків для зловмисників в комп'ютерних системах, важливим напрямом для них є інформаційні системи. Як правило, вони на сьогодні є переважно розподіленими. Тому, їх проєктування має враховувати особливості функціонування при виконанні поставлених на них задач, які можуть виконуватись при впливах зловмисного програмного забезпечення та комп'ютерних атаках. В зв'язку з цим необхідним науковим завданням є розробка спеціалізованих ІТ, в яких будуть закладені можливості протидії зловмисним проявам, що дасть змогу розробляти на їх основі стійкі до таких впливів інформаційні системи.

Аналіз досліджень та публікацій

Розглянемо відомі наукові рішення щодо забезпечення відмовостійкості в інформаційних технологіях.

В роботах [1, 2] розглянуто проблему забезпечення та підвищення надійності багатофункційної інформаційної системи, внаслідок виникнення загроз втрати або спотворення інформації, що обробляється у системі. Проведено аналіз існуючих підходів та методів забезпечення та підвищення надійності, складовою частиною якої є показники відмовостійкості та живучості комплексів програмних та технічних засобів. Обґрунтована необхідність використання методів підвищення надійності, основним з яких, автор вважає метод застосування структурної надмірності, а всі інші методи підвищення надійності можуть використовуватись як додаткові до основного.

В роботі [3] розглядаються методи високої відмовостійкості такого програмного компонента як SQL-сервер та методи її досягнення, такі як відмовостійка кластеризація; переміщення журналів (log shipping) - технологія, яка полягає в автоматизації резервного копіювання БД та її відновлення на іншому сервері.

В роботі [4] розглянута нетривіальна наукова задача самовідновлення та самоорганізації функцій кібернетичних систем, яка є фундаментальною в забезпеченні відмовостійкості та живучості ІС. Приділена увага теорії маскування та надлишковості компонент яка узагальнена W.H. Pierce у концепцію стійкості до відмов (the concept of failure tolerance) або відмовостійкості. Розглянута концепція програмної відмовостійкості, одним із ефективних методів якої є методом N – версійного програмування, що став фундаментальним для досягнення відмовостійкості в цілому.

В роботі [5] відмічається, що на сьогодні існують два типи підходів до забезпечення відмовостійкості під час роботи – архітектурний та алгоритмічний. Архітектурні, будучи алгоритмічно незалежними, можуть базуватись на ручних методах реконфігурації масиву елементів системи, відновлюючи у такий спосіб роботоздатність системи. Інший підхід полягає у маскуванні відмов шляхом негайного відновлення працездатності системи, при якому виникнення відмови не помічається. Цей спосіб ефективний для досягнення динамічної відмовостійкості і може бути реалізований за допомогою триразової апаратурної надмірності або N-кратної надмірності. Алгоритмічні підходи використовують властивості алгебри для цифрової обробки даних. Прикладом є забезпечення відмовостійкості на основі надлишкового кодування даних, що дозволяє відновити правильний результат при несправному елементі системи. Ще один підхід полягає в перебудові структури масиву елементів системи та їх алгоритмів так, щоб завдання було виконане на масиві меншого розміру. Цей підхід дозволяє будувати обчислювальні структури з амортизацією відмов.

В роботі [6] запропонована методика для порівняльної оцінки інформаційних мереж, щодо їх стійкості до відмов. Для досягнення технічного результату враховуються динаміка впливів на вузли мережі випадкових і навмисних перешкод, і навіть можливості відновлення зв'язку між транзитними вузлами. Для цього обчислюють значення показників доступності вузлів інформаційних мереж, час досягнення критичного співвідношення "небезпечних" та "безпечних" вузлів для кожного варіанта підключення абонентів, а також зв'язність суміжних "небезпечних" вузлів, що утворюють ланцюжки, що унеможливлюють обмін між абонентами.

В роботі [7] пропонується спосіб, як на базі декомпозиційного підходу отримати модель проектування технічних систем, які дозволяють об'єднати окремі показники надійності і безпеки у функцію живучості. Запропоновані сценарії одночасного проектування системи на основі критеріїв ефективності в номінальних і неномінальних режимах. Запропонована інформаційна технологія конструювання моделей дозволяє раціонально упорядковувати проектні варіанти технічних систем. Розроблені імітаційні моделі функцій живучості, що дозволяють будувати ці функції як за статистичними даними, так і за даними імітаційного моделювання і використовувати методи теорії імовірності і нечіткості в проектному аналізі.

Показано, що існуючу множину показників живучості можна подати як інтервали на шкалі вартості наслідків відмов.

В роботі [8] розглядаються методи забезпечення надійності інформаційно-автоматизованих систем на основі експертних оцінок. Розробка нової інформаційно-автоматизованої системи зазвичай супроводжується труднощами в ризиках. Процес виявлення та пом'якшення ризиків є одним із важливих напрямків розвитку програмної системи. Методи в пропонованій статті ґрунтуються на оцінці та пом'якшенні ризиків. Застосовується для підвищення надійності та відмовостійкості інформаційно-автоматизованих систем. Ризики інформаційних та автоматизованих систем дуже важливі, оскільки збій у системі може призвести до значних фінансових втрат, а іноді і до великих втрат життя тощо. Методи можуть бути використані для раннього проектування розробки програмного забезпечення та визначають найбільше ефективні стратегії пом'якшення виявлених ризиків.

В роботі [9] приводиться вичерпний огляд методів відмовостійкості для високопродуктивних обчислень. Акцент робиться на аналітичних моделях ефективності. Приводиться огляд методів загального призначення, включаючи кілька протоколів відновлення контрольних точок і відкату. Досліджуються різні джерела помилок і несправностей у великих системах; розглядається набір методів, які можна застосувати для розробки відмовостійкого програмного забезпечення, а саме: метод прогнозування, який передбачає наявність механізму, який попереджає користувача про майбутні несправності в системі; метод реплікації, який полягає в дублюванні всіх обчислень.

Загалом, автори бачать шлях до забезпечення відмовостійкості ПЗ у введені інформаційної надлишковості у дані та її підтримки під час обчислень. Крім того, в роботах [10]-[14] представлено відомі впливи зловмисного програмного забезпечення (ЗПЗ) та комп'ютерних атак на об'єкти комп'ютерних систем.

Враховуючи, що наукова задача забезпечення відмовостійкості спеціалізованих інформаційних технологій саме в умовах впливів ЗПЗ та комп'ютерних атак не розв'язана, тому вона є актуальною.

Виклад основного матеріалу

Абстрактна модель впливу зловмисного програмного забезпечення на об'єкти комп'ютерних систем

Для забезпечення стійкості ІС до впливів зловмисного програмного забезпечення та комп'ютерних атак в процесі їх функціонування, синтезуємо в ІТ сумісно з спеціалізованим функціоналом її призначання, також складові елементи, призначення яких полягатиме у підтримці працездатності ІС з виконання спеціалізованого функціоналу для виконання основного завдання в умовах впливів ЗПЗ та комп'ютерних атак. Задамо складові елементи спеціалізованої ІТ M_{IT} так:

$$M_{IT} = \left\{ F_0, F_1, F_2, \ldots, F_{N_{IT}}, A_{IT} \right\}, \tag{1}$$

де F_0 – функціонал основного завдання ІТ і обов'язково присутній в M_{IT}; F_i - i-й складовий елемент в ІТ, що забезпечує додатковий функціонал; $i = 1, 2, \ldots, N_{IT}$; N_{IT} – кількість додаткових складових в ІТ; A_{IT} – складовий елемент в ІТ, що активізує елементи $F_1, F_2, \ldots, F_{N_{IT}}$ в ІТ за настання певних подій чи запитів від елементу F_0 і він не містить додаткового функціоналу для виконання інших дій.

Оскільки сучасні ІС можуть мати різні архітектури, що впливатиме і на проєктування ІТ, а також вони переважно є розподіленими, то представлені в формулі 1 її складові елементи вважатимемо такими, що об'єднують відповідно в своїх елементах всі складові, які розміщені в різних комп'ютерних станціях, але мають складову, що відноситься до складової певної типу. Зокрема, при такому представленні матимемо такі співвідношення:

$$M_{IT} = \left\{ \begin{array}{l} F_0 \,|\, F_0 = \sum_{i=1}^{N_{IT}} F_{0,i} \\ F_1 \,|\, F_1 = \sum_{i=1}^{N_{IT}} F_{1,i} \\ \ldots \\ F_{N_{IT}} \,|\, F_{N_{IT}} = \sum_{i=1}^{N_{IT}} F_{N_{IT},i} \\ A_{IT} \,|\, A_{IT} = \sum_{i=1}^{N_{IT}} A_{IT,i} \end{array} \right\}, \tag{2}$$

де – функціонал основного завдання ІТ в i-й комп'ютерній станції і обов'язково присутній в M_{IT};

$i = 1,2,\ldots,N_{ks}$; N_{ks} – кількість комп'ютерих станцій, в яких встановлено компоненти ІС; $F_{j,i}$ - j-й складовий елемент в ІТ в i-й комп'ютерній станції, що забезпечує додатковий функціонал; $j = 1,2,\ldots,N_{IT}$; N_{IT}, i – кількість додаткових складових в ІТ; $A_{IT,i}$ – складовий елемент в ІТ в i-й комп'ютерній станції, що активізує елементи $F_{1,i}, F_{2,i}, \ldots, F_{N_{IT},i}$ в ІТ за настання певних подій чи запитів від елементу $F_{0,i}$ і він не містить додаткового функціоналу для виконання інших дій.

Не в усіх компонентах ІС, які розміщені в комп'ютерних станціях, можуть бути розміщені всі складові елементи $F_{j,i}$, де $i = 1,2,\ldots,N_{ks}$; N_{ks} – кількість комп'ютерих станцій, в яких встановлено компоненти ІС; $j = 1,2,\ldots,N_{IT}$. Також, в різних комп'ютерних станціях можуть бути різні складові елементи, які відносяться до одного і того ж типу. Зокрема, ці складові можуть відрізнятись для сервера та компнонненти в комп'ютерній станції. Але більшість складових елементів одного типу в різних компонентах комп'ютерних станцій може бути однаковою. Це крім спрощення розробки спеціалізованої ІТ дає змогу синхронізувати, також, засоби підтримки стійкості ІС при впливах зловмисного програмного забезпечення чи комп'ютерних атак за рахунок координації і взаємодії між ними напряму чи із залученням серверної частини ІС.

Це демонструє можливість до масштабування компнонентів спеіалізованих ІТ між різними комп'ютерними станціями в мережах та можливість до виконання завдання в межах однієї комп'ютерної станції.

Таким чином, спеціалізована ІТ M_{IT} представлена множиною (формула 2.) дає можливість врахувати різні архітектури при проєктуванні ІТ та різні наповнення функціоналом окремих її складових елементів, що узагальнює спеціалізовану ІТ через її таку архітектуру для використання в подальших дослідженнях щодо впливу на неї чи її компоненти в комп'ютерних станціях зі сторони ЗПЗ та комп'ютерних атак.

До складових елементів в ІТ синтезуватимемо такі: відмовостійкість, живучість, захист інформації. Ці складові елементи реалізовуватимемо як окремі завершені модулі, але з можливістю активації в умовах сигналізації про впливи та потреб, які вимагатиме функціонал основного завдання. Тобто при $N_{IT} = 3$, тоді узагальнена структура спеціалізованої ІТ матиме представлення зображене на рис. 1.

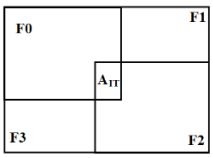

Рис. 1. Узагальнена структура спеціалізованої ІТ з елементами

Розглянемо представлення можливих впливів ЗПЗ та комп'ютерних атак на ІС в комп'ютерних мережах. Дослідження таких впливів важливо на всіх етапах функціонування ІС та в усіх комп'ютерних станціях в цілому і окремо. Впливи ЗПЗ та комп'ютетрних атак на комп'ютерні станції в мережі можуть бути спрямовані на їх різні об'єкти, як апаратні, апаратно-програмні так і на програмні. Причому, ці впливи можуть багатоетапні та одноетапні, віддалені, безпосередньо нспрямовані на об'єкт та опосередковано. Вони можуть реалізовуватись різноманітними засобами, які можуть бути звичайними засобами для роботи в мережі та комп'ютерних системах, а також спеціально створеними зловмисниками засобами. Все це урізноманітнення засобів для проведення зловмисних впливів не тільки ускладнює процес виявлення ЗПЗ та комп'ютерних атак спеціальними антивірусними засобами, але й ускладнює класифікацію саме зловмисних дій. На сьогодні для здійснення точної класифікації зловмисних впливів є невелике ознакове поле характеристик, тому спеціальні антивірусні засоби не забезпечують повного виявлення. Залишається множина ЗПЗ та комп'ютерних атак, які проникають через ці спеціальні антивірусні засоби. Тому, незважаючи на різноманітність ЗПЗ та комп'ютерних атак, не досліджуючи саме їх особливості та відображаючи їх у базах зловмисних програм та атак, що здійснюється в реалізаціях спеціальних антивірусних засобів, доцільним є дослідження можливих їх варіантів зловмисних впливів на конкретні об'єкти комп'ютерних систем. Тобто, побудова можливих впливів саме через формування множини таких впливів щодо конкретних об'єктів в комп'ютерних системах дасть змогу сформувати обмежену множину зловмисних впливів, кожен з елементів якої буде пов'язаний з певним об'єктом комп'ютерної системи, а також, наприклад, з певними елементами ІС, етапами її функціонування, включаючи початок роботи та завершення, зокрема і її різних компонент. В зв'язку з таким співвіднесенням впливів до об'єктів комп'ютерних систем з врахуванням їх часу функціонування, отримуємо зв'язки конкретних об'єктів в часі з можливими впливами на них. Крім того, впливи ЗПЗ та комп'ютерних атак можуть бути руйнуючими і неруйнуючими. Неруйнуючі впливи можуть поділятись на такі, що не досягли своєї мети і, тому, процеси, які створені ними функціонують сумісно чи паралельно з процесами створеними заданими користувачем чи комп'ютерною системою і такі, що націлені на інші об'єкти в комп'ютерній системі, тобто на об'єкти від заданої ІС і ресурсів, необхідних для її функціонування. Частина неруйнуючих вплив в певні моменти в майбутньому може перейти до категорії руйнуючих. Руйнуючі впливи можуть бути спрямовані на ІС, в яку будуть імплементовані механізми протидії, або на інші об'єкти комп'ютерної системи, які не пов'язані з ІС та ресурсами для її функціонування. Крім того, такі впливи можуть досягати як часткової мети в певний момент часу, так і в подальшому результуючої мети з виведення з ладу комп'ютерної станції, вузла мережі,

призупинки або знищення процесів, знищення інформації на жорсткому диску тощо.

Задамо впливи ЗПЗ та комп'ютерних атак множиною M_{VP} так:

$$M_{VP} = M_{VP,r} \cdot M_{VP,nr}, \quad (3)$$

де $M_{VP,r}$ – множина руйнуючих впливів; $M_{VP,nr}$ – множина неруйнуючих впливів.

Віднесення впливів до підмножин множини M_{VP} залежить від поточного моменту часу і може змінюватись. Задамо підмножини впливів переліком їх елементів так:

$$M_{VP,r} = \left\{ m_{VP,r,1}, \ldots, m_{VP,r,n_{VP,r}} \right\}, \; M_{VP,nr} = \left\{ m_{VP,nr,1}, \ldots, m_{VP,nr,n_{VP,nr}} \right\}, \quad (4)$$

де $m_{VP,r,i}$ – елемент множини $M_{VP,r}$, який означає i-й руйнуючий вплив в певний момент часу; $i = 1,2,\ldots,n_{VP,r}$; $n_{VP,r}$ – загальна кількість руйнуючих впливів; $m_{VP,r,j}$ – елемент множини $M_{VP,nr}$, який означає j-й неруйнуючий вплив в певний момент часу; $j = 1,2,\ldots,n_{VP,nr}$; $n_{VP,nr}$ – загальна кількість неруйнуючих впливів.

Частина неруйнуючих впливів множини $M_{VP,nr}$ в процесі свого здійснення може не зашкодити об'єктам комп'ютерної станції. Це може відбутись через заданих в них змістовність функціоналів так і через недосконаліть функціоналів в певному середовищі комп'ютерної станції. Інша частина неруйнуючих впливів в певний момент часу може перейти до руйнуючих. Такий розгляд впливів в динаміці є необхідним для побулови моделі впливів у співвіднесснні з об'єктами комп'ютерної станції, які динамічно змінюються.

Спрямування впливів ЗПЗ та комп'єютерних атак можу бути здійснене на ІС, для якої проєктуються механізми забезпечення стійкості при впливах і яка задана множиною M_{IT}, та ресурси, які забезпечують її функціонування. Також, спрямування впливів може бути здійснене на об'єкти комп'ютерної станції, які не пов'язані з ІС і вплив на них не впливатиме на функціонування ІС. Тому, розглядатимемо, як можливі варіанти, два типи таких впливів. Оскільки впливи динамічно можуть змінюватись з неруйнуючих в руйнуючі, то задамо множину M_{VP} переліком її елементів так:

$$M_{VP} = \{m_{VP,1}, \ldots, m_{VP,n_{VP}}\}, \quad (5)$$

де $m_{VP,i}$ – елемент множини M_{VP}, який означає i -тий вплив в певний момент часу; $i = 1,2, \ldots, n_{VP}$; n_{VP} – загальна кількість впливів.

Результатом впливів ЗПЗ та комп'ютерних атак на об'єкти комп'ютерних систем будуть наслідки, множину яких задамо так:

$$M_r = \{m_{r,1}, \ldots, m_{r,n_r}\}, \quad (6)$$

де $m_{r,i}$ – елемент множини M_r, який означає i-й наслідок впливу в певний момент часу; $i = 1,2, \ldots, n_r$; n_r – загальна кількість наслідків впливів.

Якщо впливи ЗПЗ та комп'ютерні атаки пов'язати з об'ктами комп'ютерних систем, на які вони спрямовані, і результатом таких взаємодій будуть наслідки, то ці наслідки представимо так:

$$M_r = \begin{pmatrix} m_{r,1,1} & \cdots & m_{r,1,N_{VP}} \\ \vdots & \ddots & \vdots \\ m_{r,N_{IT},1} & \cdots & m_{r,N_{IT},N_{VP}} \end{pmatrix}, \quad (7)$$

де $m_{r,i,j}$ – елемент множини наслідків впливів на об'єкти комп'ютерних систем; $i = 1,2, \ldots, N_{VP}$; $j = 1,2, \ldots, N_{IT}$.

Введемо для множини об'єктів комп'ютерної системи та впливів ЗПЗ і комп'ютерних атак алгебраїчну структуру так:

$$\Omega = \Omega_{ks}, \quad \Omega_{VP}, \; \Omega_{RVP}, \quad (8)$$

де Ω_{ks} – множина об'єктів комп'ютерної системи, на які можуть бути здійснені впливи ЗПЗ та комп'ютерних атак; Ω_{VP} – множина функцій, які реалізують впливи ЗПЗ та комп'ютерних атак; Ω_{RVP} – множина предикатів заданих на множині Ω_{ks}, які відображають успішність/неуспішність при реалізації функцій з множини Ω_{VP}; $\alpha = 1$, $\beta = 1$ – арності операцій, тому тип системи $\tau = (1, 1)$.

В якості елементів множини Ω_{ks} об'єктів комп'ютерної системи розглядатимемо всі об'єкти файлової системи, завантажувального сектору диску, оперативної пам'яті, мережні пакети, які можуть бути об'єктами впливів ЗПЗ та комп'ютерних атак. Елементами множини Ω_{VP} є одиничні елементи, які містять єдиний функціонал, реалізація якого надає змогу здійснювати зловмисний вплив ЗПЗ та комп'ютерних атак на конкретний єдиний об'єкт комп'ютерної системи та їх комбінації. Для досягнення результату щодо впливу одиничний елемент з множини Ω_{VP} може залучати деякі з інших об'єктів комп'ютерної системи, тобто здійснювати опосередкований вплив, але вплив все-рівно спрямований на один об'єкт. Тоді, комбінація таких елементів формуватиме решту елементів цієї множини Ω_{VP}. Такі елементи множини Ω_{VP} є породжуючими для решти різних елементів цієї множини. Функції з множини Ω_{VP} успішно

реалізовуватимуть впливи не завжди, тому для представимо впливи ЗПЗ та комп'ютерних атак множиною предикатів Ω_{RVP}. Вона відображатиме результат успішного / неуспішного впливу ЗПЗ та комп'ютерних атак на об'єкти комп'ютерних систем. Предикати, які належать множині Ω_{RVP} визначимо так, що вони будуть істинними, якщо результат здійснення зловмисних впливів ЗПЗ та комп'ютерних атак на об'єкт чи об'єкти комп'ютерної системи буде успішим, тобто функція з множини Ω_{VP} виконається. Інакше, результат предикату буде хибним.

Тоді, перейдемо з формули (8) до абстрактно моделі, яку задамо так:

$$\mathcal{R} = \Omega_{ks}, \Omega_{RVP}, \quad (9)$$

де Ω_{ks} – множина об'єктів комп'ютерної системи, на які можуть бути здійснені впливи ЗПЗ та комп'ютерних атак; Ω_{RVP} – множина предикатів заданих на множині Ω_{ks}, які відображають успішність / неуспішність при реалізації функцій з множини Ω_{VP}; $\alpha = 1$, $\beta = 1$ – арності операцій, тому тип системи $\tau = (1, 1)$.

Якщо впливи ЗПЗ та комп'ютерних атак будуть успішними, тоді вони матимуть наслідки, тобто відноситимуться до множини M_r, яку задано за формулою (6). В результаті функція відображення елементів множини впливів M_{VP} в множину наслідків M_r:

$$\Omega_{RVP} : M_{VP} \overset{\Omega_{VP}}{\rightarrow} M_{RVP}. \quad (10)$$

Результатом такого представлення є абстрактна модель і множина функцій, які надають можливість представити процеси, які здійснюються в комп'ютерних системах при функціонування ІС та можливих впливів ЗПЗ і комп'ютерних атак на об'єкти комп'ютерних систем. Вона поєднує такі складові, як об'єкти комп'єтерних систем, зокрема і компоненти та елементи ІС, впливи на об'єкти та таслідки впливів. Таким чином, отримана абстрактна модель надає змогу деталізувати об'єкти для впливів і можливі наслідки, що стає основою для розробки методів, які забезпечуватимуть відмовостійкість, живучість ІС та захист інформації в ІС від таких впливів. Абстрактна модель є основою для створення спеціалізованої ІТ, стійке фукнціонування якої можливе в умовах впливів ЗПЗ та комп'ютерних атак. Також, ця модель може включати особливість, яка полягатиме в розподіленні об'єктів комп'ютерних систем в комп'ютерній мережі та компонентів спеціалізованої ІТ.

Метод забезпечення відмовостійкості спеціалізованої ІТ

При забезпеченні відмовостійкості спеціалізованої ІТ механізмами, які унеможливлюватимуть вплив ЗПЗ та комп'ютерних атак розглядатимемо компоненти ІТ як такі, що поділяються на серверні та клієнтські. Якщо ж серверні частини ІТ відсутні, то результати будуть використані і для клієнтських, які розглядатимуться як такі, що можуть мати частину можливостей серверних частин. Компоненти спеціалізованої ІТ містять програмну частину та вимагають певних апаратно-програмних засобів для свого функціонування, тому розгляд впливів ЗПЗ та комп'ютерних атак потрібно враховувати до цих двох складових. Згідно даних з матриці спряження (формула (7) впливів ЗПЗ та комп'ютерних атак з об'ктами комп'ютерних систем, на які вони спрямовані, і результатом таких взаємодій будуть наслідки. Тоді, необхідно розробити метод забезпечення відмовостійкості ІТ, який би унеможливив успішне виконання відображення згідно формули (10), тобто наявність елементів в матриці спряження (формула (7)), або зменшила б їх кількість чи вірогідність появи. Таким чином, була б забезпеченна відмовостійкість ІТ в умовах впливів ЗПЗ та комп'ютерних атак. З врахуванням необхідності інтеграції механізмів протидії ЗПЗ та комп'ютерним атакам, які можуть бути застосовні однаково до змістовних елементів з матриці спряження (формула (7), представимо метод забезпечення відмовостійкості ІТ основними кроками, які відноситимуться як до клієнтської частини, так і до серверної частин.

Розглянемо перший крок методу забезпечення відмовостійкості ІТ, суть якого полягатиме у використанні блокових міток. клієнтської частини ІС при реалізації. Стосовно прикладного програмного забезпечення, до якого відносяться клієнтські автоматизовані робочі місця (АРМ), то критичні помилки, які можуть проявитись в ході експлуатації робочих місць ІС, фіксуються разом із своїми параметрами в реєстрі системи в автоматичний спосіб і, в подальшому використовується для аналізу з метою усунення причин, що їх викликали. Це стало можливим завдяки стратегії, яка базована на привнесенні деякої надмірності в програмне забезпечення АРМ ІС, по аналогії із методами забезпечення відмовостійкості апаратної частини ІС. З цією метою всі розрахункові процедури, які гіпотетично, можуть містити критичні для функціонування АРМ помилки, розробляються із дотриманням певного однотипного шаблону побудови алгоритмів їх виконання. Суть цього першого кроку методу, в подальшому кроку згідно блокових міток, відображена на рис. 2.

В структурі етапів першого кроку методу алгоритм виконання будь-якої нетривіальної процедури розділяється на два взаємодіючих блоки. В першому блоці реалізується функція процедури ІС, а в другому обробник помилок. В процесі виконання деякої процедури, яка реалізує одну із функцій АРМ ІС, обидва блоки взаємодіють між собою, передаючи управління обчислювальним процесом один одному, поки виконувана функція не завершиться.

Суть першого кроку методу згідно блокових міток полягає в тому, що алгоритм, який реалізує функцію ІС, розділяється маркерами (мітка 1, ..., мітка n на рис. 2) на фрагменти за принципом функціональної завершеності.

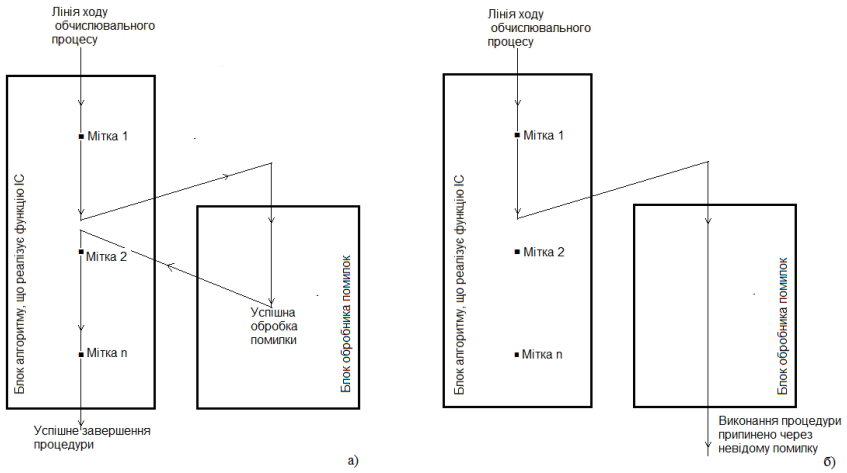

Рис. 2. Етапи першого кроку методу реалізації відмовостійкої процедури. а) для випадку успішного завершення процедури після виникнення помилки; б) для випадку, коли помилка невідома для обробника помилок

Перед початком виконання поточного фрагменту алгоритму в реєстр фатальних помилок заноситься інформація про гіпотетично можливу помилку (код екземпляра АРМ, код функції, № мітки, час і т. і.). В подальшому можливі наступні варіанти розвитку подій:

1. Фрагмент алгоритму функції успішно виконався. В цьому випадку інформація в реєстрі про помилку, що не сталась, знищується, а обчислювальний процес переходить до виконання наступного фрагмента.

2. В процесі виконання фрагменту сталась помилка, але вона успішно локалізована обробником помилок (рис. 2а). В цьому випадку інформація про помилку також може бути видалена з реєстру.

3. В процесі виконання фрагмента сталась помилка, яка не була локалізована обробником помилок (рис. 2б). В цьому випадку інформація про можливу помилку залишиться в реєстрі.

Таким чином, запропонований перший крок методу згідно блокових міток дозволяє типовим чином вирішувати задачу забезпечення відмовостійкості для всієї множини функцій клієнтської частини ІС.

Другий крок методу забезпечення відмовостійкості ІТ полягає у використанні функціонального резервування. Наступним із значимих внутрішніх факторів, що негативно впливають на відмостійкість є перевантаження апаратної платформи клієнтського ПК задачами, що може різко погіршити часові параметри виконуваних АРМ завдань, або навіть зробити неможливою його роботу, через вичерпання технічних ресурсів. Щоб нейтралізувати дію цього фактора на ІС, при розробці програмного забезпечення, а саме тієї його частини, яка відповідальна за реалізацію "бізнес-логіки" використано функціональне резервування (рис. 3). Наявність функціонального резерву "важких" розрахункових функцій дозволяє здійснювати маневр обчислювальними потужностями апаратної платформи ІС, в разі перевантаження окремих її ланок, підвищуючи таким чином відмовостійкість ІС. Оскільки процедура, яка функціонально резервується (наприклад Funk1 на рис. 3) розробляється в двох варіантах за одним і тим же алгоритмом, але в різних програмних середовищах, то для виконання на різних технічних засобах цей факт можна використати для нейтралізації такого негативного фактора, як наявність помилки в прикладному програмному забезпеченні АРМ, у випадку, коли в одному із варіантів процедури проявиться помилка. В цьому проявляється позитивна мультиплікативність ефекту функціонального резервування, що підвищує загальну відмовостійкість ІТ.

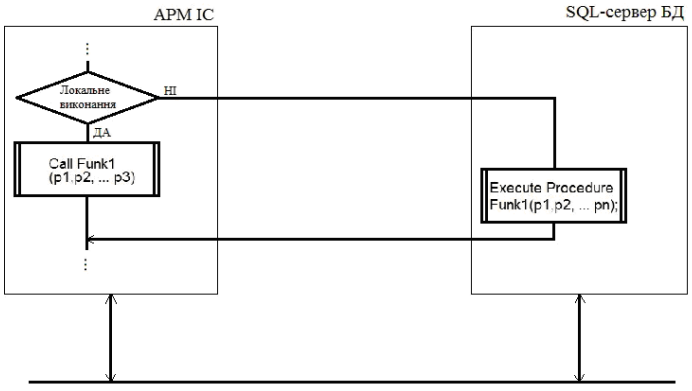

Рис. 3. Модель застосування функціонального резервування розрахункових функцій ІС в седовищі клієнт- серверної архітектуори

Третій крок методу забезпечення відмовостійкості ІТ полягає у перехресному резервуванні. Вирішення задачі, як завжди в таких випадках, полягає в створені деякого резерву. Аналіз роботи АРМ ІС показав, що деякі із них мають резерв часу та надлишковість продуктивності роботи. Тому, природнім було

рішення використовувати цей резерв в критичні моменти в роботі клієнтської частини ІС. В якості резерву тут слугує будь який інший, клієнтський ПК (рис. 4), який згідно плану подолання критичної ситуації, може взяти на себе забезпечення роботи АРМ, чий ПК вийшов з ладу. Такий підхід дозволяє не тримати в якості резерву окремий ПК, а також мати запаси комплектуючих, що зменшує експлуатаційні витрати, без втрати показників відмовостійкості системи в цілому.

Як правило, модулі програмного забезпечення, в налаштованому вигляді, зберігаються в репозитарії програмного забезпечення ІС та на тих клієнтських ПК, де вони плануються бути використаними в критичні моменти згідно плану резервування. У випадку виходу з ладу критичного обладнання комп'ютера, яке зробить неможливим виконання АРМ своїх функцій, воно переноситься на підходящий інший комп'ютер. Витрати часу на реконфігурацію клієнтської частини обчислюються хвилинами, що є прийнятною величиною для забезпечення живучості ІС, які виконують інформаційне забезпечення, наприклад, в такій предметній прикладній області, як фінансово-господарська діяльність ЗВО.

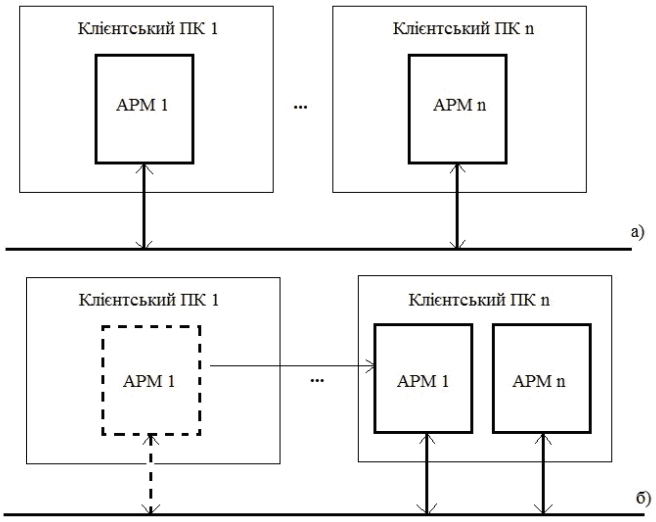

Рис. 4. Приклад структурного взаєморезервування клієнтської частини ІС. а) ІС до виходу з ладу клієнтського ПК1; б) ІС після реконфігурації системи в результаті виходу із ладу ПК1.

Така реконфігурація клієнтської частини стала можливою завдяки тому, що на клієнтських комп'ютерах, на яких виконується програмне забезпечення АРМ не зберігаються абсолютно жодні дані. При цьому, сам програмний модуль АРМ, для зручності, скомпонований в один файл і не потребує процедури інсталяції. Її достатньо скопіювати на інший комп'ютер. Після чого вона буде готовою до роботи. Такий підхід, дозволяє навіть після виходу із ладу кількох ПК, що само по собі має низьку вірогідність, зберегти повну функціональність ІС.

Є лише одне обмеження – кожен екземпляр програмного забезпечення АРМ попередньо повинен бути зареєстрований в ІС. Інакше, спроба запуску такої програми буде розглядатись як спроба несанкціонованого доступу до системи, навіть при правильних реєстраційних даних користувача. Контроль ІС за всіма екземплярами своїх АРМ дозволяє блокувати спроби зловмисників, яким вдалось оволоділи даними аккаунта користувача, отримати доступ до системи.

При цьому програма якою оволодів зловмисник, не отримує доступу до даних ІС, а сам факт спроби такої програми підключитись до системи фіксується в реєстрі фатальних помилок з відповідними даними, що дозволяє з їх використанням вжити організаційних заходів проти зловмисника.

Четвертий крок методу забезпечення відмовостійкості ІТ орієнтований на застосування в серверній частині і, тому, в клієнтській частині переважно не буде застосовний, крім випадків поєднання задач і осбливостей обох частин ІС.

Неможливо реалізувати спеціалізовану ІТ, що представляє собою ІС з БД з достатньо високими параметрами відмовостійкості, якщо вона не буде опиратись на реалізацію своєї серверної частини з достатнім рівнем резервування. З рис. 5 видно, як пропонується вирішення задачі підвищення відмовостійкості ІС, шляхом структурного резервування основних її компонентів, а саме її серверної частини. У випадку виходу з ладу основного сервера ІС, його функції може взяти на себе резервний, який має абсолютно однакові налаштування з основним.

При цьому основний та резервний сервери мають бути рознесені територіально і повинні живитись з різних ліній. Оскільки вихід з ладу зразу двох серверів є подія маловірогідна, то тим самим забезпечується висока відмовостійкість серверної частини АІС. Реконфігурація реальної системи, незважаючи на ручний режим перемикання, виконується за прийнятний відрізок часу для ІС, яка працює в ірреальному часі. Оскільки дзеркальна копія БД підтримується в актуальному стані службою реплікацій, то перемикання основної бази даних на БД - копію виконується зазвичай без втрат інформації. Але незначна втрата інформації при такій схемі все ж можлива. Це може трапитись при відмові деяких чутливих компонентів апаратної платформи сервера. Як правило, це останні запущені транзакції, виконання яких буде припинене

через відмову обладнання. І якщо це транзакції на зміну інформації в базі даних, то в цьому випадку інформація буде втрачена. Але оскільки така подія в життєвому циклі ІС сама по собі рідкісна, то такою можливою кількістю втрати інформації можна знехтувати. Після відновлення роботи серверної частини, операторам АРМ, чиї транзакції були втрачені, потрібно повторно виконати останні операції, для відновлення втраченої інформації.

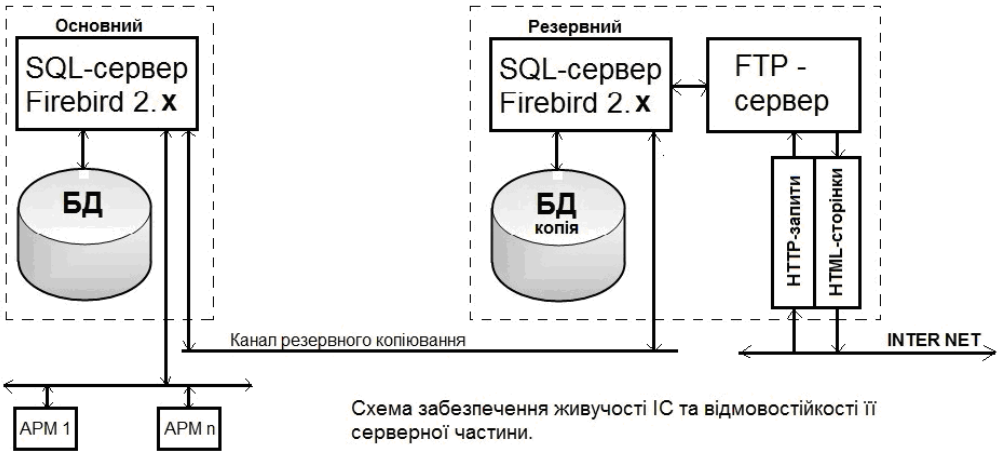

Рис. 5. Схема структурного резервування серверної частини ІС

Значно зменшити вірогідність втрати інформації можна, якщо робота серверної частини ІС буде знаходитись під постійним контролем. Для цього організується регулярне діагностування критичного обладнання сервера. Такий підхід дозволить виявляти назріваючу відмову і вчасно замінювати відповідний компонент, ще до виходу його із ладу. Наприклад, це може стосуватись дискових накопичувачів, якість дискових поверхонь котрих є надзвичайно критичними для функціонування всієї ІС. Така організація роботи дозволяє зменшити вірогідність виходу з ладу серверної частини ІС і тим самим призвести до збереження інформації.

Згідно з рис. 5 резервний сервер, окрім виконання функції резервування основного сервера, слугує джерелом даних даних для WEB-сервера, через який ІС видає інформацію для своїх віддалених користувачів. Такий крок методу згідно резервування серверної частини гарантує достатню високий рівень відмовостійкості в цілому.

Розроблений метод передбачає можливість самостійної перебудови ІС в процесі функціонування із залученням при цьому апаратно-програмних засобів. В процесі перебудови ІС виконання заданих функцій продовжується. Таким чином, метод забезпечення відмовостійкості ІТ в умовах впливів ЗПЗ та комп'ютерних атак надає змогу розширити можливості ІС в частині її адаптивності і відповідно автоматичної зміни апаратно-програмної конфігурації. Крім того, в кроках розробеного методу інтегровано два способи забезпечення відмовостійкості ІТ: залучення резервування; залучення надмірностей. Ця інтеграція поєднана з адаптивністю ІС.

Експериментальні дослідження та оцінювання ефективності методу забезпечення відмовостійкості спеціалізованої ІТ

Встановлення можливості застосування методу забезпечення відмовостійкості спеціалізованої ІТ в умовах впливів ЗПЗ та комп'ютерних атак здійснимо проведенням відповідних експериментальних досліджень та здійсненням оцінювання його ефективності. Оцінювання ефективності методу забезпечення відмовостійкості спеціалізованої ІТ в умовах впливів ЗПЗ та комп'ютерних атак здійснимо за критеріями, що відповідатимуть залученим в нього показників і відповідно функційних можливостей. Зокрема, такими досліджуваними показниками є такі: надмірності; автоматична зміна апаратно-програмного конфігурування ІС.

Здійснимо оцінювання впливу різних надмірностей на забезпечення відмовостійкості ІТ розробленим методом. Задамо множину надмірностей так:

$$M_{nd} = \left\{ m_{nd,1}, \ldots, m_{nd,p} \right\}, \tag{11}$$

де $m_{nd,i}$ – i-а надмірність у спеціалізованій ІТ; p – кількість розглядуваних надмірностей, які можуть бути реалізовані в ІТ.

Вважатимемо, що в структурі спеціалізованої ІТ, враховуючи особливості її застосування в умовах впливів ЗПЗ та клмп'ютерних атак, будуть такі надмірності: $m_{nd,1}$ - структурна; $m_{nd,2}$ - часова; $m_{nd,3}$ - інформаційна; $m_{nd,4}$ - функціональна; $m_{nd,5}$ - алгоритмічна; $m_{nd,6}$ - програмна; $m_{nd,7}$ - апаратна; $m_{nd,8}$ – багаторівнева. Задамо їх внесок в спеціалізовану ІТ в залежності від вагових коефіцієнтів:

$$O_{nd} = \alpha_i \cdot m_{nd,i}, \quad (12)$$

де – коефіцієнт ваги внеску надмірності в забезпечення відмовостійкості спеціалізованої ІТ; $m_{nd,i}$ - i-а надмірність; $i = 1,\ldots,p$; p – кількість надмірностей.

Тоді, унормуємо величину внеску надмірностей O_{nd} в забезпечення відмовостійкості спеціалізованої ІТ для встановлення її взаємозв'язку з впливами ЗПЗ і комп'ютерних атак на об'єкти комп'ютерних систем та наслідками так:

$$\mathrm{Q}_{nd} = \frac{\sum_{i=1}^{p} \alpha_i \cdot m_{nd,i}}{\sum_{i=1}^{p} m_{nd,i}}, \quad (13)$$

де $\sum_{i=1}^{p} m_{nd,i} = p$, $\sum_{i=1}^{p} \alpha_i = 1$.

З формули (13) випливає, що для певних впливів ЗПЗ та комп'ютерних атак можуть застосовуватись не всі надмірності. І це буде відображуватись відповідними величинами коефіцієнтів. Кількість успішно виконаних функцій з множини впливів Ω_{ks} буде зменшуватись при застосуванні надмірностей і залежатиме від кількості задіяних надмірностей, що виражатиметься величиною їх оцінки застосування Q_{nd}. Тому, множина предикатів Ω_{RVP}, заданих на множині Ω_{VP}, які будуть істинними зменшиться. В зв'язку з цим потрібно оцінити елемент матриці спряження (формула (7)) в контексті застосування методу, в якому використано надмірності. Для цього кожен елемент матриці спряження розглядатимемо окремо і так що до нього застосовано метод. А, також, випадок коли метод застосовний до декількох елементів матриці спряження одночасно. В цьому випадку необхідно встановити можливість втрати його ефективності. Для випадку застосування кроку методу з використанням надмірностей до одного елементу матриці спряження введемо функцію ефективності і задамо її в залежності від чинників впливу і протидії так:

$$\mathrm{Q}_{m_{r,i}} = \frac{1}{\mathrm{Q}_{nd}} \sum_{j=1}^{N_{VP}} \mathrm{Q}_{m_{VP,j}}, \quad (14)$$

де $m_{r,i}$ – елемент множини M_r, який означає i-й наслідок впливів в певний момент часу; $i = 1,2,\ldots,n_r$; n_r – загальна кількість наслідків впливів; $m_{VP,j}$. – елемент множини M_{VP}, який означає i-й вплив в певний момент часу; $i = 1,2,\ldots,N_{VP}$; N_{VP} – загальна кількість впливів; Q_{nd} - унормована величина внеску надмірностей для протидії впливам; $\mathrm{Q}_{m_{r,i}}$ – величина, яка відображає наслідок впливів після протидії впливам надмірностей; $\mathrm{Q}_{m_{VP,j}}$ – унормована величина впливів, що реалізовані функціями і виражена відповідними їх оцінками порівняно між всіма функціями впливів.

Впливів може бути декілька або один, або всі наявні. Тому, протидія їм засобами відмовостійкості може знижуватись при одночасному здійсненні широкого спектру різних впливів. Це відображено в формулі (14). Але всі ці впливи чи один вплив зосереджені на один об'єкт комп'ютерної системи в формулі (14). Результатом цієї формули (14) буде наслідок впливів відмінний від наслідку, який отримувався б без залучення надмірностей з першого кроку методу забезпечення відмовостійкості.

Якщо об'єктів комп'ютерної системи декілька і на них будуть зосереджені впливи, тоді це теж понижуватиме результат стійкості до впливів, бо засоби забезпечення відмовостійкості будуть додатково витрачати ресурси комп'ютерної системи. Тоді, результат щодо впливів оцінимо так:

$$\sum_{i=1}^{N_r} \mathrm{Q}_{m_{r,i}} = \frac{N_r}{\mathrm{Q}_{nd}} \sum_{j=1}^{N_{VP}} \mathrm{Q}_{m_{VP,j}} \quad (15)$$

Права частина рівності відображає, що загальна оцінка відмовостійкості в цьому випадку відображає зниження можливості зміни наслідків впливів. Таке оцінювання масштабуємо в межах розподіленої системи і отримує результат для сервера та комп'ютерних станцій, в які встановлено компоненти ІС.

Таким чином, отримані формули (14), (15) дають змогу оцінити вплив надмірностей щодо наслідків впливів для забезпечення відмовостійкості ІТ.

Резервування в спеціалізованій ІТ, яке впливає на забезпечення її відмовостійкості, є частиною заходів з динамічної перебудови системи і можу бути оцінене виходячи із часового використання серверних компонент, часу їх використання.

Експериментальні дослідження щодо перевірки ефективності розробленого методу забезпечення відмовостійкості ІТ проводимо в два етапи. Спочатку досліджуємо ІС без імплементованого в неї методу. Після цього на другому етапі досліджуємо ІС з імплементованим в неї розробленим методом. При

постановці такого експерименту суттєвим аспектом виступають джерела впливів. Можуть бути варіанти, коли ІС буде працювати тривалий час, щоб за тривалий час з певною ймовірністю можливо було отримати впливи, які призведуть до активізації засобів забезпечення відмовостійкості або якщо їх не імплементовано в ІС, тоді фіксації таких впливів. Але тоді вплив на ІС для експериментів для двох таких випадків не буде однаковим, бо він буде реальним і випадковим. Для проведення потрібна тривалість експерименту протягом дуже тривалого часу, наприклад року. Це пов'язано з тим, що аналіз повідомлень про комп'ютерні атаки в межах України дає статистику масованих атак приблизно три на два роки за останні 6-8 років. Для двох експериментів, можна вирішити питання проведення їх послідовно, тоді потірбно два роки, або паралельно експлуатувати дві однакових ІС, в одній з яких навявні засоби забезпечення відмовостійкості, а в іншій відсутні. Крім того, тривалість експериментів можна зменшити, створивши в закритому середовищі кіберполігон і встановити в ньому штучні джерела впливів.

Результати експериментів ІС записує в свій внутрішній формат, який за потреби після проведених експериментів може бути досліджений. На рис.6 (фрагмент файлу збереження результатів подій) зображено фрагменти з результатів роботи двох ІС. В одній ІС не було імплементовано засобів забезпечення відмовостійкості і, тому, результатом стала статистика виведення з ладу компонентів ІС в процесі функціонування в умовах впливів ЗПЗ та комп'ютерних атак. В другій ІС, в яку було імплементовано засоби забезпечення відмовостійкості, результатом стали час впливів, час залучення засобів забезпечення відмовостійкості і тривалість та результативність. В обох випадках дослідження впливів фіксувались саме щодо необхідності забезпечення подій, які викликані внутрішніми нерегламентними роботами.

Logfile001.txt [vk.com]

NPP	ARM	PVR	KVR	Error	NAMERROR	IP_BD	IP_ARM
200732	108	02.01.2021 16:10:07	02.01.2021 19:41:47			192.168.168.1	192.168.168.15
200733	40	04.01.2021 8:33:11	04.01.2021 9:02:26			192.168.168.1	192.168.168.10
200734	105	04.01.2021 8:35:43	04.01.2021 11:21:23			192.168.168.1	192.168.168.10
200735	51	04.01.2021 8:37:18	04.01.2021 11:21:26			192.168.168.1	192.168.168.10
200736	208	04.01.2021 8:42:08	04.01.2021 16:57:05			192.168.168.1	192.168.168.9
200737	83	04.01.2021 8:44:09	04.01.2021 8:44:46			192.168.168.1	192.168.168.9
200738	89	04.01.2021 8:48:36	04.01.2021 11:21:21			192.168.168.1	192.168.168.10
200739	69	04.01.2021 8:49:59	04.01.2021 13:45:48			192.168.168.1	192.168.168.9
200740	8	04.01.2021 8:55:38	04.01.2021 13:34:37			192.168.168.1	192.168.168.6

Рис. 6. Log-файл подій в ІС

Результати експериментальних досліджень підтвердили ефективність розробленого методу забезпечення відмовостійкості ІТ ЗПЗ та комп'ютерних атак. Розрахунки оціночних значень для надмірностей та резервування за даними з експерименту над розробленою ІС вказують приблизно на 87 відсотків більше порівняно з ІС, в яку не імплементовано розроблений метод.

Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Запропонована абстрактна модель дає змогу розглядати об'єкти комп'ютерної системи, на які можуть впливати ЗПЗ та комп'ютерні атаки. І, тому, вона виступає основою розробленого нового методу забезпечення відмовостійкості спеціалізованої ІТ в умовах впливів ЗПЗ та комп'ютерних атак.

В результаті, застосування розробленого методу здійснюється в системі, яка має механізми для перебудови та використовує надмірності. Для дослідження розробленого методу розроблено методику оцінювання його ефективності в частині надмірногстей та резервування. Проведені експериментальні дослідження та оціночні розрахунки підтверджують ефективність розробленого методу забезпечення відмовостійкості ІТ в умовах впливів ЗПЗ та комп'ютерних атак.

Література

1. Царегородцев А.В. Решение проблемы повышения надежности информационно - управляющих систем кластерным методом / Царегородцев А.В., Савельев И.А. // Вестник Российского университета дружбы народов. – Серия: Инженерные исследования, 2007. – С. 79-84.

2. Михеев В. А. Системный анализ методов обеспечения и повышения надежности многофункциональной информационной системы / Михеев В. А. // Известия Южного федерального университета. Технические науки. – 2009. – С. 24-34.

3. Методы достижения высокой отказоустойчивости [Електроний ресурс] // OSP – Гид по технологиям цифровой трансформации. – Режим доступу : https://www.osp.ru/winitpro/2003/12/13029028

4. Мудла Б.Г. Гарантоздатність як фундаментальний узагальнюючий та інтегруючий підхід / Б.Г. Мудла, Т.І. Єфімова, Р.М. Рудько // Математичні машини і системи. – 2010. – № 2. – С. 148–165.

5. Мартиросян А.Г. Основные методы обеспечения отказоустойчивости специализированных вычислительных устройств цифровой обработки сигналов / Мартиросян А.Г., Калмыков М.И. // Современные наукоемкие технологии. – 2014. – № 3. – С. 62-67. URL: http://www.top-technologies.ru/ru/article/view?id=34112.

6. Голуб Б.В. Методика оценки живучести распределенных информационных систем / Б.В. Голуб, Е.М. Кузнецов, Р.В. Максимов // Вестник СамГУ. Естественнонаучная серия. – 2014. – № 7(Ц8). – С. 221–232.

7. Боровська Т.М. Моделі ефективності і живучості технічних систем. / Боровська Т.М., Хомин Є.П., Северілов П.В. // Вісник Вінницького політехнічного інституту. – 2011. – № 1. – С. 89-95.

8. Boranbayev, A., Boranbayev, S., & Nurusheva, A. (2018). Development of a software system to ensure the reliability and fault tolerance in information systems based on expert estimates. *Advances in Intelligent Systems and Computing*, *869*, 924-935. https://doi.org/10.1007/978-3-030-01057-7_68

9. Jack Dongarra, Thomas Herault1, Yves Robert Fault tolerance techniques for high-performance computing. Series: Computer Communications and Networks. 2015, IX, 320 p. 113 illus. https://www.netlib.org/lapack/lawnspdf/lawn289.pdf дата звернення 23.1.22

10. Савенко О. С. Дослідження методів антивірусного діагностування комп'ютерних мереж / О. С. Савенко, С. М. Лисенко // Вісник Хмельницького національного університету. Технічні науки. – 2007. – № 2, т. 2. – С. 120–126.

11. Савенко О.С. Дослідження та аналіз блокування процесів в комп'ютерній системі / О.С. Савенко, Ю.П. Кльоц, С.В. Мостовий // Вісник Хмельницького національного університету. – 2007. – № 3, Том 1. – С. 248-251.

12. Савенко О.С. Оцінки ефективності та достовірності розподілених систем виявлення зловмисного програмного забезпечення в комп'ютерних системах локальних мережах / Савенко О.С., Нічепорук А.О., Паюк В.П. // Комп'ютерно-інтегровані технології: освіта, наука, виробництво – № 36. – 2019. – С. 134-139.

13. Lysenko S. Information technology for botnets detection based on their behaviour in the corporate area network / S. Lysenko, O. Savenko, K. Bobrovnikova, A. Kryshchuk, B. Savenko // Communications in Computer and Information Science, ISSN: 1865–0929. – 2017. – Vol. 718. – P. 166–181.

14. Pomorova O. Multi-Agent Based Approach for Botnet Detection in a Corporate Area Network Using Fuzzy Logic / Oksana Pomorova, Oleg Savenko, Sergii Lysenko, and Andrii Kryshchuk // Communications in Computer and Information Science. – 2013. – Vol. 370. – P. 243-254, ISSN: 1865-0929.

References

1. A.V. Tsaregorodtsev, I.A. Savelyev, Solving the problem of increasing the reliability of information-control systems by the cluster method. Bulletin of the Peoples' Friendship University of Russia. Series: Engineering Research, 2007, pp.79-84.

2. V. A. Mikheev. System analysis of methods for ensuring and improving the reliability of a multifunctional information system. Bulletin of the Southern Federal University. Technical Sciences, 2009, pp. 24-34.

3. Methods for achieving high fault tolerance. OSP - Guide to Digital Transformation Technologies. URL: https://www.osp.ru/winitpro/2003/12/13029028

4. Mudla B.G. Guarantee capacity as a fundamental generalizing and integrating approach / B.G. Mudla, T.I. Yefimova, RM Rudko // Mathematical Machines and Systems. - 2010. - № 2. - P. 148 - 165.

5. Martirosyan A.G., Kalmykov M.I. Basic methods for ensuring fault tolerance of specialized computing devices for digital signa1 processing // Modern science-intensive technologies. - 2014. - № 3. - P. 62-67; URL: http://www.top-technologies.ru/ru/article/view?id=34112.

6. Golub B.V. Methodology for assessing the survivability of distributed information systems / B.V. Golub, E.M. Kuznetsov, R.V. Maksimov // Bulletin of SamGU. Natural Science Series. - 2014. - № 7 (Ts8). – S. 221–232.

7. Borovska T.M. Models of efficiency and survivability of technical systems. Borovska T.M., Khomin E.P., Severilov P.V. // Bulletin of the Vinnitsa Polytechnic Institute. 2011. №1, pp. 89-95.

8. Boranbayev, A., Boranbayev, S., & Nurusheva, A. (2018). Development of a software system to ensure the reliability and fault tolerance in information systems based on expert estimates. Advances in Intelligent Systems and Computing, 869, 924-935. https://doi.org/10.1007/978-3-030-01057-7_68

9. Jack Dongarra, Thomas Herault1, Yves Robert Fault tolerance techniques for high-performance computing. Series: Computer Communications and Networks // 2015, IX, 320 p. 113 illus. https://www.netlib.org/lapack/lawnspdf/lawn289.pdf дата звернення 23.1.22

10. Savenko O.S Research of methods of antiviral diagnostics of computer networks / O.S Savenko, S.M Lysenko // Herald of Khmelnytskyi National University. Technical sciences. - 2007. - № 2, v. 2. - P. 120–126.

11. Savenko O.S., Klots Y.P, Mostoviy S.V. Research and analysis of process blocking in a computer system // Herald of Khmelnytskyi National University. Technical sciences. - 2007. - № 3, Volume 1.- P.248-251.

12. Savenko O.S., Nicheporuk A.O., Paiuk V.P. Estimates of efficiency and reliability of distributed malware detection systems in computer systems of local networks // Computer-integrated technologies: education, science, production, №36, 2019. - P.134-139.

13. Lysenko S. Information technology for botnets detection based on their behaviour in the corporate area network / S. Lysenko, O. Savenko, K. Bobrovnikova, A. Kryshchuk, B. Savenko // Communications in Computer and Information Science, ISSN: 1865–0929. – 2017. – Vol. 718. – Pp. 166–181.

14. Pomorova O. Multi-Agent Based Approach for Botnet Detection in a Corporate Area Network Using Fuzzy Logic / Oksana Pomorova, Oleg Savenko, Sergii Lysenko, and Andrii Kryshchuk // Communications in Computer and Information Science. – 2013. – Vol. 370. - PP.243-254, ISSN: 1865-0929.

Рецензія/Peer review : 26.01.2022 р. Надрукована/Printed : 27.02.2022 р.