DOI 10.31891/2307-5732-2024-331-4
УДК 004.49

РЕГІДА ПАВЛО
Хмельницький національний університет
https://orcid.org/0000-0002-6591-7069
email: pavlo.rehida@gmail.com
БАРМАК ОЛЕКСАНДЕР
Хмельницький національний університет
https://orcid.org/0000-0003-0739-9678
email: alexander.barmak@gmail.com
КАШТАЛЬЯН АНТОНІНА
Хмельницький національний університет
https://orcid.org/0000-0002-4925-9713
email: yantonina@ukr.net
МАНЗЮК ЕДУАРД
Хмельницький національний університет
https://orcid.org/0000-0002-7310-2126
email: edemasu@outlook.com

КОНЦЕПЦІЯ ЗАСТОСУВАННЯ РОЗПОДІЛЕНИХ СИСТЕМ ДЛЯ АНАЛІЗУ ПОЛІМОРФНИХ ВІРУСІВ

В даній роботі представлено модель застосування сучасних засобів для забезпечення захисту персональних даних користувачів від аномального впливу поліморфних вірусів, із залученням розподілених обчислень для ефективного виявлення загроз. Проблема виявлення зловмисного програмного забезпечення залишається актуальною протягом тривалого періоду, головним чином через значну кількість екземплярів зловмисного програмного забезпечення, що створюється сьогодні, і поширення програмного забезпечення та веб-служб, які зараз використовуються. Незважаючи на велику кількість засобів виявлення, щорічно фіксуються випадки витоку персональних даних з різних платформ, які використовуються щодня. Така ситуація виникає через те, що розробники зловмисного програмного забезпечення проводять дослідження способів виявлення зловмисного програмного забезпечення та намагаються включити в нього різні методи уникнення виявлення. У цій статті представлено аналіз сучасних методів організації розподілених систем, таких як кластерні обчислення, грід-обчислення, хмарні обчислення та периферійні обчислення, для різних цілей. Також представлено позитивні сторони застосування таких систем, а також виклики, з якими вони стикаються під час експлуатації. У статті також досліджуються ключові підходи, які можна реалізувати за допомогою розподілених обчислень для виявлення зловмисного програмного забезпечення, включаючи паралельну обробку для вивчення поведінки зловмисного програмного забезпечення, розробку ізольованих хмарних систем для тестування потенційного зловмисного програмного забезпечення та одночасне виконання того самого вірусу для отримання більш ефективних і точних результатів. Особлива увага приділяється концепції поліморфних вірусів і методам їх уникнення, які значно ускладнюють аналіз і, як наслідок, знижують шанси виявлення. Основна мета цієї статті — представити розподілений підхід, який використовує можливості високоефективного антивірусного програмного забезпечення для підвищення шансів виявлення нових шкідливих програм і загроз. Після проведеного аналізу вибирається тип системи, розглядаються основні модулі для центрального системного блоку та обчислювальних елементів. Центральний системний блок містить модуль інтерфейсу користувача, модуль для ефективного розподілу завдань (файлів і програм для аналізу) між обчислювальними елементами та модуль голосування для забезпечення вимог безпеки. Кожен обчислювальний елемент використовує спеціалізоване програмне забезпечення для зв'язку з системою та використовує одну з вибраних антивірусних програм для виконання аналізу. Запропонована концепція збільшує шанси на виявлення нових загроз завдяки поєднанню різних підходів до ідентифікації шкідливих програм із розподіленим підходом.

Ключові слова: розподілені системи, виявлення вірусів, поліморфні віруси, мультикомп'ютерні системи

REHIDA PAVLO, BARMAK OLEKSANDER, KASHTALIAN ANTONINA, MANZIUK EDUARD
Khmelnytskyi National University

THE CONCEPT OF DISTRIBUTED COMPUTING USAGE FOR THE ANALYSIS OF POLYMORPHIC VIRUSES

This work presents a model of the application of modern means to ensure the protection of personal data of users from the abnormal influence of polymorphic viruses, with the involvement of distributed computing for effective detection of threats. The challenge of detecting malware persists over an extended period, primarily due to the substantial number of malware instances being created today and the proliferation of software and web services in current use. Despite the large amount of detection tools, incidents of personal data leaks from various platforms used daily are recorded annually. This situation arises from malware developers applying research studies on how malware is detected and attempting to include various evasion techniques into malware. This paper presents an analysis of modern methods for organizing distributed systems, such as cluster computing, grid computing, cloud computing and edge computing, for various purposes. Also, the positive aspects of applying such systems are presented, as well as the challenges they face during their operation. The paper also explores key approaches achievable through distributed computing for malware detection, including parallel processing for studying malware behaviour, developing isolated cloud systems for testing potential malware, and simultaneous execution of the same virus to obtain more efficient and accurate results. Special attention is given to the concept of polymorphic viruses and their evasion techniques, which significantly complicate analysis and consequently lower the chances of detection. The primary objective of this article is to introduce a distributed approach that uses the capabilities of highly efficient antivirus software to enhance the chances of detecting new malwares and threats. Following the performed analysis, the system type is selected, and the main modules for the central system unit and computing elements are considered. Central system unit includes a user interface module, a module for efficiently distributing tasks (files and programs for analysis) among computing elements, and a voting module to ensure security requirements. Each computing element utilizes specialized software to communicate with the system and employs one of the selected antivirus programs to perform

analyses. The proposed concept increases the chances of detecting new threats by incorporating various approaches to malware identification with distributed approach.

Keywords: distributed computing, malware detection, polymorphic viruses, multicomputer systems.

Постановка проблеми

В сучасному світі використання інформаційних технологій (IT) дозволяє автоматизувати та пришвидшити рутинні повсякденні задачі за допомогою різноманітних програмних засобів та сервісів. Основною проблемою використання таких засобів є те, що вони працюють із персональним даними користувача. Користувачі можуть зберігати різні файли у хмарних сховищах, вносити персональні дані для створення облікового запису для доступу до веб-сервісу, використовувати програмні додатки, що обмінюються даними через Інтернет, тощо. Окрім усіх таких ризиків, ще визначають проблему постійного оновлення програмного забезпечення із метою впровадження нового функціоналу. Розширюючи функціонал певного застосунку, розробники можуть ненавмисно створити можливість до впливу вірусними засобами. Окрім того, різноманітні наукові лабораторії, що займаються пошуком нових загроз щорічно фіксують темпи зростання появи як нових вірусів [1], так і методів впливу на систему із метою отримання вигоди. Не зважаючи на те, що представлено велику кількість засобів виявлення [2], які ще визначають як антивірусі програмні додатки, щороку фіксуються масові витоки персональних даних навіть у великих компаніях. Саме тому, дослідження існуючих методів впливу на систему користувача та методів його виявлення є актуальною задачею. У цій роботі пропонується дослідити існуючі методи організації розподілених обчислень а також дослідити як ці системи можуть допомогти у питання виявлення нових загроз та зловмисного програмного забезпечення. Метою цієї роботи є: представити концепцію моделі розподілених систем, яка буде залучати сучасні підходи, для ефективного виявлення нових загроз.

Аналіз останніх досліджень

Розподілені обчислення використовуються для вирішення широкого спектру як наукових так і практичних завдань: моделювання складних фізичних процесів, обчислення ризиків, молекулярне моделювання, збір та обробка великих обсягів даних IoT систем, тренування моделей для задач штучного інтелекту, тощо. Для того, щоб правильно оцінити можливість залучення такої технології, проведемо аналіз сучасних методів їх організації.

Кластерні обчислювальні системи. Це одна із основних концепцій організації розподілених обчислень із метою залучити великі обчислювальні ресурси застосовуючи однорідні елементи. Усі залучені елементи взаємопов'язані і налаштовані для надання єдиного ресурсу для виконання однієї задачі. Елементи системи знаходяться поруч один із одним і використовують спеціальні провідні інтерфейси передачі даних, що дозволяє швидко передавати великі об'єми інформації (наприклад Infiniband [3]). Такі системи легко масштабуються за допомогою відносно простого залучення нових вузлів. Це дозволяє досягти високого рівня доступності системи: додаткові вузли дозволяють швидко замінити ті, що вийшли з ладу та продовжити виконувати обчислення. Усі ці аспекти визначають такі системи як провідні в якості опрацювання великих та ресурсоємних завдань. Планування та виконання правильності обчислень контролюється за допомогою центрального серверу [4]. Часто застосовуються віртуалізацію для ефективного розподілу ресурсів. Кластерні системи знаходять своє використання у багатьох сферах, наприклад моделювання наукових досліджень, серверів програмних застосунків, опрацювання великих даних [5], підтримка роботи баз даних.

Обчислювальні грід-системи. Для таких систем характерне поєднання обчислювальних елементів які можуть знаходитись віддалено один від одного. На відміну від кластерних систем, дуже часто застосовуються різні елементи з точки зору як програмного, так і апаратного забезпечення, тому їх ще визначають як гетерогенні. Для організації обміну повідомленнями між елементами найчастіше застосовуються традиційні технології підключення через мережу Інтернет. Такі системи легко масштабуються, і враховуючи їх гетерогенність, дозволяють гнучко підібрати необхідні ресурси для конкретної поставленої задачі, а також забезпечують високий рівень надійності. Керування системи характеризується як децентралізоване, тобто тут не визначений загальний керуючий модуль. Елементам системи притаманний ступінь певної автономії, що дозволяє визначати коли і та скільки обчислювальних ресурсів виділити на вирішення поставленої обчислювальної задачі перед системою. Через те, що система може залучати велику кількість елементів, потенційно такі системи можуть акумулювати велику кількість обчислювальної потужності [6]. Окремим випадком таких систем є волонтерські обчислення, які залучають елементи усіх бажаючих користувачів на добровільних засадах. Динамічність такої системи створює низку завдань, які потрібно вирішити для організації ефективних та безпечних обчислень: пошук ефективного способу розподілення завдань між елементами системи [7], організація захисту виконаних обчислень [8], оптимізація загальної продуктивності системи і її елементів та пошук алгоритмів ефективної передачі даних між її елементами.

Хмарні системи. Хмарні системи відносяться до інформаційно-технічних засобів, основною задачею яких є надання ресурсів, до яких входять: обчислювальні потужність, доступ до програмного забезпечення та способи збереження даних. Вони характеризуються в першу чергу тим, що можуть надавати свої послуги для користувачів в будь-якому місці за допомогою мережі Інтернет, без необхідності володіння потрібним апаратним обладнанням. З точки зору користувача, хмарні системи легко масштабуються відносно до потреб його задачі. Хмарні системи можуть розгортатись із різним типом доступу, і виділяють наступні: публічний (послуги надаються через глобальну мережу усім зареєстрованим користувачам), приватний (система розгортається на обладнанні компанії, та надає необхідні ресурси для її потреб, характеризується великою

захищеністю), спільна (деякі організації можуть об'єднуватись при організації хмарної системи для економії ресурсів) та гібридна (формується при об'єднанні декількох типів хмарних систем) [9]. Користувачам надаються послуги за допомогою трьох сценаріїв: IaaS (Infrastructure as a Service – надається доступ до контролю операційних систем, програм та мережевих конфігурацій), PaaS (Platform as a Service – включає в себе також бази даних і інструменти розробки та розгортання додатків), SaaS (Software as a Service – дозволяє переносити функціонал деяких додатків у браузер, що надає різні переваги) [10]. Для успішної організації функціонування такої системи, особливо визначають деякі основні завдання: забезпечення захисту системи [11] та залучення ефективних алгоритмів передачі великих наборів даних.

Граничні обчислення. Така парадигма обчислень досить нова і її поява завдячує активному використанню IoT пристроїв у сучасному світі. Ключовим аспектом цієї парадигми є переосмислення того, де зберігаються, де обробляються та як керуються дані для обчислень. Зростаюча кількість давачів збільшує відповідно і кількість інформації, що генерується для певної задачі. Через це буде збільшений час передачі інформації до центрального вузла для обробки, а також і час її обробки. В такій концепції пропонується залучати проміжні обчислювальні ресурси, які будуть проводити або часткову або повну обробку даних [12], перед тим як відправляти її на центральний вузол. Такий підхід надає перевагу у системах, які мають працювати в режимі реального часу, адже поточний результат можна отримати виконавши один запит звертаючись до керуючого елементу на відповідній ділянці. Тому, граничні обчислення знаходять застосування у комплексних системах керування IoT [13], сферах здоров'я та безпеки, автономних транспортних засобах, системах телекомунікацій та енергетичному секторі. До основних викликів для таких систем визначають: забезпечення стабільного зв'язку між елементами мережі, захисту передачі та обробки даних [14], розподіленому керуванню елементів системи [15].

Модель (концепція) для розподіленого виявлення аномальної поведінки програмного забезпечення

Розподілені обчислення посідають важливу роль у питаннях виявлення зловмисного програмного забезпечення. Основними задачами у виявленні вірусів визначають ідентифікацію та забезпечення ізольованого середовища їх виконання. Розподілені системи можуть використовуватись у наступних задачах:

- *Паралельне опрацювання*: проведення аналізу потенційного вірусу одночасно на декількох елементах системи, що дозволяє прискорювати швидкість та ефективність алгоритмів виявлення.
- *Обмін даними*: інформація про виявлену аномальну поведінку передається до усіх елементів, що значно збільшить захист інших елементів.
- *Хмарне виявлення загроз*: хмарні системи можуть бути направлені на ізольований аналіз загроз, що дозволить убезпечити елементи системи.
- *Аналіз даних поведінки*: паралельні обчислення можуть застосовуватись і при дослідженні поведінки вірусу який потрапив у систему з метою виявлення його стратегій виконання аномальних дій [16].

Одним із видів вірусів, що часто застосовуються в теперішній час є поліморфні віруси. [17-20]. Такі віруси характеризуються тим, що можуть змінювати свою структуру. Це дозволяє їм уникати виявлення за допомогою стандартних методів аналізу, що виконується антивірусними програмами. Така ситуація виникає через те, що антивірусні засоби базуються на пошуку відповідних сигнатур у власних базах даних. Також, поліморфним вірусам притаманне приховання власного коду через шифрування. Поліморфні віруси застосовують велику кількість захисних технік від виявлення: зміна порядку виконання інструкцій виконання, перетворення інструкцій а також додавання нових беззмістовних інструкцій. Усі ці заходи значно ускладнюють процес виявлення.

Для того, щоб покращити результати виявлення пропонується застосувати розподілену систему яка буде залучати різні комерційні антивірусні засоби. Відомі рішення (Avira Internet Security, Bitdefender Internet Security, F-Secure Total, Avast Antivirus, McAfee Total Protection та інші) застосовують власні розроблені методи виявлення вірусів і мають власні бази даних сигнатур. Застосування усіх рішень одночасно, дозволить із великою імовірністю виявити потенційно зловмисне програмне забезпечення.

У запропонованій системі пропонується використати концепцію грід обчислень із розробкою єдиного вхідного інтерфейсу для користувачів, що хочуть перевірити підозрілу програму або файл. В якості елементів розподіленої системи будуть виступати робочі станції, що будуть мати проміжне програмне забезпечення із центральним компонентом системи. Його мета завантажити запропонований файл системою та виконати його, та очікувати на результат опрацювання антивірусним засобом.

На рис. 1 представлено концепцію для розподіленого виявлення зловмисного програмного забезпечення.

Для забезпечення ефективного виконання поставлених цілей, в моделі передбачається використання декількох блоків, а саме: модуль планування завдань для елементів системи, модуль голосування, комунікаційний блок. Розглянемо основні завдання для кожного такого блоку:

Модуль планування розподілу завдань. Такий модуль повинний враховувати наявну кількість елементів в системі та враховувати швидкість опрацювання аналізу зважаючи на характеристики файлу чи програми. Так як концепція грід-обчислень передбачає використання елементів які не належать системі і їхня участь є не постійною, даний модуль має ефективно розподіляти завдання між учасниками враховуючи стан системи в реальному часі, а також бути готовою адаптуватись до відключення деяких елементів та перебудувати план виконання завдань.

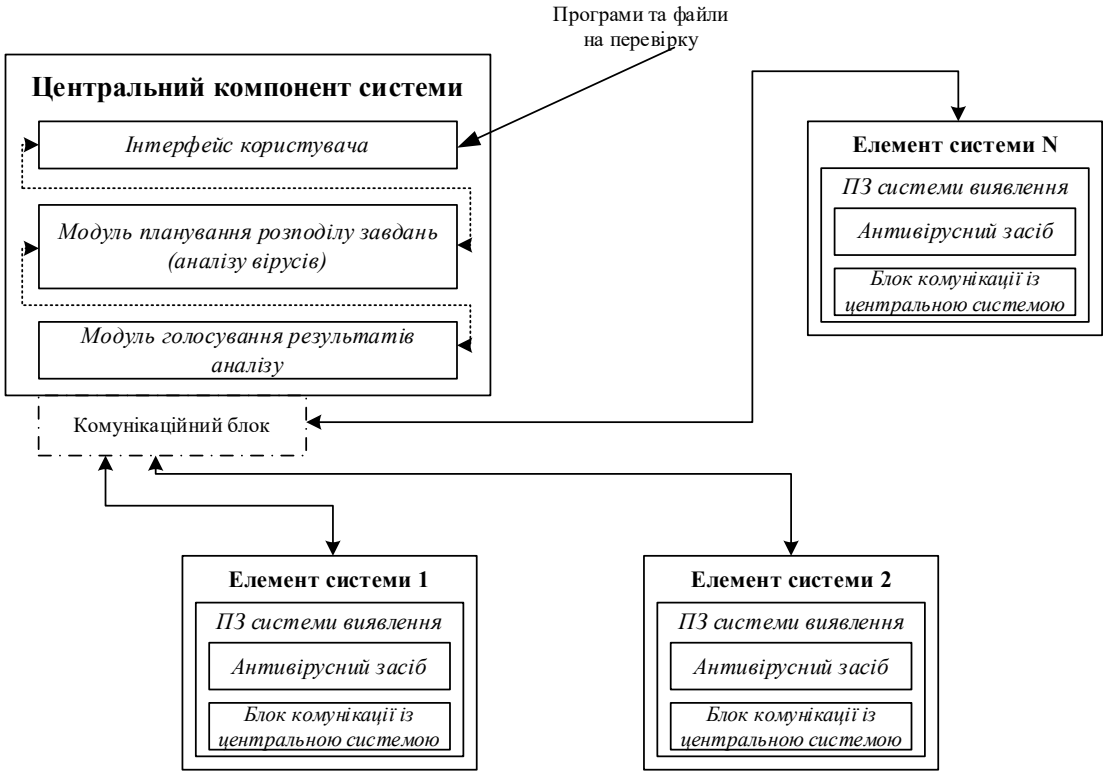

Рис. 1 – Концепція розподіленої системи виявлення поліморфних вірусів

Модуль голосування. Голосування допоможе вирішити низку викликів в такій системі, такі як прийняття рішення щодо аномальності файлу чи програми що тестуються. Тут отримані результати від елементів системи будуть опрацьовані і прийняте рішення про наступний крок. На цьому етапі можливе сповіщення користувача про результати аналізу (у разі якщо результат одностайний) або проведення повторних перевірок.

Комунікаційний блок. Відповідає за набір протоколів та алгоритмів шифрування при передачі даних від центрального елемента до обчислювальних елементів. Такі заходи дозволять передати файл чи програму на аналіз в оригінальному вигляді, бо в протилежному випадку це може негативно вплинути на результат голосування.

Таким чином, запропонована концепція з архітектурою на рис. 1 є основою для створення мультикомп'ютерних систем, які можуть виконувати функції аналізу поліморфних вірусів.

Концепція експерименту та експериментальні дослідження

Результатом реалізації запропонованої концепції будуть мультикомп'ютерні системи. Для забезпечення їх функціонування потрібно розробити проміжне програмне забезпечення. В ньому будуть реалізовані, крім функцій підтримки комунікації та функціонування безпосередньо самої системи, функції для здійснення аналізу програмного коду на наявність поліморфізму. Така мультикомп'ютерна система може бути використана для проведення експериментальних досліджень.

Важливими характеристиками, які потрібно досліджувати при проведенні експерименту саме для систем, є їх стійкість та рівновага. Оскільки, мультикомп'ютерні системи охоплюють комп'ютерні станції корпоративної мережі, а зв'язок між ними формується за рахунок проміжного програмного забезпечення, що створює мультикомп'ютерну систему, тобто розподілену систему, то стійкість таких систем та їх рівновага є важливими характеристиками і потребують дослідження [21]. Введемо три цільові функції $\mathfrak{F}_i^{\mathfrak{E}}$ ($i = 1,2,3$), які будуть характеризувати відповідно стійкість, рівновагу та результат виконання завдання спеціалізованим функціоналом системи. Результат виконання кожної з цих функцій буде відображено в проміжок [0;1]. Середньоарифметичні значення інтегрованого показника $\mathfrak{t}_{\mathfrak{F}_i^{\mathfrak{E}}}^{\mathfrak{E}}$ цих трьох функцій будуть встановлювати три коефіцієнти для характеристики системи [21]:

$$\mathfrak{t}_{\mathfrak{F}_i^{\mathfrak{E}}}^{\mathfrak{E}} = \frac{\sum_{j=1}^{n_{\mathfrak{F}_i^{\mathfrak{E}}}^{\mathfrak{E}}} \mathfrak{F}_i^{\mathfrak{E}}(t_j)}{n_{\mathfrak{F}_i^{\mathfrak{E}}}^{\mathfrak{E}}}, \quad (1)$$

де t_j – час від початку функціонування системи; j – індекс для часу, що відображає j – тий момент фіксування часу в системі; $j = 1,2,\dots,n_{\mathfrak{F}_i^{\mathfrak{E}}}^{\mathfrak{E}}$; $n_{\mathfrak{F}_i^{\mathfrak{E}}}^{\mathfrak{E}}$ – кількість фіксувань часу, які було здійснено системою; $\mathfrak{F}_i^{\mathfrak{E}}(t_j)$ ($i = 1,2,3$) - функції, які характеризують стійкість, рівновагу та результат виконання завдання спеціалізованим функціоналом системи; $\mathfrak{t}_{\mathfrak{F}_i^{\mathfrak{E}}}^{\mathfrak{E}}$ ($i = 1,2,3$) - коефіцієнти, які характеризують рівні стійкості, рівноваги та результату виконання завдання спеціалізованим функціоналом системи [21].

При плануванні експерименту з розподіленими системами потрібно враховувати стан компонентів таких систем. Результати значень трьох коефіцієнтів та проміжних величин такі: $\mathfrak{k}_{\mathfrak{F}_1}^{\mathfrak{E}} = 0{,}791$; $\mathfrak{k}_{\mathfrak{F}_2}^{\mathfrak{E}} = 0{,}687$; $\mathfrak{k}_{\mathfrak{F}_3}^{\mathfrak{E}} = 0{,}93521$; $\mathfrak{n}_1 = 100$; $\mathfrak{n}_2 = 120$; $\mathfrak{n}_{\mathfrak{F}_i}^{\mathfrak{E}} = 50$. Отримані результати є достатніми для застосування запропонованої концепції.

Висновки

У даній статті проведено аналіз існуючих методів організації розподілених систем, та задач що ними вирішуються. Проведений аналіз дозволив визначитись із типом системи для вирішення поставленої задачі. Запропоновано залучити ефективні методи виявлення розроблені компаніями, що спеціалізуються на виявленні зловмисного програмного забезпечення. Даний підхід базується на побудові моделі, що буде використовувати обчислювальні елементи як незалежні компоненти системи для аналізу. Передбачається, що система буде використовувати єдиний інтерфейс користувача, що дозволить відправляти на перевірку файл чи програму. Для ефективного розподілу задач, у системі визначений відповідний модуль, що аналізує поточний стан підключених обчислювальних елементів, враховуючи їх завантаженість та успішність виконання попередніх завдань. Окремо запропоновано реалізувати модуль голосування, який дозволить вирішувати успішність виконаного аналізу та сповіщати про його завершення користувачу або формувати запит до модуля планування для проведення повторного аналізу. Запропонований підхід акумулює успішні рішення з питань виявлення загроз що дозволить підвищити шанс виявлення нових. Напрямами подальших удосконалень запропонованої концепції є перенесення частини функцій в певні визначені компоненти розподіленої системи для покращення ефективності в цілому.

Література

1. N. Dutta. Introduction to malware analysis / N. Dutta, N. Jadav, S. Tanwar, H.K.D. Sarma, E. Pricop // Cyber Security: Issues and Current Trends, 2022 – P. 129-141.
2. A. Abusitta. Malware classification and composition analysis: A survey of recent developments. / A. Abusitta, M. Q. Li, B.C.M. Fung // Journal of Information Security and Applications, 59(2): 102828, 2021.
3. QM9700/QM9790 1U NDR 400Gb/s InfiniBand Switch Systems User Manual – Nvidia [Електронний ресурс]. – Режим доступу: https://www.sysgen.de/media/pdf/6a/98/91/QM9700_QM9790_User_Manual.pdf
4. Cluster Computing: Distributed Computing Architecture / [Mark Heynen] – CreateSpace Independent Publishing Platform, 2016 – 218p.
5. Fatima Es-Sabery. Big Data Solutions Proposed for Cluster Computing Systems Challenges: A survey / Fatima Es-Sabery, Abdellatif Hair // NISS '20: Proceedings of the 3rd International Conference on Networking, Information Systems & Security, March 2020 – P. 1-7.
6. Nane Kratzke. Volunteer Down: How COVID-19 Created the Largest Idling Supercomputer on Earth / Nane Kratzke // Future Internet, 12(6), June 2020 – P.98 – 118.
7. A Muhammed. An Estimation-Based Dynamic Load Balancing Algorithm for Efficient Load Distribution and Balancing in Heterogeneous Grid Computing Environment / A Muhammed, A Abdullah, M Hussin, S Hasan, MA Mohamed // Journal of Grid Computing, 21(1), 28 January 2023 – P.7.
8. Bradley Wood. Remote Method Delegation: a Platform for Grid Computing/ Bradley Wood, Brock Watling, Zachary Winn, Daniel Messiha, Qusay H. Mahmoud, Akramul Azim // Journal of Grid Computing 18, 28 July 2020 – P.711-725
9. Mohammad I.M. CLOUD COMPUTING-TECHNOLOGIES. / Mohammad I.M., Shahid H.W., Abreen R. // International Journal of Advanced Research in Computer Science, 9(2), 2018 – P.6.
10. Blesson V. Next generation cloud computing: New trends and research directions. / Blesson V., Rajkumar B. // Future Generation Computer Systems, 79, 2018 – P.849-861.
11. Gai K.. Blockchain Meets Cloud Computing: A Survey / Gai K., Guo J., Zhu L. Yu S. // IEEE Communications Surveys & Tutorials, 22(3), 2020 – P. 2009-2030.
12. W. Yu. A Survey on the Edge Computing for the Internet of Things / W. Yu, F. Liang, X. He, W.G. Hatcher, C. Lu, J. Lin, X. Yang // IEEE access 29, 2017 – P. 6900-6919.
13. T. Qiu. Edge Computing in Industrial Internet of Things: Architecture, Advances and Challenges. / T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman, D.O. Wu // IEEE Communications Surveys & Tutorials, 22(4), 2020 – P. 2462-2488.
14. K. Sha. A survey of edge computing-based designs for IoT security / K. Sha, T.A. Yang, W. Wei, S. Davari. // Digital Communications and Networks. 6(2). 2020 – P. 195-202.
15. X. Xia. Cost-Effective App Data Distribution in Edge Computing / X. Xia, F. Chen, Q. He, J. C. Grundy, M. Abdelrazek M, H. Jin // IEEE Transactions on Parallel and Distributed Systems, 32(1), 2020 – P. 31-44.
16. M. Kumar. Scalable malware detection system using big data and distributed machine learning approach. / M. Kumar // Soft Computing 26, 2022 – P. 3987-4003.
17. R. Badhwar. Polymorphic and Metamorphic Malware. / R. Badhwar // The CISO's Next Frontier: AI, Post-Quantum Cryptography and Advanced Security Paradigms, 2021 – P. 279-285.
18. S. K. Sahay. Evolution of malware and its detection techniques. / S. K. Sahay, A. Sharma, H.

Rathore // Information and Communication Technology for Sustainable Development: Proceedings of ICT4SD, 2018 – P. 139-150.

19. G. Markowsky, O. Savenko, S. Lysenko, A. Nicheporuk, The Technique for Metamorphic Viruses' Detection Based on its Obfuscation Features Analysis, CEUR Workshop Proceedings, Vol. 2104, 2018, pp. 680-687.

20. Pomorova, O., Savenko, O., Lysenko, S., Kryshchuk, A., Nicheporuk, A. (2014). A Technique for Detection of Bots Which Are Using Polymorphic Code. In: Kwiecień, A., Gaj, P., Stera, P. (eds) Computer Networks. CN 2014. Communications in Computer and Information Science, vol 431. Springer, Cham. https://doi.org/10.1007/978-3-319-07941-7_27

21. Каштальян А.С. Принцип синтезу мультикомп'ютерних систем з комбінованих антивірусних приманок і пасток та контролеру прийняття рішень для виявлення зловмисного програмного забезпечення та комп'ютерних атак. *Вісник Хмельницького національного університету. Технічні науки*. 2023. № 327(6). С. 386-393. https://www.doi.org/10.31891/2307-5732-2023-329-6

References

1. N. Dutta. Introduction to malware analysis / N. Dutta, N. Jadav, S. Tanwar, H.K.D. Sarma, E. Pricop // Cyber Security: Issues and Current Trends, 2022 – P. 129-141.

A. Abusitta. Malware classification and composition analysis: A survey of recent developments. / A. Abusitta, M. Q. Li, B.C.M. Fung // Journal of Information Security and Applications, 59(2): 102828, 2021.

2. QM9700/QM9790 1U NDR 400Gb/s InfiniBand Switch Systems User Manual – Nvidia [Elektronnyi resurs]. – Rezhym dostupu: https://www.sysgen.de/media/pdf/6a/98/91/QM9700_QM9790_User_Manual.pdf

3. Cluster Computing: Distributed Computing Architecture / [Mark Heynen] – CreateSpace Independent Publishing Platform, 2016 – 218p.

4. Fatima Es-Sabery. Big Data Solutions Proposed for Cluster Computing Systems Challenges: A survey / Fatima Es-Sabery, Abdellatif Hair // NISS '20: Proceedings of the 3rd International Conference on Networking, Information Systems & Security, March 2020 – P. 1-7.

5. Nane Kratzke. Volunteer Down: How COVID-19 Created the Largest Idling Supercomputer on Earth / Nane Kratzke // Future Internet, 12(6), June 2020 – P.98 – 118.

6. A Muhammed. An Estimation-Based Dynamic Load Balancing Algorithm for Efficient Load Distribution and Balancing in Heterogeneous Grid Computing Environment / A Muhammed, A Abdullah, M Hussin, S Hasan, MA Mohamed // Journal of Grid Computing, 21(1), 28 January 2023 – P.7.

7. Bradley Wood. Remote Method Delegation: a Platform for Grid Computing/ Bradley Wood, Brock Watling, Zachary Winn, Daniel Messiha, Qusay H. Mahmoud, Akramul Azim // Journal of Grid Computing 18, 28 July 2020 – P.711-725

8. Mohammad I.M. CLOUD COMPUTING-TECHNOLOGIES. / Mohammad I.M., Shahid H.W., Abreen R. // International Journal of Advanced Research in Computer Science, 9(2), 2018 – P.6.

9. Blesson V. Next generation cloud computing: New trends and research directions. / Blesson V., Rajkumar B. // Future Generation Computer Systems, 79, 2018 – P.849-861.

10. Gai K.. Blockchain Meets Cloud Computing: A Survey / Gai K., Guo J., Zhu L. Yu S. // IEEE Communications Surveys & Tutorials, 22(3), 2020 – P. 2009-2030.

11. W. Yu. A Survey on the Edge Computing for the Internet of Things / W. Yu, F. Liang, X. He, W.G. Hatcher, C. Lu, J. Lin, X. Yang // IEEE access 29, 2017 – P. 6900-6919.

12. T. Qiu. Edge Computing in Industrial Internet of Things: Architecture, Advances and Challenges. / T. Qiu, J. Chi, X. Zhou, Z. Ning, M. Atiquzzaman, D.O. Wu // IEEE Communications Surveys & Tutorials, 22(4), 2020 – P. 2462-2488.

13. K. Sha. A survey of edge computing-based designs for IoT security / K. Sha, T.A. Yang, W. Wei, S. Davari. // Digital Communications and Networks. 6(2). 2020 – P. 195-202.

14. X. Xia. Cost-Effective App Data Distribution in Edge Computing / X. Xia, F. Chen, Q. He, J. C. Grundy, M. Abdelrazek M, H. Jin // IEEE Transactions on Parallel and Distributed Systems, 32(1), 2020 – P. 31-44.

15. M. Kumar. Scalable malware detection system using big data and distributed machine learning approach. / M. Kumar // Soft Computing 26, 2022 – P. 3987-4003.

16. R. Badhwar. Polymorphic and Metamorphic Malware. / R. Badhwar // The CISO's Next Frontier: AI, Post-Quantum Cryptography and Advanced Security Paradigms, 2021 – P. 279-285.

17. S. K. Sahay. Evolution of malware and its detection techniques. / S. K. Sahay, A. Sharma, H. Rathore // Information and Communication Technology for Sustainable Development: Proceedings of ICT4SD, 2018 – P. 139-150.

18. G. Markowsky, O. Savenko, S. Lysenko, A. Nicheporuk, The Technique for Metamorphic Viruses' Detection Based on its Obfuscation Features Analysis, CEUR Workshop Proceedings, Vol. 2104, 2018, pp. 680-687.

19. Pomorova, O., Savenko, O., Lysenko, S., Kryshchuk, A., Nicheporuk, A. (2014). A Technique for Detection of Bots Which Are Using Polymorphic Code. In: Kwiecień, A., Gaj, P., Stera, P. (eds) Computer Networks. CN 2014. Communications in Computer and Information Science, vol 431. Springer, Cham. https://doi.org/10.1007/978-3-319-07941-7_27

20. Kashtalyan A.S. Synthesis principle of multi-computer systems from combined anti-virus decoys and traps and decision controller for detection of malware and computer attacks. Bulletin of the Khmelnytskyi National University. Technical sciences.2023. № 327(6). С. 386-393. https://www.doi.org/10.31891/2307-5732-2023-329-6