

УДК 004.415.2

DOI: 10.31891/2219-9365-2020-65-1-15

СТЕЦЬОК М. В., ГОРОШКО А. В., САВЕНКО Б. О.
Хмельницький національний університет

МОДЕЛЬ ЗАБЕЗПЕЧЕННЯ ЖИВУЧОСТІ ТА ВІДМОВСТІЙКОСТІ СПЕЦІАЛІЗОВАНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В УМОВАХ РУЙНУЮЧОГО ВПЛИВУ ЗЛОВМИСНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

В статті розроблений підхід до визначення ефективності інформаційних технологій (ІТ) на основі врахування кількісних величин, які характеризують відмовостійкість та живучість, та може бути розширений для врахування інших характеристичних величин. Для забезпечення відмовостійкості та живучості ІТ розроблено систему заходів, в результаті виконання яких отримано ІТ вузькоспеціалізованого використання для різних сфер застосування, де супроводжуванні процеси відносяться до ірреального або нереального часу із досить високими параметрами відмовостійкості, живучості та загалом резилентності і, в той же час, прийнятним рівнем фінансових витрат на її експлуатацію в умовах руйнуючого впливу зловмисного програмного забезпечення. В результаті використання розроблених заходів було отримано ІТ вузькоспеціалізованого використання для різних сферах застосування, де супроводжуванні процеси відносяться до ірреального або нереального часу із досить високими параметрами відмовостійкості, живучості та загалом резилентності і, в той же час, прийнятним рівнем фінансових витрат на її експлуатацію.

Ключові слова: мережа приманок, зловмисні дії, виявлення комп'ютерних атак, прогнозування, корпоративні комп'ютерні мережі.

STETSYUK M., GOROSHKO A., SAVENKO B.
Khmelnytsky National University

MODEL FOR ENSURING THE SUSTAINABILITY AND FAILURE RESISTANCE OF SPECIALIZED INFORMATION TECHNOLOGIES IN THE CONDITIONS OF THE DESTRUCTIVE INFLUENCE OF MALICIOUS

Corporate computer networks of enterprises (organizations) use specialized IT to solve production problems or automate basic or auxiliary tasks, in particular, such as accounting and document management. Computer networks, when connected to the Internet, become objects of malicious activity.

The paper analyzes that the known methods and ways to ensure fault tolerance and survivability of specialized IT are insufficiently systematized and can not always be implemented due to the specifics of use and structure of specialized IT. Therefore, further research and development of new methods that would improve the resilience and survivability of specialized IT, including cyber-attacks and malware, is needed.

As a result of using the developed measures, IT of specialized use was obtained for various applications, where the accompanying processes belong to unreal or unreal time with high parameters of fault tolerance, survivability and overall resilience and, at the same time, an acceptable level of financial costs.

Thus, an approach to determining the effectiveness of IT based on quantitative values that characterize fault tolerance and survivability has been developed, and can be extended to take into account other characteristics. To ensure fault tolerance and survivability of IT, a system of measures has been developed which resulted in highly specialized IT applications for various applications, where the accompanying processes are unrealistic or unrealistic with high parameters of fault tolerance, survivability and overall resistance and, at the same time, acceptable financial costs for its operation.

Keywords: honeynet, malicious actions, detection of computer attacks, forecasting, corporate computer networks.

Вступ. Постановка задачі дослідження. В корпоративних комп'ютерних мережах підприємств (організацій) експлуатуються спеціалізовані ІТ для вирішення виробничих задач або автоматизації основних чи допоміжних задач, зокрема, наприклад обліку і ведення документообігу. Комп'ютерні мережі будучи під'єднаними до мережі Internet стають об'єктами для зловмисних дій. Захист інформаційних ресурсів мереж вирішуються за допомогою брандмауерів (firewall), антивірусів, систем виявлення атак (Intrusion Detection System, IDS), систем контролю цілісності, криптографічних засобів захисту. Але існуючі системи забезпечення захисту корпоративних мереж не забезпечують повного надійного захисту. Тому, необхідним є пошук більш ефективних шляхів захисту інформаційних ресурсів в корпоративних мережах від зловмисних дій. Для інформаційних технологій, які забезпечують життєдіяльність установи чи підприємства, тобто є спеціалізованими, наприклад, в такій сфері, як фінансово-господарська діяльність, питання живучості та відмовостійкості є більш ніж важливими, особливо, внаслідок зростання їх кількісних параметрів функціонування (збільшення користувачів, серверів, обсягів інформації в базах даних) і рівня складності, а особливо в умовах руйнуючого впливу зловмисного програмного забезпечення.

В цих умовах актуальності набирає питання розробки таких методів створення спеціалізованих ІТ, які б могли підтримувати свою живучість та відмовостійкість в умовах руйнуючого впливу зловмисного програмного забезпечення. Це було б одним із елементів з покращення рівня безпеки в корпоративних

комп'ютерних мережах. І його розвиток дозволив би створювати функціонально живучіші та відмовостійкіші спеціалізовані ІТ. Відомі методи не забезпечують достатнього рівня живучості та відмовостійкості, тому метою роботи є подальший розвиток відомих методів і створення нових для покращення безпеки в корпоративних комп'ютерних мережах за рахунок підвищення рівнів живучості та стійкості безпосередньо спеціалізованих ІТ.

1. Відомі методи створення спеціалізованих ІТ з підвищеним рівнем живучості та відмовостійкості в умовах руйнуючого впливу зловмисного програмного забезпечення

Під відмовостійкістю вважатимемо властивість системи зберігати повну або часткову працездатність у випадках відмов окремих елементів, що не пов'язані із зовнішніми нерегламентованими діями. Під живучістю інформаційної системи розуміється її властивість залишатися працездатною з допустимим зменшенням продуктивності в умовах негативних зовнішніх впливів (нерегламентованих дій) [1]. Ці поняття визначають таку мету – забезпечення доступності ІТ, яка досягається різними шляхами. Від забезпечення цих параметрів в прямій залежності знаходиться ефективність функціонування всієї спеціалізованої ІТ.

Відомі методи та способи забезпечення відмовостійкості та живучості спеціалізованих ІТ орієнтовані на різні їх типи, сфери застосування, особливості використання, розміщення в комп'ютерних засобах та особливостях реалізації в різних компонентах ІТ. Також, актуальним напрямом, який потребує дослідження є вплив зовнішніх факторів (розподілених атак, зловмисного програмного забезпечення) на функціонування ІТ та забезпечення відмовостійкості та живучості.

В [2] запропоновано інформаційну технологію оцінки структурної надійності технічних об'єктів, структура якої відповідає одному з відомих типів нейронних мереж. В [3, 4] розглянуто забезпечення надійності інформаційних систем. Для підвищення надійності інформаційних систем використовується підхід, заснований на оцінці та нейтралізації ризиків. В [5] показано як налагоджувані програмні системи складаються з великої кількості різних, критичних, некритичних та взаємозалежних конфігурацій. В [6] розглянуто як хмари стають важливою платформою для наукових програм роботи. В роботах [7, 8] досліджено хмарні програми, які розглянуто як складові з декількох компонентів хмарних служб, що спілкуються між собою через інтерфейси веб-служб, де кожен компонент виконує визначені функціональні можливості. В [9] представлено підхід для уникнення функціональних збоїв під час виконання в компонентних прикладних системах. В [10] запропонована проактивна схема відновлення, заснована на міграції служб, описані толерантні системи. В [11] толерантність відмов є головним питанням гарантування доступності та надійності критичних послуг, а також виконання застосунків. В [12] показано як хмарні обчислення пропонують нову потужність та гнучкість для високоефективних обчислювальних програм із забезпеченням великої кількості віртуальних машин для обчислювальних інтенсивних програм. В [13–16] кібер-стійкість та кібер-життєздатність представлено як тісно пов'язані між собою поняття, що поділяють подібні технології та практику. В [17–25] показано вплив на відмовостійкість та живучість ІТ різних типів зловмисного програмного забезпечення і комп'ютерних атак.

Відомі методи та способи забезпечення відмовостійкості та живучості спеціалізованих ІТ недостатньо систематизовані та не завжди можуть бути реалізовані через специфіку використання і будови спеціалізованих ІТ. Тому, необхідним є подальше дослідження та розробка нових методів, які б дозволили покращити відмовостійкість та живучість спеціалізованих ІТ, зокрема і від кібер-атак та зловмисного програмного забезпечення.

2. Забезпечення живучості та відмовостійкості спеціалізованих ІТ з підвищеним рівнем в умовах руйнуючого впливу зловмисного програмного забезпечення

Представимо спеціалізовану ІТ в корпоративних комп'ютерних мережах множиною її компонентів:

$$S_{IT} = \{S_1, S_2, \dots, S_n\}, \quad (1)$$

де S_i – i – та компонента спеціалізованої ІТ в корпоративних комп'ютерних мережах, $i = 1, 2, \dots, n$, n – кількість компонентів.

Для кожної компоненти S_i застосуємо функцію, яка включатиме всі критерії ефективності в корпоративних комп'ютерних мережах, застосування яких при розробці ІТ необхідно для подальшого користування нею. Зокрема, серед таких критеріїв будуть, також, критерії забезпечення відмовостійкості та живучості. Задамо критерії ефективності спеціалізованих ІТ вектором, компонентами якого будуть функції ефективності, що відповідатимуть конкретним критеріям:

$$K_e = (f_1, f_2, \dots, f_m), \quad (2)$$

де f_j – j -та функція, яка задає один з критеріїв ефективності, $j = 1, 2, \dots, m$, m – кількість функцій.

Враховуючи те, що в цілому задача досягнення максимальної ефективності залежить від конкретних критеріїв, які можуть бути пов'язані між собою і відповідно впливати один на одного, при цьому

покращення ефективності одного може призводити погіршення іншого. Крім того, оскільки спеціалізовані ІТ складаються з компонентів, до яких застосовуються ті ж критерії із заданого вектору, то задача ускладнюється тим, що частина компонентів ІТ є різною і, відповідно, досягнення ефективності за тими ж наборами критеріїв буде різною. Тому, вибір оптимальних розв'язків є складною багатокритеріальною задачею. Загальну постановку задачі пошуку найкращої ефективності для спеціалізованих ІТ в корпоративних комп'ютерних мережах сформулюємо так:

$$\begin{cases} K_e(S_{IT}) \rightarrow \max; \\ f_j(S_i) \rightarrow \max, i = 1, 2, \dots, n, j = 1, 2, \dots, m \end{cases} \quad (3)$$

Введемо функцію, яка буде визначати максимальне значення критерію ефективності:

$$F: K_e(S_{IT}) \rightarrow \max; \quad (4)$$

Значення критерію ефективності задамо виразом з врахуванням вагових коефіцієнтів:

$$K_e(S_{IT}) = \sum_{i=1}^n \sum_{j=1}^m \sum_{p=0}^{n_p} \sum_{q=0}^{n_q} (\alpha_{i,j,p,q} \cdot f_{j,q}(S_{i,p})), \quad (5)$$

де $\alpha_{i,j,p,q}$ – вагові коефіцієнти.

Розглянемо досягнення максимізації критеріїв за показниками відмовостійкості та живучості в конфігураціях інформаційних технологій, які побудовані на основі архітектури «клієнт – сервер» з їх забезпеченням за усіма ланках системи: від користувачів (клієнтська частина) до критично важливої серверної частини. Вибір для розгляду саме архітектури «клієнт – сервер» залежить від її особливостей, які проявляються в наступному: базові функції клієнтського застосунку розподіляються між клієнтом та сервером; програмне забезпечення автоматизованого робочого місця клієнтського комп'ютера працює з даними через запити до серверного програмного забезпечення; здійснюється повна підтримка багатокористувацької роботи; гарантується цілісність даних. Це відрізняє її від інших архітектур і дозволяє здійснити забезпечення відмовостійкості і живучості до кожної з ланок системи окремо.

Головною властивістю відмовостійкості є прозорість відмов її окремих компонентів для кінцевого користувача. Це означає, що відмовостійка система автоматично змінює свою конфігурацію у випадку відмови. Її програмне забезпечення в процесі виконання шукає обхідні шляхи, намагаючись в умовах відмови, привести виконувану функцію до успішного завершення. Задамо функцію $f_1(S_i), i = 1, 2, \dots, n$ визначення відмовостійкості в комп'ютерних системах в кількісному вигляді так:

$$f_1(S_i) = \frac{T_{f_1(S_i),1}}{T_{f_1(S_i),1} - (T_{f_1(S_i),2} + T_{f_1(S_i),3})}, \quad (6)$$

де i – кількість компонентів спеціалізованої ІТ, $i = 1, 2, \dots, n$, $T_{f_1(S_i),1}$ – час між сусідніми збоями; $T_{f_1(S_i),2}$ – час, необхідний для виявлення збою та пошуку шляху його обходу; $T_{f_1(S_i),3}$ – час, необхідний для відновлення ІТ після збою.

Як видно з формули (6), для ІТ з автоматичною системою забезпечення відмовостійкості вона буде наближатись до максимуму, через швидкість реакції. Для побудови таких систем немає теоретичних перешкод, але в практиці при їх реалізації, потрібно враховувати ряд значимих факторів: фінансові витрати реалізації автоматичної системи забезпечення живучості та відмовостійкості; складність системи. Для ІТ, призначених для інформаційного забезпечення у вузькій спеціалізованій предметній області, наприклад фінансово-господарська діяльність закладу вищої освіти, буде доцільним відмовитись від автоматичної системи керування відмовостійкістю на користь автоматизованої. При такому підході частина дорогих функцій управління надмірностями, присутніми в ІТ, буде покладена на людину, якщо це не загрожуватиме можливими значними втратами. Тоді, згідно формули (6), відмовостійкість $f_1(S_i)$ буде нижчою, ніж в першому випадку. Але вирішенням задачі побудови ІТ (аналогічно, як і в інших задачах проектування), є не забезпечення максимально можливої відмовостійкості системи, а знаходження прийняттого балансу параметрів системи, в рамках певного технологічного базису. А, також, в тому числі враховуючи вимоги критерію «відмовостійкість \ вартість». Дослідимо вирішення питань забезпечення відмовостійкості ІТ при використанні такої стратегії. Проаналізуємо фактори, що негативно впливають на відмовостійкість ІТ зі сторони клієнта. Негативні фактори, що впливають на відмовостійкість клієнтської частини ІТ поділяються на зовнішні та внутрішні. Серед зовнішніх факторів найбільшу загрозу представляють собою збої в роботі енергосистем живлення та природні явища, які можуть призвести до відмов компонентів комп'ютерів та комп'ютерних мереж. Для уникнення таких випадків живлення клієнтських комп'ютерів ІТ необхідно

виконувати від окремої лінії, яка обладнана захисними пристроями, наприклад, розрядниками. Для захисту від грозових наводок в довгих лініях комп'ютерних мереж, також, необхідно використовувати пристрої захисту.

Перед початком виконання поточного фрагменту алгоритму в реєстр фатальних помилок заноситься інформація про гіпотетично можливу помилку (код екземпляра робочого місця клієнта, 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.

Задамо функцію $f_2(S_i)$, в якій $i = 1, 2, \dots, n$, визначення живучості в кількісних одиницях в комп'ютерних мережах виразимо так:

$$f_2(S_i) = \frac{T_{f_2(S_i),1} + T_{f_2(S_i),2}}{T_{f_2(S_i),1}}, \quad (7)$$

де $T_{f_2(S_i),1}$ – час функціонування процесі ІТ в стандартному режимі роботи, $T_{f_2(S_i),2}$ – час витрачений на процеси забезпечення живучості, $i = 1, 2, \dots, n$.

Таке визначення функції живучості надає можливість відобразити стандартний режим роботи значенням одиниці, а при виникненні потреби у забезпеченні живучості і у випадку набагато тривалішого часу, ніж час стандартного режиму роботи, значення функції відображатиме кількісну порядкову величину. На основі формул (5)–(7) отримаємо значення ефективності для ІТ з врахуванням показників відмовостійкості та живучості:

$$K_e(S_{IT}) = \sum_{j=1}^m \sum_{p=0}^{n_p} \sum_{q=0}^{n_q} \left(\alpha_{1,j,p,q} \cdot \frac{T_{f_1(S_i),1}}{T_{f_1(S_i),1} - (T_{f_1(S_i),2} + T_{f_1(S_i),3})} + \alpha_{2,j,p,q} \cdot \frac{T_{f_2(S_i),1} + T_{f_2(S_i),2}}{T_{f_2(S_i),1}} \right), \quad (8)$$

де $\alpha_{1,j,p,q}$ – коефіцієнт для значення, яке визначає відмовостійкість в кількісних одиницях; $\alpha_{2,j,p,q}$ – коефіцієнт для значення, яке визначає живучість в кількісних одиницях; $\alpha_{1,j,p,q} + \alpha_{2,j,p,q} = 1$.

Аналогічно, доданками в формулі (6) та її конкретизації формулою (9) для двох величин можуть бути інші показники, які характеризують ефективність ІТ.

В результаті використання перелічених заходів було отримано ІТ вузькоспеціалізованого використання для різних сферах застосування, де супроводжуванні процеси відносяться до ірреального або нереального часу із досить високими параметрами відмовостійкості, живучості та загалом резилентності і, в той же час, прийнятним рівнем фінансових витрат на її експлуатацію.

4. Експерименти

Для визначення на скільки ефективними є запропоновані рішення із забезпечення відмовостійкості та живучості проведемо порівняння критерія ефективності для ІТ без забезпечення відмовостійкості та живучості і з включенням цих характеристик на основі формули (8).

Значення величини критерія ефективності ІТ, в якій не забезпечуються вимоги відмовостійкості і живучості отримуємо з формули (8) так: 1) вирішення проблем, пов'язаних із відсутністю забезпечення в ІТ реалізованих відмовостійкості та живучості, покладено на оператора чи адміністратора, який постійно моніторить функціонування ІТ; вирішення проблемних ситуацій здійснюється тільки при їх виявленні. В першому випадку розрахунок за формулою (9) може бути аналогічним і значення отримає величини на порядки перевищуватиме значення критерію для ІТ, де забезпечується відмовостійкість та живучість. Якщо ж розглядати другий варіант, тоді $K_e(S_{IT}) = 1$. В цьому випадку, відношення між значеннями визначається за формулою (9) і дозволяє встановити ефективність запропонованих рішень із забезпечення відмовостійкості та живучості, а також покращувати досягнення ефективності за рахунок коригування коефіцієнтів:

$$\mu = \frac{1}{\sum_{j=1}^m \sum_{p=0}^{n_p} \sum_{q=0}^{n_q} \left(\alpha_{1,j,p,q} \cdot \frac{T_{f_1(S_i),1}}{T_{f_1(S_i),1} - (T_{f_1(S_i),2} + T_{f_1(S_i),3})} + \alpha_{2,j,p,q} \cdot \frac{T_{f_2(S_i),1} + T_{f_2(S_i),2}}{T_{f_2(S_i),1}} \right)}, \quad (9)$$

де відсутність збоїв в роботі спеціалізованої ІТ або зовнішніх впливів означатиме, що час витрачений на їх обробку дорівнюватиме нулеві і відповідно відношення стане дорівнювати одиниці.

Якщо ж відбудеться збій або зовнішнє втручання, тоді значення μ буде більше одиниці. Ефективним значенням є значення мінімально відхилене від одиниці. Результати забезпечення відмовостійкості та живучості спеціалізованої ІТ зображені в реалізованій ІТ на рис. 1.

```
16 Apr 15 09:01:02 itzz0 run-parts (/etc/cron.hourly) [17261]: starting 0anacron
17 Apr 15 09:01:02 itzz0 run-parts (/etc/cron.hourly) [17270]: finished 0anacron
18 Apr 15 10:13:16 itzz0 CROND[17274]: (root) CMD (/Stecjk/db-hourly)
19 Apr 15 11:43:16 itzz0 sshd[30314]: False error in the operation of the
20 network device from 192.168.168.2 port 43760 ssh2
21 Apr 15 11:44:01,786 DEBUG :NetworkDevice eth0:
22 DEVICE="eth0"
23 BOOTPROTO="dhcp"
24 DEFROUTE="yes"
25 HWADDR="1C:C1:DE:78:C4:4C"
26 IPV6INIT="no"
27 NAME="eth0"
28 NM_CONTROLLED="yes"
29 ONBOOT="yes"
30 PEERDNS="yes"
31 PEERROUTES="yes"
32 TYPE="Ethernet"
33 IPV4_FALSE_MISTAKE=yes
34 UUID="ee9c32a3-47c2-4217-b817-82e1d91f6a5f"
35 Apr 15 11:44:12 itzz0 sshd[29043]: pam_unix(sshd:session): session closed
36 for user swm
37 Apr 15 11:44:56 itzz0 sshd[29078]: Network device configuration required ...
38 Apr 15 11:46:02,786 DEBUG : writeIfcfgFile eth1
39 to /etc/sysconfig/network-scripts/ifcfg-eth0 not needed
40 Apr 15 11:46:21,396 DEBUG : Network.write() called
41 Apr 15 11:46:21,397 DEBUG : /etc/sysconfig/network-scripts/ifcfg-eth1:
42 DEVICE=eth1
43 TYPE=Ethernet
44 UUID=8d82e92b-3b68-4829-a09b-c76783afecaa
45 ONBOOT=yes
46 NM_CONTROLLED=yes
47 BOOTPROTO=none
48 HWADDR=1C:C1:DE:78:C4:4D
49 IPADDR=192.168.1.2
50 PREFIX=24
51 DEFROUTE=yes
52 Apr 15 11:47:41 itzz0 sshd[30314]: pam_unix(sshd:session): session opened for user swm by (uid=0)
53 IPV6INIT=no
```

Рис. 1. Вміст файлів-відомостей про збої та зовнішні впливи

Для зручності всі рядки фрагмента лог-файла були пронумеровані, а критичні позиції виділені.

В позиції 19 виявлено фатальну помилку в роботі мережевого адаптера «eth0» в момент звернення користувача з IP 192.168.168.2.

В позиції 35,36 закривається поточна сесія користувача SWM.

В позиції 37 система сповіщає, що потрібна реконфігурація мережевих пристроїв.

В позиції 38 сповіщається, що пристрій «eth0» відключається.

В позиціях 40,41 повідомляється, що активується резервний мережевий адаптер «eth1»

В позиції 52 сповіщається, що відкривається сесія користувача SWM, яка була припинена через вихід із ладу мережевого адаптера «eth0».

Напрями подальших досліджень. Важливим напрямом подальших досліджень для покращення ефективності ІТ є розробка методу забезпечення ефективного захисту інформації безпосередньо в структурі ІТ та обчислювальних процесах, що протікають в процесах обчислень. Їх врахування в загальному критерії визначенні ефективності ІТ дозволить збалансувати такі величини як живучість, відмовостійкість та захист інформації, виражені в кількісному вигляді, та стане основою розробки спеціалізованої ІТ з покращеними характеристиками.

Висновки. Таким чином, розроблений підхід до визначення ефективності ІТ на основі врахування кількісних величин, які характеризують відмовостійкість та живучість, та може бути розширений для врахування інших характеристичних величин. Для забезпечення відмовостійкості та живучості ІТ розроблено систему заходів в результаті виконання яких отримано ІТ вузькоспеціалізованого використання для різних сфер застосування, де супроводжуванні процеси відносяться до ірреального або нереального часу із досить високими параметрами відмовостійкості, живучості та загалом резилентності і, в той же час, прийнятним рівнем фінансових витрат на її експлуатацію.

Література

1. ДСТУ 3396.2-97 Protection of information. Technical protection of information. Terms and definitions. State Committee of Ukraine, Kyiv (1997) [in Ukrainian]
2. Savelyeva, O. S., Krasnozhan, O. M., Lebedeva, O. U. (2014). Using the structural fault-tolerance index in project designing. Odes'kyi Politechnichnyi Universytet. Pratsi, 2, 130–135. doi: 10.15276/opu.2.44.2014.24.
3. S. Boranbayev, S. Altayev, A. Boranbayev. Applying the method of diverse redundancy in cloud based systems for increasing reliability, in Proceedings of the 12th International Conference on Information Technology: New Generations (ITNG 2015) (Las Vegas, Nevada, 2015), pp. 796–799.
4. Boranbayev A., Boranbayev S., Yersakhanov K., Nurusheva A., Taberkhan R. (2018) Methods of Ensuring the Reliability and Fault Tolerance of Information Systems. In: Latifi S. (eds) Information Technology - New Generations. Advances in Intelligent Systems and Computing, vol 738. Springer, Cham.
5. Chinnaiah, M., Niranjan, N. Fault tolerant software systems using software configurations for cloud computing. J Cloud Comp 7, 3 (2018). <https://doi.org/10.1186/s13677-018-0104-9>.
6. Zhu X, Wang J, Guo H, Zhu D, Yang LT, Liu L (2016) Fault-tolerant scheduling for real-time scientific workflows with elastic resource provisioning in virtualized clouds. IEEE Trans Parallel Distrib Syst 27(12):3501–3517. <https://doi.org/10.1109/TPDS.2016.2543731>.
7. Liu J, Zhou J, Buyya R (2015) Software rejuvenation based fault tolerance scheme for cloud applications In: 2015 IEEE 8th International Conference on Cloud Computing, 1115–1118, New York. <https://doi.org/10.1109/CLOUD.2015.164>.
8. Liu J, Wang S, Zhou A, Kumar SAP, Yang F, Buyya R (2016) Using Proactive Fault-Tolerance Approach to Enhance Cloud Service Reliability. IEEE Trans Cloud Comput PP(99):1–1. <http://dx.doi.org/10.1109/TCC.2016.2567392>.
9. Nicolo P (2013) A frame work for self-healing software systems In: IEEE 35th International Conference on Software Engineering (ICSE), 1397–1400. <https://doi.org/10.1109/ICSE.2013.6606726>.
10. Zhao W, Wenbing Z, Melliar-Smith PM, Moser LE (2010) Fault Tolerance Middleware for Cloud Computing In: 2010 IEEE 3rd International Conference on Cloud Computing, 67–74, Miami. <https://doi.org/10.1109/CLOUD.2010.26>.
11. Bala A, Chana I (2012) Fault tolerance- challenges, techniques and implementation in cloud computing, ISSN (Online): 16940814. IJCSI Int J Comput Sci 9(1). www.IJCSL.org.
12. Egwuotuoha IP, Chen S, Levy D, Selic B (2012) A fault tolerance framework for high performance computing in cloud, Cluster, Cloud and Grid Computing (CCGrid) In: Proceedings of the 12th IEEE/ACM international symposium. 13-16 May, 709–710. <https://doi.org/10.1109/CCGrid.2012.80>.
13. S. Pitcher, "New DoD Approaches on the Cyber Survivability of Weapon Systems (25 March 2019)," 25 March 2019. [Online]. Available: <https://www.itea.org/wp-content/uploads/2019/03/Pitcher-Steve.pdf>.
14. D. J. Bodeau, R. D. Graubart, R. M. McQuaid and J. Woodill, "Cyber Resiliency Metrics and Scoring in Practice: Use Case Methodology and Examples (MTR 180449)," The MITRE Corporation, Bedford, MA, 2018.
15. D. Fitzpatrick, D. Bodeau, R. Graubart, R. McQuaid, C. Olin and J. Woodill, "(DRAFT) Cyber Resiliency Evaluation Framework for Weapon Systems: Foundational Principles and Their Potential Effects on Adversaries," The MITRE Corporation, Bedford, MA, 2019.
16. NIST, "Initial Public Draft of NIST SSP 800-160 Volume 2, Systems Security Engineering: Cyber Resiliency Considerations for the Engineering of Trustworthy Secure Systems," 21 March 2018. [Online]. Available: <https://csrc.nist.gov/CSRC/media/Publications/sp/800-160/vol-2/draft/documents/sp800-160-vol2-draft.pdf>.
17. Lysenko, S., Savenko, O., Kryshchuk, A., Kljots, Y. Botnet detection technique for corporate area network. In: Proc. of the 2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems, pp. 363-368 (2013).
18. Savenko, O., Lysenko, S., Nicheporuk, A., Savenko, B.: Metamorphic Viruses' Detection Technique Based on the Equivalent Functional Block Search. CEUR Workshop, Vol. 1844, pp. 555–569 (2017).
19. Savenko, O., Lysenko, S., Nicheporuk, A., Savenko, B.: Approach for the Unknown Metamorphic Virus Detection. In: 9-th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems. Technology and Applications, pp. 453–458 (2017).
20. Pomorova, O., Savenko, O., Lysenko, S., Kryshchuk, A. and Nicheporuk, A.: A technique for detection of bots which are using polymorphic code. In: 21st International Conference, CN, Springer, Brunów, Poland, pp. 265-276 (2014)
21. Kondratenko, Y., Kondratenko, N.: Soft Computing Analytic Models for Increasing Efficiency of Fuzzy Information Processing in Decision Support Systems. Chapter in book: Decision Making: Processes, Behavioral Influences and Role in Business Management, R. Hudson (Ed.), Nova Science Publishers, New York, 41-78 (2015)
22. Savenko O.S Research of methods of antiviral diagnostics of computer networks / O.S Savenko, S.M Lysenko // Visnyk of Khmelnytsky National University. Technical sciences. - 2007. - № 2, v. 2. - P. 120–126.(in Ukrainian)
23. Savenko O.S., Payuk V.P., Savenko B.O, Kashtalyan A.S. Models of undocumented software bookmarks in local computer networks / Measuring and computing equipment in technological processes / 2019. - №2. - P.84-90.(in Ukrainian)
24. Савенко О.С. Модель процесу пошуку троянських програм в персональному комп'ютері / О.С. Савенко, С.М. Лисенко // Радіоелектронні і комп'ютерні системи. – 2008. – №7. – С.87-92.
25. 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.

References

1. DSTU 3396.2-97 Protection of information. Technical protection of information. Terms and definitions. State Committee of Ukraine, Kyiv (1997) [in Ukrainian]
2. Savelyeva, O. S., Krasnozhan, O. M., Lebedeva, O. U. (2014). Using the structural fault-tolerance index in project designing. Odes'kyi Politechnichnyi Universytet. Pratsi, 2, 130–135. doi: 10.15276/opu.2.44.2014.24.
3. S. Boranbayev, S. Altayev, A. Boranbayev. Applying the method of diverse redundancy in cloud based systems for increasing reliability, in Proceedings of the 12th International Conference on Information Technology: New Generations (ITNG 2015) (Las Vegas, Nevada, 2015), pp. 796–799.
4. Boranbayev A., Boranbayev S., Yersakhanov K., Nurusheva A., Taberkhan R. (2018) Methods of Ensuring the Reliability and Fault Tolerance of Information Systems. In: Latifi S. (eds) Information Technology - New Generations. Advances in Intelligent Systems and Computing, vol 738. Springer, Cham.
5. Chinnaiah, M., Niranjan, N. Fault tolerant software systems using software configurations for cloud computing. J Cloud Comp 7, 3 (2018). <https://doi.org/10.1186/s13677-018-0104-9>.
6. Zhu X, Wang J, Guo H, Zhu D, Yang LT, Liu L (2016) Fault-tolerant scheduling for real-time scientific workflows with elastic resource provisioning in virtualized clouds. IEEE Trans Parallel Distrib Syst 27(12):3501–3517. <https://doi.org/10.1109/TPDS.2016.2543731>.

7. Liu J, Zhou J, Buyya R (2015) Software rejuvenation based fault tolerance scheme for cloud applications In: 2015 IEEE 8th International Conference on Cloud Computing, 1115–1118, New York. <https://doi.org/10.1109/CLOUD.2015.164>.
8. Liu J, Wang S, Zhou A, Kumar SAP, Yang F, Buyya R (2016) Using Proactive Fault-Tolerance Approach to Enhance Cloud Service Reliability. IEEE Trans Cloud Comput PP(99):1–1. <http://dx.doi.org/10.1109/TCC.2016.2567392>.
9. Nicolo P (2013) A frame work for self-healing software systems In: IEEE 35th International Conference on Software Engineering (ICSE), 1397–1400. <https://doi.org/10.1109/ICSE.2013.6606726>.
10. Zhao W, Wenbing Z, Melliar-Smith PM, Moser LE (2010) Fault Tolerance Middleware for Cloud Computing In: 2010 IEEE 3rd International Conference on Cloud Computing, 67–74, Miami. <https://doi.org/10.1109/CLOUD.2010.26>.
11. Bala A, Chana I (2012) Fault tolerance- challenges, techniques and implementation in cloud computing, ISSN (Online): 16940814. IJCSI Int J Comput Sci 9(1). www.IJCSI.org.
12. Egwuotuha IP, Chen S, Levy D, Selic B (2012) A fault tolerance framework for high performance computing in cloud, Cluster, Cloud and Grid Computing (CCGrid) In: Proceedings of the 12th IEEE/ACM international symposium. 13-16 May, 709–710. <https://doi.org/10.1109/CCGrid.2012.80>.
13. S. Pitcher, "New DoD Approaches on the Cyber Survivability of Weapon Systems (25 March 2019)," 25 March 2019. [Online]. Available: <https://www.itea.org/wp-content/uploads/2019/03/Pitcher-Steve.pdf>.
14. D. J. Bodeau, R. D. Graubart, R. M. McQuaid and J. Woodill, "Cyber Resiliency Metrics and Scoring in Practice: Use Case Methodology and Examples (MTR 180449)," The MITRE Corporation, Bedford, MA, 2018.
15. D. Fitzpatrick, D. Bodeau, R. Graubart, R. McQuaid, C. Olin and J. Woodill, "(DRAFT) Cyber Resiliency Evaluation Framework for Weapon Systems: Foundational Principles and Their Potential Effects on Adversaries," The MITRE Corporation, Bedford, MA, 2019.
16. NIST, "Initial Public Draft of NIST SSP 800-160 Volume 2, Systems Security Engineering: Cyber Resiliency Considerations for the Engineering of Trustworthy Secure Systems," 21 March 2018. [Online]. Available: <https://csrc.nist.gov/CSRC/media/Publications/sp/800-160/vol-2/draft/documents/sp800-160-vol2-draft.pdf>.
17. Lysenko, S., Savenko, O., Kryshchuk, A., Kljots, Y. Botnet detection technique for corporate area network. In: Proc. of the 2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems, pp. 363-368 (2013).
18. Savenko, O., Lysenko, S., Nicheporuk, A., Savenko, B.: Metamorphic Viruses' Detection Technique Based on the Equivalent Functional Block Search. CEUR Workshop, Vol. 1844, pp. 555–569 (2017).
19. Savenko, O., Lysenko, S., Nicheporuk, A., Savenko, B.: Approach for the Unknown Metamorphic Virus Detection. In: 9-th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems. Technology and Applications, pp. 453–458 (2017).
20. Pomorova, O., Savenko, O., Lysenko, S., Kryshchuk, A. and Nicheporuk, A.: A technique for detection of bots which are using polymorphic code. In: 21st International Conference, CN, Springer, Brunów, Poland, pp. 265-276 (2014)
21. Kondratenko, Y., Kondratenko, N.: Soft Computing Analytic Models for Increasing Efficiency of Fuzzy Information Processing in Decision Support Systems. Chapter in book: Decision Making: Processes, Behavioral Influences and Role in Business Management, R. Hudson (Ed.), Nova Science Publishers, New York, 41-78 (2015)
22. Savenko O.S Research of methods of antiviral diagnostics of computer networks / O.S Savenko, S.M Lysenko // Visnyk of Khmelnytsky National University. Technical sciences. - 2007. - № 2, v. 2. - P. 120–126.(in Ukrainian)
23. Savenko O.S., Payuk V.P., Savenko B.O, Kashtalyan A.S. Models of undocumented software bookmarks in local computer networks / Measuring and computing equipment in technological processes / 2019. - №2. - P.84-90.(in Ukrainian)
24. Savenko O.S. Model of the process of searching for Trojan programs in a personal computer / O.S. Savenko, S.M. Lysenko // Radio electronic and computer systems. - 2008. - №7. - P.87-92. (in Ukrainian)
25. Savenko O.S., Klots Y.P., Mostoviy S.V. Research and analysis of process blocking in a computer system // Visnyk of Khmelnytsky National University. - 2007. - № 3, Volume 1.- P.248-251. (in Ukrainian)

Надійшла / Paper received: 11.03.2020

Надрукована / Paper Printed : 05.06.2020