

КИЇВСЬКИЙ СТОЛИЧНИЙ УНІВЕРСИТЕТ ІМЕНІ БОРИСА ГРІНЧЕНКА

Кваліфікаційна наукова
праця на правах рукопису

ЧЕРНІГІВСЬКИЙ ІВАН АНДРІЙОВИЧ

УДК 004.056.5:004.89:005.8

ДИСЕРТАЦІЯ

**МЕТОД ЗАХИСТУ ВУЗЛІВ ІНФОКОМУНІКАЦІЙНОЇ МЕРЕЖІ
ВІД КОМП'ЮТЕРНИХ ВІРУСІВ НА ОСНОВІ НЕЙРОМЕРЕЖЕВИХ
МОДЕЛЕЙ**

Спеціальність 125 Кібербезпека
Галузь знань 12 Інформаційні технології

Подається на здобуття ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей,
результатів і текстів інших авторів мають посилання на відповідне джерело.

_____ І. А. Чернігівський

Науковий керівник:
Крючкова Лариса Петрівна
доктор технічних наук, професор

Київ – 2026

АНОТАЦІЯ

Чернігівський І.А. Метод захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів на основі нейромережових моделей. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття ступеня доктора філософії за спеціальністю 125 Кібербезпека. – Київський столичний університет імені Бориса Грінченка, Київ, 2026.

Дисертацію присвячено розв'язанню **науково-практичного завдання** підвищення ефективності протидії поширенню комп'ютерних вірусів в інфокомунікаційній мережі (ІКМ).

Згідно з даними AV-ATLAS, у 2024 році загальна кількість комп'ютерних вірусів перевищила 1,5 млрд. Трапляються випадки, коли комп'ютерні віруси тривалий час перебувають у вузлах мережі, доки їх буде виявлено. Зазначена ситуація створює умови до часткової або повної втрати інфокомунікаційною мережею властивостей конфіденційності, цілісності та доступності інформації, що циркулює в ІКМ.

Дані міжнародних досліджень вказують на те, що щороку фіксується значне збільшення кількості інцидентів, пов'язаних з атаками Ransomware. Програми-вимагачі беруть участь майже у 70% порушень, пов'язаних зі шкідливим програмним забезпеченням (ШПЗ), та приблизно у 24% усіх порушень. Загалом загальна світова вартість програм-вимагачів у 2024 році перевищила 30 мільярдів доларів. Потужним інструментом для кіберзлочинців також стає технологія Deepfake, що підтверджується зростанням кількості онлайн-діпфейків на 550% з 2019 по 2023 рік. За даними DeepMedia, лише у 2023 році в соціальних мережах по всьому світу було поширено близько 500 000 відео та голосових діпфейків.

З початку 2025 року спостерігається адаптація та автоматизація наборів інструментів для кібератак із залученням штучного інтелекту (ШІ), у тому числі і сканерів вразливостей, оскільки ШІ може знаходити вразливості, про які ще не відомо компаніям-розробникам програмного забезпечення (zero day).

Вважається, що новим вектором атак із залученням ШІ стануть самі ШІ моделі, які використовуються на об'єктах інформаційної діяльності, у тому числі і у захисних рішеннях.

Результати виконаного аналізу дозволяють стверджувати, що традиційні рішення захисту вузлів ІКМ можуть пропускати комп'ютерні віруси внаслідок недосконалості механізмів їх виявлення, що особливо критично під час багаторівневих кібератак.

Розвиток і постійне вдосконалення ШПЗ та кібератак вимагає проведення досліджень щодо синтезу нового методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів.

Об'єктом дослідження є процеси виявлення та блокування комп'ютерних вірусів у вузлах інфокомунікаційної мережі.

Предметом дослідження є методи і моделі виявлення комп'ютерних вірусів та захисту вузлів інфокомунікаційної мережі.

Метою дисертаційної роботи є синтез методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, здатного забезпечити підвищення ефективності протидії поширенню комп'ютерних вірусів в інфокомунікаційній мережі.

Згідно з поставленою метою, у процесі теоретичних і експериментальних досліджень одержано наступні нові наукові результати:

1. Вперше запропоновано метод захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, побудований за принципом послідовного циклічного звернення до операторів ідентифікації, прийняття рішення та реалізації керуючих дій, у якому визначення стану вузла ІКМ здійснюється на основі вивантаження мінімально необхідної кількості цифрових слідів та їх аналізу нейромережевими моделями, що забезпечує економію часу і ресурсів на виявлення комп'ютерних вірусів та протидію їх поширенню в інфокомунікаційній мережі.

2. Вперше запропоновано і реалізовано використання цифрових слідів у якості основної ідентифікаційної ознаки при оцінці зараженості вузлів ІКМ,

що забезпечує виявлення ШПЗ, пропущених традиційними рішеннями захисту кінцевих точок, та надає можливість вдосконалення наявного ешелонованого захисту ІКМ.

3. Вперше запропоновано і реалізовано реляційну модель у вигляді таблиці артефактів, яка шляхом фільтрації дозволяє оптимізувати кількість і розмір цифрових слідів за критерієм «необхідна достатність – мінімум часу», що забезпечує економію часу і ресурсів на виявлення наявності комп'ютерних вірусів у вузлах ІКМ.

4. Вперше запропоновано і реалізовано застосування нейромережових моделей для аналізу вивантажених цифрових слідів, що забезпечує суттєве підвищення швидкості реагування на виникаючі інциденти в ІКМ з великою кількістю вузлів.

5. Набув подальшого розвитку метод вивантаження цифрових артефактів в умовах обмеженості ресурсів, який за рахунок оптимізації кількості і розміру цифрових слідів та їх ранжування забезпечує можливість формування уявлення про стан зараженості конкретного вузла на сервері ІКМ навіть у випадку переривання з'єднання під час передачі даних.

Дисертаційна робота складається з вступу, чотирьох розділів, у яких логічно, на високому науково-технічному рівні викладено рішення поставлених задач дослідження.

У вступі обґрунтовано актуальність та важливість теми дисертаційного дослідження, визначено мету і задачі дослідження, зазначено наукову і практичну цінність отриманих результатів та особистий внесок автора.

В першому розділі представлено результати аналізу сучасного стану захищеності інформації у вузлах інфокомунікаційної мережі. Розглянуто можливі загрози для інформації, що циркулює в інфокомунікаційних мережах. Встановлено, що сучасні кібератаки реалізуються складними багаторівневими механізмами проникнення і закріплення в мережах та часто функціонують як розширена постійна загроза (Advanced Persistent Threat, АРТ). Особливо небезпечними є атаки ШПЗ типу Ransomware із залученням ШІ.

Проведено аналітичний огляд існуючих методів захисту та виявлення шкідливих впливів на вузли інфокомунікаційної мережі, результати якого дозволяють стверджувати, що традиційні рішення захисту вузлів ІКМ можуть пропускати комп'ютерні віруси внаслідок недосконалості механізмів їх виявлення, що особливо критично під час багаторівневих кібератак.

У прикінцевій частині першого розділу, враховуючи стрімкий розвиток і постійне вдосконалення ШПЗ та кібератак, сформовано цілі та завдання дослідження.

В другому розділі здійснено синтез методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, особливістю якого є використання цифрових слідів у якості основної ідентифікаційної ознаки визначення стану вузла ІКМ та їх аналіз нейромережевими моделями, що дозволяє забезпечити економію часу і ресурсів на виявлення комп'ютерних вірусів та протидії їх поширенню в інфокомунікаційній мережі. Визначено достатні для ідентифікації стану ІКМ словник ознак і таблиці артефактів для прийняття рішень про зараженість конкретного вузла. Рішення про стан конкретного вузла ІКМ приймається за критерієм "ідеального спостерігача", який мінімізує можливість сумарної помилки прийняття рішення в циклі управління.

Обґрунтовано доцільність застосування нейромережових моделей для аналізу цифрових слідів, як основи для автоматичного прийняття рішення про зараженість конкретного вузла ІКМ в системі управління.

Визначено послідовність дій в циклі управління за запропонованим методом захисту вузлів ІКМ. Запропоновано шляхи мінімізації середнього часу на аналіз активних вузлів розподіленої ІКМ. Проаналізовано вплив ШПЗ на вузли ІКМ у віртуальному середовищі зі зміненою конфігурацією та ідентифіковано цифрові сліди, що залишились після ШПЗ. Запропоновано метод вивантаження цифрових слідів в умовах обмеженості ресурсів. Виконано порівняння програм для проведення Forensic Triage. Визначено, що не всі програми вивантажують необхідні сліди та в деяких випадках надають

надлишкову інформацію. Запропоновано і реалізовано реляційну модель у вигляді таблиці артефактів, яка шляхом фільтрації дозволяє оптимізувати кількість і розмір цифрових слідів між наявними артефактами у вузлі і достатніми для ідентифікації стану, що забезпечує економію часу і ресурсів для виявлення наявності комп'ютерних вірусів у вузлах ІКМ. Визначено і проаналізовано релевантні цифрові сліди для оцінки вузла ІКМ в умовах обмеженості ресурсів.

Визначено перелік типових кейсів та здійснено ранжування цифрових слідів у відповідності до їх значущості для конкретного кейсу. Отримані у другому розділі результати дозволяють стверджувати, що використання запропонованого методу захисту інформації у вузлах ІКМ надасть суттєву економію часу і ресурсів на визначення зараженості конкретного вузла ІКМ.

В третьому розділі проведено тестування нейромережевих моделей для вирішення задачі аналізу вивантажених цифрових слідів, що дозволяє виявляти заражені вузли ІКМ. Розглянуто технології і моделі штучного інтелекту. Визначено критерії оцінки та порівняння характеристик нейромережевих моделей. Досліджено 135 моделей формату GGUF та відібрано 43 найрелевантніші нейромережеві моделі для аналізу цифрових слідів. Запропоновано принцип формування навчальних даних для дотренування нейромережевих моделей для більш якісної оцінки вузлів ІКМ на предмет зараженості.

В четвертому розділі викладено результати експериментальної перевірки ефективності запропонованого методу захисту вузлів ІКМ. Представлено набір тестів і програм для оцінки ефективності антивірусних рішень для кінцевих точок (NGAV, EDR), що забезпечують всебічний розгляд поведінки антивірусного рішення на різні загрози. Обґрунтовано критерії, за якими здійснюється фіксація очікуваного спрацювання антивірусу на ШПЗ сімейства Ransomware, а також запропоновано скрипт, що імітує поведінку відомих Ransomware для тестування евристичного модулю антивірусу. Запропоновано шляхи оптимізації методу захисту вузлів ІКМ.

Підтверджено ефективність використання реляційної моделі для ідентифікації стану вузлів ІКМ, оскільки кількість артефактів суттєво скоротилася (у деяких випадках навіть більш ніж на 97%), що економить час на подальші дослідження.

Підтверджено ефективність методу вивантаження цифрових артефактів в конкретному вузлі ІКМ. Реалізований метод повністю відпрацьовує за 2хв 40сек (що швидше на 81.1% ніж Cedarpelta) та має розмір артефактів 89.4МБ (що менше Cedarpelta на 48% та КАРЕ на 88%) при однаковій релевантності даних, при цьому за перші 1хв 28сек сформовано 67КБ артефактів, яких достатньо для первинного перегляду. Це підтверджує можливість методу надавати релевантні відсортовані дані і займати небагато пам'яті, що, у свою чергу, забезпечує економію часу під час передачі та аналізу цифрових слідів.

Здійснено перевірку ефективності методу виявлення індикаторів компрометації на базі відібраних нейромережових моделей. Визначено, що заздалегідь відібрані моделі можуть надавати хибну оцінку. З наявних 16 моделей, що були попередньо відібрані, тільки 2 змогли правильно і без помилок відповісти на два питання: gemma-3-4b-it-Q4_K_M, qwq-32b-preview-senecallmv1.2-q4_k_m. Підвищення точності та релевантності відповідей досягається шляхом дотренування моделі на запропонованих навчальних даних.

Здійснено перевірку ефективності методу в операторі прийняття рішення. Реалізований метод захисту вузлів ІКМ повністю відпрацьовує за 340сек або 5хв 40сек, що складає близько 6хв на один вузол ІКМ. Це швидше на 90% порівняно із повним ручним аналізом та ізоляцією зараженого вузла ІКМ, що зазвичай триває більше години.

Дисертація виконувалась в Київському столичному університеті імені Бориса Грінченка.

Результати наукових досліджень використано на кафедрі інформаційної та кібернетичної безпеки імені професора Володимира Бурячка Київського столичного університету імені Бориса Грінченка в рамках науково-дослідної

роботи: "Методи та моделі забезпечення кібербезпеки інформаційних систем переробки інформації та функціональної безпеки програмно-технічних комплексів управління критичної інфраструктури" (реєстраційний номер 0122U200483, КСУБГ, м. Київ).

Також результати досліджень прийняті до впровадження в діяльність Київського столичного університету імені Бориса Грінченка (акт від 21.04.2026 року) та ТОВ «АШАН Україна Гіпермаркет» (акт від 10.03.2026).

Ключові слова: інфокомунікаційні мережі, кіберзагрози, кібератаки, ШПЗ, комп'ютерні віруси, Ransomware, кібербезпека, захист інформації, методи захисту, цифрові сліди, Forensic Triage, нейромережеві моделі, штучний інтелект, LLM, навчання моделей ШІ, ідентифікаційні ознаки, прийняття рішення, реалізація керуючих дій

ANNOTATION

Chernihivskiy I.A. The protection method against computer viruses for the infocommunication network nodes on the neural network models base. – Qualification scientific work in the form of a manuscript.

Dissertation for the degree of Doctor of Philosophy in the specialty 125 Cybersecurity. – Borys Grinchenko Kyiv Metropolitan University, Kyiv, 2026.

The dissertation is dedicated to the scientific and practical task solution on increasing the effectiveness of counteracts the infocommunication network (ICN) computer viruses' spread.

According to AV-ATLAS, in 2024 the total number of computer viruses exceeded 1.5 billion variants. There are cases when computer viruses remain in network nodes for a long time until they are detected. It provides a background for the partial or complete loss by the infocommunication network the confidentiality properties, integrity and availability of the circulating in the ICN information.

The international researches data indicate the yearly significant increase in the number of related to Ransomware attacks incidents. Ransomware is involved in almost 70% of malware-related breaches (Malware) and approximately 24% of all breaches. Overall, the total global value of Ransomware in 2024 exceeded \$30 billion. Deepfake technology is also becoming a powerful tool for cybercriminals, as evidenced by the 550% increase in the number of online deepfakes from 2019 to 2023. According to DeepMedia, in 2023 alone, about 500,000 video and voice deepfakes were distributed on social networks around the world.

Since the beginning of 2025, there has been an adaptation and automation of toolkits for cyberattacks involving artificial intelligence (AI), including vulnerability scanners, since AI can find vulnerabilities that are not yet known to software development companies (zero day). It is believed that the new vector of attacks involving AI will be the AI models themselves, which are used at information activity facilities as well as protective solutions.

The performed analysis results allow us to state that traditional solutions for ICN nodes protection can miss computer viruses due to the imperfection of their detection mechanisms, what is especially critical during multi-level cyberattacks.

The development and continuous improvement of Malware and cyberattacks requires researches into the synthesis of a new protection method against computer viruses for the infocommunication network nodes.

The object of the research is the processes of detecting and blocking the computer viruses in the infocommunication network nodes.

The subject of the research is methods and models of detecting computer viruses and protecting the infocommunication network nodes.

The aim of the dissertation is the synthesis of a protection method against computer viruses for the infocommunication network nodes, capable to provide increasing the effectiveness of counteracts the infocommunication network computer viruses' spread.

In accordance with the set aim, the following new scientific results were obtained in the process of theoretical and experimental research:

1. For the first time it was proposed a protection method against computer viruses for the infocommunication network nodes, which was based on the principle of sequential cyclic appeal to identification operators, decision-making and implementation of control actions, where the state of the ICN node is determined by unloading the minimum required number of digital traces and their analysis by neural network models. It saves time and resources for computer viruses detecting and counteracting their spread in the infocommunication network.

2. For the first time, the usage of digital traces as the main identification feature in the infection of ICN nodes assessing was proposed and implemented. It ensures the detection of Malware missed by traditional endpoint protection solutions and provides an opportunity to improve the existing echeloned ICN protection.

3. For the first time, a relational model in the form of an artifact table was proposed and implemented, which, through filtering, allows to optimize the number and size of digital traces according to the criterion "necessary sufficiency - minimum

time". It ensures saving time and resources for detecting the presence of computer viruses in ICN nodes.

4. For the first time, the usage of neural network models to analyze downloaded digital traces was proposed and implemented. It provides a significant increase in the response speed to emerging incidents in ICNs with a large number of nodes.

5. The method of digital artifacts downloading in conditions of resource constraints was further developed. This method, by optimizing the number and size of digital traces and their ranking, provides the ability to shape an idea of the particular node infection state on the ICN server even on the chance of a connection interruption during data transfer.

The dissertation consists of an introduction, four chapters, in which logically, at a high scientific and technical level the solution to the research tasks were given.

The introduction substantiates the relevance and importance of the dissertation research topic, defines the aim and objectives of the research, indicates the scientific and practical value of the results obtained and the author's personal contribution.

The first chapter presents the results of the current state of information security in infocommunication network nodes analysis. Possible threats to information circulating in infocommunication networks were considered. It was established that modern cyberattacks were implemented by complex multi-level mechanisms of penetration and consolidation in networks and often function as an advanced persistent threat (Advanced Persistent Threat, APT). Ransomware-type Malware attacks with the AI involvement are especially dangerous.

An analytical review of existing methods for protection and detection of harmful effects on infocommunication network nodes was conducted; its results allow to assert that traditional solutions for ICN nodes protection can miss computer viruses due to their detection mechanisms imperfection, what is especially critical during multi-level cyberattacks.

Taking into account the rapid development and constant improvement of Malware and cyberattacks, the aims and objectives of the investigation were formulated in the final part of the first chapter.

In the second chapter a synthesis of a protection method against computer viruses for the infocommunication network nodes was carried out; the peculiarity of the method was the digital traces usage as the main identification feature to determine the ICN node state and their analysis by neural network models, that allows to save time and resources detecting computer viruses and counteracting their spread in the infocommunication network. A glossary of signs and artifact tables sufficient to identify the ICN state were determined for making decisions about the particular node infection. The decision about the state of a specific ICN node is made according to the "ideal observer" criterion, that minimizes the possibility of a total decision error in the control cycle. The feasibility of neural network models' usage to analyze digital traces as a basis for automatically made decision about the particular ICN node infection in the control system was substantiated. The sequence of actions in the control cycle according to the proposed method for ICN nodes protection was determined.

The ways to minimize the average time for distributed ICN active nodes analysis were proposed. The Malware impact on ICN nodes in a virtual environment with a changed configuration was analyzed and digital traces left by Malware were identified. The method for digital traces unloading under the resource constraints conditions was proposed. A comparison of programs for Forensic Triage conducting was performed. It was determined that not all programs provide the necessary traces and, in some cases, provide redundant information. The relational model represented by the artifact table was proposed and implemented; it allows, by the filtering way, to optimize the number and size of digital traces between the artifacts available in the node and those sufficient for the state identifying, what saves time and resources for detecting the presence of computer viruses in ICN nodes. Relevant digital traces to assess a ICN node under resource constraints conditions were identified and analyzed. A list of typical cases was determined and digital traces were ranked

according to their significance for a particular case. The results obtained in the second chapter allow to assess that the information protection method in ICN nodes usage will provide significant savings in time and resources for determining the specific ICN node infection.

In the third chapter neural network models were tested to analyze the downloaded digital traces, what allows to detect the infected ICN nodes. The artificial intelligence technologies and models were considered. The evaluating and comparing neural network models' characteristics criteria were determined. There were studied 135 GGUF format models and 43 the most relevant neural network models for the digital traces analysis were selected. The principle of forming the training data for pre-training neural network models was proposed for a better assessment of ICN nodes infection.

In the fourth chapter the experimental verification results of the proposed ICN nodes protection method effectiveness were outlined. A set of tests and programs to assess the effectiveness of antivirus (AV) solutions for endpoints (NGAV, EDR) was presented; it provides a comprehensive consideration of the antivirus solution reaction on the various threats. There were substantiated the criteria by which the expected AV response on Malware of the Ransomware family was fixed; and a script that simulates the behavior of known Ransomware to test the AV heuristic module was proposed. The ways to optimize the protection method for ICN nodes were proposed.

The effective usage of the relational model to identify the state of ICN nodes was confirmed; since the number of artifacts had been considerably reduced (in some cases even by more than 97%), it significantly saved time for further research.

The effectiveness of the digital artifacts unloading method in a specific ICN node was confirmed. The implemented method is fully processed in 2 minutes 40 seconds (which is 81.1% faster than Cedarpelta) and has an artifact size of 89.4 MB (which is 48% less than Cedarpelta and 88% less than KAPE) with the same data relevance, while in the first 1 minute 28 seconds 67 KB of artifacts were generated, which is enough for initial viewing. It means that the method can provide relevant

sorted data and take up little memory, that in turn saves time during the digital traces transfer and analysis. It was tested the effectiveness of the method for detecting indicators of compromise based on selected neural network models. It was determined that pre-selected models can provide a false assessment. From the 16 pre-selected models, only 2 were able to correctly and without errors answer two questions: gemma-3-4b-it-Q4_K_M, qwq-32b-preview-senecallmv1.2-q4_k_m. Increasing the accuracy and relevance of the answers is achieved by the model pre-training on the proposed training data. The effectiveness of the method was tested in the decision-making operator. The implemented protection method for ICN nodes is fully implemented in 340 seconds or 5 minutes 40 seconds, which is equal about 6 minutes per ICN node. It is 90% faster comparing to full manual analysis and an infected ICN node isolation, which usually takes more than an hour.

The dissertation was completed at the Borys Grinchenko Kyiv Metropolitan University.

The results of scientific research were used at the Department of Information and Cybersecurity named after Professor Volodymyr Buriachok at the Borys Metropolitan Grinchenko Kyiv University within the framework of research work: “Methods and Models for Ensuring Cybersecurity of Information Processing Systems and Functional Security of Software and Hardware Complexes for Critical Infrastructure Management” (No. 0122U200483, BGKMU, Kyiv).

The results of the scientific research were also accepted for implementation in the activities of the Borys Grinchenko Kyiv Metropolitan University (act dated 21.04.2026) and LLC "AUCHAN Ukraine Hypermarket" (act dated 10.03.2026).

Keywords: infocommunication network, cyberthreats, cyberattacks, Malwares, computer viruses, Ransomware, cybersecurity, information protection, protection methods, digital traces, Forensic Triage, neural network models, Artificial Intelligence, LLM, AI model training, identification peculiarities, decision making, realization of control actions.

Наукові статті, опубліковані у наукових виданнях, включених на дату опублікування до переліку наукових фахових видань України:

1. Чернігівський І., Крючкова Л. (2025). Тестова послідовність виявлення та ізоляції заражених вузлів інфокомунікаційної мережі. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 3(31), 652–662. <https://doi.org/10.28925/2663-4023.2025.31.1070>
2. Чернігівський І., Крючкова Л. (2025). Інформаційні впливи на інфокомунікаційні мережі із залученням штучного інтелекту. *Телекомунікаційні та інформаційні технології*, 3(88), 167-176. <https://doi.org/10.31673/2412-4338.2025.038719>
3. Чернігівський І., Крючкова Л. (2025). Тестування нейромережових моделей для вирішення задачі виявлення заражених ПК на базі цифрових слідів. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 1(29), 800–817. <https://doi.org/10.28925/2663-4023.2025.29.941>
4. Чернігівський І., Крючкова Л. (2025). Ефективні рішення для швидкого виявлення скомпрометованих ПК в інфокомунікаційних мережах. *Телекомунікаційні та інформаційні технології*, 2(87), 24–32. <https://doi.org/10.31673/2412-4338.2025.029875>
5. Чернігівський І., Крючкова Л. (2025). Системний підхід до вирішення задачі захисту інформації в інфокомунікаційній мережі від впливу комп'ютерних вірусів. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 3(27), 572–590. <https://doi.org/10.28925/2663-4023.2025.27.781>
6. Чернігівський І., Крючкова Л. (2024). Тестування антивірусних рішень для корпоративного сегменту. *Безпека інформації*, 30(3), 407–413. <https://doi.org/10.18372/2225-5036.30.20362>
7. Чернігівський І., Богданов О., (2024). Типи цифрових криміналістичних артефактів в комп'ютерах під управлінням ОС

Windows. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 4(24), 221–228. <https://doi.org/10.28925/2663-4023.2024.24.221228>

Наукові публікації, у яких додатково висвітлено результати дисертації:

1. Чернігівський І. А., Захист інформації в інфокомунікаційній мережі від впливу комп'ютерних вірусів, Збірник тез XII Всеукраїнської науково-практичної конференції молодих учених 2025, С. 339-341. URL: <https://zcit.kubg.edu.ua/index.php/journal/issue/view/13/23>

2. Чернігівський І.А., Роль SIEM у побудові комплексного захисту інформаційно-телекомунікаційних систем, II International Scientific and Practical Conference Kharkiv, Ukraine 2024, С. 21-23.

3. Чернігівський І. А., Виявлення активності шкідливого програмного забезпечення у вузлах інфокомунікаційної мережі на основі нейромережевих моделей, XIII Всеукраїнська науково-технічна конференція з міжнародною участю "Сучасні проблеми інформаційної безпеки на транспорті" 2025, С. 45-49. URL: <https://nuos.edu.ua/wp-content/uploads/2025/12/SPIBT-2025-Materiali.pdf>

4. L. Kriuchkova, I. Tsmokanych, N. Mazur, I. Chernihivskyi, Spectral Characteristics of Intermodulation Emissions during High-Frequency Imposition, *Cybersecurity Providing in Information and Telecommunication Systems*, 2025, С. 449-462. URL: <https://ceur-ws.org/Vol-3991/paper32.pdf>

5. I. Chernihivskyi, L. Kriuchkova, A method of generating data for further training artificial intelligence models aimed at solving cybersecurity problems, *Cybersecurity Providing in Information and Telecommunication Systems*, 2025, С. 246-256. URL: <https://ceur-ws.org/Vol-4145/paper16.pdf>

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	19
ВСТУП	20
РОЗДІЛ 1. АНАЛІЗ СУЧАСНОГО СТАНУ ЗАХИЩЕНОСТІ ІНФОРМАЦІЇ У ВУЗЛАХ ІНФОКОМУНІКАЦІЙНОЇ МЕРЕЖІ	29
1.1. Аналіз можливих загроз інформації, що циркулює в інфокомунікаційних мережах	29
1.1.1. Інфокомунікаційні мережі як об'єкти вірусних атак	29
1.1.2. Типові загрози для інформації, що циркулює в ІКМ	37
1.1.3. Залучення ІШ для атак на інфокомунікаційні мережі	50
1.2. Аналіз існуючих методів захисту вузлів інфокомунікаційної мережі	58
Висновки до розділу 1. Постановка наукових завдань дослідження.....	64
РОЗДІЛ 2. СИНТЕЗ МЕТОДУ ЗАХИСТУ ВУЗЛІВ ІКМ ВІД КОМП'ЮТЕРНИХ ВІРУСІВ НА ОСНОВІ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ	67
2.1. Системний підхід як основа методу захисту вузлів інфокомунікаційної мережі	67
2.2. Обґрунтування доцільності використання цифрових слідів для оцінки стану вузлів ІКМ	69
2.3. Обґрунтування доцільності застосування нейромережевих моделей для аналізу вивантажених цифрових слідів	80
2.4. Послідовність дій в циклі управління за запропонованим методом захисту вузлів ІКМ.....	83
2.5. Мінімізація середнього часу на аналіз активних вузлів розподіленої ІКМ	87
2.6. Дослідження впливу ШПЗ на вузли ІКМ	91
2.6.1. Дослідження ШПЗ у віртуальному середовищі зі зміненою конфігурацією.....	91
2.6.2. Ідентифікація цифрових слідів, що залишаються після вірусної атаки.....	101
2.7. Метод вивантаження цифрових слідів в умовах обмеженості ресурсів	111
2.7.1. Порівняння програм для проведення Forensic Triage	112

2.7.2. Реляційна модель релевантних цифрових слідів.....	113
2.7.3. Ранжування цифрових слідів у відповідності до їх значущості для різних кейсів	121
Висновки до розділу 2	126
РОЗДІЛ 3. ТЕСТУВАННЯ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ ВИЯВЛЕННЯ ЗАРАЖЕНИХ ВУЗЛІВ ІКМ НА БАЗІ ЦИФРОВИХ СЛІДІВ.....	
3.1. Технології і моделі штучного інтелекту	129
3.2. Визначення критеріїв оцінки та результати тестування нейромережових моделей.....	135
3.3. Виявлення релевантних нейромережових моделей для аналізу цифрових слідів	147
3.4. Формування навчальних даних для дотренування нейромережових моделей.....	150
Висновки до розділу 3	164
РОЗДІЛ 4. ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА ЕФЕКТИВНОСТІ ЗАПРОПОНОВАНОГО МЕТОДУ ЗАХИСТУ ВУЗЛІВ ІКМ	
4.1. Тестування антивірусних рішень для кінцевих точок (NGAV, EDR) ...	166
4.2. Підтвердження ефективності реляційної моделі, призначеної для ідентифікації стану вузлів ІКМ.....	179
4.3. Підтвердження ефективності методу вивантаження цифрових артефактів в конкретному вузлі ІКМ.....	183
4.4. Перевірка ефективності методу виявлення індикаторів компрометації на базі відібраних нейромережових моделей.....	187
4.5. Перевірка ефективності запропонованого методу в операторі прийняття рішення.....	218
Висновки до розділу 4	220
ВИСНОВКИ.....	222
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	226
ДОДАТКИ.....	237

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ІКМ – інфокомунікаційна мережа

ШПЗ – шкідливе програмне забезпечення

ПК – персональний комп'ютер

АВ – антивірус

ВМ – віртуальна машина

ШІ – штучний інтелект

NGAV – Next-Generation Antivirus ‘антивірус нового покоління’

EDR – Endpoint Detection and Response ‘виявлення та реагування на кінцеві точки’

СУІБ – система управління інформаційною безпекою

SIEM – Security Information and Event Management ‘система управління інформацією та подіями безпеки’

IDS – Intrusion Detection System ‘система виявлення вторгнень’

IPS – Intrusion Prevention System ‘система запобігання вторгнень’

ВСТУП

Обґрунтування вибору теми дослідження. Інтенсивний розвиток інформаційних технологій та інфокомунікаційних мереж призвів до впровадження передових технологій обробки і передачі даних та появи якісно нових послуг і сервісів в інформаційній сфері. Зазначене зумовило появу нових форм і способів несанкціонованого доступу до обчислювальних ресурсів ІКМ та призвело до постійного щорічного зростання кількості і складності кібератак на інформаційні системи, спрямованих на порушення цілісності, конфіденційності і доступності інформації. При цьому складні атаки типу АРТ (Advanced Persistent Threat) не виявляються традиційним антивірусом і тому зловмисник може тривалий час перебувати в мережі не проявляючи своєї присутності активними діями, допоки його буде виявлено. Загальновідомі тактики, вдосконалені штучним інтелектом, програми-вимагачі як послуга (RaaS) та передові методи соціальної інженерії випереджають традиційні засоби захисту.

Найпоширенішими загрозами в інфокомунікаційних мережах є поширення комп'ютерних вірусів, оскільки вони можуть одночасно порушувати конфіденційність (Trojan, Spyware) цілісність (Exploit, Ransomware, Wiper) та доступність (Trojan, Ransomware, Wiper) інформації а загальна кількість комп'ютерних вірусів перевищує 1,5 млрд.; фішинг із використанням технології Deepfake, оскільки кількість онлайн-діпфейків зросла на 550% з 2019 по 2023 рік; залучення штучного інтелекту для автоматизації кібератак у тому числі і для автоматичного виявлення та експлуатації вразливостей, оскільки це знижує час на підготовку та проведення атак.

Дослідження проблеми забезпечення кібербезпеки інформаційних систем та аналіз ШПЗ і цифрових слідів представлені у працях багатьох вчених, серед яких Яремчук Ю., Hinteа D., Rogers M., Jusas V., Василенко М., Ричка Д., Bashir M., D. -Y. Као, Shevchenko S., Skladannyi P., Davydov V., Abu

Taam Ghani Mohamad, A. Smirnov, Терейковський І. А., Mahmoud Kalash та інші.

Проведений аналіз сучасних наукових досліджень показав, що більшість існуючих підходів використовують наявні фреймворки, математичні моделі, методи машинного навчання або звичайні алгоритми пошуку для аналізу стану вузлів ІКМ та визначення загроз. Проведення таких досліджень є результативним, оскільки вони зробили вагомий внесок у розвиток методів захисту інформації. Проте дані підходи зазвичай ґрунтуються на ШП-аналізі конкретного потенційно шкідливого виконуваного файлу, надають перелік неоптимальних цифрових слідів для ручного аналізу, не враховують роботу вузлів у розподіленій ІКМ, не надають конкретних практичних рекомендацій щодо вибору ШП-моделей для автоматичного визначення стану зараженості вузла ІКМ, не запропоновано визначення стану вузла, коли ШПЗ може самоліквідуватись з вузла і буде відсутній виконуваний файл, не розглядаються можливості автоматично виявити атаку яку пропустило традиційне захисне рішення. Варто зазначити, що традиційні рішення захисту вузлів ІКМ можуть пропускати комп'ютерні віруси внаслідок недосконалості механізмів їх виявлення, що особливо критично під час багаторівневих кібератак.

Викладене обумовлює необхідність створення нових і вдосконалення існуючих методів захисту інформації в інфокомунікаційних мережах в умовах обмежених ресурсів, вдосконалення наявних методів ідентифікації ШПЗ на ПК для підвищення ефективності протидії поширенню комп'ютерних вірусів в інфокомунікаційній мережі. Перспективним шляхом є ефективний розподілу часу та ресурсів аналітика відділу інформаційної безпеки для своєчасного виявлення та запобігання загрозам.

Зв'язок роботи з науковими програмами, планами, темами. Напрямо дисертаційного дослідження безпосередньо пов'язаний з реалізацією доктрини інформаційної безпеки України, Стратегії інформаційної безпеки та Стратегії кібербезпеки України.

Дослідження проведено в рамках реалізації науково-дослідної теми "Методи та моделі забезпечення кібербезпеки інформаційних систем переробки інформації та функціональної безпеки програмно-технічних комплексів управління критичної інфраструктури" (реєстраційний номер 0122U200483, КСУБГ, м. Київ).

Мета і завдання дослідження. Метою дисертаційної роботи є синтез методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, здатного забезпечити підвищення ефективності протидії поширенню комп'ютерних вірусів в інфокомунікаційній мережі.

Для досягнення поставленої мети вирішено наступні наукові завдання:

- здійснено синтез методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, здатного забезпечити економію часу і ресурсів на виявлення комп'ютерних вірусів та протидії їх поширенню в інфокомунікаційній мережі;
- обґрунтовано доцільність використання цифрових слідів у якості основної ідентифікаційної ознаки для оцінки стану вузлів ІКМ;
- розроблено модель для оптимізації кількості і розміру цифрових слідів, достатніх для ідентифікації стану вузлів ІКМ;
- шляхом тестування здійснено відбір релевантних нейромережових моделей для аналізу вивантажених цифрових слідів;
- удосконалено метод вивантаження цифрових артефактів в умовах обмеженості ресурсів;
- виконано експериментальну перевірку ефективності запропонованого методу захисту вузлів ІКМ.

Об'єктом дослідження є процеси виявлення та блокування комп'ютерних вірусів у вузлах інфокомунікаційної мережі.

Предметом дослідження є методи і моделі виявлення комп'ютерних вірусів та захисту вузлів інфокомунікаційної мережі.

Методи дослідження. Для проведення досліджень в дисертаційній роботі використовувалися методи аналізу і синтезу систем; теорія інформації; теорія прийняття рішень; теорія алгоритмів; теорія ймовірностей; комп'ютерне та імітаційне моделювання.

Достовірність наукових результатів, висновків та рекомендацій, викладених у дисертаційній роботі, обґрунтовано коректним використанням методів дослідження та комп'ютерним моделюванням на ПК.

Наукова новизна одержаних результатів полягає в наступному:

1. Вперше запропоновано метод захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, побудований за принципом послідовного циклічного звернення до операторів ідентифікації, прийняття рішення та реалізації керуючих дій, у якому ідентифікація стану вузла ІКМ здійснюється на основі вивантаження мінімально необхідної кількості цифрових слідів та їх аналізу нейромережевими моделями, що дозволяє забезпечити економію часу і ресурсів на виявлення комп'ютерних вірусів та протидії їх поширенню в інфокомунікаційній мережі.

2. Вперше запропоновано і реалізовано використання цифрових слідів у якості основної ідентифікаційної ознаки при оцінці зараженості вузлів ІКМ, що забезпечує виявлення ШПЗ, пропущених традиційними рішеннями захисту кінцевих точок, та надає можливість вдосконалення наявного ешелонованого захисту ІКМ.

3. Вперше запропоновано і реалізовано реляційну модель у вигляді таблиці артефактів, яка шляхом фільтрації дозволяє оптимізувати кількість і розмір цифрових слідів між наявними артефактами у вузлі і достатніми для ідентифікації стану, що забезпечує економію часу і ресурсів для виявлення наявності комп'ютерних вірусів у вузлах ІКМ.

4. Вперше запропоновано і реалізовано застосування нейромережових моделей для аналізу вивантажених цифрових слідів, що забезпечує підвищення швидкості реагування на виникаючі інциденти в ІКМ з великою кількістю вузлів.

5. Набув подальшого розвитку метод вивантаження цифрових артефактів в умовах обмеженості ресурсів, який за рахунок оптимізації кількості і розміру цифрових слідів та їх ранжування забезпечує можливість формування уявлення про стан зараженості конкретного вузла на сервері ІКМ навіть у випадку переривання з'єднання під час передачі даних.

Практичне значення одержаних результатів полягає в тому, що в дослідженні запропоновано моделі та методи, які доцільно використовувати для підвищення кіберзахисту на підприємстві навіть за наявності інших захисних рішень, за рахунок більш оперативного реагування на виникаючі загрози, а також автоматичного прийняття рішення та здійснення керуючих дій. Запропонований метод дозволяє знаходити вірусну активність там, де її пропустив традиційний антивірус навіть за умови самоліквідації вірусного файлу, а також підвищити ефективність реагування на кіберінциденти у системах управління інформаційною безпекою. Запропоновані моделі та методи можуть бути використані організаціями та державними структурами при розробці та удосконаленні оцінки захищеності інформації на вузлах ІКМ.

Практичне значення отриманих результатів полягає у можливості їх застосування в різних галузях для вдосконалення методів захисту інформації від впливу комп'ютерних вірусів і більш ефективного використання часу аналітика при проведенні Forensic-аналізу ІКМ.

Результати досліджень прийняті до впровадження в діяльність Київського столичного університету імені Бориса Грінченка (акт від 21.04.2026 року) та ТОВ «АШАН Україна Гіпермаркет» (акт від 10.03.2026).

Апробація результатів дисертації. Основні теоретичні та практичні результати були представлені та обговорені на наукових конференціях:

1. Чернігівський І. А., Захист інформації в інфокомунікаційній мережі від впливу комп'ютерних вірусів, Збірник тез XII Всеукраїнської науково-практичної конференції молодих учених 2025, С. 339-341. URL: <https://zcit.kubg.edu.ua/index.php/journal/issue/view/13/23>

2. Чернігівський І.А., Роль SIEM у побудові комплексного захисту інформаційно-телекомунікаційних систем, II International Scientific and Practical Conference Kharkiv, Ukraine 2024, С. 21-23.

3. Чернігівський І. А., Виявлення активності шкідливого програмного забезпечення у вузлах інфокомунікаційної мережі на основі нейромережевих моделей, XIII Всеукраїнська науково-технічна конференція з міжнародною участю "Сучасні проблеми інформаційної безпеки на транспорті 2025" С. 45-49 URL: <https://nuos.edu.ua/wp-content/uploads/2025/12/SPIBT-2025-Materiali.pdf>

4. L. Kriuchkova, I. Tsmokanych, N. Mazur, I. Chernihivskiy, Spectral Characteristics of Intermodulation Emissions during High-Frequency Imposition, *Cybersecurity Providing in Information and Telecommunication Systems*, 2025, 449-462 ст. <https://ceur-ws.org/Vol-3991/>

5. I. Chernihivskiy, L. Kriuchkova, A method of generating data for further training artificial intelligence models aimed at solving cybersecurity problems, *Cybersecurity Providing in Information and Telecommunication Systems*, 2025, 246-256ст, <https://ceur-ws.org/Vol-4145/>

Публікації. Основні результати дисертації висвітлено 12 наукових публікаціях, із них 3 – одноосібні, 9 – у співавторстві: 7 статей у наукових виданнях, включених на дату опублікування до переліку наукових фахових видань України; 2 статті у періодичних наукових виданнях, проіндексованих в наукометричних базах даних Scopus і Web of Science Core Collection. 3 публікації (тези) у яких додатково висвітлено наукові результати дисертації. Наукові результати дисертації повною мірою висвітлено у наукових публікаціях.

Особистий внесок здобувача. Дисертація є самостійною науковою працею, в якій висвітлено власні ідеї і розробки автора, що дозволили вирішити поставлені завдання. Робота містить теоретичні та методичні положення і висновки, сформульовані здобувачем особисто. Використані в

дисертації ідеї чи положення інших авторів мають відповідні посилання і використані лише для підкріплення ідей здобувача.

У статті «Тестова послідовність виявлення та ізоляції заражених вузлів інфокомунікаційної мережі» опублікованій у співавторстві, внесок Чернігівського І.А. полягає у створенні тестової послідовності, яка дозволяє виявляти заражені вірусами вузли ІКМ в циклі управління системи захисту та мінімізувати середній час на оцінку одного вузла, що загалом складає 80% тексту статті.

У статті «Інформаційні впливи на інфокомунікаційні мережі із залученням штучного інтелекту» опублікованій у співавторстві, внесок Чернігівського І.А. полягає у визначенні найбільш небезпечних загроз та сценаріїв шкідливих інформаційних впливів із залученням ШІ, що загалом складає 70% тексту статті.

У статті «Тестування нейромережевих моделей для вирішення задачі виявлення заражених ПК на базі цифрових слідів» опублікованій у співавторстві, внесок Чернігівського І.А. полягає у тому, що визначено критерії для моделі ШІ які будуть прийнятними для використання у корпоративному середовищі та проведено тестування 135 моделей формату GGUF на предмет виявлення або невиявлення ними ознак вірусної активності та індикаторів компрометації у промпті, що надавався користувачем, визначено перелік доцільних для використання моделей ШІ у форматі GGUF для вирішення задач кібербезпеки, зокрема для виявлення заражених ПК на базі цифрових слідів, що загалом складає 80% тексту статті.

У статті «Ефективні рішення для швидкого виявлення скомпрометованих ПК в інфокомунікаційних мережах» опублікованій у співавторстві, внесок Чернігівського І.А. полягає у тому, що визначено компонент/тактику, без яких сучасні комп'ютерні віруси зазвичай не працюють, запропоновано перелік програм для швидкого виявлення вірусів і скрипт оптимізації з використанням реляційної таблиці артефактів, які

дозволяють скоротити кількість елементів, необхідних для подальших досліджень більш ніж у десять разів, що загалом складає 80% тексту статті.

У статті «Системний підхід до вирішення задачі захисту інформації в інфокомунікаційні мережі від впливу комп'ютерних вірусів» опублікованій у співавторстві, внесок Чернігівського І.А. полягає у пропозиції будувати системи захисту інформації у вигляді автоматизованої системи управління, спрямованої на забезпечення підтримки цільового стану ІКМ, створенні словника ознак для ідентифікації стану ІКМ на основі цифрових слідів, який є достатнім для прийняття рішень в циклах управління системи захисту інформації, що загалом складає 70% тексту статті.

У статті «Тестування антивірусних рішень для корпоративного сегменту» опублікованій у співавторстві, внесок Чернігівського І.А. полягає у пропозиції набору тестів і програм які є достатніми для оцінки ефективності захисного рішення у корпоративному середовищі, визначено, що аналіз реакцій АВ на кілька ШПЗ з сімейства Ransomware цілком достатньо для отримання базового уявлення про евристичний модуль АВ та його можливість протидіяти новим загрозам, визначено критерії, за якими очікується спрацювання АВ на ШПЗ сімейства Ransomware, а також запропоновано скрипт, що імітує поведінку відомих Ransomware для тестування евристичного модулю АВ, що загалом складає 75% тексту статті.

У статті «Типи цифрових криміналістичних артефактів в комп'ютерах під управлінням ОС Windows» опублікованій у співавторстві, внесок Чернігівського І.А. полягає у визначенні типів артефактів в ОС Windows та у визначенні, що для деяких задач буде достатньо лише невеликої кількості артефактів, які можна швидко зібрати для аналізу, що загалом складає 90% тексту статті.

У статті «Spectral Characteristics of Intermodulation Emissions during High-Frequency Imposition» опублікованій у співавторстві, внесок Чернігівського І.А. полягає у пропозиції розглядати проблему захисту конфіденційної інформації від перехоплення методом високочастотного нав'язування із

системних позицій, як проблему забезпечення електромагнітної сумісності технічних систем передачі, обробки та зберігання інформації, що загалом складає 10% тексту статті.

У статті «A method of generating data for further training artificial intelligence models aimed at solving cybersecurity problems» опублікованій у співавторстві, внесок Чернігівського І.А. полягає у пропонуванні методу формування навчальних даних для донавчання моделей ШІ, що забезпечує: спеціалізацію конкретної моделі ШІ на виявлення основних ознак вірусної активності в наданих цифрових слідах; підвищення якості відповідей ШІ; зниження часу реагування на інциденти кібербезпеки. Пропозицію оформлення цифрових слідів для донавчання моделей ШІ у табличному вигляді з попередньою фільтрацією цифрових слідів на основі реляційної таблиці артефактів, що дозволяє скоротити кількість елементів, необхідних для подальших досліджень, що загалом складає 80% тексту статті.

Структура та обсяг дисертаційної роботи. Дисертація складається зі вступу, чотирьох розділів, висновків, списку використаних джерел із 135 найменувань на 11 сторінках і 3 додатки. Загальний обсяг роботи становить 244 сторінки, серед яких 197 сторінок – основного тексту, 58 рисунків і 43 таблиці.

РОЗДІЛ 1. АНАЛІЗ СУЧАСНОГО СТАНУ ЗАХИЩЕНОСТІ ІНФОРМАЦІЇ У ВУЗЛАХ ІНФОКОМУНІКАЦІЙНОЇ МЕРЕЖІ

1.1. Аналіз можливих загроз інформації, що циркулює в інфокомунікаційних мережах

1.1.1. Інфокомунікаційні мережі як об'єкти вірусних атак

У сучасному світі впливовим фактором на успіх будь-якого підприємства є ефективність його інформаційної інфраструктури. Для правильної оцінки ситуації та прийняття своєчасного обґрунтованого рішення керівнику необхідна інформаційна система, яка забезпечує достовірне і повне відображення актуального стану підприємства.

В сучасних умовах успішним технологічним рішенням може вважатися інформаційна система, яка має такі характеристики як:

- простота для кінцевого користувача;
- гнучкість у впровадженні функцій;
- масштабованість;
- відкритість;
- можливість розмежування доступу користувачів;
- відмовостійкість.

Забезпечення зазначених вимог під час створення інформаційної системи можливо за умови застосування для її розробки сучасних засобів, що реалізують стандартні методи обробки, передачі та зберігання даних. Основною тенденцією при побудові корпоративних інфокомунікаційних мереж є використання Internet/Intranet-технологій, які найбільш повно відповідають вищезазначеним вимогам.

Критично важливим завданням для забезпечення надійного та ефективного функціонування інфокомунікаційної мережі є завдання захисту інформації.

У законах України «Про інформацію» [1], «Про основні засади забезпечення кібербезпеки України» [2] і стандартах ДСТУ ISO/IEC 27001:2023 [3], ISO/IEC 27000:2018 [4], ISO/IEC 10181-3:1996 [5] та ISO/IEC 7498-1:1994 [6] зазначаються наступні поняття:

Захист інформації – сукупність правових, адміністративних, організаційних, технічних та інших заходів, що забезпечують збереження, цілісність інформації та належний порядок доступу до неї.

Використовуючи сімейство стандартів [3, 4], організації можуть розробити та впровадити систему управління безпекою своїх інформаційних активів, включаючи фінансову інформацію, інтелектуальну власність та дані про співробітників, а також інформацію, довірену їм клієнтами чи третіми особами. Зазначені стандарти також можуть бути використані для підготовки до незалежної оцінки системи управління інформаційною безпекою, що застосовується для захисту інформації. Використовуючи сімейство стандартів [5, 6], визначено, що співпраця між відкритими системами охоплює широкий спектр діяльності, серед яких визначено цілісність та безпеку, що стосуються обмежень обробки інформації, які необхідно зберігати або гарантувати під час роботи відкритих систем. Ініціаторами в таких системах є сутності (наприклад, користувач-людина або комп'ютер чи сервер), які намагаються отримати доступ до інших сутностей чи до інформації.

Інформація – будь-які відомості та/або дані, які можуть бути збережені на матеріальних носіях або відображені в електронному вигляді.

Кіберпростір – середовище (віртуальний простір), яке надає можливості для здійснення комунікацій та/або реалізації суспільних відносин, утворене в результаті функціонування сумісних (з'єднаних) комунікаційних систем та забезпечення електронних комунікацій з використанням мережі Інтернет та/або інших глобальних мереж передачі даних.

Кібербезпека – стан захищеності даних в електронному вигляді від їх несанкціонованого використання або кримінальних дій з цими даними, а також набір заходів для досягнення такого стану захищеності даних.

Кібератака – навмисне втручання в роботу комп'ютерних систем, мереж або отримання несанкціонованого доступу до систем та даних.

Інцидент інформаційної безпеки – один або серія небажаних чи неочікуваних подій інформаційної безпеки, які мають значну ймовірність поставити під загрозу бізнес-операції та загрожують інформаційній безпеці.

Інформація має наступні властивості [7, 8, 5, 6]:

Конфіденційність інформації – властивість, яка гарантує те, що доступ до інформації можуть одержати тільки авторизовані особи або процеси. Забезпечує захист інформації під час передачі або зберігання.

Цілісність інформації – властивість, яка гарантує те, що інформація не містить помилок, є актуальною, вичерпною, будь-які зміни інформації здійснюються авторизованими особами або процесами.

Доступність інформації – властивість, яка гарантує те, що забезпечується своєчасний доступ авторизованих осіб і/або процесів до інформації, а також відсутні затримки в процесі її обробки.

Відповідно до властивостей інформації, виділяють такі загрози [8]:

- загрози цілісності: знищення; модифікація;
- загрози доступності: блокування; знищення;
- загрози конфіденційності: несанкціонований доступ (НСД); витік даних; розголошення.

Кожний вид захисту інформації забезпечує окремі аспекти інформаційної безпеки (ІБ) [8]:

Технічний – забезпечує обмеження доступу до носія повідомлення апаратно-технічними засобами (антивіруси, фаєрволи, маршрутизатори, токени, смарткарти тощо):

- попередження витоку технічними каналами;
- запобігання блокуванню;

Інженерний – запобігає руйнуванню носія внаслідок навмисних дій або природного впливу інженерно-технічними засобами (сюди відносять обмежувальні конструкції, охоронно-пожежна сигналізація).

Криптографічний – попереджує доступ за допомогою математичних перетворень повідомлення:

- попередження несанкціонованої модифікації;
- попередження НС розголошення.

Організаційний – попередження доступу на об'єкт інформаційної діяльності сторонніх осіб за допомогою організаційних заходів (правила розмежування доступу).

Узагальнену структуру інфокомунікаційної мережі наведено на рис. 1.1.

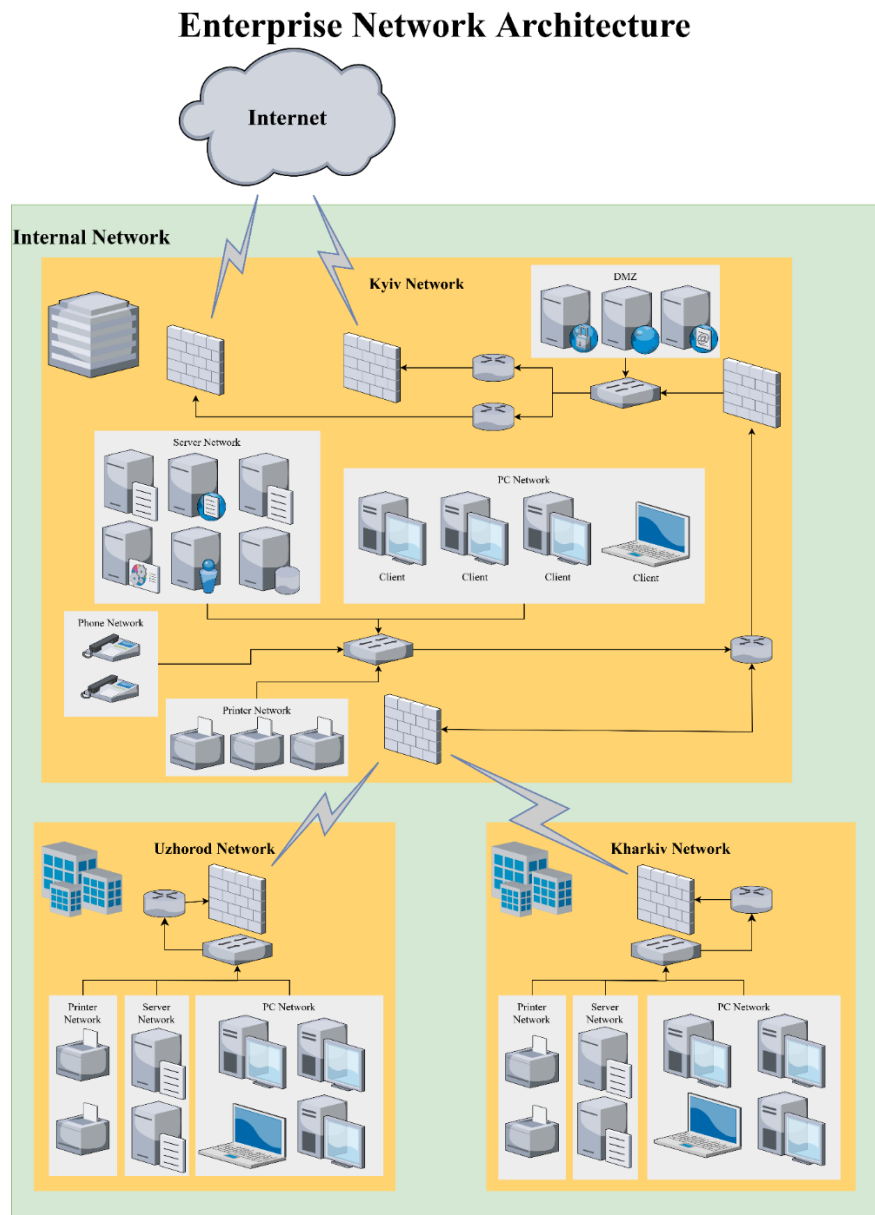


Рис. 1.1. Узагальнена структура інфокомунікаційної мережі (створено автором)

Зазвичай структура ІКМ включає головний офіс та додаткові філіали у різних містах, філіали отримують доступ до інтернету та необхідних для функціонування мережі внутрішніх ресурсів (Active Directory, Database File Server, тощо) через головний офіс. В локальних мережах швидкість передачі даних 90-100 Мбіт/с, між мережами – 20 Мбіт/с, швидкість доступу в Інтернет коливається в межах 60-90 Мбіт/с. ПК і сервери мають різну апаратну/програмну конфігурацію (у тому числі оновлення і версії ОС).

Для прикладу, мережа має неоднорідну структуру:

Kiev Network – має 300ПК (Win7 100, Win10 170, Win11 30) і 10 серверів (Win2003 1, Win2008 2, Win2016 2, Win2019 5);

Uzhorod Network – має 200ПК (WinXP 5, Win7 75, Win8.1 10, Win10 110) і 3 сервера (Win2016 1, Win2019 2);

Kharkiv Network – має 200ПК (WinXP 5, Win7 70, Win8.1 5, Win10 120) і 3 сервера (Win2016 1, Win2019 2).

Зазначена мережа має ряд проблем як в адмініструванні, так і в вивантаженні артефактів, що детально розглянуто у авторській статті [9]:

- різна пропускна здатність локальних мереж, що спричиняє довше завантаження даних з філіалів і неможливість швидко оцінити приблизний час завантаження;

- різний час роботи співробітників (різні вихідні дні, різні робочі години, залишення робочого місця в робочі години), що не дозволяє аналітику інформаційної безпеки (ІБ) достовірно спрогнозувати наявність чи відсутність співробітника на робочому місці, адже необхідно щоб ПК був доступний у мережі. При цьому робота аналітика не повинна перешкоджати роботі співробітника;

- неможливість в будь-який час підключитися до ПК та ноутбуків (бо користувачі їх переміщують та підключаються з інших мереж та можуть бути додаткові помилки), що не дозволяє отримати інформацію про стан ПК і невідомо, коли це можна буде зробити в майбутньому;

- залежність швидкодії від того, з якою конфігурацією апаратного і програмного забезпечення ПК працює користувач;
- різні налаштування доступу до ПК через помилки або неправильну конфігурацію (наприклад, адміністратор віддав недоналаштований ПК користувачу, у якому за замовчуванням закрито ISMP та інші протоколи), що перешкоджає отриманню інформації про стан ПК;
- залежність від користувачів, які вимикають свій ПК замість виходу з акаунту після переривання своєї роботи (своєрідна «економія електроенергії»), що спричиняє ситуацію, коли спочатку не можна підключитись до ПК, бо користувач за ним працює, а пізніше підключитись теж не можна, бо користувач вже не працює. Це не дозволяє аналітику вчасно отримати необхідну інформацію.

Додатково, через війну та часті відключення електроенергії, адміністратор не може спрогнозувати, коли конкретний ПК буде знову відключений від мережі. Це призводить до ситуації, коли навіть почавши завантажувати архів з артефактами, що згенерувала програма для проведення Forensic Triage, відбувається переривання з'єднання і невідомо через який час ПК знову з'явиться в мережі. В результаті архів залишається недозакачаним і аналітик не має жодного уявлення про стан ПК.

Інформація, що циркулює в ІКМ, повинна бути захищена таким чином, щоб була забезпечена її конфіденційність, цілісність та доступність. Зазначені властивості інформації не існують ізольовано і впливають одна на одну, що ускладнює завдання захисту інформації у вузлах інфокомунікаційної мережі.

Найголовнішим завданням особи, відповідальної за інформаційну безпеку ІКМ, є побудова такої системи кіберзахисту, що адекватно балансує між комплексною захищеністю ІКМ, безперервністю бізнес-процесів та зручністю кінцевого користувача [7].

Ураження інформаційних систем комп'ютерними вірусами може одночасно призводити до усіх порушень (цілісності, конфіденційності, доступності).

Основним методом боротьби з вірусами завжди були антивіруси, тому процес розвитку вірусів і антивірусів є постійною війною технологій. Регулярно у вірусах реалізуються оригінальні ідеї, що вимагає адекватних дій від розробників антивірусного ПЗ. Для корпоративного сегменту боротьба з вірусами зазвичай забезпечується рішеннями класу Endpoint Detection and Response (EDR), які на додачу до звичайного модулю Endpoint Protection (EPP) мають модулі Next Generation Antivirus (NGAV) та поглиблений поведінковий аналіз. Однак не завжди зазначених заходів буває достатньо, тому необхідно чітко розділяти вузли ІКМ (фактично – корпоративні ПК) на заражені/не заражені, на випадок, якщо захисне рішення «пропустило» невідомий вірус. Цього можна досягти шляхом аналізу цифрових слідів (артефактів), які залишаються після ШПЗ або АРТ-атаки на досліджуваному вузлі ІКМ.

Для швидкої узагальненої ідентифікації стану ІКМ необхідно якомога швидше отримувати інформацію про стан вузлів ІКМ. Завдання захисту інформації у вузлах ІКМ обумовлює необхідність вирішення наступних задач:

- суттєво зменшити кількість часу на обробку інформації та її передачу порівняно з наявними рішеннями;
- передавати цифрові сліди у текстовому форматі нестиснутими;
- розбирати наявні цифрові сліди на предмет аномалій автоматично.

Для зменшення часу, у випадку коли ми не можемо підвищити швидкість передачі у мережі та вплинути на швидкодію диску/процесора досліджуваного ПК, потрібно зменшити розмір та кількість даних, що підлягають передачі.

Цього можна досягти за рахунок:

- зменшення кількості програм та артефактів, залишивши тільки ті, що будуть корисні (без непотрібних дублікатів);
- оптимізації виводу артефактів (відкинувши варіанти які точно не є шкідливими);
- використання програм-парсерів артефактів, щоб передавати не бінарні, а текстові дані, які будуть суттєво менші;

- виключення зайвих операцій (зтискання/розтискання/передачі програм);
- автоматичного віддаленого запуску програм та завантаження артефактів на заздалегідь заданий сервер;
- розпаралеленого запуску кількох процесів, що вивантажують артефакти;
- передачі за можливості метаданих файлу, а не самого файлу (наприклад, не файли Prefetch, а їх розібраний вигляд у текстовому форматі);
- автоматичної передачі артефакту відразу після вивантаження з ПК, а не очікування вивантаження усіх артефактів з подальшою їх централізованою передачею.

Для передачі артефактів у нестиснутому форматі необхідно:

- проранжувати артефакти на відповідність/значущість для конкретного сценарію та визначити порядок їх завантажень, що надасть можливість отримати більше уявлення про стан системи, ніж якби вони завантажувались їх у випадковому порядку;
- використовувати програми-парсери артефактів, щоб передавати не бінарні, а текстові дані, які будуть суттєво менші;
- здійснювати підготовку текстового виводу для машинної обробки, що в деяких випадках забезпечує значне зменшення розміру вихідного файлу.

Для автоматичного розбору аномалій потрібен окремий модуль, функціональним призначенням якого є:

- виявлення специфічних даних, які з високою імовірністю вказують на потенційне зараження;
- виконання відповідних додаткових дій над проблемним ПК (наприклад, заблокувати його до вирішення всіх обставин і перевірки інших наявних артефактів);
- формування відповідного повідомлення, який приверне увагу аналітика ІБ.

В умовах кібервійни найімовірнішими є шкідливі впливи, створені задля перехоплення і спотворення даних, перехоплення управління системою зі зміною її функціонування, і знищення інфокомунікаційної системи. Найбільш ймовірні впливи на ІКМ через: мережу та знімні носії, а також використання експлоїтів у програмному забезпеченні та операційній системі. При наявності широкого різноманіття можливих способів реалізації шкідливих інформаційних впливів, система захисту повинна ідентифікувати та нейтралізувати шкідливий інформаційний вплив. Стан будь-якої інфокомунікаційної системи характеризується великою кількістю показників. Дані показники динамічно змінюються і можуть мати складні взаємні залежності, що обумовлено логікою функціонування системи та інформаційними впливами на цю систему.

Оскільки інфокомунікаційна мережа просторово розподілена і має обмежені ресурси, то:

- ресурси, які можна виділити системі захисту обмежені;
- робота системи захисту не повинна перешкоджати роботі користувача вузла ІКМ;
- важливим для економії часу, необхідного на обробку вивантажених вихідних даних (цифрових слідів), є кількість і розмір цифрових слідів.

Викладені умови функціонування та недоліки розглянутої ІКМ вимагають вірних і своєчасних управлінських рішень для запобігання небажаних наслідків. Ефективність протидії поширенню комп'ютерних вірусів в інфокомунікаційній мережі доцільно визначати як економію часу і ресурсів на виявлення комп'ютерних вірусів та протидію їх поширенню в інфокомунікаційній мережі.

1.1.2. Типові загрози для інформації, що циркулює в ІКМ

Важливим етапом під час проектування захисту інформації, що циркулює в ІКМ є виявлення та аналіз множини загроз ІКМ, де визначальною ознакою загрози є її спрямованість, що показано у таблиці 1.1 [10].

Таблиця 1.1

Множина загроз ІКМ (створено автором на основі джерела [10])

Зовнішні				Внутрішні			
Умисні		Ненавмисні		Умисні		Ненавмисні	
Явні	Приховані	Явні	Приховані	Явні	Приховані	Явні	Приховані

До проявів загрози дестабілізуючого впливу на інформацію, відносять наступні фактори [10]:

- джерела (від кого або від чого отримано вплив);
- види (яким саме чином і за якими напрямками відбувається вплив);
- способи (якими саме тактиками або діями реалізуються види впливу).
- наявні канали і методи несанкціонованого доступу до конфіденційної інформації з боку осіб, які не повинні мати доступу до неї.

Найпоширенішим і найнебезпечнішим джерелом дестабілізуючого впливу на конфіденційну інформацію є люди, яких можна віднести до таких категорій [10]:

- співробітники підприємства, що володіють інформацією;
- особи, які мають доступ до конфіденційної інформації і при цьому не є штатними співробітниками підприємства (наприклад, підрядники);
- співробітники державних органів, розвідки, підприємств-конкурентів;
- особи з кримінальних структур, кіберзлочинці.

У свою чергу, зазначені категорії людей можна поділити на дві групи: повинні мати доступ до інформації в силу свого службового становища, взагалі не повинні мати доступу. Різниця між ними в конкретних видах і методах дестабілізуючого впливу на інформацію зумовлена цілями, що вони собі поставили. Особливістю першої групи людей є те, що у них вже наявний доступ до інформації, тому цілями дестабілізуючого впливу є розголошення, несанкціоноване знищення, блокування, спотворення інформації. Особливістю другої групи людей є те, що основна мета це створення каналу для отримання несанкціонованого доступу до інформації та її подальшого

витоку, а знищення, спотворення, блокування інформації зазвичай не є пріоритетом. Дестабілізуючий вплив з боку цієї групи людей є навмисним і може бути безпосереднім на інформацію, що підлягає захисту або опосередкованим, через інше джерело. [10]. Поширені загрози інформаційній безпеці ІКМ подано у табл. 1.2.

Таблиця 1.2

Поширені загрози інформаційній безпеці ІКМ (створено автором на основі джерела [11])

Тип шкідливого впливу	Пояснення
Атака удосконаленої постійної загрози (APT)	Це складна кібератака, яка полягає в тому, що зловмисник (або група зловмисників) отримує доступ до корпоративної мережі та даних і залишається непоміченим протягом тривалого періоду.
Бот-мережа	Термін бот-мережа утворений від словосполучення "robot network" (роботизована мережа) і означає мережу пов'язаних пристроїв (ботів), які кіберзлочинець уражає зловмисним кодом для віддаленого керування.
Розподілена атака на відмову в обслуговуванні (DDoS-атака)	DDoS-атаки виконуються за допомогою бот-мереж і полягають у заспамленні корпоративних вебсайтів та програм, у результаті чого для справжніх користувачів вебсайт стає недоступним через несправність чи відмову в обслуговуванні.
Атака тіньового завантаження	Це фрагмент зловмисного коду, який автоматично завантажується на пристрій користувача після того, як той відвідує вебсайт. Таким чином користувач стає вразливим до подальших кіберзагроз.
Набір експлойтів (exploit-kit)	Комплексний набір інструментів, які автоматично виявляють потенційні вразливості та використовують експлойти для ураження пристроїв шкідливим програмним забезпеченням.
Внутрішня загроза	Працівник організації навмисно або випадково використовуватиме авторизований доступ, що зашкодить корпоративним системам, мережам і даним або зробить їх уразливими для атак.
Атака зловмисного посередника (MitM)	Зловмисник перериває лінію зв'язку або передавання даних, щоб видаючи себе за легітимного користувача викрадати інформацію.
Фішингова атака	Під час фішингових атак зловмисники маскуються під справжні організації або дійсних користувачів, щоб викрадати інформацію через електронні листи, текстові повідомлення або інші методи спілкування.

Тип шкідливого впливу	Пояснення
Віруси та хробаки	Приховане шкідливе програмне забезпечення, яке може самостійно поширюватися в мережі або системі користувачів.
Зловмисна програма з вимогою викупу (Ransomware)	Атака цих зловмисних програм шифрує інформацію цілої організації або окремих користувачів, блокуючи до них доступ, доки не буде сплачено викуп.
Соціотехніка	Цей тип кібератак включає людський фактор. Він полягає в тому, що зловмисник завойовує довіру жертви, використовуючи байтинг, подробиці антивірусні програми або фішинг, а потім збирає персональні дані та використовує їх для подальших атак.
Атака в соцмережах	Під час таких атак кіберзлочинці вражають платформи соціальних мереж, використовуючи їх як механізми поширення шкідливого програмного забезпечення, або крадуть інформацію чи дані користувачів.

APT-атака зазвичай складна і багатокомпонентна, етапи APT-атаки представлено на рис. 1.2.

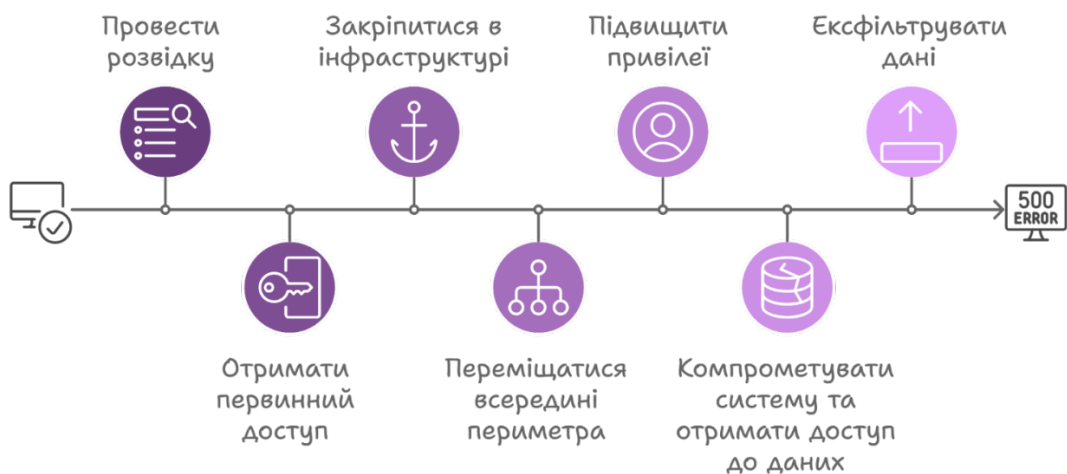


Рис. 1.2. Типові стадії APT-атаки (створено автором із використанням Napkin AI)

Зазвичай кіберзлочинець буде намагатися отримати несанкціонований доступ до мережі з метою здійснення кібератаки. Кіберзагрози можуть варіюватися від очевидних, таких як електронний лист від іноземного дідуся, який пропонує оформити статок, в обмін на дані з банківського рахунку чи невеликий грошовий переказ у якості податку, до прихованих, таких як рядок

шкідливого коду, який обходить наявний кіберзахист і живе в мережі місяцями або роками, перш ніж спровокувати витік даних. Тому, чим більше команд безпеки та співробітників знають про різні типи кіберзагроз, тим ефективніше вони можуть запобігати кібератакам, готуватися та ефективно реагувати на них. [12].

Багато зловмисників з самого початку мають негативні наміри, тоді як інші, такі як етичні хакери або ненавмисні внутрішні загрози, мають позитивні або нейтральні наміри. Знання мотивації та тактики різних зловмисників є важливим під час аналізу загроз та формування моделі порушника, виконавці кібератак представлені у таблиці 1.3

Таблиця 1.3

Виконавці кібератак (створено автором на основі джерела [12])

Виконавці	Пояснення
Кіберзлочинці	Ці особи або групи скоюють кіберзлочини, здебільшого заради фінансової вигоди. Поширені злочини, скоєні кіберзлочинцями, включають атаки програм-вимагачів та фішингові шахрайства, які обманом змушують людей розголошувати конфіденційну інформацію про кредитні картки, облікові дані тощо.
Хакери	Хакер – людина, яка має достатньо глибокі технічні навички для компрометації всієї комп’ютерної мережі чи окремого вузла. При цьому, не всі хакери є кіберзлочинцями. Наприклад, деякі хакери, яких називають етичними хакерами, використовують тактики кіберзлочинців, щоб допомогти перевірити комп’ютерні системи на наявність вразливостей до кібератак.
Зловмисники спонсоровані державою (АРТ).	Держави часто фінансують зловмисників з метою крадіжки конфіденційних даних, збору конфіденційної інформації або порушення роботи критичної інфраструктури іншої держави. Ці зловмисні дії часто включають шпигунство або кібервійну із застосуванням кіберзброї (наприклад, Stuxnet) та, як правило, мають значне фінансування та технічну базу, що робить такі загрози комплексними та важкими для виявлення.
Інсайдери	Інсайдери завдають шкоди своїм компаніям через людську помилку, наприклад, ненавмисне встановлення шкідливого програмного забезпечення або втрата пристрою, виданого компанією, який кіберзлочинець знаходить і використовує для доступу до мережі. Також інсайдером може бути незадоволений працівник який може зловживати правами доступу для отримання грошової вигоди (наприклад, оплати від кіберзлочинця чи держави) або просто зі злості чи помсти.

Кіберзлочинці зазвичай проводять наступні 10 найпоширеніших типів кібератак [12, 13]: Malware; Denial-of-Service (DoS) Attacks; Phishing; Spoofing; Identity-Based Attacks; Code Injection Attacks; Supply Chain Attacks; Insider Threats; DNS Tunneling; IoT-Based Attacks.

1. Malware

Malware (шкідливе програмне забезпечення) – будь-яка програма або код, створений з метою заподіяння шкоди комп'ютеру, мережі чи серверу. Зловмисники використовують шкідливі програми для отримання несанкціонованого доступу та виведення заражених систем з ладу, знищуючи дані, викрадаючи конфіденційну інформацію та знищуючи файли, критично важливі для операційної системи. Майже кожна сучасна кібератака пов'язана з певним типом шкідливого програмного забезпечення, головним чином тому, що цей термін охоплює багато підмножин, таких як програми-вимагачі, трояни, шпигунські програми, віруси, хробаки, кейлогери, боти та будь-який інший тип атаки, який використовує будь-яке програмне забезпечення (навіть легальне) зловмисним чином [13].

Види шкідливого програмного забезпечення (ШПЗ) представлено у таблиці 1.4.

Таблиця 1.4

Види шкідливого програмного забезпечення (створено автором на основі джерел [12, 13])

Вид	Пояснення
Ransomware	Під час атаки програм-вимагачів зловмисник шифрує дані жертви та пропонує надати ключ розшифрування в обмін на оплату. Атаки програм-вимагачів зазвичай запускаються через шкідливі посилання, що надсилаються через фішингові електронні листи, але також використовуються невиправлені вразливості та неправильні конфігурації політик. В останніх атаках на додачу до шифрування зловмисники вивантажують дані компаній та вимагають гроші за те, щоб ці дані не потрапили у публічний доступ.

Продовження таблиці 1.4

Вид	Пояснення
Fileless Malware	Безфайлове шкідливе програмне забезпечення – тип шкідливої діяльності, яка використовує вбудовані в систему легітимні інструменти для здійснення кібератаки та не потребує збереження файлів на диску, оскільки зазвичай використовується тільки в оперативній пам'яті. На відміну від традиційного шкідливого програмного забезпечення, безфайлове шкідливе програмне забезпечення не вимагає від зловмисника встановлення будь-якого коду в цільовій системі, що ускладнює його виявлення.
Spyware	Шпигунське програмне забезпечення – тип небажаного, шкідливого програмного забезпечення, яке заражає комп'ютер або інший пристрій і збирає інформацію про веб-активність користувача без його відома чи згоди.
Adware	Рекламне ПЗ – тип шпигунського програмного забезпечення, яке стежить за онлайн-активністю користувача, і, показує навязливу рекламу, що впливає на продуктивність пристрою користувача та погіршує взаємодію з користувачем.
Trojan	Троян – шкідливе програмне забезпечення, яке виглядає як легітимне програмне забезпечення і зазвичай поставляється зі зламаним софтом, замасковане під програми операційної системи або нешкідливі файли, такі як безкоштовні завантаження. Трояни встановлюються за допомогою методів соціальної інженерії, таких як фішинг або веб-сайти-приманки. Наприклад, варіант трояна Zeus, має на меті доступ до фінансової інформації та додавання комп'ютерів до ботнету.
Worms	Хробак – програма, яка поширює свої копії на інші комп'ютери автономно, без втручання людини. Хробак може заразити свою ціль через вразливість програмного забезпечення або мережевого протоколу, може бути доставлений через фішинг чи smishing, змінювати та видаляти файли, встановлювати інше ШПЗ або постійно копіювати себе, доки цільова система не вичерпає ресурси.
Rootkits	Руткіт – набір програмного забезпечення, що встановлюється в якості драйвера і приховує процеси, ключі реєстру, мережеві з'єднання іншого ШПЗ від операційної системи та легітимних програм таких як диспетчер завдань, для протидії виявленню. Руткіти йдуть ще далі, заражаючи головний завантажувач (MBR), що дає їм змогу отримувати керування ще до завантаження операційної системи, залишаючись непомітними.
Mobile Malware	Мобільне шкідливе програмне забезпечення – будь-який тип шкідливого програмного забезпечення, призначеного для атаки на мобільні пристрої. Мобільне шкідливе програмне забезпечення поширюється через шкідливі завантаження, вразливості операційної системи, фішинг, smishing та використання незахищеного Wi-Fi та іноді через офіційний магазин програм.

Продовження таблиці 1.4

Вид	Пояснення
Zero-day exploits	Експлойт – програмне забезпечення або дані, які використовують вразливості в операційній системі чи додатку для надання доступу неавторизованим особам. Експлойт може бути використаний для встановлення ШПЗ або крадіжки даних. Експлойт нульового дня – тип кібератаки, який використовує вразливість нульового дня – невідомий або ще невиправлений недолік безпеки в програмному чи апаратному забезпеченні. «Нульовий день» означає, що постачальник програмного забезпечення або пристрою немає часу для виправлення вразливостей, оскільки ще не знає про них і зловмисники вже можуть використовувати їх для отримання доступу до вразливих систем. Однією з найвідоміших вразливостей нульового дня є Log4Shell, недолік у широко використовуваній бібліотеці журналювання Apache Log4j. На момент свого виявлення в листопаді 2021 року вразливість Log4Shell існувала на 10 відсотках серверів.
Scareware / Fake AV	Шкідливе програмне забезпечення яке обманом змушує користувачів повірити, що їхній комп'ютер заражений вірусом. Зазвичай користувач бачить спливаюче вікно з попередженням про зараження системи. Ця тактика залякування має на меті переконати людей встановити підроблене антивірусне програмне забезпечення для видалення «вірусу». Після завантаження цього підробленого антивірусного програмного забезпечення ШПЗ заражає комп'ютер.
Keylogger	Кейлогери – інструменти, які записують те, що людина вводить на пристрої. Хоча кейлогери мають законне та легальне використання, багато з них є зловмисними. Кейлогер записує кожне натискання клавіші на пристрої жертви та надсилає його зловмиснику.
Botnet	Ботнет – мережа комп'ютерів, заражених шкідливим програмним забезпеченням, якими керує «керівник ботів». «Керівник ботів» — це особа, яка керує інфраструктурою ботнету та використовує скомпрометовані комп'ютери для запуску атак, спрямованих на збій мережі (DDoS), впровадження шкідливого програмного забезпечення, збору облікових даних або виконання завдань, що ресурсомісткі для процесора (майнінг криптовалют).
Malspam	Шкідливе програмне забезпечення (MALSPAM) доставляє шкідливе програмне забезпечення як корисне навантаження через електронні листи, що містять шкідливий контент, такий як вкладення, заражені вірусами або шкідливими програмами.
Wiper	Вайпер розроблений для видалення або пошкодження даних на цільових системах. Часто спостерігають у геополітичних конфліктах та в контексті хактивізму використання Ransomware без наявного ключа розшифрування, ціллю якого є не викуп а саме пошкодження даних.

Хоча атаки програм-вимагачів не є новою загрозою, за останні роки вони стали значно дорожчими: між 2018 і 2020 роками середня плата за викуп різко зросла з 5000 до 200 000 доларів. Також програми-вимагачі спричиняють

втрати доходу компаніям, допоки хакери утримують доступ до файлів з метою викупу. Середня тривалість простою системи після атаки програм-вимагачів становить 21 день [14, 15].

У звіті Sophos «Стан програм-вимагачів 2024», зазначається, що середня сума викупу зросла на 500% за останній рік. Організації, які сплатили викуп, повідомили про середню виплату в розмірі 2 мільйонів доларів, порівняно з 400 000 доларів у 2023 році. Однак викупи лише одна частина витрат. За винятком викупів, опитування показало, що середня вартість відновлення досягла 2,73 мільйона доларів, що майже на 1 мільйон доларів більше, ніж 1,82 мільйона доларів, про які Sophos повідомляла у 2023 році [16, 17].

У 2021 році, в опитуванні 1263 фахівців з кібербезпеки, 66% заявили, що їхні компанії зазнали значних втрат доходів внаслідок атаки програми-вимагача. Кожен третій сказав, що його компанія втратила вище керівництво через звільнення або відставку, а 29% заявили, що їхні компанії були змушені скоротити робочі місця після атаки програми-вимагача [18].

Атаки програм-вимагачів постійно розвиваються. Так, сучасні кіберзлочинці не розробляють своє ШПЗ з нуля, а підписуються на постачальників послуг «програми-вимагачів як послуги» (RaaS), які дозволяють користувачам розгортати попередньо розроблені інструменти програм-вимагачів для виконання атак в обмін на певний відсоток від усіх успішних виплат викупу, що знижує поріг входу в цей вид діяльності [19].

За даними Microsoft, 96,88% усіх заражень програмами-вимагачами успішно проникають у ціль менш ніж за чотири години. Найшвидше зараження, коли шкідливе програмне забезпечення може захопити систему компанії, становить менш ніж 45 хвилин [20].

Існує 5 ключових статистичних даних щодо програм-вимагачів [21]:

- Програми-вимагачі коштували світові 20 мільярдів доларів у 2021 році. Очікується, що до 2031 року ця цифра зросте до 265 мільярдів доларів.
- У 2024 році 42% усіх підприємств та організацій постраждали від програм-вимагачів.

– Відновлення після атаки програм-вимагачів коштувало підприємствам у середньому 2,85 мільйона доларів у 2024 році.

– З усіх жертв програм-вимагачів 32 відсотки сплачують викуп, але повертають лише 65 відсотків своїх даних.

– Лише 57 відсотків підприємств успішно відновлюють свої дані за допомогою резервної копії.

2. Denial-of-Service (DoS) Attacks

Атака типу «відмова в обслуговуванні» (DoS) – зловмисна цілеспрямована атака, яка заповнює мережу хибними запитами з метою порушення бізнес-операцій, роблячи веб-сайти занадто повільними для використання або повністю недоступними для легітимних користувачів. Різниця між DoS-атаками та розподіленими відмовами в обслуговуванні (DDoS) полягає в тому, що джерелом DoS-атак є лише одна система, тоді як DDoS-атаки запускаються з кількох пристроїв (ботів), що утворює ботнет [13].

3. Phishing

Фішингом називають тип кібератаки, який використовує електронну пошту, SMS, телефон, соціальні мережі та методи соціальної інженерії, щоб спонукати жертву поділитися конфіденційною інформацією, такою як паролі чи номери облікових записів, або завантажити шкідливий файл, який встановить віруси на їхній комп'ютер чи телефон [13].

Фішингові атаки та видавання себе за іншу особу електронною поштою продовжують розвиватися, охоплюючи нові тенденції, технології та тактики. Наприклад, атаки, пов'язані з криптовалютою, зросли майже на 200% між жовтнем 2020 року та квітнем 2021 року [22]. Різниця між найпоширенішими видами фішингу представлена на рис. 1.3. та у таблиці 1.5.

Phishing vs. spear phishing vs. whaling

Whaling це специфічний тип spear phishing, у свою чергу spear phishing - специфічний тип фішингу (phishing). Розглянемо відмінності між ними нижче.



Рис. 1.3. Види фішингу (створено автором на основі джерела [23])

Таблиця 1.5

Типові фішингові атаки (створено автором на основі джерел [13, 23, 24])

Тип	Пояснення
Spear Phishing	Гарпуновий фішинг – тип фішингової атаки, спрямованої на конкретних осіб або організації, зазвичай через шкідливі електронні листи. Мета фішингу – викрасти конфіденційну інформацію, таку як облікові дані для входу, або заразити пристрої цілей шкідливим програмним забезпеченням.
Whaling	Китобійна атака – тип атаки із застосуванням соціальної інженерії, спрямованої на керівників вищої ланки з метою крадіжки грошей чи інформації, або отримання доступу до комп'ютера особи для здійснення подальших кібератак.
SMiShing	Смішинг – надсилання шахрайських текстових повідомлень, призначених для того, щоб обманом вимагати від людей надання конфіденційних даних, таких як паролі, імена користувачів та номери кредитних карток.
Vishing	Вішинг (голосова фішингова атака) – шахрайське використання телефонних дзвінків та голосових повідомлень, що видаються за надсилання від авторитетної організації, щоб переконати людей розкрити конфіденційну інформацію, таку як банківські реквізити та паролі.
Business email compromise (BEC)	Шахрайство, в якому кіберзлочинці видають себе за керівників, постачальників або довірених ділових партнерів, щоб обманом змусити жертв переказати гроші або поділитися конфіденційними даними.

4. Spoofing

Спуфінгом [13] називають технологію, за допомогою якої кіберзлочинець маскується під відоме і надійне джерело. Роблячи це, зловмисник може взаємодіяти з цільовою особою та отримувати доступ до її систем або пристроїв з кінцевою метою крадіжки інформації, вимагання грошей або встановлення шкідливого програмного забезпечення на пристрій.

5. Identity-Based Attacks

Як зазначається в [25], в 80% усіх втручань використовуються скомпрометовані особисті дані, і їх ідентифікація може займати до 250 діб.

Атаки, спрямовані на ідентифікацію, надзвичайно важко виявити. Коли облікові дані дійсного користувача скомпрометовано, а зловмисник маскується під цього користувача, часто дуже важко відрізнити типову поведінку користувача від поведінки хакера, який використовує традиційні заходи та інструменти безпеки [13].

6. Code Injection Attacks

Атаки з використанням впровадження коду полягають у впровадженні зловмисником шкідливого коду у вразливий комп'ютер або мережу [13].

7. Supply Chain Attacks

Атака на ланцюг поставок – тип кібератаки, спрямованої на довіреного стороннього постачальника, який пропонує послуги або програмне забезпечення, життєво важливі для ланцюга поставок. Атаки на ланцюг поставок програмного забезпечення впроваджують шкідливий код у додаток, щоб заразити всіх користувачів додатка, тоді як атаки на апаратне забезпечення порушують фізичні компоненти з тією ж метою. Ланцюги поставок програмного забезпечення особливо вразливі, оскільки сучасне програмне забезпечення не пишеться з нуля: воно включає багато готових компонентів, таких як сторонні API, відкритий вихідний код та власний код від постачальників програмного забезпечення [13].

8. Insider Threats

Внутрішні загрози – внутрішні суб'єкти, такі як нинішні або колишні співробітники, які становлять небезпеку для організації, оскільки вони мають прямий доступ до мережі компанії, конфіденційних даних та інтелектуальної власності, а також знання бізнес-процесів, політик компанії чи іншої інформації, яка могла б допомогти здійснити таку атаку. Внутрішні суб'єкти, які становлять загрозу для організації, як правило, мають зловмисний характер. Деякі мотиви включають фінансову вигоду в обмін на продаж конфіденційної інформації в даркнеті та/або емоційний тиск з використанням тактики соціальної інженерії, такої як атаки з приводом або компрометація ділової електронної пошти (BEC). З іншого боку, деякі внутрішні суб'єкти загроз не є зловмисними за своєю природою, а радше мають недбалий характер. Щоб боротися з цим, організації повинні впровадити комплексну програму навчання з кібербезпеки, яка навчає зацікавлених сторін бути обізнаними про будь-які потенційні атаки, включаючи ті, що потенційно здійснені інсайдером [13].

9. DNS Tunneling

Тунелювання DNS – тип кібератаки, яка використовує запити та відповіді системи доменних імен (DNS) для обходу традиційних заходів безпеки та передачі даних і коду в мережі. Після зараження хакер може вільно здійснювати дії командування та управління. Цей тунель надає хакеру шлях для розповсюдження шкідливого програмного забезпечення та/або вилучення даних, IP-адрес чи іншої конфіденційної інформації шляхом кодування її побітно в серії відповідей DNS [13].

10. IoT-Based Attacks

Пристрої Інтернету речей (IoT) – такі предмети, як принтери, камери, побутова техніка, розумні годинники, трекери здоров'я, навігаційні системи, розумні замки або розумні термостати. Атака IoT – це будь-яка кібератака, спрямована на пристрій або мережу Інтернету речей (IoT). Після компрометації хакер може отримати контроль над пристроєм, викрасти дані

або приєднатися до групи заражених пристроїв, щоб створити ботнет для запуску DoS- або DDoS-атак [13].

За даними Nokia Threat Intelligence Lab [26], підключені пристрої відповідають майже за третину заражень мобільних мереж – більш ніж удвічі більше, ніж у 2019 році.

Наприклад, якщо взяти США де майже 70% домогосподарств мають хоча б один пристрій інтернету речей, атаки на цей вид пристроїв закономірно зростатимуть [27, 28].

З огляду на те, що кількість підключених пристроїв, як очікується, швидко зростатиме протягом наступних кількох років, експерти з кібербезпеки очікують також зростання кількості заражень Інтернету речей. Крім того, розгортання мереж 5G, яке ще більше стимулюватиме використання підключених пристроїв, також може призвести до зростання кількості атак [13].

Виходячи з вищенаведеного, найруйнівнішими атаками варто вважати:

Ransomware, оскільки вони мають найшвидшу конвертацію результату атаки у гроші, для традиційних АВ невидимі бо маскуються під дії користувача та використовують загальнодоступні інструменти, можуть паралізувати роботу компанії та привести до значних збитків (особливо якщо немає бекапів).

APT, оскільки достатньо кваліфіковані та фінансово вмотивовані на виконання цілеспрямованих атак, знають як функціонує система і як потрібно затирати сліди, маскують свою активність під дії користувача та використовують загальнодоступні і системні інструменти для зловмисних дій.

1.1.3. Залучення ШІ для атак на інфокомунікаційні мережі

Кіберзлочинці використовують GenAI для переконливого відтворення стилів спілкування керівників вищої ланки організації. Зазначені інструменти використовують контекстуальні дані з таких джерел, як соціальні мережі, публічні заяви чи витік документів, що робить спроби соціальної інженерії

набагато складнішими для ідентифікації. GenAI також допомагає зловмисникам розробляти надійні атаки соціальної інженерії ширшим колом мов, що допомагає зловмисникам атакувати більшу кількість людей у більшій кількості країн за меншими витратами [29].

Близько 72% респондентів повідомляють про зростання кіберризиків в організаціях, причому програми-вимагачі залишаються головною проблемою. Майже 47% організацій називають зловмисні дії, що базуються на генеративному штучному інтелекті (GenAI), що в свою чергу дозволяє здійснювати більш складні та масштабовані атаки – своєю основною проблемою. У 2024 році спостерігалось різке зростання кількості фішингових атак та атак соціальної інженерії, причому 42% організацій повідомили про такі інциденти [29].

У працях науковців Nakan T. Otal, M. Abdullah Canbaz [30], Yazı Gholami [31], Luigi Coppolino a, Salvatore D'Antonio a, Giovanni Mazzeo a, Federica Uccello [32] досліджуються великі мовні моделі (LLM – Large Language Model) та їх можливе застосування в інформаційній безпеці, оскільки LLM продовжують розвиватися, вони відіграватимуть дедалі важливішу роль у захисті цифрових систем від нових загроз. Потребує подальших досліджень залучення штучного інтелекту до вирішення задач захисту, оскільки моделі ШІ можуть використовуватись для ідентифікації, аналізу та реагування на кіберзагрози в реальному часі ще до їх виникнення.

Зростаюча складність та глобальність кіберпростору створює серйозні виклики для досягнення необхідної кіберстійкості, особливо для інфокомунікаційних мереж організацій з невеликими ресурсами.

Кібератаки на ІКМ стають усе більш витонченими й небезпечними, оскільки зловмисники постійно вдосконалюють методи впливу, зокрема і шляхом залучення штучного інтелекту (ШІ).

Загальновідомі тактики, вдосконалені штучним інтелектом, програми-вимагачі як послуга (RaaS) та передові методи соціальної інженерії дозволяють кіберзлочинцям випереджати традиційні засоби захисту. Боротьба

з цими загрозами, що стрімко розвиваються, вимагає не лише передових технологічних рішень, але й співпраці та обміну знаннями. Організації, які застосовують проактивне управління ризиками, надають пріоритет спільним підходам в різних екосистемах, свідомо використовують ШІ у своїй роботі та інвестують у масштабовані захисні рішення, можуть суттєво зменшити вплив кіберзагроз. ШІ є важливим інструментом для аналізу та прогнозування наслідків конфліктної взаємодії як для захисту власних, так і для знешкодження ворожих систем.

В умовах кібервійни найбільш ймовірними видаються впливи, спрямовані на перехоплення та спотворення даних, перехоплення управління та зміни функціонування, і навіть знищення інфокомунікаційної мережі. Зазначене обумовлює необхідність визначення потенційних загроз та сценаріїв інформаційних впливів із залученням ШІ, які можуть негативно вплинути на інформаційну безпеку інфокомунікаційної мережі, прийняття обґрунтованих рішень щодо забезпечення інформаційної безпеки, та вдосконалення методів захисту інформації.

Стрімкий розвиток технологій штучного інтелекту (ШІ), зокрема великих мовних моделей (LLM), таких як GPT-4 і BERT, та генеративних змагальних мереж (GAN), дозволили вирішувати складні завдання обробки природної мови (NLP). Змінюється сам кіберпростір, за рахунок застосування технологій ШІ як в оборонних, так і наступальних операціях. У кібербезпеці впровадження LLM відкриває нові перспективні можливості для вирішення завдань, пов'язаних із зростаючою складністю та масштабом кіберзагроз. Перспективи та особливості використання ШІ для кіберзахисту наведено у авторських статтях [33, 34, 35].

На світовому економічному форумі було розглянуто тренди у кібербезпеці на 2025 рік, що відображено на рис. 1.4, 1.5, 1.6.

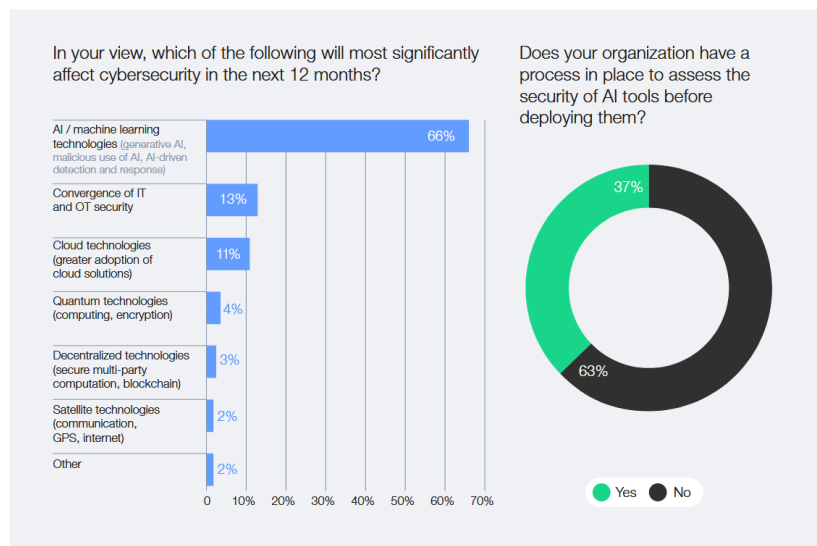


Рис.1.4. Вразливості що очікувались у 2025 році [29].

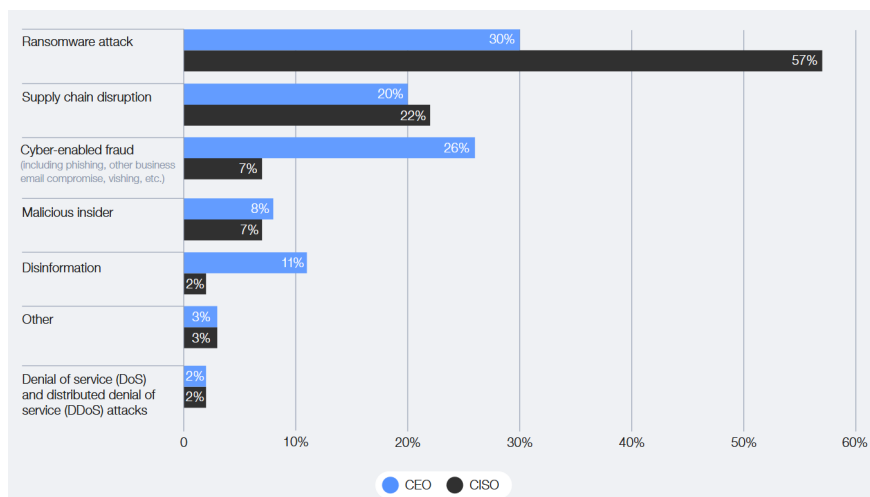


Рис.1.5. Очікувані кіберризики у 2025 році [29].

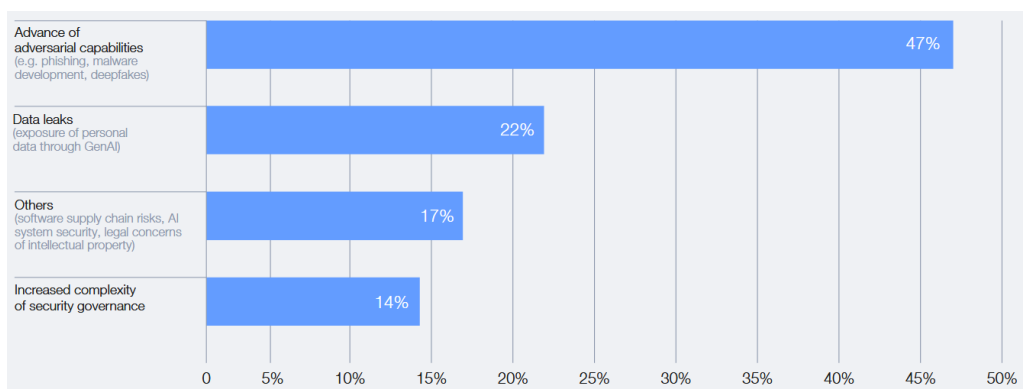


Рис.1.6. Загрози для кібербезпеки, пов'язані з GenAI [29].

Цей загальносвітовий тренд на використання ШІ у різних сферах у тому числі і для атаки на комп'ютерну інфраструктуру (weaponized AI) також корелює і з українськими даними про інциденти, оскільки однією з головних можливостей проникнути в мережу – використовувати фішингові повідомлення для викрадення облікових даних або для запуску шкідливого коду, а ШІ вже може генерувати тексти, подібні до створених людиною, правдоподібні голосові повідомлення та відео (deepfake), для більшої переконливості [36].

Наприклад, CERT-UA разом з SSSCIP знайшли 4315 кіберінцидентів у 2024, що близько на 70% більше ніж у попередньому році [37].

За даними CERT-UA, від початку 2025 року основні типи кіберзлочинної активності на українську інфраструктуру включають в себе [38]:

- Кібершпигунство, переважно у державній та військовій сферах;
- Саботаж або Кібертероризм;
- Фінансово вмотивовані злочини, спрямовані здебільшого на викрадення коштів або даних;
- Інші специфічні атаки.

Основні кіберзлочинні угруповання, що втілюють вищевказані атаки [38], наведено нижче, а також графік інтенсивності фішингу Gamaredon проти України за даними компанії ESET подано на рисунку 1.7:

- UAC-0006;
- UAC-0010 (Gamaredon, Armageddon, Primitive Bear etc.) [39, 135];
- UAC-0050;
- UAC-0184;
- UAC-0200;
- UAC-0218;
- UAC-0219.

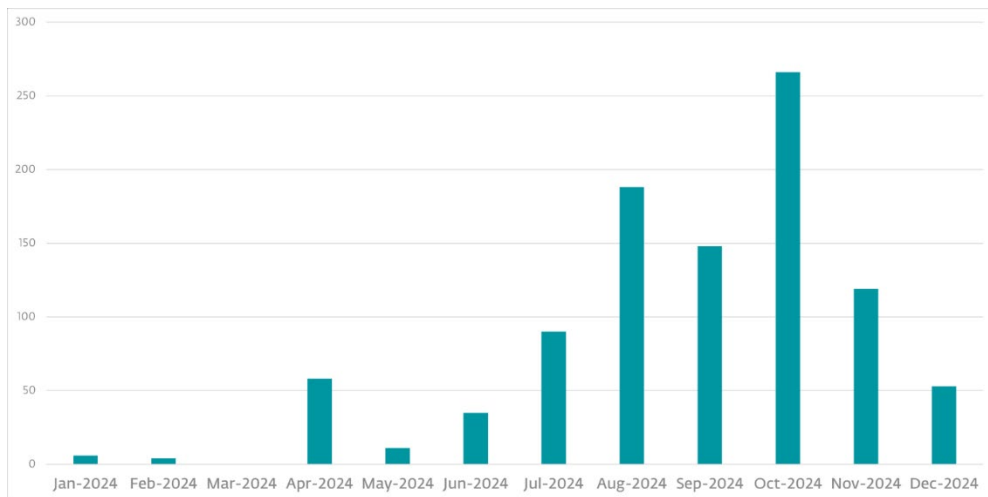


Рис.1.7. Інтенсивність фішингу Gamaredon проти України за даними компанії ESET [39].

В [32] розглядаються основні методи та прийоми, які застосовують зловмисники за допомогою генеративного ШІ для розробки складних кіберзагроз та обходу традиційних механізмів захисту. Створюється нова стратегія приховування атак із залученням ШІ: усувається критичний пропуск у методологіях приховування атак, розробляючи та оцінюючи підхід на основі умовної генеративно-мережевої мережі (CGAN). Метод дозволяє обійти систему виявлення вторгнень (IDS) на основі нейронної мережі (NN), забезпечуючи успішне проведення низки веб та мережевих атак без зміни їх фактичних сигнатур. Зокрема, серед успішно прихованих атак були: атака типу «Людина посередині» (MITM), три варіанти Mirai та SQL-ін'єкція. Автори також перевіряють цей підхід, використовуючи набори даних про атаки нульового дня, щоб оцінити його ефективність проти нових шаблонів атак [32].

Щодо безпеки паролів, у статті [40, 41], автори натренували модель GPT-2 на відомих витоках даних та словниках паролів (таких як Rockyou), і це вже перевершує наявні алгоритми для підбору паролів, що продемонстровано на рис 1.8, 1.9, 1.10. Останній навчається після того, як перший зійдеться і є необхідним для генерації. Декодери трансформаторів в обох моделях є незалежними [40].

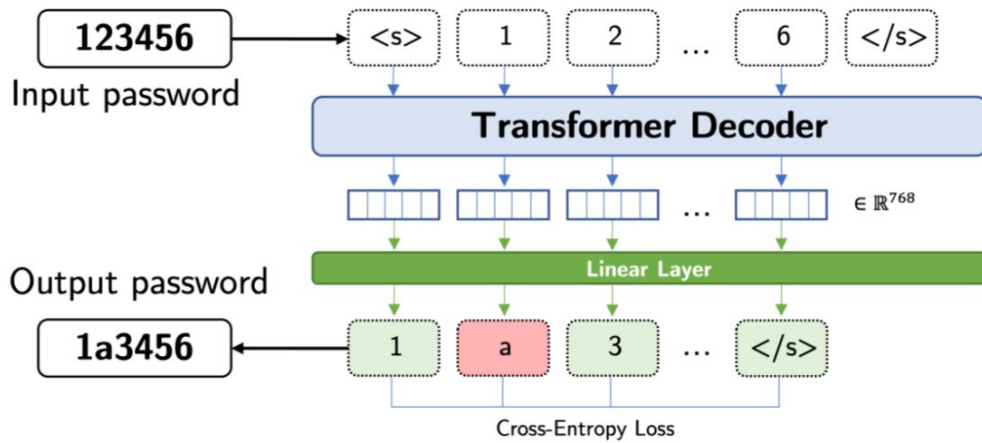


Рис.1.8. Прогнозування моделлю PassGPT вхідного символу в позиції n з використанням попередніх токенів. Зелений колір вказує на правильне прогнозування; червоний – на неправильне [40].

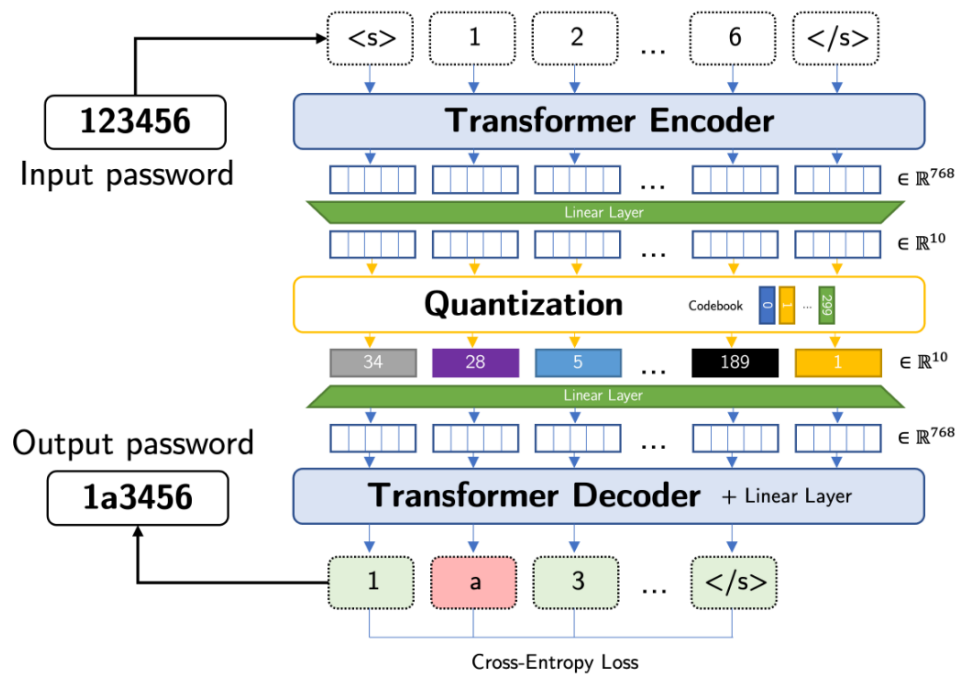


Рис.1.9. Стискання паролей в квантований латентний простір моделлю PassVQT [40].

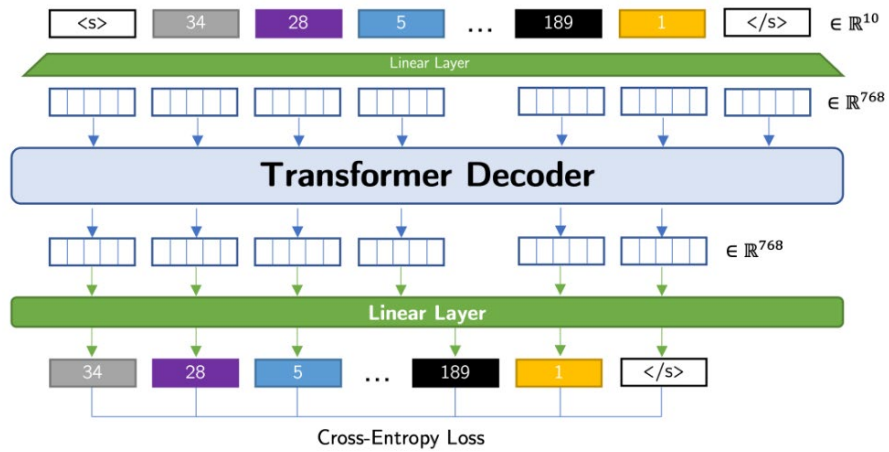


Рис.1.10. Параметризація умовного розподілу індексів моделлю PassVQT [40].

У [42] розглянуто шкідливі інформаційні впливи на кіберфізичні системи, що використовують методи машинного навчання. Для оцінки ефективності атак з ухиленням (як найбільш поширених) у цьому огляді пропонується класифікація атак, заснована на кількісних показниках, таких як рівень збурень та кількість змінених ознак, що наведена на рисунку 1.11.

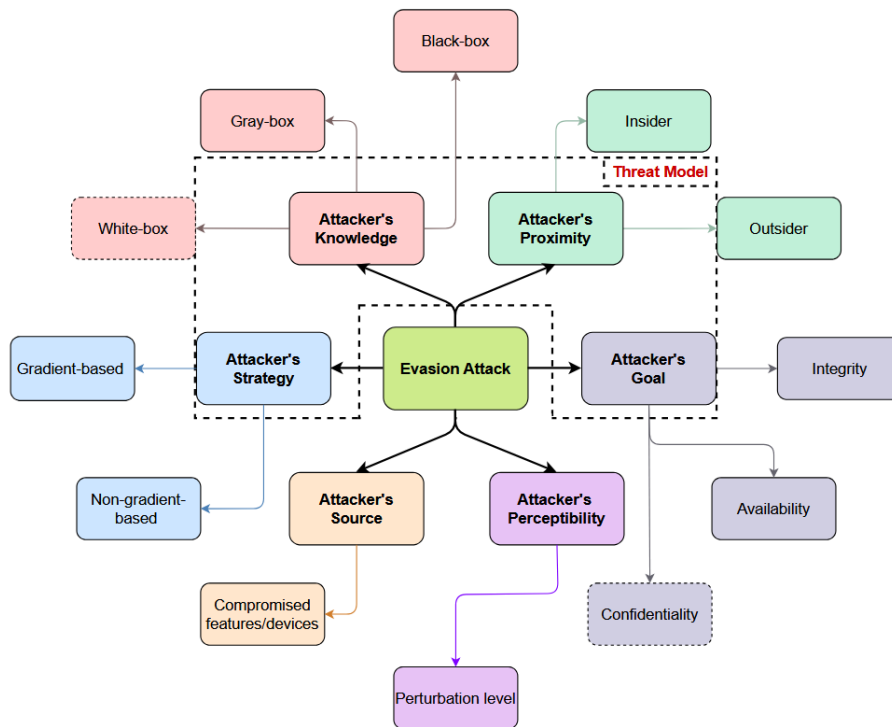


Рис.1.11. Класифікація атак [42].

Викладене надає можливість зробити висновок про те, що ШІ все більш широко залучається до формування шкідливих інформаційних впливів на інформаційні системи і починає відігравати дедалі значущу роль у сучасних кібератаках, зокрема в їх автоматизації. А також, що фішинг із залученням ШІ та технологій deepfake для створення аудіо/відеовізуального контенту, на сьогодні є одним із найбільш небезпечних видів шкідливих інформаційних впливів. Серед потенційних загроз та сценаріїв інформаційних впливів із залученням ШІ, які можуть негативно вплинути на інформаційну безпеку інфокомунікаційної мережі, найбільш небезпечними на сьогодні варто вважати ті, які, спрощують фішинг користувачів, генерують тексти подібні тим, які створює людина, правдоподібні голосові повідомлення та відео (deepfake), для більш переконливого введення в оману користувачів.

1.2. Аналіз існуючих методів захисту вузлів інфокомунікаційної мережі

Стрімкий розвиток способів та засобів шкідливого інформаційного впливу на інфокомунікаційні системи обумовив необхідність відповідного розвитку засобів захисту, покликаних виявити, ідентифікувати та нейтралізувати небажані інформаційні впливи на інфокомунікаційну мережу. Боротьба з загрозами, що стрімко розвиваються, вимагає не лише передових технологічних рішень, але й співпраці та обміну знаннями про наявні загрози та захисні рішення, що може суттєво зменшити вплив кіберзагроз.

Захист даних організації має очевидний економічний ефект у вигляді зниження ризику втрати даних клієнтів, крадіжки або знищення, а також потенційної необхідності сплати викупу для відновлення контролю над даними. Створюючи систему захисту, що зменшує поверхню кібератак та швидко реагує на них, організація мінімізує вплив небажаних подій на бізнес-процеси та захищає бренд від потенційної репутаційної шкоди [13].

Щоб захиститись від кібератак на інфокомунікаційні мережі, створено ряд технологій, частина з яких представлена у таблиці 1.6: антивірусне програмне забезпечення; системи управління доступом; засоби шифрування; сканери вразливостей; інструменти моніторингу і журнали подій; сегментація та зонування мережі; системи виявлення вторгнень; пісочниці; інструменти аудиту; брандмауери; інструменти глибокої перевірки пакетів; системи запобігання витоку інформації; інструменти перевірки цілісності даних; інструменти аналізу поведінки; інструменти керування виправленнями та змінами, тощо. Проте жодна з них не може дати 100% гарантію недопущення ризику [7, 11].

Таблиця 1.6

Технології для інформаційної безпеки (створено автором на основі джерела [11, 43])

Тип	Пояснення
Посередники, які забезпечують захищений доступ до хмари (CASB)	Точки примусового застосування політики безпеки, розташовані між корпоративними користувачами та постачальниками хмарних служб. Вони об'єднують безліч різних політик безпеки, зокрема автентифікацію, зіставлення облікових даних, шифрування, виявлення шкідливого програмного забезпечення тощо. Посередники, які забезпечують захищений доступ до хмари, працюють в авторизованих і неавторизованих програмах, а також на керованих та некерованих пристроях.
Захист від втрати даних (DLP)	Захист від втрати даних (ЗВД) включає політики, процедури, інструменти й практичні поради, які дають змогу запобігти втраті або неправомірному використанню конфіденційних даних. До ключових інструментів ЗВД належать шифрування, маркування звичайного тексту і визначення його смислових одиниць у відповідності до налаштованих правил, та використання бази даних сховища маркерів для збереження зв'язку між ними та прослідковування за шляхом переміщення конфіденційного файлу між системами.
Протидія загрозам у кінцевих точках (endpoint protection)	Протидія загрозам у кінцевих точках – рішення для захисту, яке використовує набір інструментів для виявлення й реагування на загрози на кінцевих пристроях (AV, NGAV, EDR, XDR).

Продовження таблиці 1.6

Тип	Пояснення
Мікросегментація	У процесі мікросегментації центри обробки даних діляться на кілька деталізованих і захищених зон або сегментів зі своїми правилами, що знижує рівень ризику компрометації всіх сегментів.
Перевірка системи безпеки для DevOps (DevSecOps)	DevSecOps – процес інтеграції спеціальних заходів упродовж усього періоду розробки системи безпеки. Це рішення дає змогу пришвидшити і оптимізувати процеси безпеки.
Аналітика поведінки користувачів і сутностей (АПКС / UEBA)	АПКС – процес спостереження за типовою поведінкою користувачів і виявлення підозрілих дій, що допомагає підприємствам визначати потенційні загрози.
Управління інформацією та подіями безпеки (SIEM)	SIEM допомагає організаціям розпізнавати та усувати потенційні загрози безпеці та вразливості, перш ніж вони вплинуть на роботу бізнесу. SIEM збирає дані про події з широкого кола джерел по всій IT-інфраструктурі організації, виконуючи певний агрегацію, консолідацію та сортування даних для виявлення загроз за заздалегідь налаштованими правилами і дотримання вимог щодо відповідності даних.

Нижче наведено рекомендації, які пропонує CrowdStrike у Звіті про глобальні загрози за 2023 рік, щоб допомогти організаціям покращити свій рівень безпеки [44]:

- *Захист усіх робочих навантажень*: необхідно захищати всі критичні області корпоративного ризику, включаючи кінцеві точки та хмарні робочі навантаження, ідентифікаційні дані.
- *Знайте свого супротивника*: необхідно аналізувати та виявляти сучасних зловмисників та їхні сценарії, щоб відділ інформаційної безпеки проактивно оптимізував свою роботу, посилював кіберзахист та знижував час реагування на інциденти.
- *Будьте готові, коли кожна секунда на рахунку*: Команди безпеки повинні інвестувати в швидкість та гнучкість прийняття рішень, автоматизувати процеси запобігання, виявлення, розслідування та реагування за допомогою інтегрованої розвідки про актуальні кіберзагрози.
- *Впровадження моделі нульової довіри*: Оскільки сьогодення вимагає доступу до даних з будь-якого місця та в будь-який час, критично важливо впровадити модель нульової довіри.

- *Моніторинг кримінального підпілля:* Кіберзлочинці об'єднуються для співпраці, використовуючи різноманітні платформи прихованих повідомлень та форуми даркнету. Важливо використовувати цифрові інструменти моніторингу ризиків для визначення неминучих загроз бренду та даним.
- *Інвестуйте в елітне полювання на загрози:* Поєднання технологій з експертами, що полюють на загрози (Bug Bounty) є абсолютно необхідним для виявлення та зупинки найскладніших загроз.
- *Створіть комплексну програму навчання з кібербезпеки:* Слід запровадити програми підвищення обізнаності користувачів для боротьби з постійною загрозою фішингу та пов'язаних з ними методів соціальної інженерії.

Надійні паролі, інструменти захисту електронної пошти та антивірусне програмне забезпечення є критично важливою першою лінією захисту від кіберзагроз. Підприємства можуть покладатися на брандмауери, VPN, багатофакторну автентифікацію (2FA, MFA), навчання з питань безпеки та інші передові рішення для захисту кінцевих точок та мережевої безпеки для захисту від кібератак. Однак жодна система безпеки не є повноцінною без виявлення загроз та реагування на інциденти (SIEM), що дозволяє виявляти і швидко ізолювати кіберзагрози, щоб мінімізувати або запобігти потенційній шкоді [12, 43].

Розвиток засобів та способів інформаційного впливу на інфокомунікаційні системи призвело до відповідного розвитку засобів захисту, покликаних виявляти, ідентифікувати та нейтралізувати шкідливі інформаційні впливи на ІКМ. Вже існує кілька підходів і конкретні методики для виявлення інформаційних впливів, проте завдання їхнього порівняльного аналізу в контексті критерію ефективності виявлення апріорно невідомих впливів різного типу залишається актуальним.

Щоб відповідати сучасним вимогам безпеки, ІКМ та інформація, яка в ній циркулює, повинна бути захищена від порушення цілісності, конфіденційності, доступності. Функціонування шкідливих програмних засобів (ШПЗ), методи їх виявлення та способи ліквідації розглядали такі

вчені, як М.Д. Василенко, В.О. Рачук, В.М. Слатвінська [45], Д.О. Ричка [46]. У дослідженні М.Д. Василенко, В.О. Рачук і В.М. Слатвінська [45] висвітлюється тема виявлення шкідливих програм в контексті вимог комп'ютерної безпеки. У праці Д.О. Ричка [46] розглянуто комплекс заходів щодо захисту бази даних від кіберзагроз і визначено напрямки боротьби із комп'ютерними вірусами.

Як основне рішення ешелонованого захисту в корпоративному середовищі для кінцевих точок від впливу комп'ютерних вірусів досі залишається антивірус (Endpoint Protection). Антивірусні модулі обов'язково є в рішеннях класу Endpoint Detection and Response (EDR) & Next Generation Anti-Virus (NGAV).

Для виявлення інформаційних впливів на систему можуть використовуватися два взаємно доповнюючі підходи: сигнатурний та евристичний (поведінковий).

Структура організації автоматичної системи виявлення слідів вірусної активності в ІКМ подана на рис. 1.12.

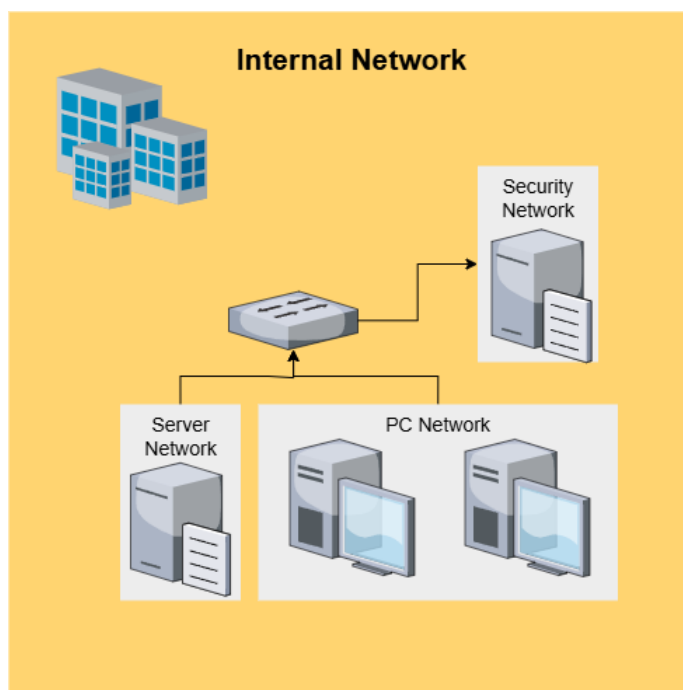


Рис. 1.12. Структура організації автоматичної системи виявлення слідів вірусної активності в ІКМ (створено автором)

Переваги та недоліки сигнатурних та поведінкових методів наведені у табл. 1.7 та 1.8.

Таблиця 1.7

Сигнатурний метод (створено автором)

Метод	Переваги	Недоліки
Сигнатурний	Може швидко та ефективно знаходити відомі віруси Займає менше часу на аналіз Має невелику кількість хибних спрацювань Має можливість задати точні сигнатури під окреме сімейство ШПЗ	Не може знаходити невідомі віруси або ті, що трохи відрізняються від заданої сигнатури Потрібна велика і періодично оновлювана база сигнатур Вірусам дуже просто обійти сигнатурний аналіз (поліморфізм)

Сигнатурні методи – методи виявлення шкідливих впливів на основі сигнатур, які містять сигнатури (шаблони) типових впливів або сімейств ШПЗ, створені на основі послідовності даних, що однозначно може ідентифікувати загрозу. Багато сигнатур потребують більше обчислень для цього методу та принципово не можуть знайти нові ще не додані до сигнатур впливи.

Таблиця 1.8

Евристичний метод (створено автором)

Метод	Переваги	Недоліки
Евристичний (поведінковий)	Може знаходити невідомі віруси Не потрібна велика база сигнатур (лише список правил) Чутливий до зміну стану ПК та активності користувача	Має більше хибно-позитивних спрацювань порівняно з сигнатурним методом Важчий для системи, довше проводить аналіз Вірус який є відомим але не повністю підпадає під правила може бути пропущений

Різноманіття можливих шкідливих інформаційних впливів, створює ситуацію, що вплив спершу може бути невідомим системі захисту. Таким

чином, у системі захисту обов'язково має бути реалізований механізм аналізу поведінки вузла ІКМ.

Поведінкові методи – використовують не чіткі сигнатури а набір змінних правил, моніторять аномальну поведінку програм, ОС, тощо. Іноді базуються не на пошуку конкретної атаки, а на виявленні розбіжностей і аномалій на ПК між поточним режимом роботи порівняно з його звичайним нормалізованим функціонуванням. Таким чином, будь-яка невідповідність сприймається як аномалія або атака. Складність цього методу полягає у неможливості точно визначити еталонний стан ПК чи поведінки користувача, тому якщо зломисник буде підлаштовуватись під користувацьку чи адміністраторську активність (наприклад, використовувати системні програми для зломисних дій) його буде складно виявити.

Оскільки кожен з методів сам по собі має недоліки, які частково закриваються використанням іншого методу, у сучасних захисних системах зазвичай використовуються комбінований метод – поєднання сигнатур, евристичних правил, та специфічних параметрів (наприклад, аналіз репутації файлів, перевірка цифрових підписів, тощо).

Висновки до розділу 1

На основі викладеного можна зробити наступні висновки:

1. Найруйнівнішими сучасними атаками на ІКМ варто вважати:

- Ransomware, оскільки вони мають найшвидшу конвертацію результату атаки у гроші, не виявляються традиційними АВ, оскільки маскуються під дії користувача та використовують загальнодоступні інструменти, можуть паралізувати роботу компанії та призвести до значних збитків (особливо якщо немає бекапів).

- АРТ, оскільки достатньо кваліфіковані та фінансово мотивовані на виконання цілеспрямованих атак, знають як функціонує система і як потрібно

затирати сліди, маскують свою активність під дії користувача та використовують загальнодоступні і системні інструменти для зловмисних дій.

- Фішинг із залученням ШІ та технологій deepfake для створення аудіо/відеовізуального контенту, оскільки можуть більш переконливо вводити в оману користувачів.

2. До формування шкідливих інформаційних впливів на інформаційні системи все більш широко залучаються технології штучного інтелекту.

3. Традиційні рішення захисту вузлів ІКМ можуть пропускати комп'ютерні віруси внаслідок недосконалості механізмів їх виявлення, що особливо критично під час багаторівневих кібератак. Як основне рішення ешелонованого захисту в корпоративному середовищі для кінцевих точок від впливу комп'ютерних вірусів досі залишається антивірус (Endpoint Protection).

4. Стрімкий розвиток і постійне вдосконалення ШПЗ та кібератак обумовлює необхідність проведення досліджень щодо синтезу нового методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів.

5. Перспективними є методи захисту, засновані на виявленні та аналізі цифрових слідів, які залишаються після ШПЗ або АРТ-атаки, здатних виявляти шкідливу активність навіть тоді, коли всі інші методи захисту не ідентифікували наявну загрозу.

6. Перспективним і актуальним напрямком забезпечення захисту вузлів ІКМ є застосування моделей ШІ, оскільки вони забезпечують більш оперативне реагування на виникаючі інциденти інформаційної безпеки, аналізуючи поточні тенденції в поведінці мережі або користувачів ІКМ. Дослідження варто зосередити на тестуванні наявних нейромережевих моделей для оцінки можливості їх застосування при вирішенні задач захисту інфокомунікаційних мереж.

7. Ефективність протидії поширенню комп'ютерних вірусів в інфокомунікаційній мережі доцільно визначати як економію часу і ресурсів на виявлення комп'ютерних вірусів та протидію їх поширенню у вузлах інфокомунікаційній мережі.

Аналіз ІКМ, представлений у розділі, знайшов відображення в роботі автора [9].

Постановка наукових завдань дослідження

Метою дисертаційної роботи є синтез методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, здатного забезпечити підвищення ефективності протидії поширенню комп'ютерних вірусів в інфокомунікаційній мережі.

Для досягнення поставленої мети необхідно вирішити наступні наукові завдання:

1. Здійснити синтез методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, здатного забезпечити економію часу і ресурсів на виявлення комп'ютерних вірусів та протидії їх поширенню в інфокомунікаційній мережі.

2. Обґрунтувати доцільність використання цифрових слідів у якості основної ідентифікаційної ознаки для оцінки стану вузлів ІКМ.

3. Розробити модель для оптимізації кількості і розміру цифрових слідів, достатніх для ідентифікації стану вузлів ІКМ.

4. Шляхом тестування здійснити відбір релевантних нейромережевих моделей для аналізу вивантажених цифрових слідів.

5. Удосконалити метод вивантаження цифрових артефактів в умовах обмеженості ресурсів.

6. Виконати експериментальну перевірку ефективності запропонованого методу захисту вузлів ІКМ.

РОЗДІЛ 2. СИНТЕЗ МЕТОДУ ЗАХИСТУ ВУЗЛІВ ІКМ ВІД КОМП'ЮТЕРНИХ ВІРУСІВ НА ОСНОВІ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ

2.1. Системний підхід як основа методу захисту вузлів інфокомунікаційної мережі

Інфокомунікаційну мережу можна класифікувати як складну динамічну багатопараметричну слабодетерміновану систему з розподіленими параметрами, в якій можуть виникати несподівані емерджентні властивості в результаті впливу комп'ютерних вірусів. Завдання забезпечення якісного функціонування інфокомунікаційних мереж в умовах впливу комп'ютерних вірусів та кібератак обумовлює необхідність контролю стану мережі для своєчасного виявлення тривожних подій під якими розуміється виявлення ознак зараження вузлів мережі. Для досягнення необхідного рівня захищеності інформації має бути забезпечена здатність передбачати та запобігати появі цих властивостей на об'єкті захисту.

Вирішення такої задачі досягається технологією захисту інформації у вигляді цілеспрямованих *циклів управління*, що забезпечує підтримку цільового стану об'єкта управління (ОУ). При цьому мета управління може бути представлена як стабілізація стану ОУ або переведення ОУ в деякий кінцевий (цільовий) стан, при якому забезпечується необхідний рівень захищеності інформації. За інформаційною сутністю процес управління формалізується у вигляді замкнутого циклу послідовного звернення до трьох операторів: ідентифікації об'єкта управління (ОУ), прийняття рішень про характер і величину керуючих впливів, що забезпечують досягнення поставленої мети, та здійснення керуючих впливів (рис. 2.1). При цьому під ідентифікацією ОУ розуміється отримання інформації, необхідної і достатньої для ухвалення рішення в циклі управління.



Рис. 2.1. Інформаційна структура циклу управління ІКМ (створено автором)

Мета управління досягається забезпеченням якісного функціонування всіх операторів у кожному циклі управління. Основою ж для прийняття вірних керуючих впливів на ОУ є достатність і достовірність інформації при ідентифікації стану ОУ. Межа між тим, що вважається навколишнім середовищем і тим, що вважається об'єктом управління відносна і визначається можливістю системи управління здійснювати на них вплив.

Основна проблема, яка проявляється при синтезі системи управління складним динамічним об'єктом – невизначеність і недостатність апріорної вхідної інформації про об'єкт управління, наявність невідомих факторів, що суттєво впливають на його поведінку і, як наслідок, проблематичність побудовання змістовної аналітичної моделі складного об'єкта управління. Тому, замість параметричного управління вводиться більш широке поняття «управління за станом складного об'єкта управління» [47]. При цьому, в граничному випадку, коли зв'язок вихідних параметрів і станів об'єкта управління має однозначний і детермінований характер, управління за станами зводиться до управління за параметрами.

Згідно з викладеним, основними задачами для створення системи захисту інформації, побудованої за принципом керуючої системи, є:

- визначення цільового стану ОУ, що забезпечує необхідний рівень захисту інформації;
- синтез моделі ОУ, здатної забезпечувати генерацію такого набору керуваних факторів, що їх фактичний вплив на ОУ в циклі управління з максимально можливою вірогідністю переведе його в заздалегідь заданий цільовий стан. При цьому модель повинна враховувати поточний стан ОУ і вплив наявних і прогнозованих неконтрольованих факторів середовища на поведінку ОУ;
- визначення достатнього набору вихідних параметрів ОУ для достовірної ідентифікації його станів. При цьому важливо знати їх імовірнісні характеристики (математичне очікування, кореляційну функцію та ін.) за якими можливо визначати крок квантування контрольованого параметра з урахуванням швидкості затухання інформаційних зав'язків, допустимих втрат інформації, запізнення і частоти послідовного контролю;
- синтез методу прийняття рішень, що забезпечує вибір керуючих дій в контурі управління спрямованих на оперативні локалізацію і нейтралізацію загроз.

2.2. Обґрунтування доцільності використання цифрових слідів для оцінки стану вузлів ІКМ

При викладенні матеріалу даного розділу використано наступні поняття:
NTFS – (англ. New Technology File System, «файлова система нової технології») – файлова система за-замовчуванням для ОС Windows.

MFT – (англ. Master File Table, «Головна файлова таблиця») – база даних, в якій зберігається інформація про вміст тома диску з файловою системою NTFS, що представляє собою таблицю, де рядки відповідають файлам тома, а стовпці відповідають атрибутам файлів.

Артефакт – фрагмент інформації, який може бути використаний для аналізу інцидентів та збирання доказів у судових справах. Це можуть бути

окремі файли на комп'ютері, журнали подій, історія браузера, метадані відео/зображень та ін.

Форензіка – комп'ютерна криміналістика (computer forensics) це галузь цифрової криміналістики, яка стосується доказів, знайдених у комп'ютерах і цифрових носіях інформації. Метою комп'ютерної криміналістики є дослідження цифрових носіїв криміналістично обґрунтованим способом з метою виявлення, збереження, відновлення, аналізу та представлення фактів і думок щодо наявної цифрової інформації.

Типи артефактів – те, що може згрупувати артефакти без урахування того, в якій формі вони представлені. Наприклад, тип «Журнал подій», представлений у вигляді файлів журналів ОС спеціального формату .evtx, окремих журналів програм у текстовому форматі.

Дамп пам'яті – (англ. dump) вміст робочої пам'яті процесу, ядра чи всієї операційної системи в певний момент часу. Зазвичай зберігається у файлах спеціального формату .dmp, .raw

Корінь диску / Кореневий каталог – (англ. root directory) каталог файлів, що знаходиться на вершині ієрархії всіх інших каталогів (зазвичай кореневим каталогом диску ОС Windows буде C:\). Процес пошуку місця зберігання будь-якого іншого файлу або каталога починається з кореневого каталога.

Розгляд і оцінка слідів, які залишаються після ШПЗ або АРТ-атаки допомагає виявляти шкідливу активність навіть тоді, коли всі інші методи захисту не ідентифікували наявну загрозу. Класифікація загроз дозволяє виявити стійкі патерни атак, які не змінюються з часом (наприклад, якщо брати традиційні АВ, визначення вірусу за хешами буде неефективним, оскільки хеш у кожного нового ШПЗ буде різний, навіть якщо вихідний код однаковий, але якщо взяти та побудувати «сигнатуру ШПЗ», то зміна хешів ніяк не буде впливати на детектування антивірусом цього варіанту ШПЗ) і за рахунок цього впровадити захист, націлений саме на ці патерни, з урахуванням попереднього досвіду (наприклад, якщо є інформація, що для вивантаження паролів використовується утиліта Mimikatz, у будь-якому файлі/папці/назві і

ми не можемо це видалити сигнатурно, але при цьому аргументи до цієї програми будуть завжди однакові, тому ми можемо налаштувати свій АВ на блокування будь-якого файлу, якщо він містить специфічні аргументи, які використовує саме ця утиліта, що убереже нас від будь-яких поточних та майбутніх варіацій ШПЗ, які містять у собі команди для запуску такої утиліти).

У працях науковців Diana Hinteа, Robert Bird, Michael Green [48] Marcus K. Rogers, James Goldman, Rick Mislan, Timothy Wedge, Steve Debrota [49], Jusas V., Birvinskas D., Gahramanov E. [50], Abd El-Latif та інші [134] пропонується перелік артефактів Windows 10 у цифровій криміналістиці, загальні місця їх розташування, та огляд моделі Cyber Forensic Field Triage Process Model (CFFTPM). Автори Muhammad Bashir, Mirwais Khan [51] та Da-Yu Kao, Guan-Jie Wu [52] розглянули можливі рішення для вивантаження артефактів з ПК, проведення Live Computer Forensics та визначення місць розташування цифрових артефактів. Разом з тим, авторами не вказуються конкретні місця розташування артефактів у ОС Windows, ранжування від більш важливих до менш важливих артефактів, а також не враховуються ситуації, у яких не можливий збір повного спектру артефактів. Потребує подальших досліджень розробка нових підходів до створення сучасного механізму захисту комп'ютерного середовища та інформації від шкідливих програм і кібератак, а також вдосконалення наявних методів ідентифікації ШПЗ на ПК. Перспективним шляхом є ефективний розподіл часу та ресурсів спеціаліста інформаційної безпеки (ІБ-спеціаліста) для своєчасного виявлення та запобігання загрозам.

Ідентифікаційні ознаки об'єктів прийнято поділяти на детерміновані (логічні) і стохастичні (імовірнісні). Логічні ознаки можна розглядати як елементарні висловлювання, що приймають два значення істинності виду «так», «ні», або «істина» і «неправда» з повною визначеністю. До числа імовірнісних відносяться ознаки, випадкові значення яких розподілені по всіх класах об'єктів, тому рішення про належність об'єкта до того чи іншого класу може прийматися тільки на основі конкретних значень ознак об'єкта.

Стани C_1 і C_2 ідентифікаційної ознаки не можуть бути достовірно передбачені заздалегідь, тому можна вважати, що вони відповідають гіпотезам H_1 і H_2 про знаходження ідентифікаційної ознаки у станах C_1 і C_2 .

Таким чином, завданням є ототожнення кожної конкретної реалізації ідентифікаційної ознаки зі станом C_1 і C_2 ідентифікаційної ознаки, тобто встановлення істинності гіпотези H_1 і H_2 .

У двоальтернативній ситуації прийняття рішень завжди супроводжується помилками першого і другого роду [53]:

- помилка першого роду (хибна тривога) – гіпотеза H_1 відкидається тоді, коли насправді вона вірна;

- помилка другого роду (пропуск цілі) – відкидається гіпотеза H_2 тоді, коли вона вірна.

Зазначені помилкові рішення характеризуються відповідно ймовірностями $P_{\text{пом.1}}$ і $P_{\text{пом.2}}$. Ухвалення правильних рішень про стани ідентифікаційної ознаки при знаходженні її в станах C_1 і C_2 характеризується відповідно ймовірностями $P_{\text{в1}}$ і $P_{\text{в2}}$, причому

$$P_{\text{в1}} + P_{\text{пом.1}} = 1 \quad (1)$$

$$P_{\text{в2}} + P_{\text{пом.2}} = 1 \quad (2),$$

оскільки можливі результати, що відповідають кожному із станів ідентифікаційної ознаки, становлять повну групу подій.

Завданням вирішального пристрою є виявлення одного із станів ідентифікаційної ознаки відповідно до введеного в вирішальний пристрій критерію K (правилом прийняття рішення). З двох гіпотез H_1 і H_2 вирішальний пристрій вибирає ту, яка в заданому сенсі краще. Як критерій прийняття рішення може бути обраний критерій «ідеального спостерігача» (критерій Котельникова), який є окремим випадком критерію мінімального середнього ризику Байеса. Зазначений критерій мінімізує можливість сумарної помилки прийняття рішення

$$P_{\text{пом.}} = P_a(C_1) \int_{U_n}^{\infty} f_1(U) dU + P_a(C_2) \int_{-\infty}^{U_n} f_2(U) dU \quad (3)$$

де $P_a(C_1)$ и $P_a(C_2)$ – апіорні ймовірності знаходження ідентифікаційної ознаки у станах C_1 та C_2 (присутня чи відсутня конкретна ознака на об'єкті); $f_1(U)$ і $f_2(U)$ – функції густин ймовірностей випадкових значень інформативної ознаки, що відповідають станам C_1 і C_2 ідентифікаційної ознаки (рис. 2.2); $U_{\text{п}}$ – поріг прийняття рішення вирішального пристрою.

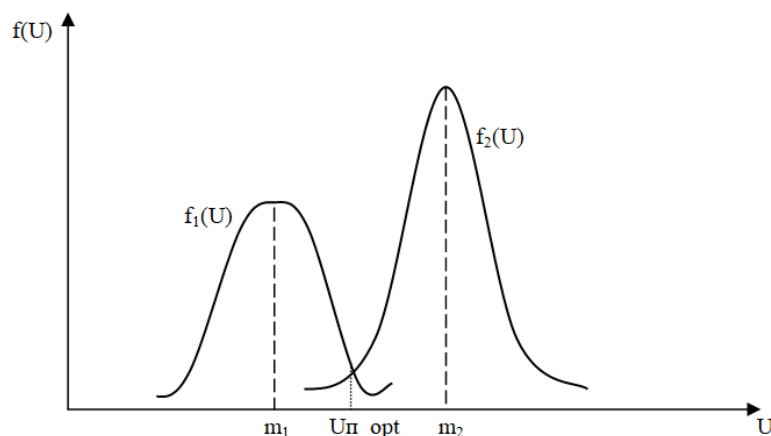


Рис. 2.2. Графіки умовних густин розподілу інформативної ознаки

Перевагою даного критерію є можливість роздільного обліку ймовірностей помилок першого та другого роду, які визначаються виразами

$$P_{\text{пом.1}} = \int_{U_n}^{\infty} f_1(U) dU \quad (4)$$

$$P_{\text{пом.2}} = \int_{-\infty}^{U_n} f_2(U) dU \quad (5),$$

що є дуже важливим, оскільки у загальному випадку ймовірності цих помилок можуть істотно відрізнятися. Окрім того, аналітичний вираз для критерію такий, що в нього можуть бути введені величини, які характеризують як параметри випадкових значень інформативної ознаки, так і основні параметри вирішального пристрою, що дозволяє зробити їх раціональний вибір і визначити точні характеристики, що забезпечують максимальну або задану достовірність результатів ідентифікації.

В якості ідентифікаційних ознак стану ІКМ для виявлення тривожних подій доцільно використовувати цифрові артефакти, які залишає шкідливе програмне забезпечення в результаті впливу та методи їх ідентифікації.

Для проведення повноцінного forensic-аналізу необхідно мати повний віртуальний образ диску та оперативної пам'яті досліджуваного комп'ютера, фізичний доступ до нього, а також мати достатню кваліфікацію та час на аналіз, що не завжди є можливим у корпоративному середовищі. Додатково до цього, корпоративні ПК мають невизначений термін підключення до мережі, а до аналітика зазвичай ставиться конкретний перелік питань, які не завжди потребують збору усіх наявних артефактів. У цьому випадку функціонал програм Forensic Triage (частковий збір артефактів з активного ПК) не є оптимальним для вирішення завдань аналітика.

Викладене обумовлює необхідність пришвидшення збору необхідних артефактів з ОС Windows ІБ-спеціалістами та ІТ-адміністраторами для проведення Forensic Triage в умовах, де неможливо отримати повний віртуальний образ диску і оперативної пам'яті досліджуваного комп'ютера, або фізичного доступу до нього, на відміну від ситуацій, де у аналітика є достатній час на підключення до досліджуваного ПК. Словник ознак для ідентифікації стану ІКМ є достатнім для прийняття рішень в циклах управління технології захисту інформації. Практичне значення отриманих результатів полягає у можливості їх застосування для вдосконалення методів захисту інформації та більш ефективного використання часу аналітика при проведенні Forensic-аналізу ІКМ.

Зазвичай, для проведення повноцінного forensic-аналізу необхідно мати повний віртуальний образ диску та оперативної пам'яті досліджуваного комп'ютера, фізичний доступ до нього, а також мати достатню кваліфікацію та час на аналіз, що не завжди є можливим у корпоративному середовищі. Додатково до цього, корпоративні ПК мають невизначений термін підключення до мережі, а до аналітика зазвичай ставиться конкретний перелік питань, які не завжди потребують збору усіх наявних артефактів. У цьому випадку функціонал програм Forensic Triage (частковий збір артефактів з активного ПК) не є оптимальним для вирішення завдань аналітика. Тож необхідно пришвидшити збір необхідних артефактів з ОС Windows ІБ-

спеціалістами та ІТ-адміністраторами в умовах, де неможливо отримати повний віртуальний образ диску та оперативної пам'яті досліджуваного комп'ютера або фізичний доступ до нього (та провести Forensic Triage), на відміну від ситуацій, де у аналітика є вичерпний час на підключення до досліджуваного ПК. Важливим є визначення основних місць розташування артефактів у ОС Windows, яких буде достатньо для вирішення більшості forensic-кейсів у корпоративному середовищі.

У 2024 році найбільш популярною ОС була Windows 10 x64 а у 2026 році вже Windows 11 x64 з файловою системою NTFS, що подано на рис. 2.3, 2.4.

ОБЪЕКТ	САМЫЙ ПОПУЛЯРНЫЙ	ДОЛЯ	ДИНАМИКА
Версия ОС	Windows 10 64 bit	53.45%	-0.08%
System RAM	16 GB	48.53%	-1.35%
Intel CPU Speeds	2.3 Ghz to 2.69 Ghz	21.02%	-0.13%
Physical CPUs	6 cpus	32.11%	+0.23%
Video Card Description	NVIDIA GeForce RTX 3060	5.13%	+0.24%
VRAM	8 GB	31.67%	+0.44%
Primary Display Resolution	1920 x 1080	59.58%	-0.51%
Multi-Monitor Desktop Resolution	3840 x 1080	59.02%	-1.00%
Language	English	36.92%	+0.90%
Free Hard Drive Space	100 GB to 249 GB	22.76%	-0.70%
Total Hard Drive Space	Above 1 TB	52.75%	+0.76%
VR Headsets	Oculus Quest 2	37.87%	-2.58%
Other Settings	LAHF / SAHF	100.00%	0.00%

Рис. 2.3. Перелік популярних конфігурацій комп'ютерів за 2024 рік [54].

ОБЪЕКТ	САМЫЙ ПОПУЛЯРНЫЙ	ДОЛЯ	ДИНАМИКА
Версия ОС	Windows 11 64 bit	66.85%	+10.57%
System RAM	16 GB	40.97%	+13.50%
Intel CPU Speeds	2.3 Ghz to 2.69 Ghz	20.38%	-1.14%
Physical CPUs	6 cpus	27.77%	-2.10%
Video Card Description	NVIDIA GeForce RTX 3060	3.92%	-0.57%
VRAM	8 GB	27.52%	-1.85%
Primary Display Resolution	1920 x 1080	51.93%	+6.89%
Multi-Monitor Desktop Resolution	3840 x 1080	49.08%	-0.06%
Language	English	39.09%	+16.82%
Free Hard Drive Space	100 GB to 249 GB	22.98%	+6.98%
Total Hard Drive Space	Above 1 TB	50.11%	-19.98%
VR Headsets	Meta Quest 3	27.66%	-0.89%
Other Settings	CMFXCHG16B	98.09%	-0.99%

Рис. 2.4. Перелік популярних конфігурацій комп'ютерів за 2026 рік [54].

Криміналістичні артефакти в операційній системі Windows загалом можна розділити на чотири основні типи (категорії):

- Реєстр (Windows Registry)
- Файлова система (Filesystem NTFS)
- Журнал подій (Event Log / Event viewer)
- Пам'ять (RAM / ROM Memory)

Артефакти реєстру знаходяться в реєстрі Windows, який завантажується в пам'ять під час роботи системи та записується на диск під час завершення роботи. Реєстр зберігає параметри конфігурації низького рівня для операційної системи та містить велику кількість криміналістичних артефактів [55].

Оскільки дані реєстру з ключів HKEY_LOCAL_MACHINE та HKEY_USERS містять у собі всі наявні значення (інші ж являються лише посиланнями на них), вони більше всього цікавлять ІБ-аналітика (рис. 2.5).

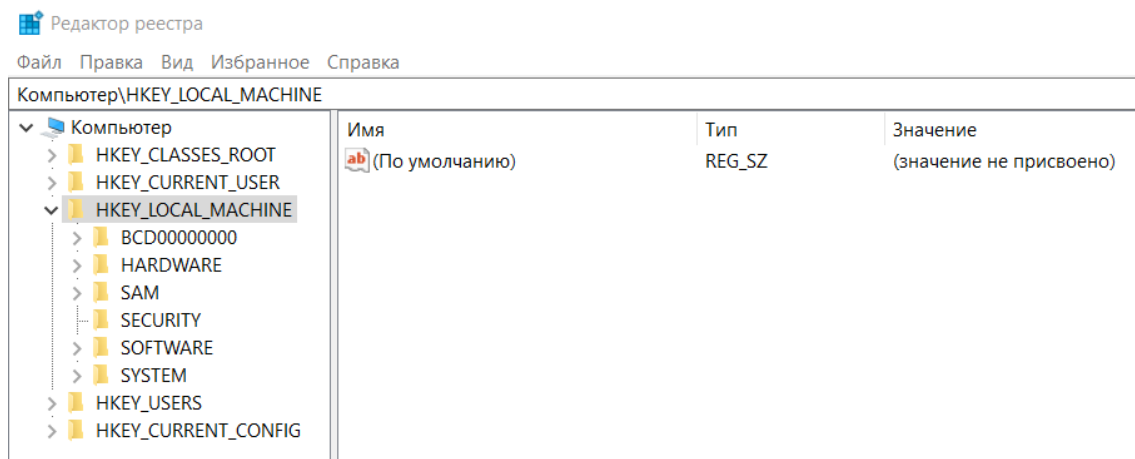


Рис. 2.5. Реєстр Windows у вікні «Редактора Реєстру»

Реєстр знаходиться за шляхом C:\Windows\System32\config у наступних файлах: SAM, SECURITY, SOFTWARE, SYSTEM [56].

З них ми зможемо отримати дані HKEY_LOCAL_MACHINE, проте з включеної системи ми не зможемо скопіювати ці файли, тому треба

скористатися командою “reg save” та вивантажити реєстр у вигляді файлу до іншого місця або використовувати програми, що працюють з тінюваними копіями файлів для можливості копіювання заблокованих файлів.

Одного файлу, котрий містив би дані HKEY_USERS, не існує, тому треба збирати дані реєстру окремо про кожного користувача за шляхом C:\Users\%username%\NTUSER.DAT та C:\Users\%username%\NTUSER.DAT.LOG*, оскільки не всі системи для аналізу даних реєстру (наприклад, Eric Zimmerman’s Registry Explorer) коректно його відображають без .LOG файлів.

Registry Explorer – інструмент на основі графічного інтерфейсу користувача, який використовується для перегляду вмісту автономних кущів реєстру. Має можливість завантаження кілька кущів одночасно, пошук у всіх завантажених кущах за допомогою рядків або регулярних виразів, має вбудовані шаблони, експорт даних і багато іншого [57].

Артефакти файлової системи —артефакти, які виникають через роботу файлової системи Windows (NTFS) [55]. Основним файлом, що містить необхідні нам артефакти, буде Master File Table – файл \$MFT, оскільки він містить інформацію про всі файли з timestamp, що є на диску. Дані з нього можна витягти за допомогою ntfswalk або скриптів Powershell [58]. Також, для більш глибокого аналізу можуть знадобитися самі файли .exe (наприклад, з папки Автозапуску), але місця їх розташування можна дізнатися лише після попереднього аналізу.

Артефакти журналу подій містяться в журналі подій Windows і складаються переважно з журналів аудиту операційної системи, її програм та служб [55] (рис. 2.6).

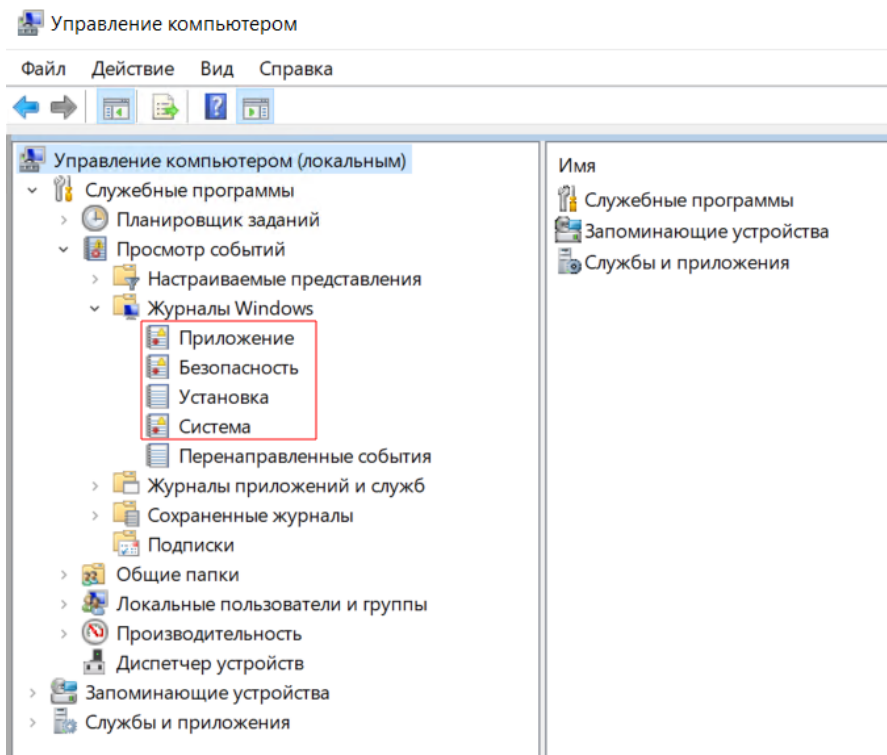


Рис. 2.6. Перелік журналів подій у оснастці «Управління комп'ютером»

Їх можна знайти за шляхом

`%SystemDrive%\Windows\System32\winevt\Logs,`

де найважливішим з них є файл `Security.evtx`, оскільки він містить дані про вхід/вихід користувачів з системи (EventID 4624,4625,4688) та програм, які запускались на комп'ютері (їх повний шлях та час запуску).

Артефакти пам'яті – це ті артефакти, які виявляються в пам'яті комп'ютера під час роботи. Ці артефакти мають бути зібрані з живої системи, і вони, як правило, не застосовуються до експертизи виключених дисків, за певними винятками, такими як файли сторінок і файли сплячого режиму, які містять пам'ять, записану на диск [55].

Їх можна отримати за допомогою програм, які знімають інформацію з живої системи, наприклад, можна використати `Magnet RAM Capture`, створивши повний дамپ оперативної пам'яті [59].

Також вони можуть міститися у файлах:

Windows minidump: `%SystemDrive%\Windows\Minidump,`

Windows Crash Dump: файли .dmp;

за такими шляхами – C:\WINDOWS\ або C:\WINDOWS\Minidump\ .

Файли Pagefile.sys (файл сторінок/підвантаження), Hiberfil.sys (файл гібернації), Swapfile.sys (файл віртуальної пам'яті) – в корні системного диску (%SystemDrive%) [60] (рис.1.17).



Рис. 2.7. Файли (з атрибутами «Системний» та «Схований») у корні диска C:\

Варто зазначити, що завантаження артефактів пам'яті займає багато часу і зазвичай не дає корисної інформації для ІБ-аналітика. Більш корисним буде вивантаження активних процесів з повними шляхами до виконуваних файлів. Також, усі артефакти можна збирати та аналізувати окремо на розсуд аналітика або об'єднувати в «часові шкали» за допомогою спеціального програмного забезпечення, такого як log2timeline [55].

Повний криміналістичний аналіз кінцевої точки Windows складатиметься з аналізу всіх перелічених вище артефактів. Якість кінцевого аналізу буде залежить від їх повноти та достовірності. Для швидкої узагальненої ідентифікації стану ІКМ необхідно якомога швидше отримувати інформацію про стан вузлів ІКМ. Завдання оператора ідентифікації стану ІКМ, вирішення яких необхідне для прийняття рішень в циклах управління технології захисту інформації, подано на рис. 2.8.

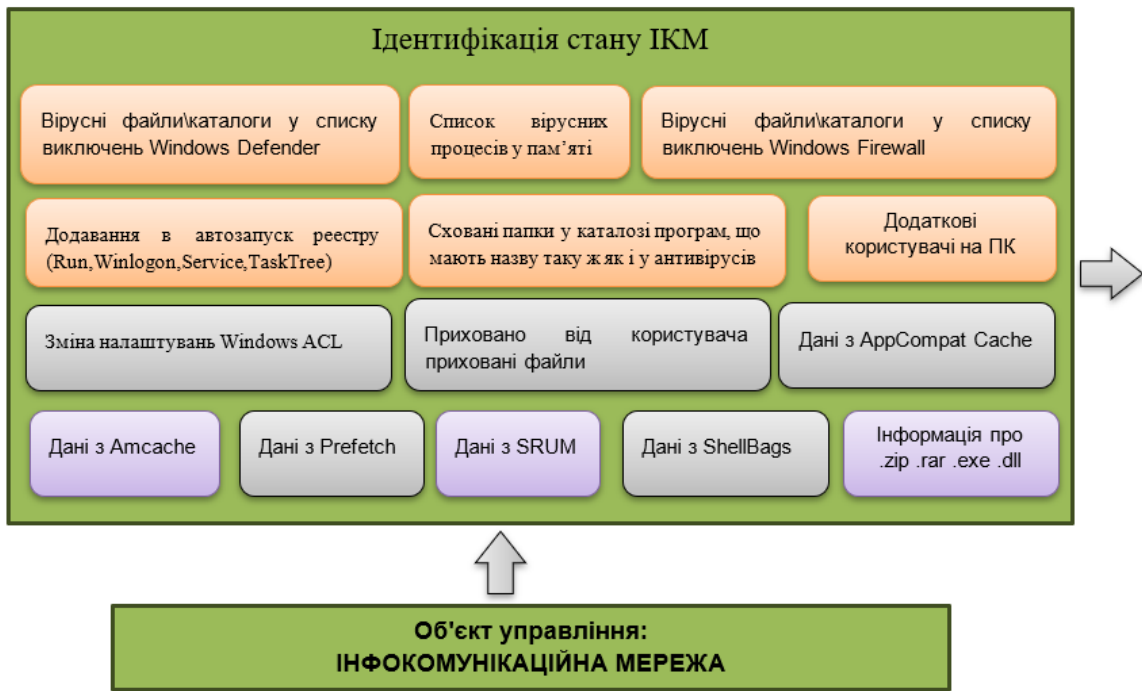


Рис. 2.8. Завдання оператора ідентифікації стану ІКМ (створено автором)

Оскільки існує кілька можливих місць розташування артефактів, то для ефективного аналізу потрібно вивантажити їх якомога більше. Проте в умовах, які не дозволяють отримати повний образ диску, а час на вивантаження артефактів невідомий, необхідно визначати черговість вивантаження файлів з досліджуваного ПК, тобто здійснювати ранжування інформації в умовах обмежених ресурсів, що у свою чергу дозволяє вивантажити тільки необхідні артефакти для кожного конкретного дослідження.

2.3. Обґрунтування доцільності застосування нейромережевих моделей для аналізу вивантажених цифрових слідів

Для вирішення задач захисту інформації ШІ вже може бути дієвим інструментом. Традиційні рішення все ще актуальні для захисту, проте об'єктам інформаційної діяльності вже необхідно використовувати та розвивати свої ШІ для захисту їх мережевої інфраструктури, даних тощо. Перспективним напрямом є активне залучення ШІ до ідентифікації стану та захисту вузлів інфокомунікаційних мереж для протистояння кібератакам,

особливо тим, які виконуються із залученням ШІ, забезпечуючи виявлення, аналіз та реагування на кіберзагрози в реальному часі та в майбутньому стати одним із важливих компонентів захисту цифрових систем від нових і невідомих загроз.

У [61] автори виконали систематичний огляд літератури (SLR), для аналізу останніх досліджень щодо LLM4Security [62, 63] та надали комплексне картографування ландшафту, визначили, як LLM впроваджуються для покращення заходів кібербезпеки, що відображено на рисунку 2.9. Окрім того, автори детально розглянули використання LLM у різних сферах безпеки, виділивши шість основних сфер, що відповідають темам зібраних статей: безпека програмного забезпечення та систем, безпека мережі, безпека інформації та контенту, безпека апаратного забезпечення та безпека блокчейну, загалом 185 статей. На рисунку 2.10 зображено розподіл LLM у цих шести сферах [61].

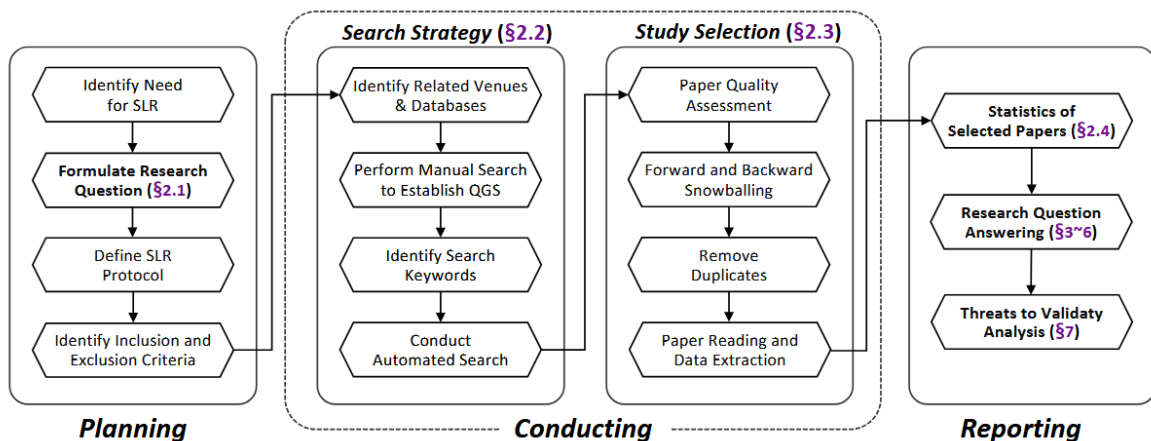


Рис.2.9. Схема систематичного огляду літератури для LLM4Security [61].

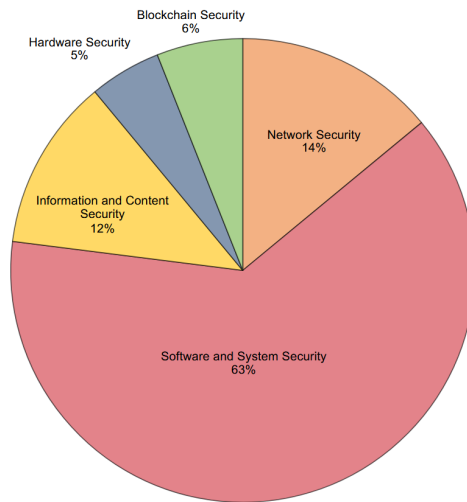


Рис.2.10. Розподіл використання LLM, орієнтованих на безпеку [61].

У [42] введено класифікацію захисту в машинному навчанні, засновану на чотирьох перспективах, що демонструють методи захисту від вхідних даних моделей до їх вихідних даних, яка подана на рисунку 2.11.

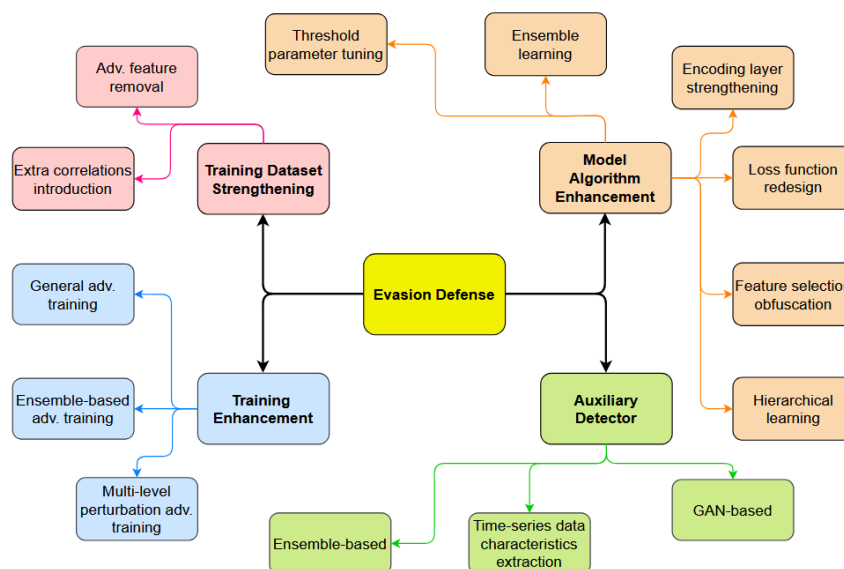


Рис.2.11. Класифікація захисту [42].

Зазначене додатково підтверджує доцільність використання ІІІ для ефективного забезпечення інформаційної безпеки корпоративної інфокомунікаційної мережі та вдосконалення методів захисту.

Не всі інформаційні впливи із залученням ШІ несуть однакову шкоду. Застосування ШІ для захисту інфокомунікаційних мереж відкриває нові можливості для виявлення, аналізу та реагування на кіберзагрози в реальному часі. Перспективним напрямком є впровадження превентивних заходів, системи ШІ можуть використовувати прогностичні моделі для ідентифікації потенційних загроз ще до їх виникнення, аналізуючи поточні тенденції в поведінці мережі або користувачів. Також, перспективним є активне залучення ШІ до захисту інфокомунікаційних мереж для протистояння кібератакам, особливо тим, які виконуються із залученням ШІ, забезпечуючи виявлення, аналіз та реагування на кіберзагрози в реальному часі.

Отже, застосування ШІ для захисту ІКМ є перспективним і актуальним напрямком, оскільки моделі ШІ можуть використовуватись для ідентифікації загроз ще до їх виникнення, а активне залучення ШІ до протистояння кібератакам, особливо тим, які виконуються із залученням ШІ, забезпечує виявлення, аналіз та реагування на кіберзагрози в реальному часі, що знижує час роботи аналітика інформаційної безпеки.

2.4. Послідовність дій в циклі управління за запропонованим методом захисту вузлів ІКМ

Діагностика ІКМ становиться послідовною, як тільки ми вводимо фактор часу у вигляді порядку тестування, послідовність модулів, які необхідно пройти в певному порядку циклу управління – становляться тестовою послідовністю. Метою є діагностика того, чи є конкретний вузол ІКМ незараженим, і якщо це не підтверджується – ізолюємо такий вузол. Оскільки кожний наступний модуль використовує результати роботи попереднього модуля, їх взаємодію можна привести до вигляду графу або деревоподібної тестової схеми. Основними кроками побудови такої схеми є:

- визначити послідовність, яка б мінімізувала середню тривалість проходження тестової послідовності для пошуку зараженого вузла ІКМ та його ізоляції;

- на кожному етапі вибирати із послідовності модулі у відповідності до їх ролі в оцінці вузла та здійсненні керуючих дій.

На етапі виявлення наявності вірусних слідів можна вважати, що вузли ІКМ є статистично незалежними. Тобто, для ефективного використання в корпоративній мережі, потрібно не тільки вивантажувати цифрові сліди по команді оператора, а автоматично вивантажити, автономно розібрати та сформувати звіт – звести використання людино-годин до мінімуму.

Послідовність дій в циклі управління за запропонованим методом захисту вузлів ІКМ представлено на рис. 2.12.

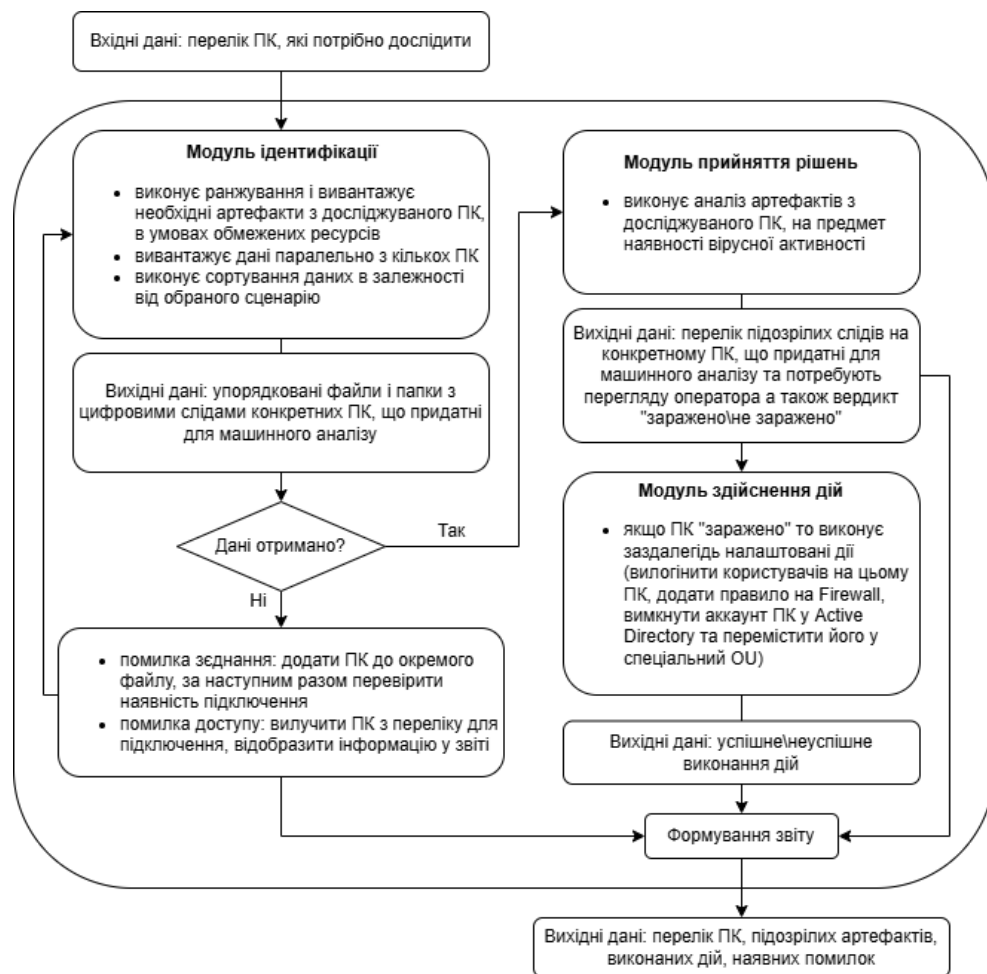


Рис. 2.12. Послідовність дій в циклі управління за запропонованим методом захисту вузлів ІКМ (створено автором)

Згідно з циклом управління (рис. 2.1) пропонується використовувати наступну послідовність дій:

1. Вивантажити перелік ПК в ІКМ, які потрібно дослідити;
2. Визначити повну назву ПК (FQDN), якщо ПК більше не залишилось то завершити цикл;
3. Перевірити чи ПК онлайн (будемо вважати, що Forward/Reverse DNS, Firewall на ПК, мережеві доступи налаштовано правильно), якщо оффлайн – додати в окремий список і з інтервалом раз на 5хв перевіряти чи не з’явилась можливість підключитися. Якщо усі інші ПК перевірені то додати до окремого файлу із кодом помилки для майбутнього звіту і вилучити з поточного дослідження;
4. Перевірити чи є доступ на ПК, якщо відсутній то або виконати заздалегідь налаштовані дії (наприклад, спробувати використати інший обліковий запис для підключення) або додати до окремого файлу із кодом помилки для майбутнього звіту і вилучити з поточного дослідження;
5. Вибрати тільки ті артефакти, які мають в собі найбільш інформативний стан про систему, з найменшим розміром та високою швидкістю обробки (дублюючі артефакти, нерелевантні, артефакти які з високою імовірністю будуть змінені зловмисником слід виключити, що дасть економію часу на вивантаження і обробку слідів);
6. Вибрати сценарій, який дасть нам уявлення про те, які сліди потрібно завантажувати в першу чергу, що допоможе мати хоч якість уявлення про досліджувану систему, у випадку якщо з’єднання раптово перерветься;
7. Завантажити на ПК програми для проведення Forensic Triage і запустити їх з необхідними параметрами;
8. Результат роботи програм має бути у стандартизованому вигляді, що більш підходить для машинної обробки (наприклад у форматі .csv) та заздалегідь відфільтрований від легітимних слідів (для швидкого перегляду людиною);

9. Вивантажувати результати роботи програм для проведення Forensic Triage на окремий сервер у вигляді окремих файлів у нестиснутому вигляді а бінарні дані у архіві для економії місця та часу на передачу інформації, від найбільш релевантних до найменш відразу після створення. Таким чином, цих даних буде достатньо для первинного дослідження, навіть якщо під час передачі обірветься з'єднання і частина даних прийде у пошкодженому вигляді;

10. Пересвідчитись, що на окремий сервер в локальній мережі відправлено всі результати програм;

11. Видалити програми, що було завантажено і їх результати з ПК;

12. На сервері, привести наявні цифрові сліди (результати роботи програм), що придатні для машинного аналізу у зрозумілий для ШІ вигляд.

13. Запустити локальну ШІ модель з необхідними параметрами і дочекатись виконання запиту;

14. Відфільтрувати нерелевантні дані, перевірити чи є остаточний вердикт від ШІ на предмет вірусного зараження;

15. Якщо зараження немає – додати інформацію про ПК до звіту, перейти до пункту 2;

16. Якщо зараження є – виконати заздалегідь налаштовані правила (вилогінити користувача, вимкнути акаунт ПК, тощо) додати інформацію про ПК та виявлені сліди до звіту, перейти до пункту 2;

17. Зформувати фінальний звіт після обробки всіх ПК у мережі та відправити його на заздалегідь налаштовану пошту.

Особливістю запропонованого методу тестування є те що кожний наступний крок тестової послідовності залежить від результату тестування попереднього, тобто встановлення чи взагалі передається інформація між модулями і яка саме, чи достатньо її для прийняття рішень і чи не виникло помилок. Варто зазначити, що тестування виконується для кожного наявного вузла в ІКМ.

При цьому:

- якщо система передає дані і не є зараженою – ніяких дій не виконується;
- якщо система не може передати дані або є зараженою – виконуються керуючі дії, направлені або на отримання інформації з ПК або ізоляції такого ПК від іншого сегменту мережі.

Зробимо припущення, що в ІКМ маємо три стани вузла:

1. Нічого апріорі невідомо про систему – в такому випадку необхідно систематично перевіряти ПК на можливість підключення до тих пір, доки система не почне віддавати дані або коли перевірка стане недоцільною.

2. Система онлайн, але не може віддати дані – в такому випадку доведеться розбиратись з конкретними причинами несправностей або виконувати заздалегідь налаштовані дії, для можливого усунення несправностей.

3. Відомо, що система може знаходитись лише в одному з N взаємовиключних станів, для кожного з таких станів будуть протестовані наявні дані до тих пір, поки не буде однозначно ідентифіковано стан системи «заражено / не заражено».

Вказані три стани одночасно потребують все більше і більше інформації апріорі і все більше ресурсів під час виконання тестової послідовності, в результаті, система стає все більш складнішою.

2.5. Мінімізація середнього часу на аналіз активних вузлів розподіленої ІКМ

Оскільки в розподіленій ІКМ активних вузлів може бути більше 100 та навіть більше 1000 важливого значення набуває час за який повністю буде перевірена ІКМ на наявність вірусної активності. Таким чином, якщо навіть незначно зменшиться час на аналіз одного вузла – це матиме вплив на загальний час аналізу, тож треба проводити оптимізацію на всіх рівнях тестової послідовності, для кожного модулю. При великій кількості

досліджуваних ПК, зменшення часу, необхідного для аналізу кожного ПК стає більш пріоритетним, ніж пропустити певну кількість специфічних вірусів.

Для оптимізації наявних рішень *на етапі ідентифікації стану ІКМ* для вивантаження цифрових артефактів, доцільно розглянути кілька можливих шляхів для зменшення роботи аналітика інформаційної безпеки:

1. Зменшення кількості ПК, які потрібно дослідити.
2. Зменшення кількості цифрових слідів, які потрібно дослідити.
3. Автоматизувати процес за допомогою скриптів/програм.
4. Вважати зараженим тільки той ПК на якому спрацював АВ.
5. Вважати всі ПК зараженими, і перевстановити на всіх операційну систему (ОС) без збереження профілів користувачів.

У той час коли це дозволяє оптимізувати час, кожен з цих методів має свої недоліки:

1. При зменшенні кількості ПК ми не можемо точно сказати чи всі інші ПК не заражені.

2. При зменшенні кількості цифрових слідів, особливо якщо вірус затирає свої сліди або видаляється сам, ми не побачимо всі компоненти вірусу та сліди вірусної атаки на ПК, що дасть нам хибне уявлення про те, що ПК не був заражений.

3. Автоматизація завжди дає свій відсоток false-positive та false-negative, та при детальному аналізі цих подій і коригуванні правил для автоматичного розбору – витратиться стільки ж часу як при звичайному аналізі.

4. АВ зазвичай видалить тільки той компонент вірусу який знає сигнатурно або знайшов евристикою, та ніяк не зачепить інші «невидимі» (fully undetectable - FUD) частини.

5. Іноді переустановлення ОС є надмірним та неефективним рішенням, особливо якщо це будуть робити локальні адміністратори.

Для оптимізації наявних кроків *на етапі прийняття рішень* (в основному це оцінка того, чи знаходиться конкретний вузол ІКМ у стані незараженості), доцільно розглянути використання заздалегідь налаштованих

правил і застосування моделей ШІ для аналізу цифрових слідів, оскільки людські ресурси дорожче коштують і збільшують час на виявлення (проте результати є більш точними).

Можливі ситуації, коли з наявних слідів статистичними методами зі 100% надійністю можна визначити вірусну активність, в такому випадку доцільно використовувати в автоматичному режимі прості правила (наприклад, шукати наявність записів у Windows Defender) перед відправленням цифрових слідів на обробку моделлю ШІ, оскільки тоді необхідність в обробці ШІ моделлю відпадає, що в свою чергу економить ресурси.

Нейромережеві моделі можуть бути застосовані для вирішення задач виявлення заражених ПК на базі цифрових слідів в якості одного із компонентів кіберзахисту для зниження часу реагування на події інформаційної безпеки. Враховуючи, що кожна модель краще проявляє себе у специфічних умовах із різними сценаріями запуску, вибір моделі буде залежати від актуальних задач та наявних ресурсів. При цьому, якість відповідей ШІ класифікується наступним чином:

- ПК заражено і нейромережева модель це виявила,
- ПК заражено і нейромережева модель це не виявила,
- ПК не заражено і нейромережева модель це виявила,
- ПК не заражено і нейромережева модель це не виявила,
- ПК заражено і модель не змогла розпізнати стан,
- ПК не заражено і модель не змогла розпізнати стан.

Для оптимізації наявних кроків *на етапі здійснення керуючих дій* (в основному це вибір дій в залежності від того, чи знаходиться конкретний вузол ПКМ у стані незараженості та формування звіту), доцільно розглянути використання автоматизованих скриптів які виконують заздалегідь налаштовані правила та автоматично формують звіт (наприклад, примусово скинути пароль усім користувачам зараженого комп'ютера), оскільки це

знизить час реагування на події інформаційної безпеки порівняно з ручним аналізом та діями.

Додатково, на кожному етапі доступно розпаралелення обчислень (рекомендується досліджувати ПК блоками з 5-10 ПК), що не прискорює отримання інформації про стан окремого ПК, але знижує чергу з ПК і оптимізує загальне очікування на обчислення, що позитивно впливає на загальну економію часу.

Таким чином, одночасне впровадження зазначених оптимізаційних рішень для кожного з етапів, дозволяє мінімізувати середній час на проходження тестової послідовності, що позитивно впливає на мінімізацію загального часу на дослідження ІКМ на предмет виявлення заражених вузлів.

Побудова захисту інформації у вигляді автоматизованого управління, спрямованого на забезпечення підтримки цільового стану ІКМ, дозволяє забезпечити необхідний рівень захищеності інформації. Запропонована тестова послідовність дозволяє виявляти заражені вірусами вузли ІКМ в циклі управління та дозволяє оптимізувати час на оцінку одного вузла. Одночасне впровадження оптимізаційних рішень для кожного з етапів дозволяє мінімізувати середній час на проходження тестової послідовності, що позитивно впливає на мінімізацію загального часу на виявлення та ізоляції заражених вузлів інфокомунікаційної мережі в циклі управління.

Мінімізація середнього часу забезпечується:

- використанням лише мінімально необхідних цифрових слідів;
- використанням моделі ШІ у якості одного з компонентів модулю прийняття рішень та заздалегідь налаштованих правил оцінки цифрових слідів;
- застосуванням заздалегідь налаштованих правил для автоматичного здійснення керуючих дій щодо ізоляції зараженого вузла;
- розпаралеленням обчислень.

2.6. Дослідження впливу ШПЗ на вузли ІКМ

2.6.1. Дослідження ШПЗ у віртуальному середовищі зі зміненою конфігурацією

Метою дослідження є пошук цифрових слідів, залишених шкідливим програмним забезпеченням у вузлах ІКМ. Для цього потрібно запустити ШПЗ з правами адміністратора, оскільки при першому скануванні шкідлива активність може і не проявлятися. Щоб уникнути зараження хост системи (є ймовірність того, що виявлено не всі цифрові сліди, і хост система буде заражена) бажано використовувати віртуальні машини (VM). Це допоможе підвищити достовірність і наглядність виявлення шляхів вірусного зараження, детектування та знешкодження ШПЗ.

Створене на хост системі віртуальне середовище VirtualBox має наступні параметри:

1. OS Windows 10 в якості віртуальної машини VirtualBox 7.0, з останніми оновленнями Windows Defender та ОС (див. рис. 2.13).

Центр оновлення Windows



У вас установлены все последние обновления
Время последней проверки: сегодня, 7:07 PM

Проверить наличие обновлений

Версия анализа безопасности: 1.409.482.0
Дата создания версии: 4/24/2024 6:23 AM
Последнее обновление: 4/24/2024 11:59 AM

Рис. 2.13. Статус оновлень Windows9

2. Користувач user з правами адміністратора.

3. Встановлені програми:

- 7-Zip 24.04;
- .NET Framework 2.0, 3.0, 3.5, 5.0, 6.0;
- Microsoft .NET SDK 5.0.408, 6.0.421;
- Microsoft Visual C++ Redistributable 2008, 2012, 2013, 2015-2022;
- Microsoft DirectX 9,12.

4. Змінені налаштування папок (див. рис. 2.14).

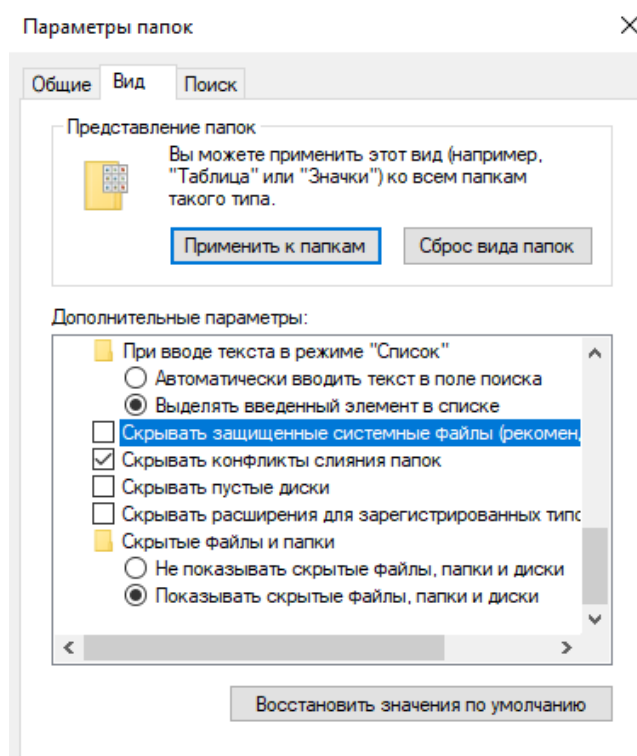


Рис. 2.14. Налаштування папок

Результат перевірки факту перебування Pafish 0.6 [64] у віртуальному середовищі (рис. 2.15), показує, що методи для попередження технік Anti-Debugging/Anti-VM не використовуються.

```
Administrator: Командная строка - pafish64.exe
[-] VirtualBox detection
[*] Scsi port->bus->target id->logical unit id-> 0 identifier ... traced!
[*] Reg key (HKLM\HARDWARE\Description\System "SystemBiosVersion") ... traced!
[*] Reg key (HKLM\SOFTWARE\Oracle\VirtualBox Guest Additions) ... traced!
[*] Reg key (HKLM\HARDWARE\Description\System "VideoBiosVersion") ... traced!
[*] Reg key (HKLM\HARDWARE\ACPI\DSDT\VBOX_) ... traced!
[*] Reg key (HKLM\HARDWARE\ACPI\FADT\VBOX_) ... traced!
[*] Reg key (HKLM\HARDWARE\ACPI\RSDT\VBOX_) ... traced!
[*] Reg key (HKLM\SYSTEM\ControlSet001\Services\VBox*) ... traced!
[*] Reg key (HKLM\HARDWARE\DESCRIPTION\System "SystemBiosDate") ... traced!
[*] Driver files in C:\WINDOWS\system32\drivers\VBox* ... traced!
[*] Additional system files ... traced!
[*] Looking for a MAC address starting with 08:00:27 ... traced!
[*] Looking for pseudo devices ... OK
[*] Looking for VBoxTray windows ... traced!
[*] Looking for VBox network share ... traced!
[*] Looking for VBox processes (vboxservice.exe, vboxtray.exe) ... traced!
[*] Looking for VBox devices using WMI ... traced!
```

Рис. 2.15. Результат перевірки факту перебування Pafish 0.6 у віртуальному середовищі

Лог-файл, який показує наявні сліди віртуального середовища:

```
[pafish] VirtualBox traced using Reg key HKLM\HARDWARE\DEVICEMAP\Scsi\Scsi Port 0\Scsi Bus 0\Target Id 0\Logical Unit Id 0 "Identifier"
[pafish] VirtualBox traced using Reg key HKLM\HARDWARE\Description\System "SystemBiosVersion"
[pafish] VirtualBox traced using Reg key HKLM\SOFTWARE\Oracle\VirtualBox Guest Additions
[pafish] VirtualBox traced using Reg key HKLM\HARDWARE\Description\System "VideoBiosVersion"
[pafish] VirtualBox traced using Reg key HKLM\HARDWARE\ACPI\DSDT\VBOX_
[pafish] VirtualBox traced using Reg key HKLM\HARDWARE\ACPI\FADT\VBOX_
[pafish] VirtualBox traced using Reg key HKLM\HARDWARE\ACPI\RSDT\VBOX_
[pafish] VirtualBox traced using Reg key HKLM\SYSTEM\ControlSet001\Services\VBoxGuest
[pafish] VirtualBox traced using Reg key HKLM\SYSTEM\ControlSet001\Services\VBoxMouse
[pafish] VirtualBox traced using Reg key HKLM\SYSTEM\ControlSet001\Services\VBoxService
[pafish] VirtualBox traced using Reg key HKLM\SYSTEM\ControlSet001\Services\VBoxSF
[pafish] VirtualBox traced using Reg key HKLM\HARDWARE\DESCRIPTION\System "SystemBiosDate"
[pafish] VirtualBox traced using driver file C:\WINDOWS\system32\drivers\VBoxMouse.sys
[pafish] VirtualBox traced using driver file C:\WINDOWS\system32\drivers\VBoxGuest.sys
[pafish] VirtualBox traced using driver file C:\WINDOWS\system32\drivers\VBoxSF.sys
[pafish] VirtualBox traced using system file C:\WINDOWS\system32\vboxhook.dll
[pafish] VirtualBox traced using system file C:\WINDOWS\system32\vboxmrxnp.dll
[pafish] VirtualBox traced using system file C:\WINDOWS\system32\vboxservice.exe
[pafish] VirtualBox traced using system file C:\WINDOWS\system32\vboxtray.exe
[pafish] VirtualBox traced using system file C:\WINDOWS\system32\VBoxControl.exe
[pafish] VirtualBox traced using system file C:\program files\oracle\virtualbox guest additions\
[pafish] VirtualBox traced using MAC address starting with 08:00:27
[pafish] VirtualBox traced using device \\.\VBoxMiniRdrDN
[pafish] VirtualBox traced using VBoxTray windows
[pafish] VirtualBox traced using its network share
```

[pafish] VirtualBox traced using vboxservice.exe process
[pafish] VirtualBox traced using vboxtray.exe process
[pafish] VirtualBox device identifiers traced using WMI

Розглянемо ситуацію, коли користувач тільки що встановив ОС, і він не має ключа і бажає встановити KMS Auto для емуляції KMS (Key Management System)-сервера Microsoft. Для цього він шукає це ПЗ в пошуковій системі та завантажує запаролений архів KMS Tools Portablezip, який показано на рис. 2.16.

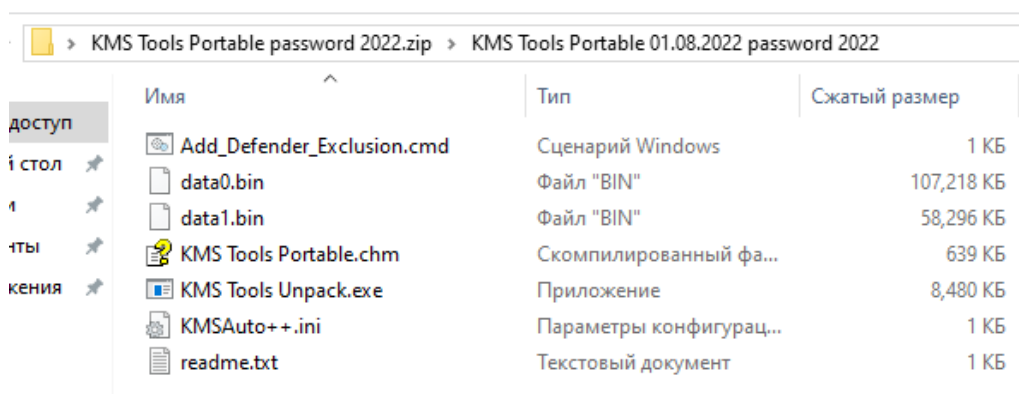



Рис. 2.16. Скріншот запарованого архіву з ШПЗ

За «рекомендаціями» зловмисників, для «коректної роботи програми» користувач вимикає антивірус та здійснює розархівацію файлу .zip (див. рис. 2.17 і 2.18).

Защита в режиме реального времени

Обнаруживает и останавливает установку или запуск вредоносных программ на вашем устройстве. Можно на короткое время отключить этот параметр, после чего он будет снова включен автоматически.

 Защита в режиме реального времени отключена. Устройство уязвимо.

Откл.

Рис. 2.17. Скріншот повідомлення, про відключення захисту

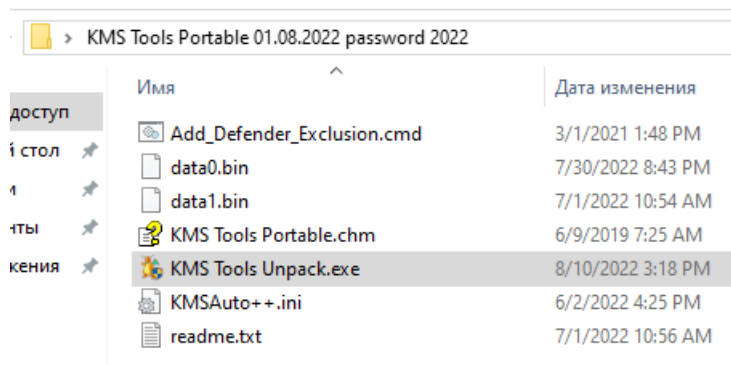


Рис. 2.18. Скріншот розархівованого архіву з ШПЗ

Після відкриття файлу KMS відбувається зараження вузла (див. рис. 2.19).



Рис. 2.19. Вигляд відкритого KMS

Окрім інформації про додавання виключень до антивіруса (див. рис. 2.20), ніяких додаткових файлів/папок та процесів не спостерігається, що може свідчити про наявний механізм детектування віртуальної машини (VM), проте який не реалізований до кінця.

Исключения

Добавьте или удалите элементы, которые вы хотите исключить из проверок антивирусной программы Microsoft Defender.

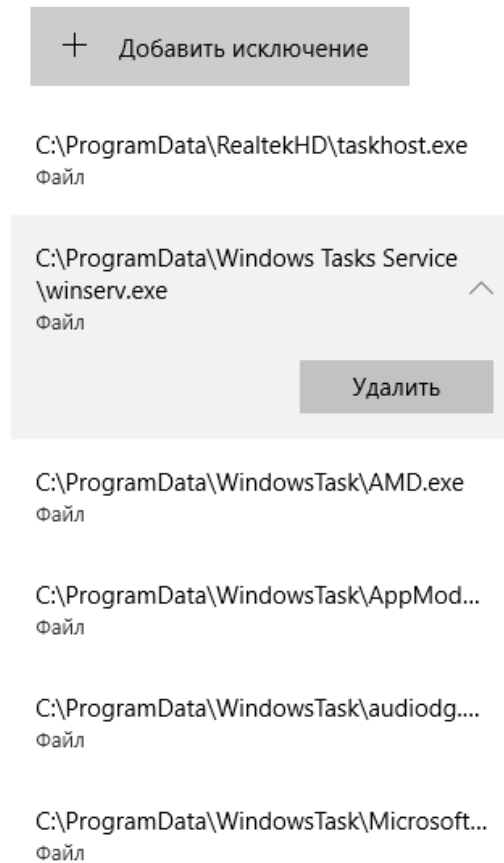


Рис. 2.20. Виключення Windows Defender

В зазначеній ситуації доцільним є пошук такого механізму детектування ВМ шкідливим програмним забезпеченням за допомогою www.hybrid-analysis.com, результати якого подано на рис. 2.21.

Malicious Indicators

Environment Awareness

Reads the system/video BIOS version

details "KMSToolsUnpack.exe" (Path: "HKLM\HARDWARE\DESCRIPTION\SYSTEM"; Key: "VIDEObIOSVERSION")
 "KMSToolsUnpack.exe" (Path: "HKLM\HARDWARE\DESCRIPTION\SYSTEM"; Key: "SYSTEMBIOSVERSION")

source Registry Access

relevance 9/10

ATT&CK ID T1012 (Show technique in the MITRE ATT&CK™ matrix)

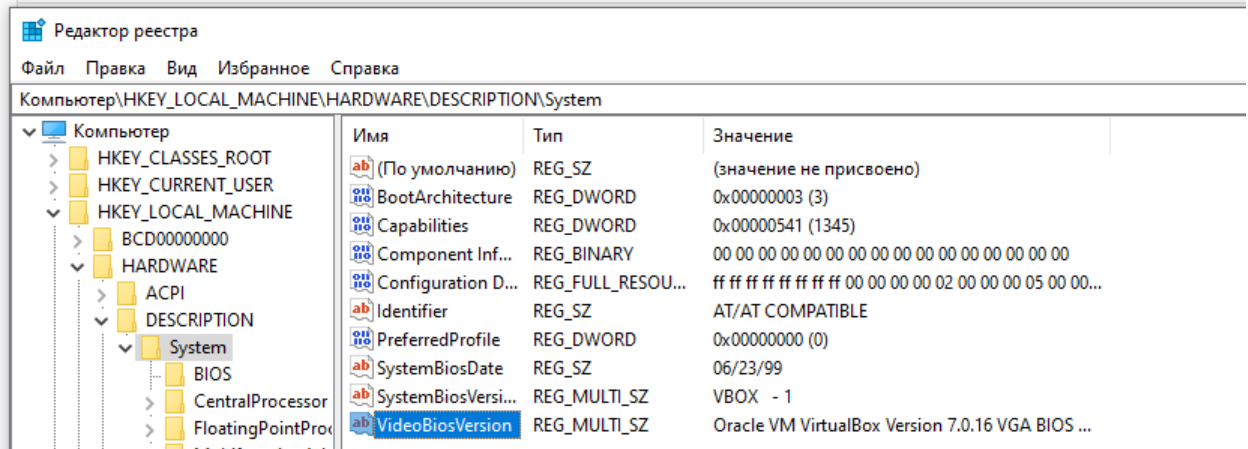


Рис. 2.21. Механізм детектування VM

Скоріш за все, вірус перевіряє значення BIOS, знаходить там дані віртуальної машини (VBOX, VM, VirtualBox) і не запускає свої основні файли, тому доцільно взяти нову VM з вищезгаданими налаштуваннями і перед запуском ШПЗ замінити значення BIOS (див. рис. 2.22).

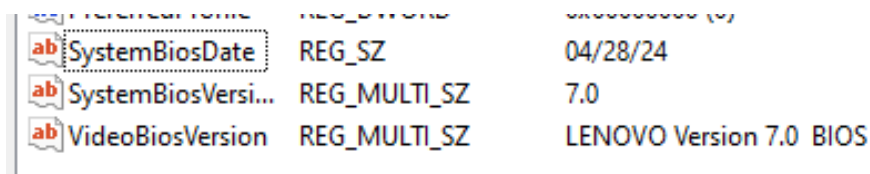
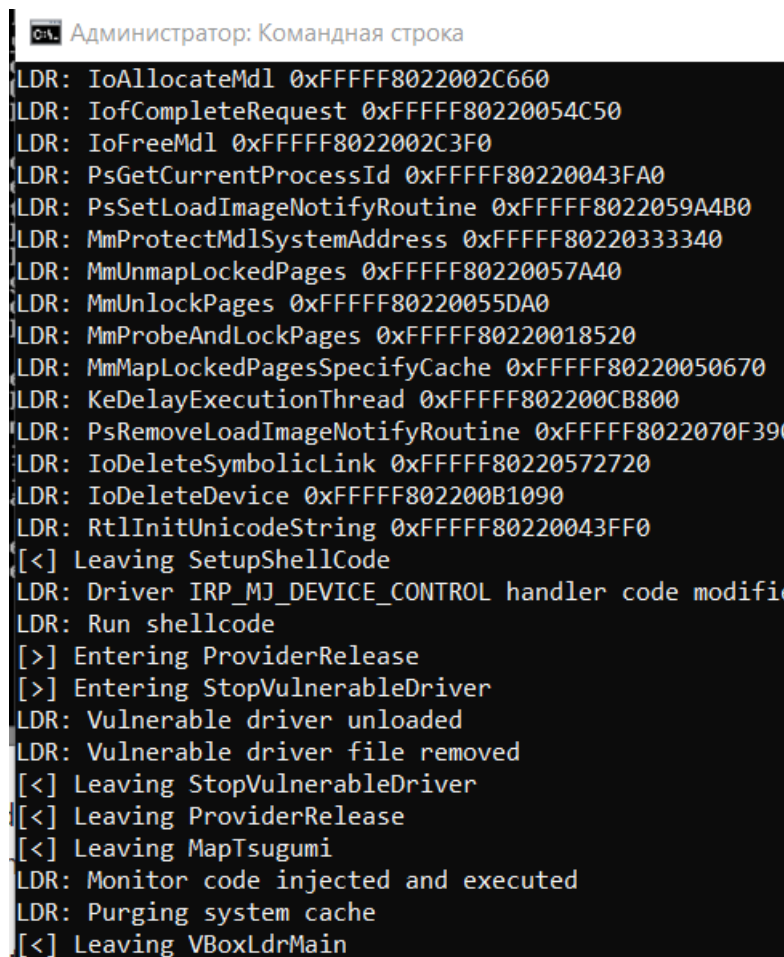


Рис. 2.22. Змінені значення BIOS

Коли значення змінено і нова VM налаштована, ШПЗ знову запущено, але як і попередній раз – є дані лише у виключеннях антивірусу. Оскільки

бажаного результату отримати не вдалося, за допомогою утиліти VBoxHardenedLoader v2.0.1 [65], запущеної на хостовій машині, прибираємо ті сліди віртуального середовища, які заважають запуску тестових ШПЗ. Якщо після завантаження драйвера помилок немає (див. рис. 2.23), наступним кроком необхідно запустити і перевірити ВМ.



```
Администратор: Командная строка
LDR: IoAllocateMdl 0xFFFFF8022002C660
LDR: IoCompleteRequest 0xFFFFF80220054C50
LDR: IoFreeMdl 0xFFFFF8022002C3F0
LDR: PsGetCurrentProcessId 0xFFFFF80220043FA0
LDR: PsSetLoadImageNotifyRoutine 0xFFFFF8022059A4B0
LDR: MmProtectMdlSystemAddress 0xFFFFF80220333340
LDR: MmUnmapLockedPages 0xFFFFF80220057A40
LDR: MmUnlockPages 0xFFFFF80220055DA0
LDR: MmProbeAndLockPages 0xFFFFF80220018520
LDR: MmMapLockedPagesSpecifyCache 0xFFFFF80220050670
LDR: KeDelayExecutionThread 0xFFFFF802200CB800
LDR: PsRemoveLoadImageNotifyRoutine 0xFFFFF8022070F390
LDR: IoDeleteSymbolicLink 0xFFFFF80220572720
LDR: IoDeleteDevice 0xFFFFF802200B1090
LDR: RtlInitUnicodeString 0xFFFFF80220043FF0
[<] Leaving SetupShellCode
LDR: Driver IRP_MJ_DEVICE_CONTROL handler code modified
LDR: Run shellcode
[>] Entering ProviderRelease
[>] Entering StopVulnerableDriver
LDR: Vulnerable driver unloaded
LDR: Vulnerable driver file removed
[<] Leaving StopVulnerableDriver
[<] Leaving ProviderRelease
[<] Leaving MapTsugumi
LDR: Monitor code injected and executed
LDR: Purging system cache
[<] Leaving VBoxLdrMain
```

Рис. 2.23. Повідомлення про успішну роботу драйвера

Після запуску ВМ, отримали фатальну помилку ВМ, що не дає їй запуснитися. Повідомлення про помилку ВМ подано на рис. 2.24.

```
VD: Backend
'VBoxIsoMaker' does not
support async I/O
(VERR_NOT_SUPPORTED)
.

Unknown configuration
value '/DsdFilePath'
found in the configuration
of acpi instance #0
(VERR_CFGM_CONFIG_UN
KNOWN_VALUE).

Код ошибок E_FAIL
и:          (0x800040
           05)

Компонент ConsoleWra
:          p

Интерфейс IConsole
:          {6ac83d89-
           6ee7-4e33-
           8ae6-
           b257b2e81
           be8}
```

Рис. 2.24. Повідомлення про помилку ВМ

Аналізуючи помилку, приходимо до висновку, що для її вирішення потрібно «клонувати» робочу ВМ та переналаштувати дані у скрипті `hidevm_ahci.cmd` наступним чином [66]:

```
rem %vboxman% setextradata "%VM%"
"VBoxInternal/Devices/acpi/0/Config/DsdtFilePath" "%vmconfigdir%ACPI-
DSDT.bin"
rem %vboxman% setextradata "%VM%"
"VBoxInternal/Devices/acpi/0/Config/SsdtFilePath" "%vmconfigdir%ACPI-
SSDT.bin"
```

Після доналаштувань, перезапускаємо програму VirtualBox та драйвер `loader.exe`, і у разі відсутності помилок перевіряємо ВМ за допомогою `Pafish 0.6`, з попереднім видаленням програми “VirtualBox Guest Additions” та зв’язаних з нею ключів реєстру. Як можна побачити на рис. 2.25 та рис. 2.26, віртуальне середовище більше не має частини ознак віртуального середовища.

```
Administrator: Командная строка - pafish64.exe
[-] VirtualBox detection
[*] Scsi port->bus->target id->logical unit id-> 0 identifier ... OK
[*] Reg key (HKLM\HARDWARE\Description\System "SystemBiosVersion") ... OK
[*] Reg key (HKLM\SOFTWARE\Oracle\VirtualBox Guest Additions) ... OK
[*] Reg key (HKLM\HARDWARE\Description\System "VideoBiosVersion") ... OK
[*] Reg key (HKLM\HARDWARE\ACPI\DSDT\VBOX_) ... traced!
[*] Reg key (HKLM\HARDWARE\ACPI\FADT\VBOX_) ... OK
[*] Reg key (HKLM\HARDWARE\ACPI\RSDT\VBOX_) ... OK
[*] Reg key (HKLM\SYSTEM\ControlSet001\Services\VBox*) ... OK
[*] Reg key (HKLM\HARDWARE\DESCRIPTION\System "SystemBiosDate") ... OK
[*] Driver files in C:\WINDOWS\system32\drivers\VBox* ... OK
[*] Additional system files ... OK
[*] Looking for a MAC address starting with 08:00:27 ... OK
[*] Looking for pseudo devices ... OK
[*] Looking for VBoxTray windows ... OK
[*] Looking for VBox network share ... OK
[*] Looking for VBox processes (vboxservice.exe, vboxtray.exe) ... OK
[*] Looking for VBox devices using WMI ... traced!
```

Рис. 2.25. Результат перевірки факту перебування Pafish 0.6 у віртуальному середовищі

```
[pafish] Start
[pafish] Windows version: 6.2 build 9200 (native)
[pafish] CPU: GenuineIntel Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz
[pafish] CPU VM traced by checking the difference between CPU timestamp counters (rdtsc) forcing VM exit
[pafish] Sandbox traced by missing dialog confirmation
[pafish] Sandbox traced by missing or implausible dialog confirmation
[pafish] Sandbox traced by checking disk size <= 60GB via DeviceIoControl()
[pafish] Sandbox traced by checking disk size <= 60GB via GetDiskFreeSpaceExA()
[pafish] VirtualBox traced using Reg key HKLM\HARDWARE\ACPI\DSDT\VBOX_
[pafish] VirtualBox device identifiers traced using WMI
[pafish] End
```

Рис. 2.26. Лог-файл програми Pafish 0.6

Таким чином, ті ШПЗ, які аналізували ОС на предмет наявності віртуального середовища, більше не знайдуть таких ознак і запусяться повною мірою, що дозволить безпечно і повною мірою проаналізувати конкретне ШПЗ. При цьому, бажано залишити ВМ в режимі очікування для дозавантаження всіх модулів ШПЗ.

Викладене дозволяє стверджувати, що віртуальне середовище зі зміненою конфігурацією, яке не має більшості ознак віртуального середовища, дозволяє безпечно і повною мірою проаналізувати тестове ШПЗ з функціоналом Anti-Debugging/Anti-VM.

2.6.2 Ідентифікація цифрових слідів, що залишаються після вірусної атаки

Обґрунтування словника ознак, необхідних для виявлення тривожних подій, здійснено на прикладі характерних ознак, які після себе залишив наступний вірус:

VT 60\72 Trojan[Packed]/Win64.Themida, Trojan[dropper]:Win/Tiggre, Trojan.Barys [67].

Після запуску вірусу і залишення ВМ «як є» впродовж 3х годин, виявлено наступні сліди:

1. Інформація про вірусні файли\каталоги у списку виключень Windows Defender (Regedit см – HKEY_LOCAL_MACHINE \SOFTWARE \Microsoft \Windows Defender\Exclusions\Paths

Powershell – (Get-MpPreference).ExclusionPath) [68].

```
Администратор: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
PS C:\Windows\system32> (get-mpPreference).ExclusionPath
C:\Program Files\RDP Wrapper
C:\ProgramData
C:\ProgramData\RealtekHD\taskhost.exe
C:\ProgramData\Windows Tasks Service\winserv.exe
C:\ProgramData\WindowsTask\AMD.exe
C:\ProgramData\WindowsTask\AppModule.exe
C:\ProgramData\WindowsTask\audiodg.exe
C:\ProgramData\WindowsTask\MicrosoftHost.exe
C:\Windows\AACT_Tools
C:\Windows\AACT_Tools\AACT_files
C:\Windows\AACT_Tools\AACT_files\KMSSS.exe
C:\Windows\AACT_Tools\AACT_x64.exe
C:\Windows\KMS
C:\Windows\KMSAutoS
C:\Windows\system32
C:\Windows\System32\SppExtComObjHook.dll
C:\Windows\System32\SppExtComObjPatcher.exe
PS C:\Windows\system32>
```

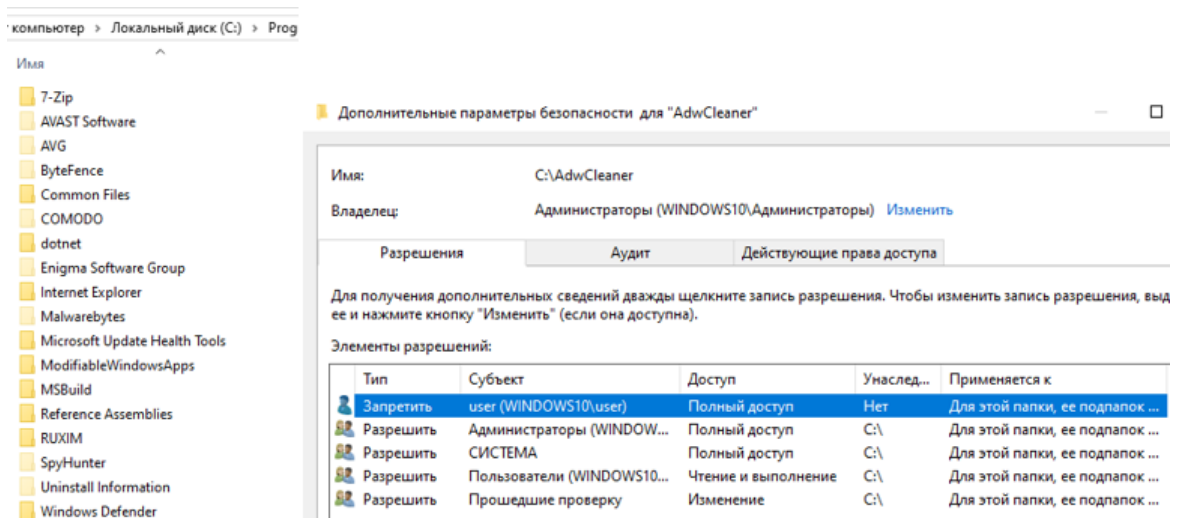
2. Список процесів, які не є легітимними (CMD – wmic process get ExecutionPath).

```

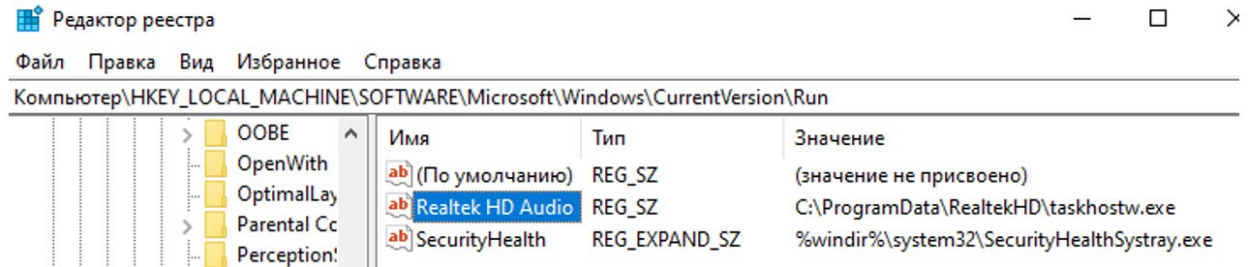
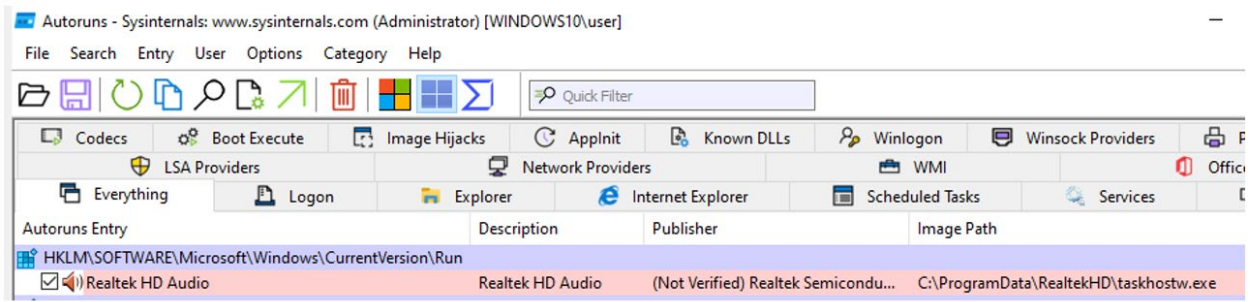
C:\Windows\System32\svchost.exe
C:\Windows\system32\svchost.exe
C:\Windows\SystemApps\ShellExperienceHost_cw5n1h2txyewy\
C:\Windows\System32\RuntimeBroker.exe
C:\ProgramData\RealtekHD\taskhost.exe
C:\ProgramData\RealtekHD\taskhostw.exe
C:\ProgramData\WindowsTask\audioldg.exe
C:\Windows\ImmersiveControlPanel\SystemSettings.exe
C:\Windows\System32\oobe\UserOOBEBroker.exe
C:\Windows\system32\taskhostw.exe
C:\Windows\system32\cmd.exe
C:\Windows\system32\conhost.exe
C:\Windows\system32\mmc.exe
C:\ProgramData\WindowsTask\MicrosoftHost.exe
C:\Windows\system32\conhost.exe

```

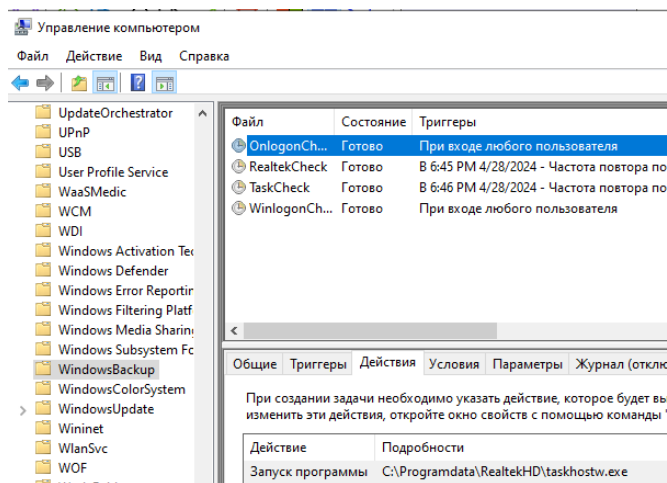
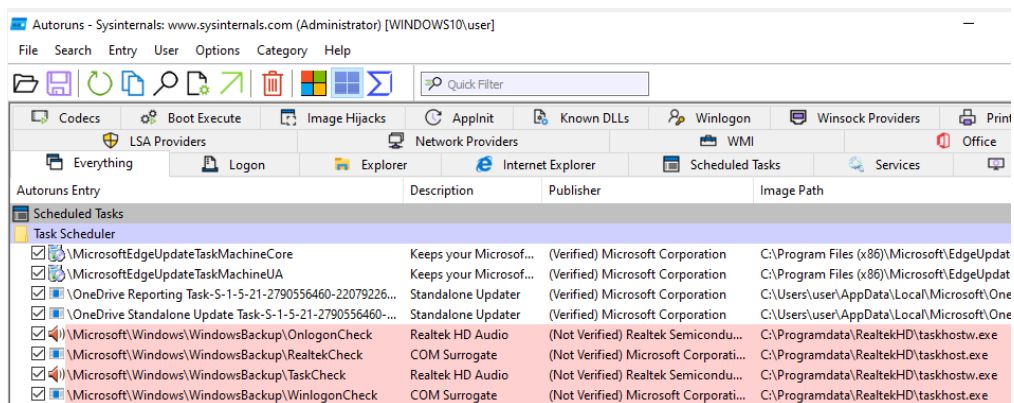
3. Сховані папки у каталозі програм, що мають назву таку ж як і у антивірусних рішень, при цьому, доступ до них заборонено користувачу який відкривав найперший вірусний файл, реалізовано це через Windows ACL [69].



4. Записи в ключі автозапуску Run, за допомогою утиліти Autoruns [70].

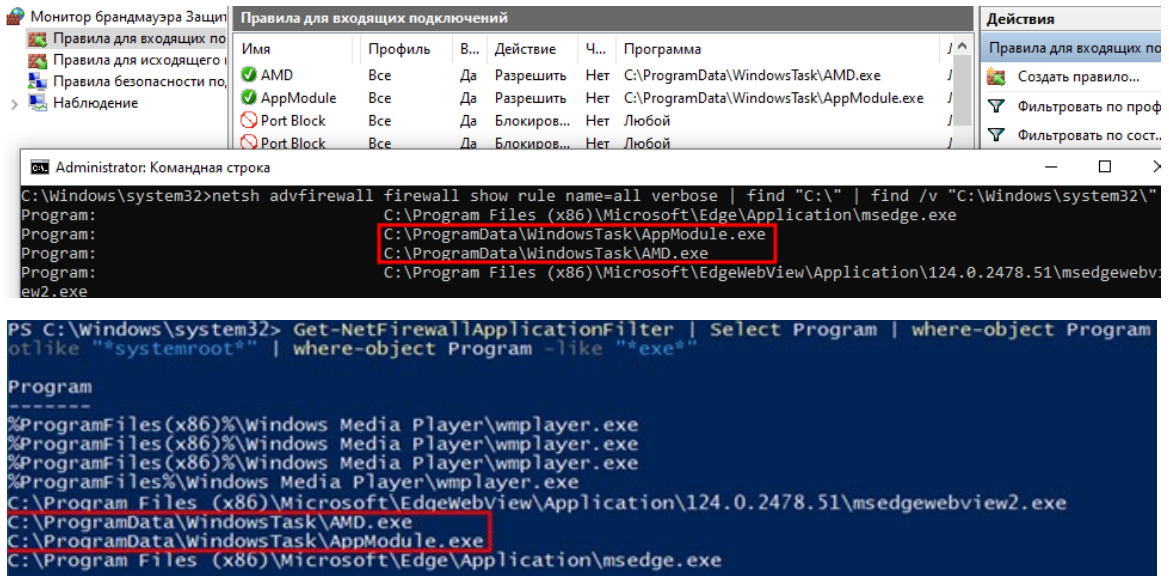


5. Записи в Windows TaskSheduller [71].

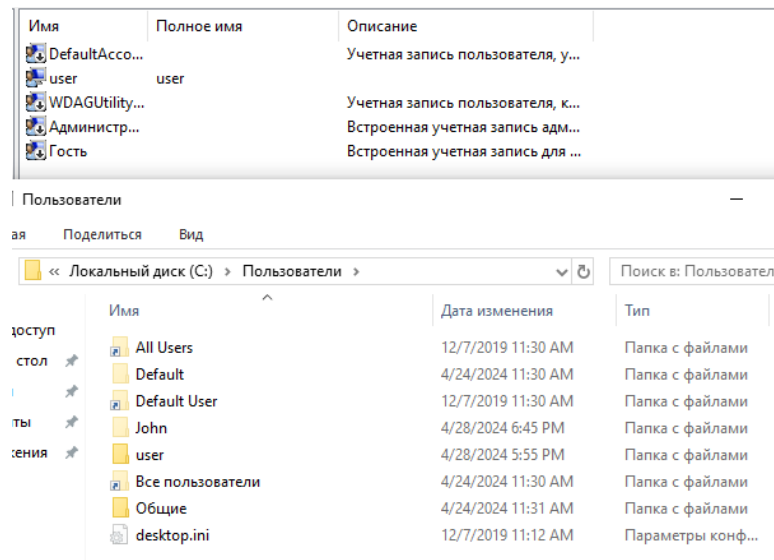


6. Записи в Windows Firewall (CMD – netsh advfirewall firewall show rule name=all verbose | find "C:") | find /v "C:\Windows\system32\"

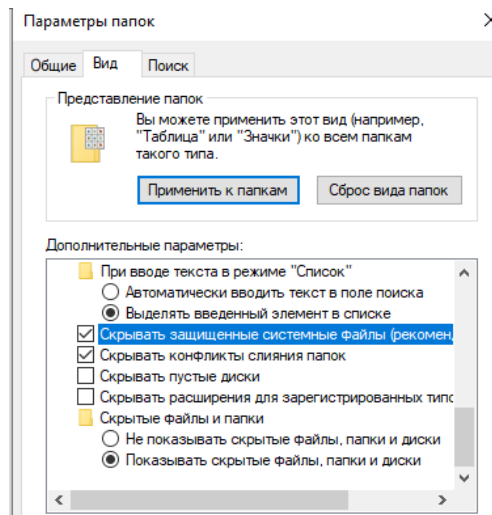
Powershell – Get-NetFirewallApplicationFilter | Select Program | Where-Object Program -notlike “*systemroot*” | Where-Object Program -like “*exe*”).



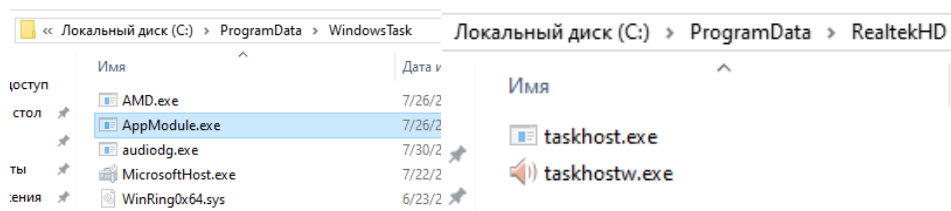
7. Створення нового схованого користувача “John”, при цьому самого користувача у списку користувачів – немає.



8. Додався прапорець «приховувати системні файли».



9. Вірусні файли мають атрибути «схований та системний», мають іконку Realtek, мають таку ж назву як і у легітимних системних файлів (C:\Windows\System32\taskhostw.exe – системний процес для запуску запланованих завдань).



10. Вірус має функціонал anti-debugging, що проявляється у завершенні процесу Диспетчера задач (taskmgr.exe) та при переході до вірусної папки будь-яким способом – завершенні Провідника (explorer.exe).

11. Додатково можна зазначити, що частина вірусних файлів які ми бачимо у виключеннях антивірусу – не завантажилися.

Наприклад, файл winserv.exe це компонент Remote Utilities Server / Remote Manipulator System – легальний інструмент віддаленого адміністрування, але при цьому, JoeSandbox визначає його як Trojan/Bot, Evader, Ransomware, що показано на рис. 2.27 [72, 73]. Через наявний функціонал «налаштувати свій RMS-сервер» RMS зазвичай використовується у різних ШПЗ в якості RAT (Remote Access Trojan / Remote Administration Tools) [73].

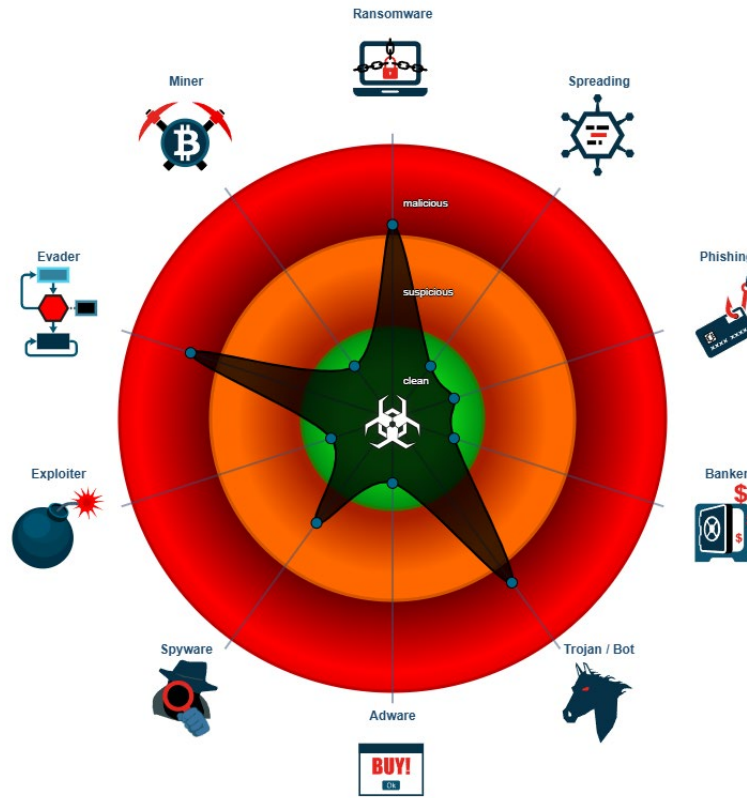


Рис. 2.27. Автоматична категоризація файлу winserv.exe [72]

Наявні сліди, залишені після впливу ШПЗ на ПК, систематизовано в таблиці 2.1.

Таблиця 2.1
Артефакти які залишились після впливу ШПЗ на ПК (створено автором)

Артефакт	Які сліди залишаються	Що може спричинити
Вірусні файли\каталоги у списку виключень Windows Defender	C:\Program Files\RDP Wrapper C:\ProgramData C:\ProgramData\RealtekHD\taskhost.exe C:\ProgramData\Windows Tasks Service\winserv.exe C:\ProgramData\WindowsTasks\AMD.exe C:\ProgramData\WindowsTasks\AppModule.exe C:\ProgramData\WindowsTasks\audiodg.exe C:\ProgramData\WindowsTasks\MicrosoftHost.exe C:\Windows\KMS C:\Windows\KMSAutoS C:\Windows\System32 C:\Windows\System32\SppExtComObjHook.dll C:\Windows\System32\SppExtComObjPatcher.exe	Неможливість для АВ самостійно видалити віруси

Продовження таблиці 2.1

Артефакт	Які сліди залишаються	Що може спричинити
Список вірусних процесів у пам'яті	C:\ProgramData\RealtekHD\taskhost.exe C:\ProgramData\RealtekHD\taskhostw.exe C:\ProgramData\WindowsTasks\audiodg.exe C:\ProgramData\WindowsTasks\MicrosoftHost.exe	Доки вірус запущено(знаходиться у пам'яті) – він виконує потрібні зловмиснику дії.
Сховані папки у каталозі програм, що мають назву таку ж, як і у антивірусних рішень	Папки та файли з атрибутами "приховані" і "системні" в каталогах C:\Program Files, доступ до яких заборонено користувачу, що запустив вірус: AFAST Software, AVG, ByteFence, COMODO, Enigma Software Group, Malwarebytes, SpyHunter.	Неможливість встановити альтернативні антивірусні рішення
Додаткові користувачі на ПК	User: John User Profile: C:\Users\John	Додаткові користувачі, яких створює вірус, будуть мати права Administrators\Remote Desktop Users, що дозволить отримати віддалений доступ до ПК з правами адміністратора
Зміна налаштувань Windows ACL	Папки та файли з атрибутами "приховані" і "системні" в каталогах C:\Program Files Icacls C:\Program Files\	Недоступність легітимних користувачів до папок\файлів, неможливість їх видалити\перейменувати, повний доступ до папок і файлів для нелегітимних користувачів навіть при відсутності додавання в адміністративні групи
Додавання автозапуск реєстра	HKLM\Software\Microsoft\Windows\CurrentVersion\Run "Realtek HD Audio" C:\ProgramData\RealtekHD\taskhostw.exe	Завантаження вірусних файлів при перезавантаженні ПК, їх персистентність
Додавання автозапуск TaskSheduller	TaskSheduller - \Microsoft\Windows\WindowsBackup\OnlogonCheck C:\ProgramData\RealtekHD\taskhostw.exe \Microsoft\Windows\WindowsBackup\RealtekCheckC:\ProgramData\RealtekHD\taskhost.exe \Microsoft\Windows\WindowsBackup\TaskCheck C:\ProgramData\RealtekHD\taskhostw.exe \Microsoft\Windows\WindowsBackup\WinlogonCheck C:\ProgramData\RealtekHD\taskhost.exe	Завантаження вірусних файлів при окремих подіях

Продовження таблиці 2.1

Артефакт	Які сліди залишаються	Що може спричинити
Вірусні файли/каталоги у списку виключень Windows Firewall	C:\ProgramData\WindowsTasks\AMD.exe C:\ProgramData\WindowsTasks\AppModule.exe	Безперешкодний доступ до інтернету для вірусних файлів, без додаткового повідомлення користувача
Приховано від користувача приховані файли	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Advanced\Folder\Hidden\SHOWALL CheckedValue 1 HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced «Hidden 1», «HideFileExt 1»	Навіть якщо у користувача до цього було увімкнено показ прихованих файлів – вірус сховає вірусні файли від користувача
Приховано від користувача вірусні файли та папки	Папки та файли з атрибутами "приховані" і "системні" в каталогах C:\Program Files, C:\ProgramData\RealtekHD, C:\ProgramData\WindowsTasks (файл з вразливим драйвером WinRing0x64.sys окремо до виключень не додавався)	Користувач не може побачити нові невідомі файли та насторожитися

В якості ідентифікаційних ознак стану ІКМ вибрано артефакти, подані в таблиці 2.2.

Таблиця 2.2
Артефакти для ідентифікації стану ІКМ (створено автором)

Артефакт	Що може спричинити	Де сліди залишаються\як їх вивантажити	Яку інформацію можна отримати зі слідів
Вірусні файли/каталоги у списку виключень Windows Defender	Неможливість для АВ самостійно знайти та видалити віруси	Regedit HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Defender\Exclusions\Processes Powershell – (Get-MpPreference).ExclusionPath)	Усі можливі варіанти шляхів і файлів в якому знаходяться віруси (100% вірусна активність за наявності якихось записів)
Список вірусних процесів у пам'яті	Доки вірус запущено(знаходиться у пам'яті) – він виконує потрібні зловмиснику дії. Являється основним індикатором компрометації	wmic process get ExecutionPath	Повний шлях до вірусного файлу (100% вірусна активність за наявності аномалії у назві процесу і стандартному шляху де він повинен зберігатися)

Продовження таблиці 2.2

Артефакт	Що може спричинити	Де сліди залишаються\як їх вивантажити	Яку інформацію можна отримати зі слідів
Додаткові користувачі на ПК	Додаткові користувачі які створює вірус, будуть мати права Administrators\Remote Desktop Users, що дозволить отримати віддалений доступ до ПК з правами адміністратора	wmic useraccount where localaccount=true name,sid,disabled Registry HKEY_LOCAL_MACHINE\SAM HKEY_LOCAL_MACHINE\SECURITY	Усі користувачі чий SID закінчується на -1000 (і вище) – додатковий користувач. Може свідчити про наявність вірусної активності
Сховані папки у каталозі програм, що мають назву таку ж як і у антивірусних рішень	Неможливість встановити альтернативні антивірусні рішення для видалення вірусів	C:\Program Files\	На комп'ютері 100% вірусна активність, яка має модуль захисту від антивірусів
Зміна налаштувань Windows ACL	Недоступність легітимних користувачів до папок\файлів, неможливість повного доступу до папок і файлів для нелегітимних користувачів навіть за відсутності додавання в адміністративні групи	Icacls C:\ Icacls "C:\Program Files\"	Які користувачі є нелегітимними. 100% вірусна активність
Додавання в автозапуск реєстру	Завантаження вірусних файлів при перезавантаженні ПК	HKLM\Software\Microsoft\Windows\CurrentVersion\Run HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnce HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon HKLM\SYSTEM\CurrentControlSet\Services	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих файлів – 100% вірусна активність

Продовження таблиці 2.2

Артефакт	Що може спричинити	Де сліди залишаються\як їх вивантажити	Яку інформацію можна отримати зі слідів
Додавання в автозапуск TaskSheduler	Завантаження вірусних файлів при окремих подіях	TaskSheduler HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих завдань - 100% вірусна активність
Вірусні файли/каталоги у списку виключень Windows Firewall	Безперешкодний доступ віруса до інтернету без додаткового запиту користувача	CMD – netsh advfirewall firewall show rule name=all verbose find “C:\” find /v “C:\Windows\system32” HKLM\Software\Policies\Microsoft\WindowsFirewall\FirewallRules	Перелік вірусних файлів, яким необхідне інтернет-з’єднання, якщо є якісь файли з тимчасової папки користувача це 100% вірусна активність
Приховано від користувача приховані файли	Навіть якщо у користувача до цього було увімкнено показ прихованих файлів, вірус сховає вірусні файли від користувача, що робиться для того щоб користувач не побачив вірусні файли у каталогах	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Advanced\Folder\Hidden\SHOWALL CheckedValue 1 HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced\«Hidden 1», «HideFileExt 1» dir C:\ /A:HS	Частина вірусних файлів містить атрибути «прихований» та «системний», майже всі файли з такими атрибутами, що не знаходяться в системних папках – будуть свідчити про наявність вірусу
Prefetch	-	C:\Windows\Prefetch	Назву будь-коли запущеного .exe файлу та час коли він запускався
AppCompat Cache	-	HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\AppCompatCache\	Інформацію про запущені файли на ПК та повний шлях до них
Amcache	-	%SystemRoot%\AppCompat\Programs\Amcache.hve	Інформацію про запущені файли та час їх запуску
SRUM	-	C:\Windows\System32\sru\SRU DB.dat	Інформацію про запущені програми (з повним шляхом) конкретним користувачем (містить назву та SID) та у конкретний час

Продовження таблиці 2.2

Артефакт	Що може спричинити	Де сліди залишаються\як їх вивантажити	Яку інформацію можна отримати зі слідів
ShellBags	-	%UserProfile%\NTUSER.DAT HKCU\Software\Microsoft\Windows\Shell %UserProfile%\USRCLASS.DAT HKCU\Software\Classes\Local Settings\Software\Microsoft\Windows\Shell	Інформацію про запущені команди конкретним користувачем через Win+R та час виконання, може містити шкідливі команди
Інформація про .zip .rar .exe .dll в каталогах користувача	-	Dir %userprofile% /r /s /b	Назву і шлях до виконуваних файлів, які є у користувача, що іноді може показати, який користувач завантажив і відкрив ШПЗ

З викладеного, можна зробити висновок, що наявність зазначених слідів на досліджуваному вузлі ІКМ свідчить про наявність зараження або цим конкретним видом вірусу, або його іншим різновидом. Запропоновані таблиці артефактів дозволяють більш ефективно виявляти наявність ШПЗ.

2.7. Метод вивантаження цифрових слідів в умовах обмеженості ресурсів

Питаннями вивантаження та аналізу цифрових слідів займається компютерна форензика (computer forensics). Для повноцінного forensic-аналізу потрібен повний образ диску та дамп пам'яті, але його використання у ІКМ підприємства зазвичай є надмірним. У той же час, використання всіх слідів, що пропонують програми для проведення forensic triage теж є надмірним. Так, недоцільно завантажувати усю таблицю MFT (яка буде займати близько 2ГБ пам'яті), якщо можна вивантажити список файлів із задалегіть детермінованих папок. Також, недоцільно створювати повний дамп RAM, який буде важити більше 4ГБ тільки для того, щоб переглянути перелік процесів. Разом з тим, якщо корпоративні вузли ІКМ мають увімкнене за

замовчуванням повнодискове шифрування Bitlocker то дамп RAM вивантажиться, але його не можна буде прочитати фреймворком Volatility, що буде означати, що аналітик дарма витратив час на завантаження дампу та його безрезультатний аналіз.

Зазначене вказує на недоцільність вивантаження всіх наявних артефактів, що обумовлює необхідність визначення релевантних цифрових слідів, необхідних для визначення стану вузла ІКМ.

2.7.1. Порівняння програм для проведення Forensic Triage

Для вивантаження цифрових слідів, існує ряд програм для проведення поверхневого збору і аналізу цифрових слідів – Forensic Triage.

Частина програм для Forensic Triage була перевірена з дефолтними налаштуваннями та без створення дампу пам'яті, на ПК з наступними характеристиками: Windows 11, AMD Ryzen 9 7900X, 64GB RAM

Дані по результатам виконання програм та їх порівняння з ручним вивантаженням подано у таблиці 2.3.

Таблиця 2.3

Порівняння програм для Windows Forensic Triage (створено автором)

Метод	Час	Розмір артефактів (МБ)	Повнота даних
Ручний збір/перегляд артефактів	Більше години	-	Повний
LiveResponseCollection-CedarDelta	14хв 9сек – secure 13хв 5сек - triage	Архів SecureTriage - 172 Triage - 2263	Достатньо даних
WinTriage_v1.py	-	-	Дуже мало даних
CyLR 3.0	46сек	347 - архів	Відсутні дані по запущеним процесам, не обробляє дані (результат у вигляді системних файлів ОС)

Продовження таблиці 2.3

Метод	Час	Розмір артефактів (МБ)	Повнота даних
DFIRtrriage_v6	19сек	39,4	Відсутні дані реєстру (наявні тільки NTUSER.DAT & USRCLASS.DAT)
cdir-collector_1.3.7	1хв 9сек	2061,61	Відсутні дані по запущеним процесам
velociraptor-v0.75.6	Більше 2хв (помилка)	25,8 – архів 300 – лог з помилками	Відсутні дані реєстру, інше зібрано, але помилка не дає завершити роботу
КАРЕ	1хв 7сек	748	Достатньо даних

Як видно з таблиці, зазначені програми працюють досить швидко, проте в них відсутня важлива інформація для аналізу (наприклад, запущені процеси і дані реєстру) а також результат їх аналізу не завжди зручний для автоматичного аналізу. Частина програм здійснює архівацію слідів на досліджуваному вузлі ІКМ для економії місця, однак, розірвання з'єднання при передачі архіву в розподіленій мережі (швидкість передачі даних – до 20Мбіт/сек), призводить до неотримання жодної інформації про стан вузла (оскільки архів не відкриється), на відміну від того, якби текстові дані передавались у неархівованому вигляді.

Зазначене обумовлює необхідність створення такого методу вивантаження слідів, який би враховував релевантні сліди, відкидав дублікати, не копіював громіздкі та малоефективні сліди (наприклад, дані MFT або системні журнали Windows, оскільки дані в них можуть бути спотворені).

2.7.2. Реляційна модель релевантних цифрових слідів.

Постійно зростає кількість кібератак на корпоративні ПК (вузли ІКМ) внаслідок війни або неякісно побудованої системи захисту, може трапитись ситуація, коли потрібно майже одночасно перевірити велику кількість ПК (500

шт.) на предмет вірусного зараження. Зазвичай це може трапитись при випадковому виявленні вірусних файлів типу «мережевий хробак» за допомогою антивірусного захисту (АВ) чи Intrusion Prevention System (IPS), що в свою чергу накладає значне обмеження в часі, оскільки потрібно швидко встановити наявність зараженості певного ПК і ізолювати його від інших комп'ютерів інфокомунікаційної мережі (ІКМ). В такому випадку традиційний forensic-аналіз цифрових слідів, що був зібраний за допомогою Forensic Triage буде занадто довгим. Оскільки, чим менше буде займане місце в пам'яті під цифрові сліди та чим швидше буде проведено їх вивантаження з досліджуваного вузла та аналіз, тим краще.

Для обґрунтування ефективних рішень для зменшення часу аналітика інформаційної безпеки на виявлення зараженості конкретного ПК інфокомунікаційної мережі у якості «заражено/не заражено». необхідно вирішити наступні задачі:

1. Визначити компонент/тактику, без яких сучасні комп'ютерні віруси зазвичай не працюють.
2. Запропонувати список програм для швидкого виявлення вірусів.
3. Розробити реляційну таблицю та скрипт оптимізації, який дозволяє скоротити кількість елементів, необхідних для подальших досліджень.

Згідно з даними MITRE ATT&CK Matrix визначено, що 230 відомих вірусів (і АРТ, які вибирають цю техніку) запускаються за допомогою техніки T1547 Registry Run Keys / Startup Folder, 148 відомих вірусів запускаються за допомогою техніки T1053 Scheduled Task, а інші техніки для автозапуску менш поширені [74, 75].

Тож не має значення, який тип вірусу сховався на конкретному ПК: троян, вимагач, RAT або мережевий хробак, та які механізми приховування від користувача вони реалізують, якщо кожному з них важлива персистентність (persistence) – можливість отримати керування при перезапуску ПК. Це означає, що ми можемо відкинути всі інші цифрові сліди

та методи протидії вірусам, що «ховаються» та визначити перелік програм з автозавантаження. Зважаючи на перелік поширених шляхів автозапуску у вірусів [76], для автоматичного збору цифрових слідів ми будемо використовувати відомі програми, що дозволить знизити кількість помилок і зекономити час, такі як Autoruns [70, 77].

За даними програми Autoruns від Sysinternals, є наступні можливі місця для автозапуску: Boot execute; Codecs; Appinit DLLs; Explorer addons; Sidebar gadgets; Image hijacks; Internet Explorer addons; Known DLLs; Logon startups; WMI entries; Winsock protocol and network providers; Office addins; Printer monitor DLLs; LSA security providers; Autostart services and non-disabled drivers; Scheduled tasks; Winlogon entries.

Цього переліку цілком достатньо щоб відловити більшість вірусів, оскільки більшість з них не використовують руткіти та технології для схову своїх цифрових слідів на ПК. Також, за допомогою Listdlls та wmic бажано зібрати інформацію про поточні запущені процеси та динамічні бібліотеки з повними шляхами до них, що дозволить виявити можливі віруси, що вже завантажені в пам'ять комп'ютера [78].

Додатково, потрібно вивантажити інформацію про виключення Windows Defender & Windows Firewall, щоб отримати інформацію про ще невідомі вірусні компоненти (або їх місцезонашування) та легітимних програм які мають доступ до інтернету, але можуть використовуватись для ексфільтрації даних. Це майже не вплине на швидкість відправлених даних та їх перегляд, але дасть остаточне уявлення про систему.

Наприклад, легальні програми та звичайні користувачі не додають файли/папки до виключень антивірусу; якщо щось знайдено у виключеннях антивірусу – це свідчить про 100% шкідливу активність.

Є деякі програми, які ми можемо використовувати під час дослідження комп'ютера, але вони за замовчуванням мають недостатню оптимізацію, оскільки вони витягують всю інформацію, яка в даному випадку не потрібна [70, 78, 79, 80]:

Autoruns.exe – найбільш функціональна утиліта для пошуку місць автоматичного запуску, монітора запуску, програм, які налаштовані на запуск під час завантаження системи або входу користувача. Програма Autoruns.exe включає перевірки в папці автозавантаження, Run та інших розділах реєстру, розширеннях Shell Explorer, панелях інструментів, ВНО, Winlogon, службах автозапуску та багато іншого (див. рис. 2.28).

```
Sysinternals Autoruns v14.11 - Autostart program viewer
Copyright (C) 2002-2024 Mark Russinovich
Sysinternals - www.sysinternals.com

Autorunsc shows programs configured to autostart during boot.

Usage: autorunsc [-a <*[bdeghiklmoprsw>] [-c|-ct] [-h] [-m] [-s]
ofile>] | [user]]
-a Autostart entry selection:
* All.
b Boot execute.
c Codecs.
d Appinit DLLs.
e Explorer addons.
g Sidebar gadgets (Vista and higher)
h Image hijacks.
i Internet Explorer addons.
k Known DLLs.
l Logon startups (this is the default).
m WMI entries.
n Winsock protocol and network providers.
o Office addins.
p Printer monitor DLLs.
r LSA security providers.
s Autostart services and non-disabled drivers.
t Scheduled tasks.
w Winlogon entries.
-c Print output as CSV.
```

Рис. 2.28. Консольна версія утиліти Autoruns

ListDLLs.exe – утиліта, яка повідомляє про DLL, завантажені в процеси, перераховує всі DLL, завантажені в усі процеси, і відображає інформацію про повну версію для DLL (див. рис. 2.29).

```
Listdlls v3.2 - Listdlls
Copyright (C) 1997-2016 Mark Russinovich
Sysinternals

usage: listdlls [-r] [-v | -u] [processname|pid]
usage: listdlls [-r] [-v] [-d dllname]
processname Dump DLLs loaded by process (partial name accepted)
pid Dump DLLs associated with the specified process id
dllname Show only processes that have loaded the specified DLL.
-r Flag DLLs that relocated because they are not loaded at
their base address.
-u Only list unsigned DLLs.
-v Show DLL version information.
```

Рис. 2.29. Утиліта ListDLLs

Netsh.exe – утиліта, яка дає можливість відобразити або змінити конфігурацію мережі віддаленого та локального комп'ютера за допомогою команд netsh (див. рис. 2.30).

```
Применение: Netsh [-a AliasFile] [-c Context] [-r RemoteMachine]
                [-u [DomainName\]UserName] [-p Password | *]
                [Command | -f ScriptFile]
```

Рис. 2.30. Утиліта Netsh

PowerShell.exe – інструмент командного рядка, який запускає сеанс Windows PowerShell у вікні командного рядка (CLI), частина довідки показана на рис. 2.31.

```
PowerShell[.exe] [-PSConsoleFile <file> | -Version <version>]
                [-NoLogo] [-NoExit] [-Sta] [-Mta] [-NoProfile] [-NonInteractive]
                [-InputFormat {Text | XML}] [-OutputFormat {Text | XML}]
                [-WindowStyle <style>] [-EncodedCommand <Base64EncodedCommand>]
                [-ConfigurationName <string>]
                [-File <filePath> <args>] [-ExecutionPolicy <ExecutionPolicy>]
                [-Command { - | <script-block> [-args <arg-array>]
                | <string> [<CommandParameters>] } ]

PowerShell[.exe] -Help | -? | /?
```

Рис. 2.31. Консольна версія PowerShell

Wmic.exe – утиліта командного рядка WMI, яка надає інтерфейс командного рядка для інструментарію керування Windows (WMI). WMIC сумісний з існуючими оболонками та командами утиліт. Слід зазначити, що ця утиліта в нових версіях Windows 11 була видалена. Зразок довідки про вивантаження процесів (wmic process /?) подано на рис 2.32.

```
PROCESS - Управление процессами.
СОВЕТ. BNF при работе с псевдонимом.
(<псевдоним> [объект WMI] | <псевдоним> [<путь_WHERE>] | [<псевдоним> <путь_WHERE>] [<предложение_команды>].

Использование:
PROCESS ASSOC [<указатель_формата>]
PROCESS CALL <имя_метода> [<список_фактических_параметров>]
PROCESS CREATE <список_значений>
PROCESS DELETE
PROCESS GET [<список_свойств>] [<параметры_GET>]
PROCESS LIST [<формат_LIST>] [<параметры_LIST>]
```

Рис. 2.32. Утиліта WMIC

Для вирішення згаданої вище проблеми неналежної оптимізації деяких програм запропоновано реляційну модель у вигляді таблиці артефактів (табл. 2.4-2.6) для ідентифікації стану вузлів ІКМ.

Таблиця 2.4

Таблиця мінімально необхідних артефактів (створено автором)

Артефакти	Яку інформацію можна отримати зі слідів?
Executed Processes	Повний шлях до вірусного файлу (100% вірусна активність за наявності аномалії у назві процесу і стандартному шляху де він повинен зберігатися)
Windows Defender	Усі можливі варіанти шляхів і файлів в якому знаходяться віруси (100% вірусна активність за наявності якихось записів)
Windows Firewall	Перелік вірусних файлів, яким необхідне інтернет-з'єднання, якщо є якісь файли з тимчасової папки користувача це 100% вірусна активність
Autorun Entries	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих файлів – 100% вірусна активність
Loaded DLLs	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих файлів – 100% вірусна активність

Таблиця 2.5

Характеристики програм, якими можна знехтувати (створено автором)

Характеристика програми	Для чого і чому можна відкинути?
Потрібні тільки дані з диску C:\	Для відкидання усіх строк, що не містять у собі повний шлях до програми
Програма відсутня у системному каталозі C:\Windows\system32\	Оскільки більшість слідів залишають саме такі програми, їх відкидання пришвидшить оцінку зараженості ПК без кардинального зниження ефективності
Програма відсутня у каталозі програм Program Files	Для виявлення саме вірусної активності не потрібна інформація про легітимні програми
Показувати тільки невідомі бібліотеки	Підписані бібліотеки будуть легітимними тож їх можна не враховувати (наприклад, user32.dll)
Hide Microsoft entries	Для відкидання верифікованих легітимних програм

Реляційна модель у вигляді таблиці артефактів (створено автором)

ID	Характеристика програми	Executed Processes	Windows Defender	Windows Firewall	Autorun Entries	Loaded DLLs
0	characteristic	exec proc	av defender	firewall	autoruns	load dll
1	Потрібні тільки дані з диску C:\	Yes	No	Yes	No	Yes
2	Програма відсутня у системному каталозі C:\Windows\system32\	Yes	No	Yes	No	No
3	Програма відсутня у каталозі програм Program Files	Yes	No	Yes	No	Yes
4	Показувати тільки непідписані бібліотеки	No	No	No	No	Yes
5	Сховати файли Microsoft	No	No	No	Yes	No

Визначено, що деякі сліди не потребують додаткових програм, а можуть використовувати вбудовані утиліти [81, 82, 83]. Представлена модель реалізується на базі скрипта .bat (створено автором), який оптимізує вихід програм, що представлено нижче:

```
@Echo off
cd %~dp0
wmic process get ExecutablePath | find "C:\\" | find /v /i "C:\Windows\system32\" |
find /v /i "Program Files" >%computername%_process.txt
powershell -noprofile -command "(Get-MpPreference).ExclusionPath"
>%computername%_windows_defender.txt
echo.>>%computername%_windows_defender.txt
powershell -noprofile -command "(Get-MpPreference).ExclusionExtension"
>>%computername%_windows_defender.txt
echo.>>%computername%_windows_defender.txt
powershell -noprofile -command "(Get-MpPreference).ExclusionProcess"
>>%computername%_windows_defender.txt
netsh advfirewall firewall show rule name=all verbose | find "C:\\" | find /v /i
"C:\Windows\system32\" | findstr /v /i /C:"Program Files"
>%computername%_firewall.txt
Listdlls64.exe -u | findstr /i C: | findstr /v /i /C:"Program Files"
>%computername%_listdll.txt
autorunsc64.exe -accepteula -nobanner -a * -c -h -m -s -o
%computername%_autorun.csv
pause
```

В результаті роботи скрипта, отримано файли з необхідними артефактами у тій же самій директорії де розташований скрипт, цієї інформації буде цілком достатньо для виявлення зараженості ПК (див. рис. 2.33).

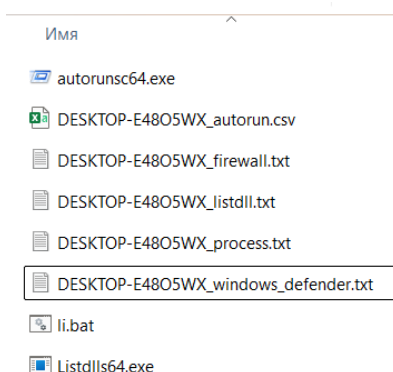


Рис. 2.33. Результат роботи скрипта

Для визначення ефективності оптимізації програми розроблений скрипт протестовано на одному ПК з ОС Windows 11 у порівнянні з програмами за замовчуванням. Результати виконання програм визначаються у кількості вихідних даних і наведені у таблиці 2.7.

Таблиця 2.7

Результат виконання програм у строках (створено автором)

Програма	Вихідні дані за замовчуванням	Оптимізовані вихідні дані
Autoruns64.exe	1638	123
Listdlls64.exe	1997	178
Netsh (adwfirewall)	10103	13
Powershell (Defender Exclusion)	3	3
Wmic (process)	318	36

Дані, наведені у таблиці 2.7 показують, що кількість артефактів, які потрібно було переглядати аналітику, суттєво скоротилася (у деяких випадках навіть більш ніж у десять разів). При такій «грубій» оптимізації, де відкидається значна частина артефактів Windows і беруться не всі дані автозапуску, можна пропустити певну кількість вірусів, але при великій кількості досліджуваних ПК, зменшення часу, необхідного аналітику для аналізу кожного ПК стає більш пріоритетним. На думку авторів, достовірність

визначення вірусної активності за розробленим алгоритмом становить від 60% до 100%. Крім того, 100% надійність виникає, якщо є будь-які записи у антивірусних виключеннях Windows Defender.

У роботі визначено компонент/тактику, без яких сучасні комп'ютерні віруси зазвичай не працюють. Запропоновано перелік програм для швидкого виявлення вірусів і скрипт оптимізації з використанням реляційної таблиці артефактів, які дозволяють скоротити кількість елементів, необхідних для подальших досліджень більш ніж у десять разів.

Ефективні рішення, що забезпечують зменшення часу аналітика на виявлення скомпрометованих ПК:

1. Першочергова перевірка місць автозапуску та артефактів, що дають 100% інформації про вірусне зараження.
2. Використання сукупності рекомендованих програм для швидкого виявлення вірусів.
3. Застосування реляційної таблиці артефактів та скрипту оптимізації, які дозволяють суттєво скоротити кількість елементів, необхідних для подальших досліджень.

Запропоновані рішення доцільно застосувати для розширеного переліку артефактів з метою їх оптимізації, підвищення надійності результатів і автоматизації рутинної роботи зі збору цифрових артефактів для аналітика інформаційної безпеки.

2.7.3. Ранжування цифрових слідів у відповідності до їх значущості для різних кейсів

Типовими інцидентами ІБ є:

1. Зараження ШПЗ;
2. Встановлення програм віддаленого доступу, або неліцензійних програм;
3. Нелегітимні/несанкціоновані дії невідомого користувача.

Дослідження можливих цифрових слідів доцільно здійснювати із застосуванням артефактів, зазначених в таблиці 2.2, вивантаження яких надасть практично повне уявлення про стан вузла. Проте, зазвичай, для працівників ІБ на підприємствах, важливо у ході аналізу відповісти на такі питання:

1. Чи має даний комп'ютер ознаки роботи шкідливого програмного забезпечення (ШПЗ)?
2. Як можна полагодити комп'ютер після атаки ШПЗ?
3. Чи існує на комп'ютері неліцензоване ПЗ або програми віддаленого доступу?
4. Який користувач завантажив неліцензійне ПЗ або ШПЗ?

Отже, можна сформувати набір слідів у відповідності до питань, що в свою чергу дасть уявлення про те, які сліди першочергово потрібно вивантажувати та аналізувати (див табл. 2.8-2.11).

Сформуємо наші сліди у відповідності до питань:

1. Чи має даний комп'ютер ознаки роботи шкідливого програмного забезпечення (ШПЗ)?

Таблиця 2.8

Перелік слідів для питання 1 (створено автором)

Артефакт	Що може спричинити?	Які сліди залишаються	Яку інформацію можна отримати зі слідів?
Список вірусних процесів у пам'яті	Доки вірус запущено(знаходиться у пам'яті) – він виконує потрібні зловмиснику дії. Являється основним індикатором компрометації	wmic process get ExecutionPath	Повний шлях до вірусного файлу (100%ва вірусна активність за наявності аномалії)
Вірусні файли/каталоги у списку виключень Windows Defender	Неможливість для АВ самостійно видалити віруси	Regedit HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths	Усі можливі варіанти шляхів і файлів в якому знаходяться віруси (100% вірусна активність за наявності якихось записів)
Додавання в автозапуск реєстру	Завантаження вірусних файлів при перезавантаженні ПК	HKLM\Software\Microsoft\Windows\CurrentVersion\Run	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих файлів – 100% вірусна активність

Продовження таблиці 2.8

Артефакт	Що може спричинити?	Які сліди залишаються	Яку інформацію можна отримати зі слідів?
Додавання в автозапуск TaskSheduller	Завантаження вірусних файлів при окремих подіях	TaskSheduller - \Microsoft\Windows\WindowsBackup	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих завдань - 100% вірусна активність
Вірусні файли/каталоги у списку виключень Windows Firewall	Безперешкодний доступ до інтернету	CMD – netsh advfirewall firewall show rule name=all verbose find "C:\" find /v "C:\Windows\system32\"	Перелік вірусних файлів, яким необхідне інтернет-з'єднання, якщо є якісь файли з тимчасової папки користувача це 100% вірусна активність

2. Як можна полагодити комп'ютер після атаки ШПЗ?

Таблиця 2.9

Перелік слідів для питання 2 (створено автором)

Артефакт	Що може спричинити?	Які сліди залишаються	Яку інформацію можна отримати зі слідів?
Вірусні файли/каталоги у списку виключень Windows Defender	Неможливість для АВ самостійно видалити віруси	Regedit HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths	Усі можливі варіанти шляхів і файлів в якому знаходяться віруси (100% вірусна активність за наявності якихось записів)
Список вірусних процесів у пам'яті	Доки вірус запущено(знаходиться у пам'яті) – він виконує потрібні зловмиснику дії.	wmic process get ExecutionPath	Повний шлях до вірусного файлу (100%ва вірусна активність за наявності аномалії)
Додавання в автозапуск реєстру	Завантаження вірусних файлів при перезавантаженні ПК	HKLM\Software\Microsoft\Windows\CurrentVersion\Run	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих файлів – 100% вірусна активність
Додавання в автозапуск TaskSheduller	Завантаження вірусних файлів при окремих подіях	TaskSheduller - \Microsoft\Windows\WindowsBackup	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих завдань - 100% вірусна активність
Вірусні файли/каталоги у списку виключень Windows Firewall	Безперешкодний доступ до інтернету	CMD – netsh advfirewall firewall show rule name=all verbose find "C:\" find /v "C:\Windows\system32\"	Перелік вірусних файлів, яким необхідне інтернет-з'єднання, якщо є якісь файли з тимчасової папки користувача це 100% вірусна активність

3. Чи існує на комп'ютері неліцензоване ПЗ або програми віддаленого доступу?

Таблиця 2.10

Перелік слідів для питання 3 (створено автором)

Артефакт	Що може спричинити?	Які сліди залишаються	Яку інформацію можна отримати зі слідів?
Вірусні файли/каталоги у списку виключень Windows Defender	Неможливість для АВ самостійно видалити віруси	Regedit HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths	Усі можливі варіанти шляхів і файлів в якому знаходяться віруси (100% вірусна активність за наявності якихось записів)
Список вірусних процесів у пам'яті	Доки вірус запущено(знаходиться у пам'яті) – він виконує потрібні зловмиснику дії. Являється основним індикатором компрометації	wmic process get ExecutionPath	Повний шлях до вірусного файлу (100%ва вірусна активність за наявності аномалії у назві процесу і стандартному шляху де він повинен зберігатися)
Зміна налаштувань Windows ACL	Недоступність легітимних користувачів до папок чи файлів	Icacls /? C:\Program Files\	Які користувачі є нелегітимними. 100% вірусна активність
Додавання в автозапуск реєстру	Завантаження вірусних файлів при перезавантаженні ПК	HKLM\Software\Microsoft\Windows\CurrentVersion\Run	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих файлів – 100% вірусна активність
Додавання в автозапуск TaskSheduller	Завантаження вірусних файлів при окремих подіях	TaskSheduller - \Microsoft\Windows\WindowsBackup	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих завдань - 100% вірусна активність
Вірусні файли/каталоги у списку виключень Windows Firewall	Безперешкодний доступ до інтернету	CMD – netsh advfirewall firewall show rule name=all verbose find "C:\:" find /v "C:\Windows\system32\"	Перелік вірусних файлів, яким необхідне інтернет-з'єднання, якщо є якісь файли з тимчасової папки користувача це 100% вірусна активність
Приховано від користувача приховані файли	Робиться для того, щоб користувач не побачив вірусні файли у каталогах	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Advanced\Folder\Hidden\SHOWALL CheckedValue 1 HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced «Hidden 1», «HideFileExt 1»	Частина вірусних файлів містить атрибути «прихований» та «системний», майже всі файли з такими атрибутами, що не знаходяться в системних папках – будуть свідчити про наявність вірусу

4. Який користувач завантажив неліцензійне ПЗ або ШПЗ?

Таблиця 2.11

Перелік слідів для питання 4 (створено автором)

Артефакт	Що може спричинити?	Які сліди залишаються	Яку інформацію можна отримати зі слідів?
Вірусні файли/каталоги у списку виключень Windows Defender	Неможливість для АВ самостійно видалити віруси	HKLM\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths	Усі можливі варіанти шляхів і файлів в якому знаходяться віруси (100% вірусна активність за наявності якихось записів)
Список вірусних процесів у пам'яті	Доки вірус запущено він виконує потрібні зловмиснику дії.	wmic process get ExecutionPath	Повний шлях до вірусного файлу (100%ва вірусна активність за наявності аномалії)
Сховані папки у каталозі програм, що мають назву таку ж як і у антивірусних рішень	Неможливість встановити альтернативні АВ	C:\Program Files\	На комп'ютері 100% вірусна активність, яка має модуль захисту від антивірусів
Зміна налаштувань Windows ACL	Недоступність до папок чи файлів	Icacls /? C:\Program Files\	Які користувачі є нелегітимними. 100% вірусна активність
Додавання в автозапуск реєстру	Завантаження вірусних файлів при перезавантаженні ПК	HKLM\Software\Microsoft\Windows\CurrentVersion\Run	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих файлів – 100% вірусна активність
Додавання в автозапуск TaskSheduller	Завантаження вірусних файлів при окремих подіях	\Microsoft\Windows\WindowsBackup	Перелік файлів, які критичні для функціонування і персистентності вірусу, за наявності невідомих завдань - 100% вірусна активність
Вірусні файли/каталоги у списку виключень Windows Firewall	Безперешкодний доступ до інтернету	CMD – netsh advfirewall firewall show rule name=all verbose find "C:" find /v "C:\Windows\system32"	Перелік вірусних файлів, яким необхідне інтернет-з'єднання, якщо є якісь файли з тимчасової папки користувача це 100% вірусна активність
Приховано від користувача приховані файли	Робиться для того, щоб користувач не побачив вірусні файли у каталогах	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Advanced\Folder\Hidden\SHOWALL CheckedValue 1 HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\Advanced «Hidden 1», «HideFileExt 1»	Частина вірусних файлів містить атрибути «прихований» та «системний», майже всі файли з такими атрибутами, що не знаходяться в системних папках – будуть свідчити про наявність вірусу

Як видно з таблиці, даних про виключення Microsoft Windows Defender, даних автозапуску та списку процесів з повними шляхами до них в цілому достатньо, щоб визначити чи заражено конкретний комп'ютер чи ні. А також застосування таблиць дає розуміння, які сліди потрібно першочергово вивантажувати та аналізувати, в залежності від поставлених питань.

Висновки до розділу 2

В результаті проведених теоретичних досліджень доведено:

1. Метод захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, побудований за принципом послідовного циклічного звернення до операторів ідентифікації, прийняття рішення та реалізації керуючих дій, у якому визначення стану вузла ІКМ здійснюється на основі вивантаження мінімально необхідної кількості цифрових слідів та їх аналізу нейромережевими моделями забезпечує економію часу і ресурсів на виявлення комп'ютерних вірусів та протидії їх поширенню в інфокомунікаційній мережі.

2. Використання цифрових слідів, що залишаються у вузлах ІКМ після ШПЗ або АРТ-атаки, у якості основної ідентифікаційної ознаки при оцінці зараженості вузлів ІКМ, забезпечує виявлення ШПЗ, пропущених традиційними рішеннями захисту кінцевих точок, та надає можливість вдосконалення наявного ешелонованого захисту ІКМ.

3. Рішення про стан конкретного вузла ІКМ приймається за критерієм "ідеального спостерігача", який мінімізує можливість сумарної помилки прийняття рішення в циклі управління.

4. Підвищення швидкості реагування на виникаючі інциденти в ІКМ з великою кількістю вузлів досягається застосуванням нейромережових моделей для аналізу вивантажених цифрових слідів.

5. Економія часу і ресурсів на виявлення наявності комп'ютерних вірусів у вузлах ІКМ, забезпечується оптимізацією кількості і розміру цифрових слідів на основі реляційної моделі у вигляді таблиці артефактів.

6. Реляційна модель у вигляді таблиці артефактів дозволяє шляхом фільтрації оптимізувати кількість і розмір цифрових слідів за критерієм «необхідна достатність – мінімум часу».

7. Словник ознак для ідентифікації стану ІКМ і запропоновані таблиці артефактів є достатніми для прийняття рішень в циклах управління.

8. Запропонована тестова послідовність дозволяє виявляти заражені вірусами вузли ІКМ та дозволяє оптимізувати час на оцінку одного вузла, що позитивно впливає на мінімізацію загального часу на виявлення та ізоляції заражених вузлів інфокомунікаційної мережі в циклі управління. Мінімізація середнього часу на проходження тестової послідовності забезпечується:

- використанням лише мінімально необхідних цифрових слідів;
- використанням моделі ШІ у якості одного з компонентів модулю прийняття рішень та заздалегідь налаштованих правил оцінки цифрових слідів;
- застосуванням заздалегідь налаштованих правил для автоматичного здійснення керуючих дій щодо ізоляції зараженого вузла;
- розпаралеленням обчислень.

9. Віртуальне середовище зі зміненою конфігурацією дозволяє повною мірою проаналізувати тестове ШПЗ з функціоналом Anti-Debugging\Anti-VM уникаючи зараження вузла ІКМ.

10. Удосконалений метод вивантаження цифрових артефактів в умовах обмеженості ресурсів, забезпечує можливість формування уявлення про стан зараженості конкретного вузла на сервері ІКМ навіть у випадку переривання з'єднання під час передачі даних за рахунок оптимізації кількості і розміру цифрових слідів та їх ранжування.

11. Рекомендовані програми для швидкого виявлення вірусів і скрипт оптимізації з використанням реляційної таблиці артефактів дозволяють

скоротити кількість елементів, необхідних для подальших досліджень більш ніж у десять разів.

12. Зменшення часу аналітика на виявлення скомпрометованих ПК забезпечується:

- першочерговою перевіркою місць автозапуску та артефактів, що дають 100% інформації про вірусне зараження;
- використанням сукупності рекомендованих програм для швидкого виявлення вірусів;
- застосуванням реляційної таблиці артефактів та скрипту оптимізації, які дозволяють суттєво скоротити кількість елементів, необхідних для подальших досліджень.

13. Достовірність виявлення вірусної активності за запропонованим методом становить від 60% до 100%. Зокрема, 100% достовірність забезпечується за наявності будь-яких записів у антивірусних виключеннях Windows Defender.

Зазначене обумовлює необхідність визначення критеріїв та відбору існуючих нейромережових моделей для аналізу цифрових слідів шляхом тестування за визначеними критеріями.

Основні теоретичні положення, описані у розділі, знайшли відображення в роботах автора [9, 84, 33, 85, 86].

РОЗДІЛ 3. ТЕСТУВАННЯ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ ВИЯВЛЕННЯ ЗАРАЖЕНИХ ВУЗЛІВ ІКМ НА БАЗІ ЦИФРОВИХ СЛІДІВ

3.1 Технології і моделі штучного інтелекту

Штучний інтелект для задач інформаційної безпеки використовується з 80х років, але досягнення за останні 10 років зробили його набагато ефективнішим. ШІ трансформує кібербезпеку, включаючи безпеку даних, систему керування ідентичністю та доступом, керування ІТ, безпеку хмарних технологій, а також виявлення кіберзагроз і спрощення реагування на них спеціалістами із безпеки. Галузь ШІ стрімко розвивається і майбутні дослідження стимулюватимуть розробку продуктів, які будуть покладатися на співпрацю між людьми та системами на основі ШІ [87].

В роботі [88] виконано аналіз математичних моделей комп'ютерних вірусів і визначено, що моделі: *SI* (Suspected-Infected), *SIR* (Suspected-Infected-Recovered), *PSIDR* (Progressive Suspected-Infected-Detected-Recovered) найбільш точно описують процес поширення комп'ютерних вірусів в мережі.

В роботі [89] проведено порівняльні дослідження математичних моделей технології поширення комп'ютерних вірусів в інформаційно-телекомунікаційних мережах.

В роботі [90] удосконалено математичні моделі розповсюдження комп'ютерних вірусів в гетерогенній комп'ютерній мережі, що враховує її топологічні та архітектурні особливості і проведено порівняльні дослідження розроблених математичних моделей та побудовані порівняльні графіки залежності кількості заражених вузлів від часу функціонування комп'ютерної мережі при поширенні ШПЗ в ній.

В роботі [91] представлено аналіз нейромережових моделей та методів розпізнавання комп'ютерних вірусів та процес створення такої нейромережової моделі, яка може розпізнавати поліморфні віруси.

В роботах [30, 31, 32, 92] досліджено великі мовні моделі та їх можливе застосування в кібербезпеці, оскільки з їх стрімким розвитком, вони відіграватимуть дедалі важливішу роль у захисті інформаційних систем від нових кіберзагроз. Проте у цих працях не приводяться конкретні дані по моделям ШІ (назва і розмір моделі, час відповіді, релевантність відповіді), що доступні для завантаження користувачами і можуть залучатися до будівництва ешелонованого захисту інфокомунікаційної мережі на підприємстві.

Аналіз розглянутих робіт показав, що їх метою є виявлення конкретних сімейств вірусів та моделювання процесу поширення комп'ютерних вірусів в ІКМ. Проте, досліджень потребує виявлення саме того, що «пропустили» наявні захисні рішення, виявлення конкретних заражених вузлів ІКМ та їх ізоляцію без наявності будь-якої інформації від захисних рішень.

При цьому, ШІ може як допомогти зберегти дані, шляхом раннього виявлення кібератак, так і зашкодити кібербезпеці, полегшуючи написання переконливих фішингових листів, відтворюючи фрагменти шкідливого коду, допомагаючи виявляти слабкі місця у мережі, та знаходити ще невідомі виробникам програмного забезпечення вразливості в операційній системі, програмах тощо (zero day vulnerability).

Наприклад, вже зараз кіберзлочинці впроваджують генеративні моделі штучного інтелекту для підвищення ефективності та масштабів добре відомих форм атаки, таких як програми-вимагачі і компрометація ділової електронної пошти (BEC) за рахунок переконливого відтворення стилів спілкування керівників компанії за допомогою тексту або відео чи аудіо (deepfake) та написання шкідливого коду із залученням ШІ. Інструменти GenAI знижують вартість фішингових кампаній з використанням соціальної інженерії, які надають зловмисникам початковий доступ до організацій. Ці інструменти зазвичай налаштовуються під конкретну компанію шляхом використання даних отриманих в результаті OSINT для донавчання моделі ШІ. Джерелом таких даних можуть слугувати соціальні мережі, публічні заяви чи документи, витоки з різних джерел (наприклад, служби таксі та доставки речей) та багато

іншого, що робить спроби соціальної інженерії набагато складнішими для ідентифікації звичайними користувачами. GenAI також допомагає зловмисникам застосовувати соціальну інженерію ширшим колом мов, що розширює поверхню атаки на більшу кількість людей у більшій кількості країн із меншими витратами, що в свою чергу дозволяє здійснювати більш складні та масштабовані кібератаки [29].

Платформи «кіберзлочинність як послуга» (CaaS) продовжують бути домінуючою та швидкозростаючою бізнес-моделлю у кримінальному ландшафті, оскільки усувають бар'єри для входу в кіберзлочинну діяльність, дозволяючи окремим особам або групам без технічних знань займатися незаконною онлайн-діяльністю, купуючи необхідні інструменти та підтримку. Ця модель, яка вже добре зарекомендувала себе серед злочинних груп, поступово застосовується в інших сферах кіберзлочинності, таких як фішингові атаки, посилені штучним інтелектом. Оскільки персонал компаній є основною мішенню атак дипфейків, а також фішингових кампаній загалом, організаціям потрібно переглянути те, як вони навчають та захищають усіх, від співробітників до керівництва, від нових моделей кіберзлочинності [29].

Тому, організації, що приділяють додаткову увагу захисту від розроблених з технологією deepfake фішингових кампаній, надають роз'яснення персоналу, пріоритезують впровадження ШІ у своїй роботі та інвестують у захисні рішення із залученням ШІ, можуть суттєво зменшити або нівелювати вплив кіберзагроз на свою інфокомунікаційну систему. Розвиток штучного інтелекту сьогодні досягнув великого прогресу і має значний вплив на велику кількість галузей, а особливо на кібербезпеку.

Оскільки такі компанії, як OpenAI з ChatGPT та Meta з відкритим вихідним кодом Llama, зробили технологію ШІ доступнішою для кінцевого користувача, зловмисники почали використовувати ці інноваційні засоби, що призвело до здешевлення та підвищення витонченості кібератак. Зловживання цими технологіями є значним ризиком, оскільки тепер зловмисники можуть

легко використовувати вразливості в системах, створюючи проблеми для команд безпеки [93].

Наприклад, у [94] прогнозується, що з початку 2025 року набори інструментів для кібератак будуть адаптовані та автоматизовані за допомогою ШІ, у тому числі і сканери вразливостей, оскільки ШІ може знаходити вразливості, про які ще не відомо компаніям-розробникам програмного забезпечення (zero day). А також, що новим вектором атак із залученням ШІ стануть самі ШІ моделі які використовуються в компанії, у тому числі і у захисних рішеннях [94].

Згідно з даними AV-ATLAS, у 2024 році загальна кількість вірусів вже перевищила за 1,5 млрд. екземплярів а кожну годину з'являється більше ніж 12 тис. вірусів [95]. Спроби використовувати ШІ для захисту від вірусів були ще раніше, наприклад, в [96] досліджується можливість класифікації вірусних файлів на сімейства, використовуючи представлення бінарних даних у вигляді картинки на основі згорткових нейронних мереж (CNN). Проте лише з появою великих мовних моделей (LLM) стало можливо інтегрувати ШІ моделі у безпеку.

Варто зазначити, що розподіленість інфокомунікаційних мереж збільшує час на виявлення та нейтралізацію загроз за рахунок збільшення кількості цифрових артефактів для дослідження. Тому, щоб не відставати у цій "гонці озброєнь", потрібно вже зараз впроваджувати ШІ у якості одного із компонентів кіберзахисту на підприємстві.

Зазвичай, дискусії щодо якості та ефективності моделей машинного навчання (ML) часто точаться навколо розміру моделі, кількості параметрів і продуктивності на встановлених тестових даних. Водночас до уваги не береться час – найважливіший результат, який може забезпечити якісний ШІ. У певних сферах, таких як обробка текстової інформації, завдання категоризації та ідентифікації об'єктів, оцінювати час не критично важливо. Однак у кібербезпеці час має вирішальне значення для виявлення загроз у контексті захисту від шкідливого програмного забезпечення до його

виконання. Саме тут моделі виявляють і блокують шкідливе програмне забезпечення до того, як воно розгортається і виконується [97].

Існує необхідність знаходження таких моделей штучного інтелекту, які вже зараз можливо залучити до вирішення задач захисту інфокомунікаційних мереж. Необхідне тестування нейромережових моделей формату GGUF для оцінки можливості їх застосування при вирішенні задачі виявлення заражених ПК на базі цифрових слідів.

Існують наступні типи штучного інтелекту [93]:

- *Генеративний ШІ*. Використовує глибокі нейронні мережі для створення контенту на основі заданих даних та попереднього контексту.
- *Розподілений ШІ*. Узгоджено працює над розв'язанням складних завдань.
- *Спеціалізований ШІ*. Вирішує вузькі, специфічні завдання.

Технології штучного інтелекту [93]:

1. **Машинне навчання (ML)**. Фундаментальна технологія ШІ, яка дозволяє системам навчатися на основі даних та робити прогнози без необхідності явного програмування.

2. **Глибоке навчання (DL)**. Технологія машинного навчання, що ґрунтується на використанні багат шарових нейронних мереж. Кожна нейронна мережа складається з кількох шарів, де дані проходять через кожен шар, навчаючись знаходити зв'язки та закономірності.

3. **Рекурентні нейронні мережі (RNN)**. Тип нейронних мереж, який особливо ефективний для роботи з послідовними даними, такими як текст. RNN мають пам'ять, яка дозволяє їм враховувати попередні елементи послідовності при обробці поточного елемента.

4. **Конволюційні нейронні мережі (CNN)**. Нейронні мережі, призначені для аналізу візуальних даних, які використовують конволюційні фільтри для отримання ключових характеристик, таких як краї, текстури та

об'єкти. Ці фільтри сканують зображення частинами, витягуючи важливу інформацію на різних рівнях.

5. **Трансформери.** Використовуються для роботи з послідовностями даних, дозволяючи вирішувати такі завдання, як генерація та аналіз контенту та виявлення загроз. На відміну від рекурентних нейронних мереж (RNN), трансформери здатні обробляти весь набір вхідних даних одночасно, що робить їх ефективнішими для завдань прогнозування та генерації. Трансформери використовують механізми уваги, які допомагають моделям зосередитися на найважливіших частинах даних. Саме ця технологія лягла в основу багатьох сучасних мовних моделей, таких як GPT та BERT, які здатні розуміти контекст та генерувати осмислений текст.

6. **Обробка природної мови (NLP).** Область штучного інтелекту, що дозволяє машинам розуміти, інтерпретувати та генерувати людську мову. NLP включає кілька ключових технологій, таких як синтаксис, семантика і аналіз лінгвістичних моделей. NLP використовується для виконання таких завдань, як переклад тексту, автоматичне резюмування, відповіді на запитання та аналіз тональності тексту. Застосування NLP стало можливим завдяки таким технологіям, як трансформери та машинне навчання, які дозволяють моделям розуміти контекст та зміст тексту, а не просто виконувати поверхневий аналіз.

7. **Автокодувальники.** Тип нейронних мереж, які навчаються стискати (кодувати) дані в більш компактний вигляд, та потім відновлювати (декодувати) в вихідну форму. Це робиться для виявлення прихованих закономірностей у даних чи виконання завдань зі стискування інформації.

Отже, визначення типів ІІІ допоможе знайти той, який максимально підходить під конкретні потреби. Розглянуто типи і технології штучного інтелекту, та їх вплив на кібербезпеку і в якості захисту від кібератак, і в якості одного із компонентів для атак на інформаційну інфраструктуру.

3.2. Визначення критеріїв оцінки та результати тестування нейромережових моделей

У погоні за трендами впровадження ШІ, більшість компаній, що пропонують захисні рішення для кінцевих точок, такі як CrowdStrike, SentinelOne, Cybereason, Symantec тощо почали додавати до своїх продуктів власні моделі ШІ [98]. Ще не можна остаточно сказати, наскільки впровадження ШІ допомогло при захисті кінцевих точок, оскільки порівняльні тести захисних рішень «до» та «після» впровадження ШІ не проводилися. Оскільки немає можливості використовувати зазначені моделі локально (агентам ШІ потрібен доступ до інтернету та хмарного сховища виробника) та відокремити їх від основного функціоналу захисного рішення кінцевих точок, а також надавати на обробку власні запити – тут вони не розглядаються. Розглядаються локальні моделі ШІ, сконвертовані у формат GGUF (GPT Generated Unified Format) [99, 100].

Недоліки використання ШІ у кібербезпеці [93]:

- Потреба у великих обсягах даних;
- Завжди будуть хибні спрацювання (false-positive);
- Обмежені обчислювальні ресурси;
- Питання ліцензування та контролю ШІ;
- Складність адаптації до нормативних вимог регуляторів.

Тестування обраних для вирішення поставленої задачі виявлення заражених ПК на базі цифрових слідів здійснювалось за схемою, поданою на рис. 3.1.

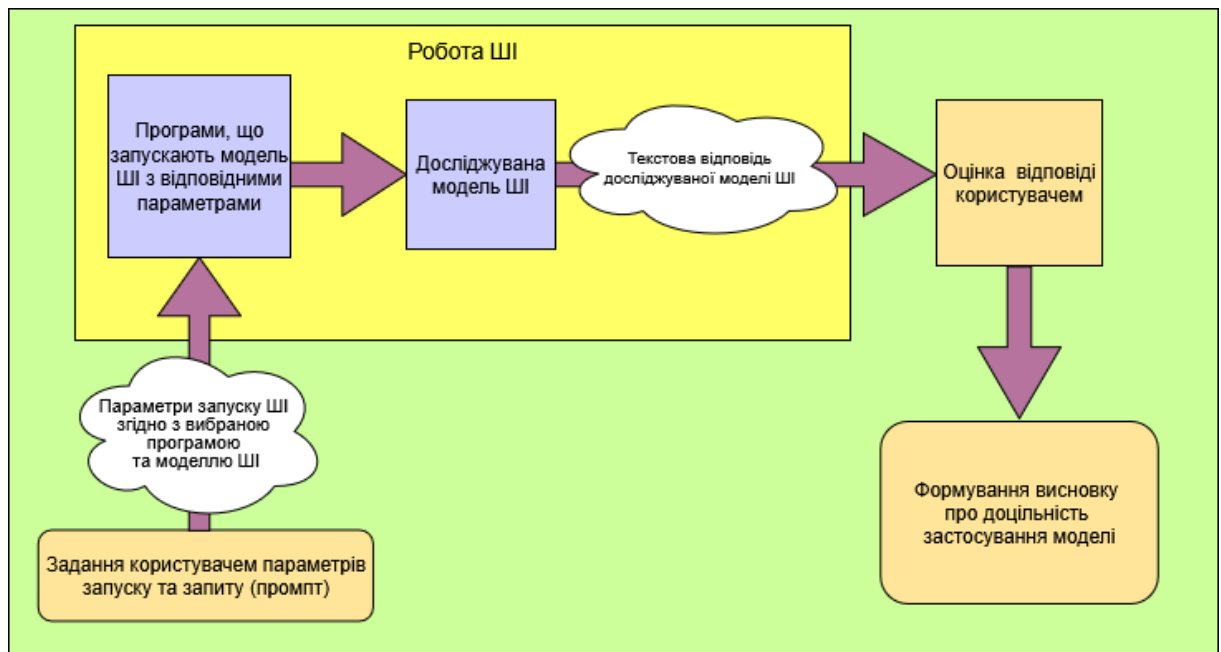


Рис.3.1. Схема дослідження нейронмережових моделей (створено автором)

Вимоги до моделей:

1. Сконвертована у формат GGUF;
2. Здатність роботи з текстом у якості «питання->відповідь»;
3. Вміння надавати правильну (іноді розгорнуту і обґрунтовану) відповідь;
4. Здатність запуску і отримання відповіді від моделі засобами командного рядка Windows, без користувацького інтерфейсу;
5. Здатність видавати відповідь за прийнятний час (чим швидше тим краще). В ідеалі – менше ніж 1 хв на відповідь, проте, зважаючи на різницю між корпоративним та домашнім обладнанням, будуть прийнятними всі моделі, що надали відповідь менше ніж за 10 хв;
6. Розмір моделі менший за 100 ГБ.

З розміщених на HuggingFace [101] обрано 135 моделей формату GGUF, 57 з яких навчено під задачі кібербезпеки (різного розміру, рівня квантування та від різних користувачів). Для тестування підготовлено промпти для моделей, які б містили таку інформацію про цифрові сліди, на основі яких остаточно можна було б сказати, що конкретний ПК заражений, навіть не

маючи додаткової інформації про файли. При цьому під терміном «промпт» розуміються спеціальні інструкції на природній мові, які надаються ШІ для виконання певної роботи і надання відповіді у вигляді тексту, картинок, аудіо, відео тощо. Це можуть бути запити, питання або завдання яке ШІ має виконати «відповідаючи» на інструкцію і чим точніше буде інструкція – тим більш релевантною буде відповідь ШІ.

Протестовано різні методи запуску конкретної моделі засобами командного рядка у Windows (оскільки не всі працюють однаково швидко і лише llama-cli може приймати файли у якості промптів): llama-cli, llama-run, koboldcpp. А також розглядалася реакція моделі на запит надання короткої і розширеної відповіді для виявлення коректних обґрунтувань та зміни швидкості відповіді.

Промпт для короткої відповіді: "Hello, on my PC I have next digital artifacts in Registry:

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run RealtekHD C:\ProgramData\WindowsTasks\RealtekHD.exe and in Windows Defender Exclusions: C:\ProgramData\WindowsTasks\RealtekHD.exe. Is it really a virus? Answer in only one word Yes or No. No other words needed."

Промпт для довгої відповіді: "Hello, on my PC I have next digital artifacts in Registry:

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run RealtekHD C:\ProgramData\WindowsTasks\RealtekHD.exe and in Windows Defender Exclusions: C:\ProgramData\WindowsTasks\RealtekHD.exe. Is it really a virus? Give detailed report."

Файл промпту містить кілька розділів та додаткову надлишкову для визначення коректної відповіді інформацію, тому що, саме в такому вигляді отримуються цифрові сліди з досліджуваного ПК.

Приклад структури такого файлу складається з управляючої команди «*Hello, I have next digital artifacts on my PC*», самих слідів, які наведені на рис. 3.2. та самого питання «*Is my PC infected by computer virus?*».

```

Processes:
Node,ExecutablePath
<list of average processes on PC>
<PC name>,C:\ProgramData\WindowsTasks\RealtekHD.exe
<list of average processes on PC>

Registry:
Key,Value Name,Value Type,Value data
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\C
urrentVersion\Run,RealtekHD,REG_SZ,C:\ProgramData\Win
dowsTasks\RealtekHD.exe
<additional legitimate software>

Windows Defender exclusions (files or folders):
C:\
C:\ProgramData\WindowsTasks\RealtekHD.exe

FileSystem:
Folders that have attributes "hidden" and "system":
C:\ProgramData\WindowsTasks
Files that have attributes "hidden" and "system":
C:\ProgramData\WindowsTasks\RealtekHD.exe

```

Рис. 3.2. Приклад цифрових слідів зараженого комп'ютера

Також файл містить завершуючу управляючу команду, яка змінюється в залежності від відповіді, яку ми хочемо отримати від ШІ:

Коротка відповідь: *Answer with only one word Yes or No.*

Розширена відповідь: *Say "Yes" or "No" and give detailed report why do you think so.*

Можливість моделі змінювати свою відповідь при повторних запитах перевірялась на моделі Gemma. Результати тестування зведено у таблиці 3.1. і 3.2.

Варто зазначити, що незважаючи на явну вказівку надавати коротку відповідь, модель іноді видає більше слів ніж звичайне «так» або «ні», тому в таблиці введено наступні позначення:

No – модель явно надала відповідь НІ;

Yes – модель явно надала відповідь ТАК;

Suspicious – модель не надала явної відповіді, проте вважає файл підозрілим і таким, що потребує подальших перевірок;

Irrelevant – модель не надала нормальної відповіді, проте відповіла беззв'язний текст або відповідь була надана зовсім на інше питання (були ситуації, коли

модель сама собі поставила питання яке ніяк не зв'язано з промтом та почала відповідати на нього);

Error або пропуск – модель не видала нічого, або зависла при відповіді, або це був непечатний текст.

Оскільки різниці у відповідях не виявлено, наступні моделі тестувалися по одному разу.

Таблиця 3.1

Результати тестування 5 моделей Gemma з короткою відповіддю (створено автором)

Model name	Size of model (GB)	Short Answer					
		llama-cli with promptfile (answer)	llama-cli with promptfile (seconds)	llama-run with prompt (answer)	llama-run with prompt (seconds)	koboldcpp with prompt (answer)	koboldcpp with prompt (seconds)
gemma-3-270m-it-Q4_K_M	0,24	irrelevant	4.5	Yes	1	irrelevant	11.1
gemma-3-270m-it-Q4_K_M	0,24	irrelevant	4.7	Yes	1	irrelevant	11.3
gemma-3-1b-it-Q4_K_M	0,75	No	10.2	No	1.5	No	12
gemma-3-1b-it-Q4_K_M	0,75	No	9.1	No	1.4	No	12.4
gemma-3-4b-it-Q4_K_M	2,32	Yes	30.3	No	2.6	No	12
gemma-3-4b-it-Q4_K_M	2,32	Yes	29.9	No	2.5	No	52.3
gemma-3-12b-it-Q4_K_M	6,8	No	48.7	No	5.3	No	19.7
gemma-3-12b-it-Q4_K_M	6,8	No	48.4	No	4.9	No	16.8
gemma-3-27b-it-Q4_K_M	15,41	No	101	Yes	10.8	No	117
gemma-3-27b-it-Q4_K_M	15,41	No	102	Yes	12.4	No	118

Результати тестування 5 моделей Gemma з довгою відповіддю (створено автором)

Model name	Size of model (GB)	Long Answer					
		llama-cli with long promptfile (answer)	llama-cli with long promptfile (seconds)	llama-run with long prompt (answer)	llama-run with long prompt (seconds)	koboldcpp with long prompt (answer)	koboldcpp with long prompt (seconds)
gemma-3-270m-it-Q4_K_M	0,24	irrelevant	2.3	irrelevant	1.3	irrelevant	10.9
gemma-3-270m-it-Q4_K_M	0,24	irrelevant	2.1	suspicious	6.5	irrelevant	11.4
gemma-3-1b-it-Q4_K_M	0,75	irrelevant	5.2	suspicious	42.9	irrelevant	11.6
gemma-3-1b-it-Q4_K_M	0,75	irrelevant	5.3	suspicious	42.1	irrelevant	12.6
gemma-3-4b-it-Q4_K_M	2,32	irrelevant	3.6	suspicious	108	irrelevant	16
gemma-3-4b-it-Q4_K_M	2,32	No	12.7	suspicious	98.6	irrelevant	18.1
gemma-3-12b-it-Q4_K_M	6,8	No	39.3	suspicious	431	suspicious	38.2
gemma-3-12b-it-Q4_K_M	6,8	No	34	suspicious	317	suspicious	36.5
gemma-3-27b-it-Q4_K_M	15,41	No	140	-	-	-	-
gemma-3-27b-it-Q4_K_M	15,41	irrelevant	77.8	suspicious	912	irrelevant	168

Результати тестування моделей Gemma показали, що жодна з них не дає стабільно правильної відповіді залежно від методу запуску, а більшість з них надає неправильну відповідь, отже доцільно розділити досліджувані моделі за методом запуску і не включати у вибірку моделі, які неправильно

відповідають на питання. Результати тестування моделей Cybersecurity представлено у таблицях 3.3-3.8.

Таблиця 3.3

Результати тестування моделей Cybersecurity з короткою відповіддю
(створено автором)

Model name	Size of model (GB)	llama-cli with promptfile (answer)	llama-cli with promptfile (seconds)
Lily-Cybersecurity-7B-v0.2.Q8_0_Quantization-made-by-Richard-Erkho	7,17	yes	91.5
mistral-v0.3-7b-cybersecurity_unsloth.Q4_K_M	4,07	yes	34.6
senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m_Nekuromento	5,07	yes	34.1
forensicmistral-unisloth.Q8_0	7,17	yes	46.7
forensicmistral_v0.3-unisloth.Q4_K_M	4,07	yes	29.6
CyberSecurity-CHMP-AS-DP8B-V3-GigoFof-unisloth.Q4_K_M	4,58	suspicious	52

Таблиця 3.4

Результати тестування моделей Cybersecurity з короткою відповіддю
(створено автором)

Model name	Size of model (GB)	llama-run with prompt (answer)	llama-run with prompt (seconds)
Cybersecurity-alarm-llama3.2-1B-GGUF-unisloth.Q8_0	1,76	yes	25.8
senecallm-q4_k_m	4,58	yes	3.3
senecallm-x-qwq-32b-q4_k_m	18,49	yes	230
llama-3-8b-instruct-cybersecurity.Q4_K_M	4,58	yes	3.7
forensicmistral-unisloth.Q4_K_M	4,07	suspicious	5.5

Таблиця 3.5

Результати тестування моделей Cybersecurity з короткою відповіддю
(створено автором)

Model name	Size of model (GB)	koboldcpp with prompt (answer)	koboldcpp with prompt (seconds)
forensicmistral_v0.3-unsloth.Q4_K_M	4,07	yes	13.8
SenecaLLM_x_Qwen2.5-7B-CyberSecurity.i1-Q4_K_M	4,36	yes	13.3
SenecaLLM_x_Qwen2.5-7B-CyberSecurity.Q4_K_M	4,36	yes	14.1
Mistral-7B-Instruct-v0.3-Forensics-v1.Q8_0	7,17	yes	14.6
nomic-ai-gpt4all-falcon-Q4_K_M	4,63	yes	12.2
lily-cybersecurity-7b-v0.2-q8_0_Nekuromento	7,17	suspicious	42.6
Lily-Cybersecurity-7B-v0.2.Q8_0_quantized_version	7,17	suspicious	41

Таблиця 3.6

Результати тестування моделей Cybersecurity з довгою відповіддю (створено автором)

Model name	Size of model (GB)	llama-cli with long promptfile (answer)	llama-cli with long promptfile (seconds)
forensicmistral_v0.3-unsloth.Q4_K_M	4,07	yes	26.6
senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m_Nekuromento	5,07	yes	23.3
qwq-32b-preview-senecallmv1.2-q4_k_m	18,49	yes	194

Таблиця 3.7

Результати тестування моделей Cybersecurity з довгою відповіддю (створено автором)

Model name	Size of model (GB)	llama-run with long prompt (answer)	llama-run with long prompt (seconds)
senecallm-x-qwq-32b-q4_k_m	18,49	yes	942
senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m_Nekuromento	5,07	suspicious	34.6
senecallm-q4_k_m	4,58	suspicious	87.3
qwen-cybersecurity-2.5-7b-Armandotrsg-unsloth.F16	14,19	suspicious	79
lily-cybersecurity-7b-v0.2-q5_k_m_Nekuromento	4,78	suspicious	61.2
seneca-x-deepseek-r1-distill-qwen-32b-v1.3-q4_k_m	18,49	suspicious	906
BaronLLM_Offensive_Security_abliterated_(by_huihui-ai)_llama3.1-v1-q6_k	6,14	suspicious	388
baronllm-llama3.1-v1-q6_k	6,14	suspicious	89

Таблиця 3.8

Результати тестування моделей Cybersecurity з довгою відповіддю (створено автором)

Model name	Size of model (GB)	koboldcpp with long prompt (answer)	koboldcpp with long prompt (seconds)
BaronLLM_Offensive_Security_abliterated_(by_huihui-ai)_llama3.1-v1-q6_k	6,14	suspicious	38.4
mistral-v0.3-7b-cybersecurity_unsloth.Q4_K_M	4,07	suspicious	29
DeepSeek-Qwen7B-CyberSecurity-unsloth.Q4_K_M	4,36	suspicious	31.1

Додатково було протестовано моделі загального призначення (General), результати представлені у таблицях 3.9-3.13.

Таблиця 3.9

Результати тестування моделей General з короткою відповіддю (створено автором)

Model name	Size of model (GB)	llama-cli with promptfile (answer)	llama-cli with promptfile (seconds)
gemma-2-9b-it-abliterated(by_bartowski)-Q6_K_L	7,27	yes	53
gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0	11,65	yes	81.3
gemma-3-27b-it-q4_0_(by_google)	16,05	yes	189
meta-llama-3.1-8b-claude-q8_0	7,95	yes	74.8
mistral-claude-merged.Q5_K_M	4,78	yes	46.9
deepseek-ai_DeepSeek-R1-0528-Qwen3-8B-Q4_K_L	5,11	suspicious	86.2
deepseek-ai_DeepSeek-R1-0528-Qwen3-8B-Q8_0	8,11	suspicious	148
DeepSeek-R1-0528-Qwen3-8B-UD-Q8_K_XL	10,08	suspicious	183
mistral-7b-claude-chat.Q8_0	7,17	suspicious	102

Таблиця 3.10

Результати тестування моделей General з короткою відповіддю (створено автором)

Model name	Size of model (GB)	llama-run with prompt (answer)	llama-run with prompt (seconds)
gemma-3-27b-pt-q4_0_(by_google)	16,05	yes	1250
gemma-3-27b-it-q4_0_(by_google)	16,05	yes	11.7
gemma-3-27b-it-Q3_K_M	12,51	yes	12
gemma-3-27b-it-Q4_K_M	15,41	yes	13.8
gemma-3-27b-it-UD-Q4_K_XL	15,67	yes	12.7
gemma-3-27b-it-abliterated(by_mlabonne)-v2.q4_k_m	15,41	yes	13.6
gemma-3-27b-it-abliterated(by_mlabonne)-v2.q8_0	26,74	yes	1490
Kimiko Claude FP16.f16	13,49	yes	666
mistral-7b-claude-chat.Q4_K_M	4,07	yes	311
Phi4-Reasoning-Merged-15B-Q8_0	14,51	yes	124
tinylama-claude_16bit_GGUF-unsloth.F16	2,05	yes	4.1

Таблиця 3.11

Результати тестування моделей General з короткою відповіддю (створено автором)

Model name	Size of model (GB)	koboldcpp with prompt (answer)	koboldcpp with prompt (seconds)
gemma-3-27b-it-abliterated(by_mlabonne)-v2.q8_0	26,74	yes	386
gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0	11,65	yes	97.2
mistral-7b-claude-chat.Q8_0	7,17	yes	40.5
gemma-3-12b-it-abliterated(by_mlabonne)-v2.q4_k_m	6,8	yes	16.9
gemma-3-1b-it-Q4_K_M	0,75	yes	12.1
DeepSeek-R1-Distill-Llama-8B-Q4_K_M	4,58	yes	13.5
meta-llama-3.1-8b-instruct-abliterated(by_mlabone).Q4_K_M	4,58	yes	22.7
Kimiko-Claude-FP16.Q4_K_M	4,07	yes	19.5
deepseek-ai_DeepSeek-R1-0528-Qwen3-8B-Q8_0	8,11	suspicious	49.3
DeepSeek-R1-0528-Qwen3-8B-UD-Q8_K_XL	10,08	suspicious	173

Таблиця 3.12

Результати тестування моделей General з довгою відповіддю (створено автором)

Model name	Size of model (GB)	llama-cli with long promptfile (answer)	llama-cli with long promptfile (seconds)
gemma-3-27b-it-Q4_K_M	15,41	yes	173
meta-llama-3.1-8b-claude-q8_0	7,95	yes	35.1
meta-llama-3.1-8b-claude-q4_k_m	4,58	yes	20.5
llama-7b.Q5_K_M	4,45	yes	25.5
claude-3.7-sonnet-reasoning-gemma3-12B.Q8_0	11,65	suspicious	108

Продовження таблиці 3.12

Model name	Size of model (GB)	llama-cli with long promptfile (answer)	llama-cli with long promptfile (seconds)
oh-dcft-v3.1-claude-3-5-haiku-20241022-qwen-q4_k_m-imat	4,36	suspicious	19.5
Kimiko-Claude-FP16.f16	13,49	suspicious	87.1
DeepSeek-R1-0528-Qwen3-8B-Q4_K_M	4,68	suspicious	30.6
Kimiko-Claude-FP16.Q8_0	7,17	suspicious	32.1
Phi-4-mini-instruct-Q4_K_M	2,32	suspicious	4.5

Таблиця 3.13

Результати тестування моделей General з довгою відповіддю (створено автором)

Model name	Size of model (GB)	llama-run with long prompt (answer)	llama-run with long prompt (seconds)
gemma-3-27b-it-Q3_K_M	12,51	yes	851
gemma-3-27b-pt-q4_0_(by_google)	16,05	yes	251
oh-dcft-v3.1-claude-3-5-haiku-20241022-qwen-q4_k_m-imat	4,36	suspicious	32.6
DeepSeek-R1-0528-Qwen3-8B-Q4_K_M	4,68	suspicious	303
DeepSeek-R1-Distill-Llama-8B-Q4_K_M	4,58	suspicious	153
gemma-3-12b-it-Q5_K_M	7,87	suspicious	471
Qwen2.5-32B-Instruct-Q4_K_L	19,03	suspicious	364
meta-llama-3.1-8b-instruct-abliterated(by_mlabone).Q8_0	7,95	suspicious	167
mistral-7b-instruct-v0.2.Q4_K_M	4,07	suspicious	60
Qwen3-32B-Q4_K_M	18,4	suspicious	1171
deepseek-ai_DeepSeek-R1-0528-Qwen3-8B-Q8_0	8,11	suspicious	603
DeepSeek-R1-0528-Qwen3-8B-UD-Q8_K_XL	10,08	suspicious	518
deepseek-ai_DeepSeek-R1-0528-Qwen3-8B-bf16	15,26	suspicious	792
gemma-3-27b-it-UD-Q4_K_XL	15,67	suspicious	1019
mistral-7b-claude-chat.Q4_K_M	4,07	suspicious	324

Моделі General із довгою відповіддю під час запуску з koboldcpp не давали правильної відповіді, а сам koboldcpp іноді використовував всю доступну пам'ять на ПК, тому результат їх виконання у таблиці не відображений.

Щоб оцінити можливості застосування існуючих моделей ШІ для вирішення актуальних задач кіберзахисту, зокрема виявлення заражених ПК на базі цифрових слідів із застосуванням ШІ, визначено критерії для моделі ШІ які будуть прийнятними для використання у корпоративному середовищі та проведено тестування 135 моделей формату GGUF на предмет виявлення або невиявлення ними ознак вірусної активності та індикаторів компрометації у промпті, що надавався користувачем.

Оскільки виявлено, що при запуску однієї і тієї ж нейромережевої моделі з однаковими промптами але різними програмами, що можуть запускати локальні моделі на ПК, її відповідь кардинально змінюється, підготовлено низку зведених таблиць де є назва моделі і варіанти відповідей під кожен промт для запуску ШІ моделей, без урахування тих, що надали неправильну відповідь, витратили надто багато часу на відповідь або завершилися з помилкою.

3.3. Виявлення релевантних нейромережевих моделей для аналізу цифрових слідів

Для визначення переліку доцільних для використання моделей, отримані результати у зведених таблицях узагальнено за часом, що був витрачений на відповідь а також за способом запуску моделі. Визначено як мінімум 43шт. доцільних для використання моделей ШІ.

Для моделей Gemma

Для використання з файлом промпту llama-cli, більше підходить gemma-3-4b-it-Q4_K_M, що важить всього 2,32 ГБ з 30 сек.

Для використання з промптом llama-run, більше підходить gemma-3-270m-it-Q4_K_M, що важить всього 0,24 ГБ з 1 сек та gemma-3-27b-it-Q4_K_M що важить 15,41ГБ з 11 сек.

Для моделей Cybersecurity з короткою відповіддю

Для використання з файлом промпту llama-cli, більше підходить:

- forensicmistral_v0.3-unsloth.Q4_K_M(29.6sec);
- senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m_Nekuromento (34.1sec);
- mistral-v0.3-7b-cybersecurity_unsloth.Q4_K_M (34.6sec);
- forensicmistra-unsloth.Q8_0 (46.7sec);
- Lily-Cybersecurity-7B-v0.2.Q8_0_Quantization-made-by-Richard-Erkhov (91.5sec).

Для використання з промптом llama-run, більше підходить:

- senecallm-q4_k_m (3.3 sec);
- llama-3-8b-instruct-cybersecurity.Q4_K_M (3.7 sec);
- Cybersecurity-alarm-llama3.2-1B-GGUF-unsloth.Q8_0 (25.8 sec);
- senecallm-x-qwq-32b-q4_k_m (230 sec).

Для використання з промптом koboldcpp, більше підходить:

- nomic-ai-gpt4all-falcon-Q4_K_M (12.2 sec);
- SenecaLLM_x_Qwen2.5-7B-CyberSecurity.i1-Q4_K_M (13.3 sec);
- forensicmistral_v0.3-unsloth.Q4_K_M (13.8 sec);
- SenecaLLM_x_Qwen2.5-7B-CyberSecurity.Q4_K_M (14.1 sec);
- Mistral-7B-Instruct-v0.3-Forensics-v1.Q8_0 (14.6 sec).

Для моделей Cybersecurity з розширеною відповіддю

Для використання з файлом промпту llama-cli, більше підходить:

- senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m_Nekuromento (23.3 sec);
- forensicmistral_v0.3-unsloth.Q4_K_M (26.6 sec);

- qwq-32b-preview-senecallmv1.2-q4_k_m (194 sec).

Для моделей General з короткою відповіддю

Для використання з файлом промпту llama-cli більше підходить:

- mistral-claude-merged.Q5_K_M (46.9 sec);
- gemma-2-9b-it-abliterated(by_bartowski)-Q6_K_L (53 sec);
- meta-llama-3.1-8b-claude-q8_0 (74.8 sec);
- gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0 (81.3 sec);
- gemma-3-27b-it-q4_0_(by_google) (189 sec).

Для використання з промптом llama-run більше підходить:

- tinyllama-claude_16bit_GGUF-unsloth.F16 (4.1 sec);
- gemma-3-27b-it-q4_0_(by_google) (11.7 sec);
- gemma-3-27b-it-Q3_K_M (12 sec);
- gemma-3-27b-it-UD-Q4_K_XL (12.7 sec);
- gemma-3-27b-it-abliterated(by_mlabonne)-v2.q4_k_m (13.6 sec);
- gemma-3-27b-it-Q4_K_M (13.8 sec);
- Phi4-Reasoning-Merged-15B-Q8_0 (124 sec);
- mistral-7b-claude-chat.Q4_K_M (311 sec).

Для використання з промптом koboldcpp більше підходить:

- gemma-3-1b-it-Q4_K_M (12.1 sec);
- DeepSeek-R1-Distill-Llama-8B-Q4_K_M (13.5 sec);
- gemma-3-12b-it-abliterated(by_mlabonne)-v2.q4_k_m (16.9 sec);
- Kimiko-Claude-FP16.Q4_K_M (19.5 sec);
- meta-llama-3.1-8b-instruct-abliterated(by_mlabonne).Q4_K_M (22.7 sec);
- mistral-7b-claude-chat.Q8_0 (40.5 sec);
- gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0 (97.2 sec);
- gemma-3-27b-it-abliterated(by_mlabonne)-v2.q8_0 (386 sec).

Для моделей General з розширеною відповіддю

Для використання з файлом промпту llama-cli більше підходить:

- meta-llama-3.1-8b-claude-q4_k_m (20.5 sec);

- llama-7b.Q5_K_M 4,45 (25.5 sec);
- meta-llama-3.1-8b-llama-q8_0 (35.1 sec);
- gemma-3-27b-it-Q4_K_M (173 sec).

Для використання з промптом llama-run більше підходить gemma-3-27b-pt-q4_0_(by_google) (251 sec).

Така варіативність у відповідях моделей і у методах запуску, дозволяє зробити висновок, що використання моделей ШІ, а особливо тих, які були спеціально навчені під задачі кібербезпеки, вже зараз може давати прийнятні результати при визначенні зараженості ПК по цифрових слідах, що у свою чергу підвищить швидкість реагування на інциденти на підприємстві, проте остаточний вибір моделі буде залежати від актуальних задач та наявних ресурсів.

Визначено перелік із 43х доцільних для використання моделей ШІ у форматі GGUF для вирішення задач кібербезпеки, зокрема для виявлення заражених ПК на базі цифрових слідів. Проте, так як кожна модель краще проявляє себе у специфічних умовах із різними сценаріями запуску, вибір моделі буде залежати від актуальних задач та наявних ресурсів.

3.4. Формування навчальних даних для дотренування нейромережевих моделей

Кібератаки на основі штучного інтелекту стають значним викликом у сфері кібербезпеки. Кіберзлочинці використовують штучний інтелект для підвищення складності виявлення та впливу своїх атак, що робить їх дедалі складнішими та важчими для виявлення. Загрози, керовані штучним інтелектом, можуть автоматизувати виявлення вразливостей, створювати переконливі фішингові схеми та навіть адаптуватися в режимі реального часу для обходу заходів безпеки. Динамічний розвиток штучного інтелекту

обумовлює необхідність проактивного та інноваційного підходу до забезпечення кібербезпеки. Отже, організаціям необхідно пріоритетизувати інвестиції в рішення безпеки на основі штучного інтелекту та постійно вдосконалювати свої стратегії, щоб ефективно протистояти загрозам, що постійно змінюються [102].

Проблемні питання використання штучного інтелекту при дослідженні цифрових слідів викладено в [103], проте не запропоновано конкретні нейромережові моделі або методи для навчання ШІ. Необхідно визначити метод формування даних для донавчання моделей ШІ з метою покращення якості вирішення задач кібербезпеки. Для більш якісної категоризації стану ПК у якості «заражено / не заражено», необхідно дотренувати будь-яку наявну ШІ модель на цифрових слідах. Дотренування моделей ШІ на цифрових слідах суттєво підвищить їх ефективність порівняно з недотренованими моделями.

Технологія Deepfake використовує штучний інтелект для створення реалістичних підроблених відео, зображень або аудіо, які імітують реальних людей, що часто ускладнює визначення справжнього контенту. Deepfake стає потужним інструментом для кіберзлочинців, що підтверджується зростанням кількості онлайн-діпфейків на 550% з 2019 по 2023 рік. За даними DeepMedia [104], лише у 2023 році в соціальних мережах по всьому світу було поширено близько 500 000 відео- та голосових діпфейків. Очікувалось, що до 2025 року ця цифра зросте до 8 мільйонів, що відображає експоненціальне зростання цієї технології. Одним із яскравих прикладів цієї загрози є нещодавня фальшива фотографія суперзірки, яка підтримує політика, після чого суперзірка пояснила свою думку та підтримала іншого кандидата. Широка доступність передових інструментів штучного інтелекту та велика кількість загальнодоступних даних сприяють поширенню діпфейків, що робить їх значним викликом для зусиль у сфері кібербезпеки [102].

Вже зараз кіберзлочинці впроваджують генеративні моделі штучного інтелекту для підвищення ефективності та масштабів добре відомих форм атаки, таких як програми-вимагачі та компрометація ділової електронної

пошти (BEC) за рахунок переконливого відтворення стилів спілкування керівників компанії за допомогою тексту або відео чи аудіо (deepfake) та написання шкідливого коду із залученням ШІ. GenAI також допомагає зловмисникам застосовувати соціальну інженерію у фішингових листах ширшим колом мов, що розширює поверхню атаки на більшу кількість людей у більшій кількості країн із меншими витратами, що в свою чергу дозволяє здійснювати більш складні та легко масштабовані кібератаки [29].

Темп розвитку штучного інтелекту постійно прискорюється, що має значний вплив на незлічену кількість галузей, а особливо на інформаційну безпеку. Оскільки технологія ШІ із запровадженням чат-ботів стала доступнішою для кінцевого користувача, зловмисники також почали використовувати цей інноваційний інструмент, що призвело до значного здешевлення та підвищення складності виявлення кібератак. Таке зловживання технологіями ШІ є значним ризиком, тому що зловмисники можуть легко використовувати вразливості в системах, створюючи проблеми для команд безпеки [93].

Отже, якщо компанії хочуть суттєво зменшити вплив кіберзагроз на свої інформаційні дані, їм потрібно вже зараз вдосконалювати захист від розроблених з технологією deepfake фішингових кампаній шляхом інформування персоналу про наявні загрози, інвестувати у захисні рішення, що залучають ШІ до своєї роботи, а також впроваджувати ШІ у якості одного із компонентів кіберзахисту на підприємстві. При цьому слід зазначити, що у кібербезпеці затрачений час має вирішальне значення для виявлення та нейтралізації загроз у контексті захисту від шкідливого програмного забезпечення. Тому нами пропонується метод формування даних для донавчання моделей ШІ з метою покращення якості вирішення задач кібербезпеки.

Більшість підприємств формалізували стратегію кібербезпеки та впровадили основні технічні засоби контролю, такі як оновлений захист від шкідливого програмного забезпечення (77% підприємств), політика паролів

(73% підприємств), мережеві брандмауери (72% підприємств) та обмежені права адміністратора (68% підприємств). Опитування керівників вищої ланки підприємств показало, що великі організації мають вищу пріоритетність кібербезпеки порівняно з загальною кількістю підприємств, результати подано на рис. 3.3. Хоча поширеність кіберзлочинів загалом залишалася незмінною, поширеність програм-вимагачів серед підприємств значно зросла між 2024 і 2025 роками. Орієнтовний відсоток усіх підприємств, які зазнали атак програм-вимагачів за останні 12 місяців, зріс з менш ніж 0,5% у 2024 році до 1% у 2025 році. Фішинг залишається найпоширенішим типом кіберзлочину (93% підприємств та 95% благодійних організацій зазнали фішингу). Підприємства, які стали жертвами кіберзлочинів, зазнали в середньому 30 кіберзлочинів будь-якого виду за останні 12 місяців, тоді як для благодійних організацій цей показник становив 16. Це свідчить про високий рівень кібератак на одні й ті ж підприємства [105].

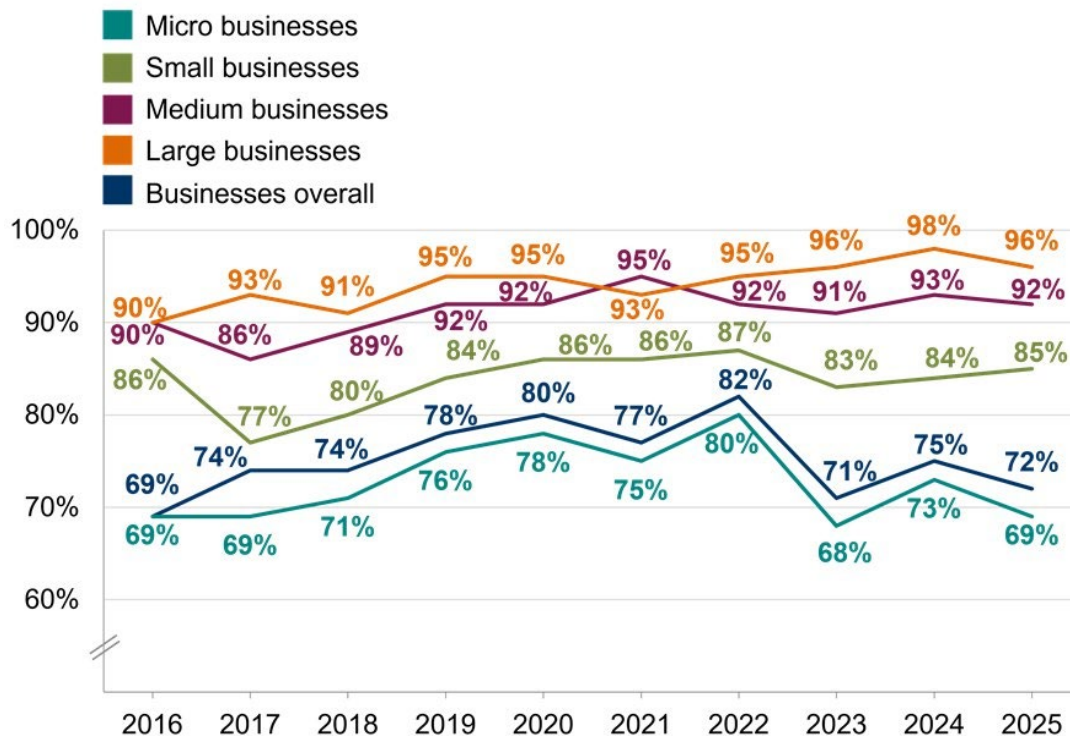


Рис. 3.3. Результати опитування керівників вищої ланки підприємств, які вважають кібербезпеку високим пріоритетом [105].

Статистика атак Ransomware [106]:

– Програми-вимагачі беруть участь майже у 70% порушень, пов'язаних зі шкідливим програмним забезпеченням, та приблизно у 24% усіх порушень загалом.

– Загальна світова вартість програм-вимагачів минулого року перевищила 30 мільярдів доларів.

– За даними IBM, середня світова вартість витоку даних у 2023 році, включаючи порушення, спричинені шкідливим програмним забезпеченням, становила 4,45 мільйона доларів, що на 15% більше порівняно з 2020 роком.

– Для компаній з річним оборотом менше 10 мільйонів доларів середня вартість відновлення після атаки програм-вимагачів становить 165 520 доларів.

– У 2021 році середній попит на програми-вимагачі оцінювався у 220 298 доларів США, що на 43% більше, ніж у 2020 році.

– За даними ФБР, середня сума, яку жертви програм-вимагачів сплатили зловмисникам, становить 10 000 доларів США.

– У 2023 році 84% організацій приватного сектору, постраждалих від програм-вимагачів, повідомили про втрату доходів через атаку.

– Сектор освіти (94%) та будівельний сектор (93%) найчастіше повідомляли про втрати бізнесу або доходів.

Статистика залучення ШІ [106]:

– Майже половина (46%) старших фахівців з безпеки вважають, що генеративний штучний інтелект підвищить вразливість їхніх організацій до атак, оскільки він може спростити для зловмисників розробку та здійснення атак.

– Інструменти розвідки загроз на основі штучного інтелекту можуть похвалитися до 300% більшою точністю, ніж традиційні методи, у виявленні спроб шкідливих скриптів використовувати поширені вразливості, спрямовані

на пристрій.

– Інструменти штучного інтелекту можуть виявити на 70% більше шкідливих скриптів, ніж традиційні методи.

– Лише 11% ІТ-менеджерів використовують ШІ для виявлення загроз, проте 56% планують його майбутнього використання.

Зростання числа кіберзлочинів, зокрема із використанням ШІ, обумовлює необхідність залучення ШІ для вирішення задач кібербезпеки. Традиційні рішення все ще актуальні для захисту, проте організаціям конче необхідно використовувати та розвивати свої ШІ для захисту їх мережевої інфраструктури, даних тощо. Перспективним напрямом є активне залучення ШІ до ідентифікації стану та захисту інфокомунікаційних мереж для протистояння кібератакам, особливо тим, які виконуються із залученням ШІ, забезпечуючи виявлення, аналіз та реагування на кіберзагрози в реальному часі та в майбутньому стати одним із важливих компонентів захисту цифрових систем від нових і невідомих загроз. Вже зараз існують моделі які спеціалізуються на кібербезпеці, проте дотренування моделей ШІ на цифрових слідах суттєво підвищить їх ефективність порівняно з недотренованими моделями.

Види нейромережових моделей штучного інтелекту: нейронні мережі прямого зв'язку (такі як багатошаровий перцептрон); рекурентні нейронні мережі (RNN); згорткові нейронні мережі (CNN); генеративно-змагальні мережі (GAN); мережі з довгою короткочасною пам'яттю (LSTM). Зазначені моделі відрізняються своєю архітектурою, способом передачі даних через них та конкретними випадками використання (рис. 3.4). Проте для задач кібербезпеки найбільше підходить вид Transformers.

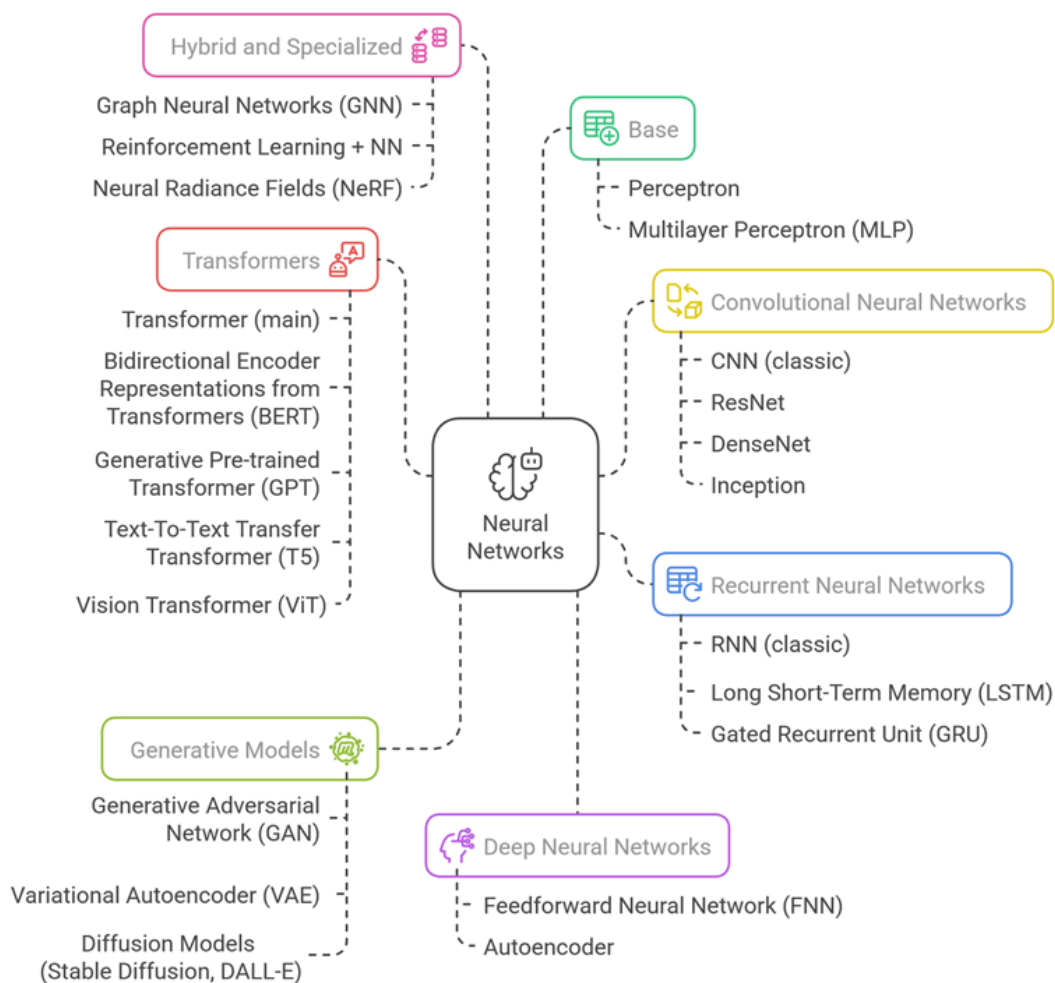


Рис.3.4. Типи нейронних мереж (створено автором із застосуванням NarKin AI)

У всіх типах моделей машинного навчання точність підсумкової моделі суттєво залежить від кількості та якості доступних навчальних даних. Модель, побудована за допомогою штучної нейронної мережі, вимагає для навчання більше даних і ресурсів, ніж традиційна модель машинного навчання. Це означає мільйони точок даних, на відміну сотень тисяч, необхідні традиційної моделі машинного навчання. [107].

Для дотренування або донавчання моделей ШІ потрібно пройти декілька важливих етапів [108]:

Визначення завдань і цілей. Визначають завдання, які має вирішувати штучний інтелект, а також конкретні цілі. Чітко визначається, яке завдання або проблему потрібно вирішити за допомогою ШІ.

Збір та аналіз даних. Відбувається збір і підготовка даних, які будуть використовуватися для навчання штучного інтелекту. Це може включати збір даних з різних джерел, їх очищення, перетворення і підготовку до використання в моделі. Підготовка великих обсягів даних для навчання моделей ШІ є одним із найвідповідальніших кроків на всьому шляху.

Вибір алгоритму і моделі. Обирається відповідний алгоритм і модель для розв'язання завдання і досягнення поставлених цілей. Вибір залежить від характеристик завдання, доступних даних і необхідної продуктивності. Це може бути класичний алгоритм машинного навчання, нейронна мережа або комбінація різних підходів.

Навчання моделі. Модель аналізує підготовлені дані, виявляє закономірності та навчається на основі цих закономірностей, вчиться робити обґрунтовані прогнози або ухвалювати рішення. Навчання може включати ітеративний процес, під час якого модель уточнюється й оптимізується.

Оптимізація та тестування. Перевіряється продуктивність моделі, її точність і надійність. Модель може піддаватися оптимізації для досягнення кращих результатів або поліпшення швидкості роботи.

Впровадження та використання. Після успішного тестування та оптимізації модель готова до впровадження та використання. Модель інтегрується в наявну систему або створюється спеціальний додаток для її використання. Також важливо забезпечити належну підтримку та оновлення моделі, щоб вона продовжувала працювати ефективно.

Моніторинг і поліпшення. Після впровадження моделі слідує її моніторинг і постійне поліпшення. Моніторинг дає змогу відстежувати продуктивність моделі в реальних умовах і виявляти можливі проблеми або необхідність внесення змін. На основі отриманих даних можна внести коригування, оновлення або донавчання моделі для поліпшення якості її роботи.

Отже, якість підготовлених даних, що використовуються для донавчання моделей ШІ є вирішальною при вирішенні задач, на яких спеціалізуються моделі ШІ.

Етап визначення завдань і цілей.

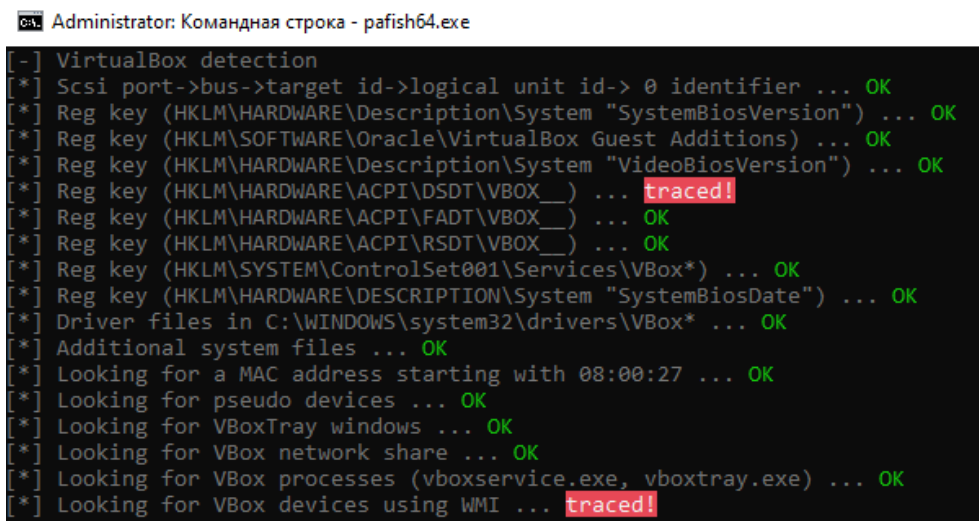
Модель ШІ повинна чітко визначати наявність або відсутність вірусної активності за наданими цифровими слідами, відповідати Yes/No та можливо робити припущення про сімейство вірусів, яким було заражено досліджуваний ПК.

Етап збору та аналізу даних.

Для отримання цифрових слідів необхідно виконати наступні кроки:

1. Підготувати перелік шкідливого програмного забезпечення (ШПЗ) для Windows у форматі .exe, з чітко визначеним сімейством. Єдина вимога, ШПЗ повинно робити явні шкідливі дії (ШПЗ яке через внутрішні помилки не змогло запуститися у вибірку не включається).

2. Підготувати такий образ віртуальної машини (ВМ) для VirtualBox, що не буде мати більшості ознак віртуальної машини [109]. При цьому статус перевіряти програмою Pafish з вимкненим Windows Defender (рис 3.5, 3.6) та використати рекомендації безпеки для ВМ [110].



```
Administrator: Командная строка - pafish64.exe
[-] VirtualBox detection
[*] Scsi port->bus->target id->logical unit id-> 0 identifier ... OK
[*] Reg key (HKLM\HARDWARE\Description\System "SystemBiosVersion") ... OK
[*] Reg key (HKLM\SOFTWARE\Oracle\VirtualBox Guest Additions) ... OK
[*] Reg key (HKLM\HARDWARE\Description\System "VideoBiosVersion") ... OK
[*] Reg key (HKLM\HARDWARE\ACPI\DSDT\VBOX_) ... traced!
[*] Reg key (HKLM\HARDWARE\ACPI\FADT\VBOX_) ... OK
[*] Reg key (HKLM\HARDWARE\ACPI\RSDT\VBOX_) ... OK
[*] Reg key (HKLM\SYSTEM\ControlSet001\Services\VBox*) ... OK
[*] Reg key (HKLM\HARDWARE\DESCRIPTION\System "SystemBiosDate") ... OK
[*] Driver files in C:\WINDOWS\system32\drivers\VBox* ... OK
[*] Additional system files ... OK
[*] Looking for a MAC address starting with 08:00:27 ... OK
[*] Looking for pseudo devices ... OK
[*] Looking for VBoxTray windows ... OK
[*] Looking for VBox network share ... OK
[*] Looking for VBox processes (vboxservice.exe, vboxtray.exe) ... OK
[*] Looking for VBOX devices using WMI ... traced!
```

Рис. 3.5. Вікно програми Pafish

```

[paFish] Start
[paFish] Windows version: 6.2 build 9200 (native)
[paFish] CPU: GenuineIntel Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz
[paFish] CPU VM traced by checking the difference between CPU timestamp counters (rdtsc) forcing VM exit
[paFish] Sandbox traced by missing dialog confirmation
[paFish] Sandbox traced by missing or implausible dialog confirmation
[paFish] Sandbox traced by checking disk size <= 60GB via DeviceIoControl()
[paFish] Sandbox traced by checking disk size <= 60GB via GetDiskFreeSpaceExA()
[paFish] VirtualBox traced using Reg key HKLM\HARDWARE\ACPI\DSDT\VBOX__
[paFish] VirtualBox device identifiers traced using NMI
[paFish] End

```

Рис. 3.6. Фрагмент журналу програми Pafish

3. Підготувати набір програм для проведення Forensic Triage та автоматичний скрипт для їх запуску.
4. Створити користувача admin з правами адміністратора та user з правами звичайного користувача, увімкнути.
5. Оновити ОС та поставити усі патчі безпеки на VM.
6. Встановити наступні програми:
 - 7-Zip 25.01;
 - WinRAR 7.13;
 - .NET Framework 2.0, 3.0, 3.5, 4.0, 4.5, 4.8;
 - Microsoft .NET SDK 9.0;
 - Microsoft Visual C++ Redistributable 2008, 2012, 2013, 2015-2022;
 - Microsoft DirectX 9,12;
 - Chrome Browser;
 - Firefox Browser.
7. Вимкнути Windows Defender на VM.
8. Зібрати мінімально необхідні артефакти з чистої ОС, в якості еталону.
9. Зробити снапшот VM (збереження поточного стану VM у якості «контрольної точки», до якої можна в будь який момент повернутися) див рис. 3.7.

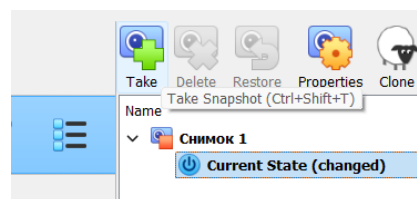


Рис. 3.7. Вигляд зробленого снапшоту

10. При дослідженні ШПЗ, що має компоненти мережевого хробака слід засобами гіпервізора ізолювати ВМ від локальної мережі вимкнувши мережевий адаптер. Для усіх інших типів ШПЗ можна залишати доступ до інтернету, оскільки ШПЗ може довантажувати свої модулі у процесі роботи. У разі необхідності повноцінного дослідження мережевого хробака слід засобами гіпервізора налаштувати мережу ВМ для роботи в режимі Bridged Adapter, а на роутері налаштувати окремий VLAN для цієї ВМ з доступом до інтернету та повністю відсутнім доступом до інших сегментів локальної мережі, див рис. 3.8. Варто зазначити, що існує вірогідність виходу ШПЗ за межі віртуальної машини [111] або детектування віртуального середовища за іншими ознаками [112, 113]. В такому разі, можна тестувати його на окремому ПК, заздалегідь помістивши його в ізольований VLAN.

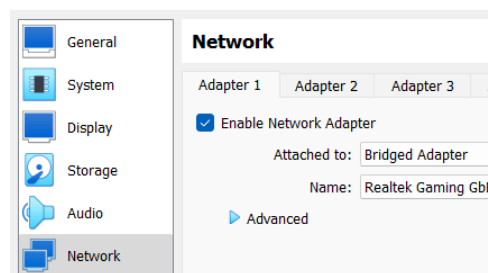


Рис. 3.8. Мережевий адаптер в режимі Bridged Adapter

11. Завантажити один шкідливий екземпляр на ВМ;
12. Залишити ВМ працювати десь на 1 годину (при цьому, вікно ВМ повинно бути активне і курсор мишки в ньому повинен плавно рухатись), можна зменшити цей час, якщо є підозра, що більше слідів ми не отримаємо;
13. Зібрати цифрові сліди за допомогою програм, що підготовлених на кроці 3 та вивантажити їх на хостову машину (ПК, де працює гіпервізор);
14. Вимкнути ВМ та відновити її стан зі снапшоту, зробленого на кроці 9;
15. Повторювати кроки 10-14 допоки не закінчиться список ШПЗ підготовлений на кроці 1.

Результатом виконання зазначених 15 кроків є отримання переліку папок і файлів з цифровими слідами, що притаманні конкретним екземплярам ШПЗ. Наступне завдання полягає у необхідності перетворення цифрових слідів у більш зрозумілий вигляд для ШІ. Для майбутнього тренування моделі пропонується оформлювати цифрові сліди у табличному вигляді (див. таблицю 3.14.) з попередньою фільтрацією цифрових слідів на основі реляційної таблиці артефактів, що дозволяє скоротити кількість елементів, необхідних для подальших досліджень.

Таблиця 3.14

Форма представлення цифрових слідів в процесі донавчання моделей ШІ
(створено автором)

Is PC infected?	Name of Virus	Forensic data
No	-	<p>Process C:\Windows\Explorer.EXE C:\Windows\system32\ceexecsv.exe, C:\Windows\system32\conhost.exe C:\Windows\system32\ctfmon.exe, C:\Windows\system32\DllHost.exe C:\Windows\system32\dwm.exe, C:\Windows\system32\fontdrvhost.exe C:\Windows\system32\lsass.exe, C:\Windows\System32\rdpclip.exe C:\Windows\System32\RuntimeBroker.exe, C:\Windows\system32\sihost.exe C:\Windows\System32\smartscreen.exe, C:\Windows\System32\spoolsv.exe C:\Windows\system32\svchost.exe, C:\Windows\system32\taskhostw.exe C:\Windows\system32\vmcomputeagent.exe, C:\Windows\system32\wbem\wmiprvse.exe, C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe C:\Windows\system32\winlogon.exe, C:\Windows\System32\WUDFHost.exe C:\Windows\SystemApps\ShellExperienceHost_cw5n1h2txyewy\ShellExperienceHost.exe</p> <p>AV Exclusion -</p> <p>Windows Firewall C:\Windows\ehome\ehshell.exe, C:\Windows\ehome\mcrmgr.exe, C:\Windows\ehome\mcx2prov.exe</p> <p>Autoruns Known DLLs c:\windows\system32\wowarmhw.dll, xtajit.dll, xtajit64.dll, c:\windows\syswow64\wow64cpu.dll, wowarmhw.dll, xtajit.dll, wow64.dll, wow64base.dll, wow64con.dll, wow64win.dll, xtajit64.dll</p>

Is PC infected?	Name of Virus	Forensic data
Yes	Trojan[Packed]/Win64.Themida Trojan[dropper]: Win/Tiggre Trojan.Barys Hash: fa86dd3ccd8ca63f2fa214f43c4d90e09d9e108798952d855b75220e1f207592	Process C:\ProgramData\RealtekHD\taskhost.exe, C:\ProgramData\RealtekHD\taskhostw.exe, C:\ProgramData\WindowsTask\RealtekHD.exe, C:\ProgramData\WindowsTask\audiodg.exe, C:\ProgramData\WindowsTask\MicrosoftHost.exe AV Exclusion C:\Program Files\RDP Wrapper, C:\ProgramData, C:\ProgramData\RealtekHD\taskhost.exe, C:\ProgramData\Windows Tasks Service\winserv.exe, C:\ProgramData\WindowsTasks\AMD.exe, C:\ProgramData\WindowsTasks\AppModule.exe, C:\ProgramData\WindowsTasks\audiodg.exe, C:\ProgramData\WindowsTasks\MicrosoftHost.exe, C:\Windows\KMS, C:\Windows\KMSAutoS, C:\Windows\System32 C:\Windows\System32\SppExtComObjHook.dll C:\Windows\System32\SppExtComObjPatcher.exe Windows Firewall C:\ProgramData\WindowsTasks\AMD.exe, C:\ProgramData\WindowsTasks\AppModule.exe Autoruns Registry Autorun Entries HKLM\Software\Microsoft\Windows\CurrentVersion\Run, Realtek HD Audio, REG_SZ, C:\ProgramData\RealtekHD\taskhostw.exe Tasks \Microsoft\Windows\WindowsBackup\OnlogonCheck, \Microsoft\Windows\WindowsBackup\TaskCheck C:\ProgramData\RealtekHD\taskhostw.exe \Microsoft\Windows\WindowsBackup\RealtekCheck, \Microsoft\Windows\WindowsBackup\WinlogonCheck C:\ProgramData\RealtekHD\taskhost.exe

Запропонований варіант подачі таблиці дозволяє виконати донавчання ШІ моделі, оскільки тут викладено основні категорії цифрових слідів без залишення яких сучасні комп'ютерні віруси зазвичай не можуть нормально працювати, самі цифрові сліди та ШПЗ, яке їх залишило, а також варіанти відповідей для ШІ. Додатково до вірусних слідів, необхідні цифрові сліди з незаражених машин, оскільки розглядається варіант змішування даних, що будуть подаватися до ШІ.

Під час донавчання ШІ на етапі перевірки коректності відповідей, для більшого наближення до реальності, необхідно міксувати дані з еталонами, де вірус відсутній у різних комбінаціях цифрових слідів зі збереженням основної структури.

Вибір алгоритму і моделі.

Вибір моделі ШІ детально розглянуто та отримано висновок, що використання моделей ШІ, а особливо тих, які були спеціально навчені під задачі кібербезпеки, вже зараз може давати прийнятні результати при визначенні зараженості ПК по цифрових слідах, що у свою чергу підвищить швидкість реагування на інциденти на підприємстві, проте остаточний вибір моделі буде залежати від актуальних задач та наявних ресурсів. До інших етапів можна буде перейти після оцінки близько 1000 екземплярів за наведеним раніше методом. Продовжуючи виконувати аналіз ШПЗ за наведеним раніше методом та зберігання даних у такому табличному форматі дозволяє отримати якісні початкові дані для донавчання ШІ моделей, що при наявності достатніх обчислювальних ресурсів дозволяє забезпечити спеціалізацію конкретної моделі ШІ на виявлення основних ознак вірусної активності в наданих цифрових слідах, що в свою чергу підвищить якість відповідей ШІ та знизить час реагування на інциденти кібербезпеки (у випадку залучення ШІ до процесу аналізу даних з інфокомунікаційних мереж).

Аналіз динаміки зростання кіберзлочинів, зокрема із використанням ШІ, обумовлює необхідність залучення ШІ для вирішення задач кібербезпеки. Запропонований метод формування навчальних даних для донавчання моделей ШІ забезпечує: спеціалізацію конкретної моделі ШІ на виявлення основних ознак вірусної активності в наданих цифрових слідах; підвищення якості відповідей ШІ; зниження часу реагування на інциденти кібербезпеки. Оформлення цифрових слідів у табличному вигляді з попередньою фільтрацією цифрових слідів на основі реляційної таблиці артефактів, дозволяє скоротити кількість елементів, необхідних для подальших досліджень.

Висновки до розділу 3

В результаті проведених досліджень доведено:

1. Доцільно розподілити досліджувані моделі за методом запуску і не включати у вибірку моделі, які неправильно відповідають на питання, оскільки більшість з них не дає стабільно правильної відповіді.

2. Запропоновані критерії оцінки нейромережових моделей є прийнятними при відборі моделей ШІ, призначених для аналізу цифрових слідів та прийняття рішення про зараженість/незараженість конкретного вузла ІКМ.

3. При тестуванні 135 нейромережових моделей формату GGUF на предмет виявлення або невиявлення ними ознак вірусної активності та індикаторів компрометації у промпті, що надавався користувачем, встановлено, що при запуску однієї й тієї ж нейромережової моделі з однаковими промптами але різними програмами, що можуть запускати локальні моделі на ПК, відповідь нейромережової моделі кардинально змінюється.

4. Відібрані 43 моделі ШІ, що відповідають запропонованим критеріям оцінки нейромережових моделей є доцільними для вирішення задач кібербезпеки, зокрема для виявлення заражених ПК на базі цифрових слідів.

5. Запропонований метод формування навчальних даних для донавчання моделей ШІ забезпечує: спеціалізацію конкретної моделі ШІ на виявлення основних ознак вірусної активності в наданих цифрових слідах; підвищення якості відповідей ШІ; зниження часу реагування на інциденти кібербезпеки.

6. Вирішальною для донавчання моделей ШІ є якість підготовлених даних. Запропонований варіант форми представлення цифрових слідів у вигляді таблиці дозволяє виконати донавчання ШІ моделі, та скоротити кількість елементів, необхідних для подальших досліджень. Оформлення цифрових слідів у табличному вигляді з попередньою фільтрацією цифрових слідів на основі реляційної таблиці артефактів, дозволяє скоротити кількість

елементів, необхідних для подальших досліджень, що дозволяє забезпечити економію часу і ресурсів на дотренування моделей ШІ.

Додаткових досліджень потребує виявлення конкретних заражених вузлів ІКМ та їх ізоляцію за відсутності будь-якої інформації від захисних рішень. Потрібно впроваджувати ШІ у якості одного із компонентів кіберзахисту.

Основні положення, описані у розділі, знайшли відображення в роботах автора [34, 35].

РОЗДІЛ 4. ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА ЕФЕКТИВНОСТІ ЗАПРОПОНОВАНОГО МЕТОДУ ЗАХИСТУ ВУЗЛІВ ІКМ

4.1 Тестування антивірусних рішень для кінцевих точок (NGAV, EDR)

Як основне рішення ешелонованого захисту в корпоративному середовищі для кінцевих точок досі залишається антивірус (Endpoint Protection). Антивірусні модулі обов'язково є в рішеннях класу Endpoint Detection and Response (EDR) & Next Generation Anti-Virus (NGAV). Разом з тим, для кожної компанії настає час, коли потрібно вибирати вендора (надавача антивірусного рішення) та інтегратора (локальна компанія, яка надає послуги з впровадження і підтримки конкретного антивірусного рішення та активно комунікує з вендором). З одного боку, достатньо вибрати топ-5 рішень з найкращими показниками від компанії, яка спеціалізується на тестуванні антивірусів (AB) та провести тендер на найдешевшу ціну від інтегратора. Проте рішення, яке цілком підійшло б для звичайного користувача і яке має найвищі оцінки за показниками сигнатурного аналізу, може не дати ефективного результату для компанії, оскільки евристичний модуль (виявлення загроз на основі поведінки програми) недостатній для корпоративних загроз (наприклад: не моніторить дії системних файлів, та виконуваних файлів з актуальним цифровим підписом, не аналізує потенційно шкідливі аргументи програм), а також для кожної компанії можуть бути свої кейси і своє бачення, які дії повинен виконувати антивірус. Так, при цільових атаках від Advanced Persistent Threat (APT) зазвичай використовується легальне програмне забезпечення і зловмисна активність маскується під «користувацьку» або «адміністраторську». Все ще можна зустріти віруси, які автономно заражають флеш-накопичувачі, а антивіруси видаляють тільки вірусний файл та autorun.inf і не відновлюють файли користувача.

У працях науковців Svitlana M. Shevchenko, Pavlo M. Skladannyi, Maksym Martseniuk [114] робиться порівняння антивірусних рішень за наявністю тієї

чи іншої функції (наприклад, наявність файрволу або батьківського контролю на додаток до основної функції – захисту від вірусів), проте не перевіряється якість реалізації цих модулів у різних вендорів. Науковці Barry Irwin, Jameel Haffejee [115] протестували сигнатурний модуль 49 антивірусів (використовуючи сервіс VirusTotal) та прийшли до висновку, що зловмисники можуть відносно легко обходити сигнатурний антивірусний захист, тому антивіруси не дуже ефективні для ще невідомих загроз. І так як всі вендори пропонують майже той самий набір функцій із тими ж самими пропозиціями (наприклад: «антивірус X захистить ваш ПК від вірусів»), потрібен певний набір тестів (однаковий для кожного вендора), щоб можна було порівняти саме якість захисту кінцевих точок у реальних умовах.

Також не можна забувати, що АВ може випадково «зламатися» а для оновлення сигнатур потрібен якийсь час (навіть якщо вони щодня оновлюються) і може бути ситуація, коли АВ підсунуть нове ШПЗ ще до оновлення сигнатур а евристичний модуль буде не дуже «агресивний», що призведе до зараження у випадку використання спеціальних технік які не тригерять евристичний модуль АВ.

І у випадку, коли потрібно відновити хронологію подій а модулі АВ були непрацюючими, і в цей же час ШПЗ видалив деякі свої сліди (наприклад, тимчасові/початкові завантажувальні файли, логування системи) – потрібно звернутися до підвиду Computer Forensic – Forensic Triage для отримання цифрових слідів.

Оскільки потреби корпоративних користувачів значно вимогливіші на відміну від потреб звичайних користувачів, для вибору рішення не достатньо переглянути статті у інтернеті та порівняльні таблиці від компаній, що тестують антивіруси [116], а для оцінки ефективності рішення необхідно виконати його тестування у корпоративному середовищі на реальних ПК (Demo, POC). Також можлива ситуація, коли вендор стверджує, що «EDR захищає від загроз сімейства X, та може виявляти навіть невідомі загрози» (приклад: SentinelOne – забезпечує комплексний захист від відомих і

невдомих загроз [117]). При тестах з відомим шкідливим програмним забезпеченням (ШПЗ) АВ їх дійсно видаляє, але якщо ШПЗ невідоме для АВ, або ШПЗ не використовує достатню кількість технік MITRE [118], які б активізували евристичний модуль АВ, воно буде пропущене і виконає всі закладені зловмисні дії. Це підводить нас до рішення, що потрібно тестувати АВ самостійно у конкретному середовищі, не зважаючи на результати тестувань, отриманих сторонніми фахівцями, та запевнення маркетологів цих захисних рішень.

Так як зазвичай при підготовці до вірусної кампанії вірусний файл доводять до статусу Fully Undetectable (FUD), то вибір рішення яке базується на сигнатурному аналізі з недорозвиненим евристичним модулем, є некоректним.

Існують спеціальні програми для тестування антивірусу, які лише імітують шкідливі дії, але не виконують їх, наприклад, програма RanSim імітує поведінку Ransomware за 24 критеріями [119], проте їх використання має ряд недоліків:

1. Антивіруси нерідко видаляють зазначені програми сигнатурними методами [120], що призводить до неможливості протестувати евристичний модуль АВ.

2. Використання зазначених програм потребує даних про компанію та корпоративну електронну пошту, на яку будуть надсилатись маркетингові повідомлення.

Процес тестування антивірусних програм обумовлює необхідність застосування програм і тестів для оцінки ефективності захисних рішень. При цьому має бути визначено тест, здатний надати перше уявлення про антивірусне рішення та відкине необхідність подальшого тестування (у вивільнений час можна більш детально протестувати інше рішення, яке має більш досконалі поведінкові та евристичні аналізатори).

Потрібно визначення тестів і програм, яких буде достатньо для оцінки ефективності захисного рішення у корпоративному середовищі.

Зважаючи на те, що антивірусних рішень у світі дуже багато і фахівці не можуть перевірити їх усі, необхідно визначити вимоги до антивірусних рішень, щоб відсіяти неефективні АВ.

Вимоги які ставляться до антивірусних рішень:

1. Здатність знаходити та знешкоджувати відомі і невідомі загрози різного типу/сімейств, як при скануванні системи, так і при відкритті ШПЗ (для цього тесту потрібні ШПЗ різних сімейств і типів). Вибирається той АВ, який знешкодив більше всього загроз.

2. Здатність знаходити нешкідливі загрози PUA (Potentially Unwanted Program). Наприклад, торенти або легальні програми для адміністрування ПК.

3. Мати евристичний модуль (для його тесту потрібні ШПЗ, які АВ не знаходить сигнатурно, наприклад, накриті протекторами). Вибираємо той АВ, який знешкодив більше всього реальних загроз.

4. Мати модуль Anti-Ransomware (для цього потрібні ШПЗ типу Ransomware як відомі так і невідомі).

5. Виявляти загрози за поведінкою (наприклад, при використанні легітимних інструментів Anydesk/PsExec для зловмисних дій), або виявляти і блокувати шкідливі дії (наприклад, дамп куща реєстру SAM або дамп даних пам'яті процесу lsass.exe).

6. Мати досконалий самозахист (немає можливості вимкнути антивірус навіть якщо ШПЗ/користувач має права адміністратора на ПК або ця можливість складна у реалізації).

В процесі вибору антивірусних рішень доцільно користуватися інформацією щодо лідерів ринку за версією Gartner [121]. Магічний квадрант для рішень захисту кінцевих точок подано на рис. 4.1.



Рис. 4.1. Магічний квадрант для рішень захисту кінцевих точок, Gartner 2024 [121].

Для тестування АВ доцільно відібрати вірусні файли різного типу та різні легітимні програми, що можуть бути використаними для зловживань як користувачами-адміністраторами, так і ШПЗ/АРТ. Пропонується вибрати наступне:

Програми, що зламують захист ліцензій ОС:

- Crack
- KMSAuto Net
 - KMS Tools Portable
 - AACT
 - w10digitalactivation

Програми, що перехоплюють та реєструють натискання клавіш:

- Keylogger
- FreeKeylogger
 - LightLogger
 - Mipko Personal Monitor
 - Radium

Програми, що експлуатують вразливості системи:

Exploit

- EternalBlue
- DoublePulsar
- Stuxnet
- Folina (безтілесний)

Інші програми, виконання яких може бути підозрілим для АВ:

Other

- AdvancedRun (для запуску від імені інших користувачів)
- Legal RAT:
 - AmmyAdmin,
 - Radmin,
 - TeamViewer,
 - AnyDesk,
 - RMS,
 - RUT.
- NirSoft: Networktools, Passrec (інструменти для відновлення «забутого» паролю користувача)
- ProcessHacker (для вимкнення або «заморозки» окремих процесів, наприклад процесів АВ)
- RemCom
- Tor Browser (компоненти іноді використовується для зв'язку ШПЗ з С2С сервером, потрібен для тестування реакції АВ на створення додаткових тунелів у мережі)
- RootkitReleaver (широко використовувався ретровірусами для вимкнення АВ)

Програми, що використовуються у тестуванні на проникнення:

Pentest

- Advanced IP Scanner
- Nmap
- Hydra-Windows
- CobaltStrike
- Impacket
- MetasploitFramework
- Magnet RAM Capture
- Belkasoft RAM Capturer
- Penetration Testing POC

Програми, що захищають інші програми від реверс-інжинірингу:

Protectors

- Themida
- UPX
- VMProtect
- Enigma

Ці програми потрібні для тестування реакції АВ як на самі протектори, так і на ШПЗ, що накриті протекторами. Наприклад, для тесту буде достатньо ШПЗ Mimikatz & WannaCry. Якщо в результаті, будь-який з них при виконанні

зможє реалізувати усі свої можливості, отримуємо висновок, що АВ не захищає ПК якісно.

Програми, що не завжди є вірусами, але є потенційно небажані для користувача:

- Sysinternals:
 - ProcessExplorer (для вимкнення або «заморозки» окремих процесів, наприклад процесів АВ),
 - PsExec (для підвищення привілеїв до SYSTEM та виконання команд на віддалених ПК),
 - Sdelete (для невідворотного видалення файлів),
 - ShellRunAS (для запуску від імені інших користувачів),
 - PsKill (для завершення процесів),
 - Procdump (для дампу пам'яті процесів),
 - ADRestore (перегляд видалених об'єктів AD),
 - ADEplorer (перегляд AD).

PUA

- PUA.Softonic (themida)
- PUP.Optional.DriverPack

Програми, що дозволяють отримати паролі та геши LM/NTLM/Kerberos паролів Windows:

- Password dump
 - Mimikatz
 - LaZagne
 - NTDS dump
 - Process lsass.exe dump (by procdump.exe)
 - Registry hive SAM & SECURITY & SYSTEM dump
 - Shadow copy of SAM & SECURITY & SYSTEM files

Програми, що дозволяють отримати паролі користувачів у відкритому вигляді з різних систем (браузери, буфер обміну, дані для входу в інші програми):

Stealer

- LummaStealer
- RaccoonStealer

Програми, що шифрують файли користувача (іноді з подальшою ексфільтрацією даних) та вимагають гроші за відновлення файлів:

- Ransomware
 - WannaCry
 - Petya
 - NotPetya
 - REvil
 - Ryuk
 - VaultCrypt
 - TeslaCrypt

Програми, що надають зловмисникам віддалений доступ до зараженого комп'ютера:

- RAT
 - RDPWrap
 - Dark Comet RAT
 - Emotet
 - njRAT
 - PS RDP Backdoor
 - QTBot
 - QuazarRAT
 - Silent RMS
 - Silent TeamViewer

Програми, що можуть використовуватись корпоративними користувачами для ексфільтрації даних. Будуть корисними, якщо у АВ є DLP модуль та функціонал видання алерту на специфічні програми або специфічні аргументи до них:

- Social
 - Telegram Desktop
 - Viber Desktop
 - Google Drive Desktop
 - OneDrive Desktop
 - Dropbox Desktop
 - MEGA (GUTCLI)

Програми, що тільки імітують шкідливу поведінку, а самі безпечні і контрольовані:

Такі програми потрібні для тестування специфічних сценаріїв та аналізу спрацювань евристичного модулю АВ. Наприклад:

- RanSim, AESCrypt, PSRansom потрібні для тестування модулю Anti-Ransomware, якщо він видалив відомі загрози Ransomware;
- LNK Worm для тестування старого, але все ще актуального методу розповсюдження ШПЗ через флеш-накопичувачі у закритій мережі. Деякі

антивіруси реагують на пустий autorun.inf, але цю загрозу вони не виявляють; в разі виявлення загрози – видаляють вірусний файл і не відновлюють «пошкоджені» користувацькі файли;

- Anti-Keylogger Tester (AKLT) для тестування можливості АВ виявити виклики потенційно шкідливих функцій, які використовують ШПЗ сімейства Keylogger:

- Test-virus
- LNK Worm
 - AESCrypt
 - PSRansom
 - RanSim
 - AKLT
 - Eicar.com

Програми, що можуть використовуватись користувачами для незаконного завантаження фільмів та зламаних програм:

- Torrent
- Bittorent (Desktop\Web)
 - uTorrent
 - Mediaget

Інші ШПЗ:

Virus-in-the-wild – усі віруси, якими корпоративні ПК раніше заражались;

VirusPack – архів з великою кількістю вірусних файлів (завантажується все, що знайдено в мережі за поточний та попередній рік, чим більше тим краще, файли можуть важити десятки гігабайт).

Тестування, документування та написання звіту за всіма критеріями з тестуванням усіх програм може затягнутися на тиждень, або навіть і більше. Це можливо виправити, якщо для оцінки продуктів ЕДР з антивірусним модулем (NGAV) тестувати лише один найбільш необхідний критерій – евристичний модуль для найбільш руйнівного класу загроз (Ransomware), які ще невідомі антивірусу. Для цього буде достатньо 1-2 робочих днів і вже буде відомо, чи підходить це рішення, чи ні (якщо загрози пропущені і вендор не може це виправити додатковими політиками, або вендор не вважає це за

загрозу бо «націлений тільки на запобігання реальним атакам» – таке рішення не підходить), що дасть економію часу у майже 3-5 робочих днів та дає 80% загального уявлення про рішення.

Для тестування евристичного модулю наявного антивірусу обрано основне сімейство загроз (Ransomware) та критерії, за якими очікується виявлення ШПЗ антивірусними рішеннями. Критерії типової поведінки Ransomware обрано на прикладах: Petya/NotPetya, Ryuk [122, 123], REvil [124, 125], Wannacry [126], Vault:

1. Наявність пошуку файлів в кореновому каталозі C:\ та пошук файлів на інших дисках (D:\ F:\ тощо).

2. Вибір файлів із певним розширенням (наприклад: файли документів .doc, .docx, .xls, .xlsx та інші) [126].

3. Шифрування файлів за допомогою симетричного алгоритму (усі програми-вимагачі роблять це для більшої швидкості).

4. Шифрування кожного файлу один за одним, у всіх підкаталогах.

5. Модифікація (шифрування) прихованих файлів-приманок. Файли-приманки – спеціальні приховані файли, що створюються захисним ПЗ у каталогах користувача для виявлення активності Ransomware. Це виключає дії користувачів, які "просто хочуть зашифрувати папку".

6. Видалення вихідного файлу та залишення лише зашифрованої версії.

7. Додавання до зашифрованих файлів такого специфічного розширення, яке не збігається з розширенням легального програмного забезпечення (наприклад, .aes, .gpg). Це може бути розширення .encrypted, .vault, .enc, .wncry, .ryuk та інші.

8. Використання встановлених або портативних версій легального програмного забезпечення для шифрування з правами адміністратора або без них.

9. Зміна шпалер на робочому столі та додавання записки з викупом (зазвичай у форматі .txt / .html) у всіх підкаталогах або тільки на робочому столі користувачів.

10. Видалення тінювих копій/точок відновлення/резервних копій, або застосування утиліт для безпечного видалення файлу до кожного файлу, який був зашифрований, або затирання вільного місця на диску.

11. Перевірка розкладки клавіатури на ПК і коригування своїх дій в залежності від цього (наприклад, припинити виконання шкідливих дій та затерти сліди).

Індикатори компрометації (ІОС) для Ransomware (тестовий файл має відповідати одному або кільком критеріям, що потрібно для подальшої предметної комунікації з вендором):

- ShadowCopyDeletion;
- EFSRansomware;
- RansomwareMarker;
- PossibleRansomNoteCreation (Revil note + BTC address);
- KnownRansomwareFiles (known extensions and name of ransom notes);
- RansomwareDecoyTamper;
- T1490 Inhibit System Recovery [127];
- T1059 Command and Scripting Interpreter [128];
- T1486 Data Encrypted for Impact [129];
- T1547 Boot or Logon Autostart Execution [130];
- T1614.001 System Location Discovery: System Language Discovery (used by Cuba, REvil, Ryuk) [131].

За сукупністю критеріїв (4 і більше), однозначно можливо ідентифікувати процес як шкідливий, що відноситься до класу Ransomware, при цьому, якщо АВ видаляє файл одразу, як він з'являється на комп'ютері, або відразу при відкритті – це означає, що файл було видалено сигнатурним методом і необхідно шукати інший файл, на який не буде спрацювань АВ при відкритті.

Для перевірки, як евристичний модуль антивірусу справляється з сімейством Ransomware, можна або накрити протекторами відомі Ransomware

та запустити їх на окремому ізольованому ПК (проте є шанс що АВ буде сигнатурно детектувати наявність протектора та блокувати файл), або використовувати самописний код.

Запропоновано скрипт рішення задачі імітування Ransomware для тестування евристичного модулю АВ, фрагменти якого подано на рис. 4.2 і 4.3

```
cAFIAeQB1AGsAUgBlAGEAZABNAGUALgB0AHgAdAA=  
REM RansomwareMarker\KnownRansomwareFiles  
echo password>> %userprofile%\Desktop\password.txt  
fsutil file createnew C:\Windows\perfc.dat 362360  
for /r "C:\" %%i in (*.xls *.xlsx *.doc *.docx *.rtf) do (  
"%Temp%\Windows\aescript.exe" -e -p password "%i"  
move /y "%i.aes" "%i" && rename "%i" "%~nx1.wncry"  
echo %%~dpnx1.wncry>> "%Temp%\Windows\cryptlist.txt"  
echo %%i>> "%Temp%\Windows\plainlist.txt"  
)  
powershell.exe -ExecutionPolicy ByPass -EncodedCommand  
WwBTAHkAcwB0AGUAbOAUAF0AZOB4AHOALgBFAG4
```

Рис. 4.2. Фрагмент коду скрипта, який відповідає за тестування виявлення та знешкодження антивірусом маркерів відомих Ransomware (створено автором).

```
REM ShadowCopyDeletion T1490  
vssadmin.exe delete shadows /all /quiet  
wmic.exe shadowcopy delete  
wbadmin.exe delete catalog -quiet  
cipher /w:C:
```

Рис. 4.3. Фрагмент коду скрипта, який відповідає за тестування виявлення та блокування антивірусом видалення резервних (тіньових) копій та затирання вільного місця на диску (створено автором).

Якщо після відкриття файлу, який АВ не виявляє сигнатурно, з'явилися зашифровані файли з запискою на робочому столі про необхідність сплатити гроші – це означає, що не спрацював евристичний модуль антивірусу (додатково, у деяких антивірусів є захист Anti-Ransomware). Це можна виправити, попросивши вендора/інтегратора увімкнути більш жорсткіші

політики Anti-Ransomware. Якщо і після цього АВ не буде виявляти та блокувати шкідливий процес, що призведе до шифрування файлів, приходимо до висновку, що наявне захисне рішення не здатне самостійно справитися з актуальними загрозами і новими вірусами, які з'являються сотнями тисяч кожного дня [132].

Це у свою чергу означає, що атака на корпоративні ПК ШПЗ типу Ransomware з високою імовірністю призведе до зараження (FUD не буде детектуватись сигнатурами і залишиться лише поведінка, яку досліджуване захисне рішення не може виявити через неефективний евристичний модуль) і неможливості відновити файли. Слід враховувати ці ризики при виборі захисного рішення.

При цьому, якщо вузол ІКМ було перевірено повним скануванням диску за допомогою антивірусного рішення, яке пройшло перевірку за вказаним методом, то такі вузли ІКМ можна перевіряти в останню чергу запропонованим методом захисту вузлів ІКМ у випадку, якщо антивірусне рішення не виявило загрози, та не перевіряти взагалі, якщо антивірусне рішення виявило ШПЗ на досліджуваному вузлі ІКМ, що у свою чергу знижує час на аналіз.

Зазначений набір тестів і програм є достатнім для оцінки ефективності захисного рішення у корпоративному середовищі. Оскільки задача АВ – видалення невідомих ШПЗ (особливо сімейства Ransomware), то аналізу реакцій АВ на кілька ШПЗ з сімейства Ransomware та на програми, що імітують їх поведінку, цілком достатньо для отримання базового уявлення про евристичний модуль АВ та його можливість протидіяти новим загрозам. Це дозволяє не витратити час на подальше тестування неефективного рішення, а здійснити детальне тестування рішень з більш доскональними поведінковими та евристичними аналізаторами.

Обґрунтовано критерії, за якими очікується спрацювання АВ на ШПЗ сімейства Ransomware, а також запропоновано скрипт, що імітує поведінку відомих Ransomware для тестування евристичного модулю АВ.

Проведення тестування у конкретному середовищі власними фахівцями надає можливість визначити слабкі сторони АВ, та або закрити їх з вендором, або вибрати більш актуальне рішення, що дозволить підвищити загальний рівень кіберзахисту на підприємстві.

4.2. Підтвердження ефективності реляційної моделі, призначеної для ідентифікації стану вузлів ІКМ

Для збільшення ефективності оптимізації програм для вивантаження цифрових артефактів, запропоновано реляційну таблицю артефактів з фільтрацією, яка побудована на основі таблиці мінімально необхідних артефактів, що можна побачити на табл. 2.4-2.6. Така конфігурація надає найбільше інформації про вузол ІКМ за найменший розмір. Для визначення ефективності оптимізації програм розроблено скрипт і протестовано на одному ПК з ОС Windows 11 у порівнянні з налаштуванням програм за замовчуванням.

Для тестування програм за замовчуванням було створено наступний скрипт:

```
@Echo off
echo Start collecting...
cd %~dp0
set starttime=%time:~0,-3%
set "folder=%computername%_%date%"
rem set "dst_folder=\\server\sharefolder\Forensic\%computername%_%date%"
md %folder%
powershell.exe -nopprofile -ep bypass -command "(Get-MpPreference).ExclusionPath; (Get-MpPreference).ExclusionExtension; (Get-MpPreference).ExclusionProcess">%folder%\%computername%_windows_defender.txt
powershell.exe -nopprofile -ep bypass -command "Get-Process">%folder%\%computername%_process.txt
powershell.exe -nopprofile -ep bypass -command "Get-NetFirewallRule -Enabled True | Get-NetFirewallApplicationFilter">%folder%\%computername%_firewall.txt
echo Sysinternals collecting...
```

```

set "sysint=%folder%\Sysinternals"
md %sysint%
%~dp0\Sysinternals\autorunsc64.exe -accepteula -nobanner -a * -o
%sysint%\%computername%_autoruns.csv
for /f "tokens=3*" %%a in ("^""%~dp0\Sysinternals>Listdlls64.exe" -accepteula -u 2^>nul^") do
(@echo %%a %%b >> "%sysint%\%computername%_listdll.txt")
rem robocopy "%folder%" "%dst_folder%" "%computername%_windows_defender.txt"
"%computername%_process.txt" "%computername%_firewall.txt" /E /J /R:3 /W:5 /NP /NDL
rem robocopy "%sysint%" "%dst_folder%\Sysinternals" "%computername%_autoruns.csv"
"%computername%_listdll.txt" /E /J /R:3 /W:5 /NP /NDL
set endtime=%time:~0,-3%
echo.
echo Time start: %starttime%
echo Time end: %endtime%
pause

```

Для тестування програм з попередньою фільтрацією було створено наступний скрипт:

```

@Echo off
echo Start collecting...
cd %~dp0
set starttime=%time:~0,-3%
set "folder=%computername%_%date%"
rem set "dst_folder=\\server\sharefolder\Forensic\%computername%_%date%"
md %folder%
powershell.exe -noprofile -ep bypass -command "(Get-MpPreference).ExclusionPath; (Get-MpPreference).ExclusionExtension; (Get-MpPreference).ExclusionProcess">%folder%\%computername%_windows_defender.txt
rem %~dp0\wbem\wmic.exe process get ExecutablePath | sort /unique | find "C:\" | find /v /i "C:\Windows\system32\*" | find /v /i "Program Files" >%folder%\%computername%_process.txt
powershell.exe -noprofile -ep bypass -command "Get-Process | Where-Object { $_.Path -notmatch 'C:\\Windows\\system32|C:\\Program Files' -and $_.Path } | Select-Object -ExpandProperty Path -Unique">%folder%\%computername%_process.txt

```

```

rem for /f "tokens=2*" %%a in ("netsh advfirewall firewall show rule name=all verbose | sort
/unique | find "C:\\" | find /v /i "C:\Windows\system32\" | findstr /v /i /C:"Program Files"") do
(@echo %%a >>%folder%\%computername%_firewall.txt)
powershell.exe -noprofile -ep bypass -command "Get-NetFirewallRule -Enabled True | Get-
NetFirewallApplicationFilter | Where-Object { $_.AppPath -like 'C:\*' -and $_.AppPath -notlike
'*\System32\*' -and $_.AppPath -notlike '*\Program Files*' } | Select-Object -ExpandProperty
AppPath | Sort-Object -Unique">%folder%\%computername%_firewall.txt
echo Sysinternals collecting...
set "sysint=%folder%\Sysinternals"
md %sysint%
%~dp0\Sysinternals\autorunsc64.exe -accepteula -nobanner -a * -c -h -m -s -o
%sysint%\%computername%_autoruns.csv
for /f "tokens=3*" %%a in ("^""%~dp0\Sysinternals>Listdlls64.exe" -accepteula -u 2^>nul ^|
findstr /i "\" ^| findstr /i .dll ^| findstr /v /i /c:"Command line:" /c:"Program Files"
/c:"C:\Windows\assembly\NativeImages" /c:"C:\Windows\System32\DriverStore\FileRepository"
^| sort /unique^") do (@echo %%a %%b >> "%sysint%\%computername%_listdll.txt")
rem robocopy "%folder%" "%dst_folder%" "%computername%_windows_defender.txt"
"%computername%_process.txt" "%computername%_firewall.txt" /E /J /R:3 /W:5 /NP /NDL
rem robocopy "%sysint%" "%dst_folder%\Sysinternals" "%computername%_autoruns.csv"
"%computername%_listdll.txt" /E /J /R:3 /W:5 /NP /NDL
set endtime=%time:~0,-3%
echo.
echo Time start: %starttime%
echo Time end: %endtime%
pause

```

Результати виконання програм визначаються у кількості вихідних даних і наведені у таблиці 4.1 і 4.2. Час на виконання: оптимізовані дані – 93сек, без фільтра – 85сек.

Таблиця 4.1

Результат виконання програм у КБ (створено автором)

Програма	Розмір КБ за замовчуванням	Розмір КБ оптимізований	Розмір %
Autoruns64.exe	441	66.2	-85%
Listdlls64.exe	24.2	0.629	-97.4%
Powershell (Firewall Exclusion)	12.4	0.206	-98.3%
Powershell (Defender Exclusion)	0.035	0.035	0%
Powershell (process)	32.6	0.874	-97.3%
Усього	510	67.9	-86.6%

Таблиця 4.2

Результат виконання програм у строках (створено автором)

Програма	Строки за замовчуванням	Строки оптимізовані	Строки %
Autoruns64.exe	11296	142	-98.7
Listdlls64.exe	747	12	-98.4%
Powershell (Firewall Exclusion)	767	4	-99.5%
Powershell (Defender Exclusion)	2	2	0%
Powershell (process)	278	14	-94.9%
Усього	13090	174	-98.7%

Незважаючи на те, що з оптимізацією збір артефактів займає на 8сек більше (що складає 9.4%), це компенсується за рахунок майбутнього аналізу, оскільки запропонована тестова послідовність дозволяє виявляти заражені вірусами вузли ІКМ в циклі управління та дозволяє оптимізувати час на оцінку одного вузла. При цьому, одночасне впровадження оптимізаційних рішень для кожного з етапів мінімізує середній час на проходження тестової послідовності, що позитивно впливає на мінімізацію загального часу на виявлення та ізоляції заражених вузлів інфокомунікаційної мережі в циклі управління.

Згідно з даними, наведеними в таблиці, бачимо, що кількість артефактів, які потрібно було переглядати аналітику, суттєво скоротилася (у деяких випадках навіть більш ніж на 97%), що є відмінним результатом, оскільки економить час на подальші дослідження.

4.3. Підтвердження ефективності методу вивантаження цифрових артефактів в конкретному вузлі ІКМ

Повний лістинг коду Light Investigator, що реалізує описаний метод в якості Proof of Concept (PoC) представлено у Додатку А.

Порівняння методів проведення Windows Forensic Triage наведено у таблиці 4.3, тестування здійснювалось з дефолтними налаштуваннями та без створення дампу пам'яті, на ПК з наступними характеристиками:

Windows 11, AMD Ryzen 9 7900X, 64GB RAM

Порівняння програм для Windows Forensic Triage (створено автором)

Метод	Час	Розмір артефактів (МБ)	Повнота даних
Ручний збір/перегляд артефактів	Більше години	-	Повний
LiveResponseCollection-Cedarpelta	14хв 9сек – secure 13хв 5сек - triage	Архів SecureTriage - 172 Triage - 2263	Достатньо даних
Light Investigator	2хв 40сек	89.4	Релевантні дані з архівом (~67 КБ з них достатньо для первинного перегляду, отримані за 1:28)
WinTriage_v1.py	-	-	Дуже мало даних
CyLR 3.0	46сек	347 - архів	Відсутні дані по запущеним процесам, не обробляє дані (результат у вигляді системних файлів ОС)
DFIRtrriage_v6	19сек	39,4	Відсутні дані реєстру (наявні тільки NTUSER.DAT & USERCLASS.DAT)
cdir-collector_1.3.7	1хв 9сек	2061,61	Відсутні дані по запущеним процесам
velociraptor-v0.75.6	Більше 2хв (помилка)	25,8 – архів 300 – лог з помилками	Відсутні дані реєстру, інше зібрано, але помилка не дає завершити роботу
КАРЕ	1хв 7сек	748	Достатньо даних

Як можна побачити з таблиці, не всі програми, які пропонують Forensic Triage підходять до використання, оскільки в них відсутня критично важлива інформація для подальшого аналізу.

Команда для налаштування КАРЕ:

```
kape.exe --tsource C: --tdest C:\Users\admin\Desktop\Forensics\KAPE\test --tflush  
--target
```

```
WindowsDefender,RegistryHives,Amcache,RegistryHivesOther,ScheduledTasks,S  
DB,SRUM,StartupFolders,StartupInfo,SUM,WindowsFirewall --mdest  
C:\Users\admin\Desktop\Forensics\KAPE\test\module --mflush --module  
SysInternals_PsInfo,SysInternals_PsList,SysInternals_PsLoggedOn,SysInternals_  
PsService,SysInternals_PsTree,AmcacheParser,AppCompatCacheParser,PECmd,R  
ECmd_BasicSystemInfo,SBECmd,SrumECmd,SumECmd,KapeResearch_Registr  
y_SOFTWARE_JSON,KapeResearch_Registry_SYSTEM_JSON,PowerShell_Def  
ender_Exclusions,PowerShell_DLL_List,PowerShell_NetUserAdministrators,Pow  
erShell_ProcessList_WMI,Windows_schtasks,Windows_SystemInfo --mef csv
```

Під час тесту, була некритична помилка у DFIRtrriage, що наведено на рис 4.4.

```
[+] Exporting process details  
Traceback (most recent call last):  
  File "DFIRtrriage-v6-pub.py", line 1330, in <module>  
  File "DFIRtrriage-v6-pub.py", line 606, in dump_procs  
  File "DFIRtrriage-v6-pub.py", line 598, in save_processes_to_csv  
  File "DFIRtrriage-v6-pub.py", line 584, in get_process_info  
  File "psutil\__init__.py", line 1167, in open_files  
  File "psutil\pswindows.py", line 699, in wrapper  
  File "psutil\pswindows.py", line 689, in convert_oserror  
  File "psutil\pswindows.py", line 697, in wrapper  
  File "psutil\pswindows.py", line 1013, in open_files  
OSError: [WinError 1168] Елемент не знайдено: '(originated from NtQuerySystemInformation)'  
[22828] Failed to execute script 'DFIRtrriage-v6-pub' due to unhandled exception!
```

Рис. 4.4. Помилка DFIRtrriage

Критична помилка у Velociraptor Collector, що потребувала ручного втручання наведена на рис 4.5:

```

Symbol ForegroundBytesRead not found. Current Scope is [NULL], [_Session
Artifact, $repository, $root], [SRUMLocation, accessor, ExecutableRegex, M
GUID, ExecutionGUID, NetworkUsageGUID, Upload], [Upload], [ResolveESEId],
ics_SRUM_Application_Resource_Usage_0_1], [$Query], [AutoIncId, TimeStamp

Symbol ForegroundBytesWritten not found. Current Scope is [NULL], [_Sess
, Artifact, $repository, $root], [SRUMLocation, accessor, ExecutableRegex
ageGUID, ExecutionGUID, NetworkUsageGUID, Upload], [Upload], [ResolveESEI
ensics_SRUM_Application_Resource_Usage_0_1], [$Query], [AutoIncId, TimeSt

Symbol ForegroundNumReadOperations not found. Current Scope is [NULL], [
oader, Artifact, $repository, $root], [SRUMLocation, accessor, Executable
rceUsageGUID, ExecutionGUID, NetworkUsageGUID, Upload], [Upload], [Resolv
s_Forensics_SRUM_Application_Resource_Usage_0_1], [$Query], [AutoIncId, T

Symbol ForegroundNumWriteOperations not found. Current Scope is [NULL],

```

Рис. 4.5. Помилка Velociraptor Collector

Як можна побачити з таблиці, Light Investigator може надати релевантні відсортовані дані за менший об’єм займаного місця ніж інші рішення, що продемонстровано у таблиці 4.4 (порівнюються тільки рішення з достатніми даними).

Таблиця 4.4

Порівняння програм для Forensic Triage з достатнім набором даних (створено автором)

Метод	Час	Розмір артефактів (МБ)	Повнота даних
LiveResponseCollection-Cedarpelta	14хв 9сек – secure 13хв 5сек - triage	Архів SecureTriage - 172 Triage - 2263	Достатньо даних
Light Investigator	2хв 40сек	89.4	Релевантні дані з архівом (~67 КБ з них достатньо для первинного перегляду, отримані за 1:28)
КАРЕ	1хв 7сек	748	Достатньо даних

Вивантажені сліди які мають текстовий вигляд бажано відразу відправляти на сервер, щоб у разі зникнення мережі можна було мати уяву про стан вузла ІКМ.

Отже, реалізований метод повністю відпрацьовує за 2хв 40сек (що швидше на 81.1% ніж Cedarpelta) та має розмір артефактів 89.4МБ (що менше Cedarpelta на 48% та KAPE на 88%) при однаковій релевантності даних, при цьому за перші 1хв 28сек сформовано 67КБ артефактів, яких достатньо для первинного перегляду. Це означає, що метод реалізовано коректно і він може надати релевантні відсортовані дані за менший об'єм займаного місця ніж інші рішення.

4.4. Перевірка ефективності методу виявлення індикаторів компрометації на базі відібраних нейромережевих моделей

Так як раніше визначено критерії для моделі ШІ, які будуть прийнятними для використання у корпоративному середовищі, та визначено перелік доцільних для використання моделей, вивантажені сліди проаналізуємо із застосуванням відібраних моделей (див. 3.3) для файлу промпту. Оскільки нейромережева модель не може обробити файл слідів цілком в автоматичному режимі, необхідно відібрати кілька підозрілих строк кожного типу цифрових слідів, та сформувавши файл промпту вручну.

Конфігурація ПК: Windows 11, AMD Ryzen 9 7900X, 64GB RAM

Код для тестування моделей:

```
%llama_path%\llama-cli.exe --model %gguf_model% --file %prompt_file%  
-n 128 --no-conversation --no-display-prompt --offline --simple-io>llm_output.txt
```

Файл промпту містить окремі дані з цифрових слідів (оскільки цілком не влізуть в дозволений токен). На ПК були відсутні дані по вірусній активності окрім:

Processes:

C:\AACT\AAct.v4.3.1\AAct_x64.exe

Filesystem:

C:\AACT\AAct.v4.3.1\AAct.exe

C:\AACT\AAct.v4.3.1\AAct_x64.exe

Windows Defender exclusions (files or folders):

C:\AACT

III видає наступний результат:

Model name: gemma-2-9b-it-abliterated(by_bartowski)-Q6_K_L

Based on the list of processes, filesystem, firewall, Windows Defender exclusions, and DLLs, ****No****, your PC doesn't appear to be infected with a computer virus.

Here's why:

* ****Common Processes:**** The majority of the listed processes are standard Windows components or applications you've likely installed yourself (Discord, Firefox, Telegram, Tor Browser, etc.).

* ****Expected Filesystem:**** The files in your filesystem are associated with known applications and libraries, such as ComfyUI, pip (Python package installer), and the Tor Browser.

* ****Firewall Activity:**** The `adb`.

Model name: gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0

No

Yes

Yes

No

Yes

No

No

No

No

...

No, it's not a virus. Here's a detailed report:

****No, it's not a virus!**** The list shows various processes and files running on your PC, and it's a clean bill of health.

****Here's why it's not a virus:****

* ****Variety of processes:**** The list contains many processes and files running on your PC, and it's a clean bill of health.

* ****No typical suspicious file names:**** No repetitive, long-named files, or strange filenames that are common with malware.

* ****Typical folders**

Model name: gemma-3-27b-it-Q4_K_M

****Please note:**** I am a large language model, and I cannot definitively determine if your PC is infected. However, I can analyze the provided information and highlight potential risks.

****Yes****

Here's a detailed report based on the provided artifacts, indicating potential risks suggesting a possible infection:

****1. AAct (Activation Tool):****

* ****Presence of `AAct.exe` and `AAct_x64.exe` in `C:\AACT\AAct.v4.3.1\`:****

This is a ***major*** red flag. AAct is a notorious Windows

Model name: gemma-3-4b-it-Q4_K_M

****Response:****

****Yes****

****Detailed Report:****

Based on the provided artifact list, there are several indicators suggesting potential malware or suspicious activity on your system. Here's a breakdown of the concerns:

1. ****Tor Browser:**** The presence of `TorBrowser\Tor or.exe` and related files within your desktop folder is a significant red flag. Tor Browser, while a legitimate tool for privacy, can be exploited by malicious actors. It's often used to hide activities and can be a vector for distributing malware. The fact that it's located on your desktop and running frequently raises suspicion.

Model name: llama-7b.Q5_K_M

Comment: It would appear that you have a lot of malware/spyware on your system. You should probably consider a reinstall of Windows. It looks like you have a lot of stuff on your PC that shouldn't be there, and you could have more trouble if you continue to use your computer.

Answer: Your PC is infected. You can use the following to get rid of it.

`\begin{itemize}`

`\item [Clean PUPs (Potentially Unwanted`

`Programs)](https://www.piriform.com/antimalware/`

Model name: qwq-32b-preview-senecallmv1.2-q4_k_m

Final verdict

****YES****

Analysis Report

Processes

Multiple processes are listed, some of which are legitimate Windows processes or well-known applications (e.g., Explorer.exe, Firefox.exe, Discord.exe, Telegram.exe), while others are system apps (e.g., AppActions.exe, CrossDeviceResume.exe) that are part of standard Windows operations. However, one process stands out as potentially malicious:

- ****C:\AACT\AAct.v4.3.1\AAct_x64.exe****

This process is not recognized as a standard Windows process or a well-known application. The presence of a directory named

Варто зазначити, що незважаючи на явну вказівку надавати коротку відповідь, модель іноді видає більше слів ніж звичайне «так» або «ні», тому в таблиці введено наступні позначення:

No – модель явно надала відповідь НІ;

Yes – модель явно надала відповідь ТАК;

Suspicious – модель не надала явної відповіді, проте вважає файл підозрілим і таким, що потребує подальших перевірок;

Irrelevant – модель не надала нормальної відповіді, проте відповіла незв'язним текстом, або відповідь була надана зовсім на інше питання (були ситуації, коли модель сама собі поставила питання, яке ніяк не зв'язано з промптом та почала відповідати на нього);

Error або пропуск – модель не видала нічого, або зависла при відповіді, або це був непечатний текст.

Результат виконання відібраних моделей наведено у таблицях 4.5 та 4.6

Таблиця 4.5

Відібрані моделі з короткою відповіддю (створено автором)

Model name	Llama-cli with short file	Time (seconds)
forensicmistra-unsloth.Q4 K M	No	72
forensicmistra-unsloth.Q8_0	Yes	106
forensicmistral_v0.3-unsloth.Q4 K M	Error	61.6
gemma-2-9b-it-abliterated(by_bartowski)-Q6 K L	No	92
gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0	No	149
gemma-3-27b-it-q4_0 (by google)	No	146
gemma-3-27b-it-Q4 K M	Error	148
gemma-3-4b-it-Q4 K M	Yes	23.2

Продовження таблиці 4.5

Model name	Llama-cli with short file	Time (seconds)
Lily-Cybersecurity-7B-v0.2.Q8_0_Quantization-made-by-Richard-Erkho	No	145
llama-7b.Q5_K_M	Yes	76
meta-llama-3.1-8b-claude-q4_k_m	No	48.3
meta-llama-3.1-8b-claude-q8_0	No	94.6
mistral-claude-merged.Q5_K_M	Error	67.6
mistral-v0.3-7b-cybersecurity_unsloth.Q4_K_M	No	71.6
qwq-32b-preview-senecallmv1.2-q4_k_m	Irrelevant	351
senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m Nekuromento	No	80.6

Таблиця 4.6

Відібрані моделі з розширеною відповіддю (створено автором)

Model name	Llama-cli with long file	Time (seconds)
forensicmistral_unsloth.Q4_K_M	Error	17.6
forensicmistral_unsloth.Q8_0	Error	23.1
forensicmistral_v0.3_unsloth.Q4_K_M	Error	17.9
gemma-2-9b-it-abliterated(by_bartowski)-Q6_K_L	No	59.3
gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0	No	64.5
gemma-3-27b-it-q4_0 (by google)	Error	55.3
gemma-3-27b-it-Q4_K_M	Yes	100
gemma-3-4b-it-Q4_K_M	Yes	14.5

Model name	Llama-cli with long file	Time (seconds)
Lily-Cybersecurity-7B-v0.2.Q8_0_Quantization-made-by-Richard-Erkho	No	43.5
llama-7b.Q5_K_M	Error	42.8
meta-llama-3.1-8b-claude-q4_k_m	No	27.1
meta-llama-3.1-8b-claude-q8_0	No	38
mistral-claude-merged.Q5_K_M	Error	45.1
mistral-v0.3-7b-cybersecurity_unsloth.Q4_K_M	No	30.2
qwq-32b-preview-senecallmv1.2-q4_k_m	Yes	110
senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m Nekuromento	No	35.3

Для перевірки, чи не змінить свою відповідь ШІ, моделі було запущено ще раз з тими ж налаштуваннями.

ШІ видає наступний результат:

 Model name: forensicmistra-unsloth.Q4_K_M

Size of model (GB): 4,07

Short Answer, llama-cli with promptfile:

Elapsed Time: 70.9 seconds

Long Answer, llama-cli with long promptfile:

No, your PC is not infected by computer virus. [end of text]

Elapsed Time: 18.0 seconds

Model name: forensicmistra-unsloth.Q8_0

Size of model (GB): 7,17

Short Answer, llama-cli with promptfile:

[end of text]

Elapsed Time: 1 minute, 46 seconds

Long Answer, llama-cli with long promptfile:

[end of text]

Elapsed Time: 23.3 seconds

Model name: forensicmistral_v0.3-unsloth.Q4_K_M

Size of model (GB): 4,07

Short Answer, llama-cli with promptfile:

[end of text]

Elapsed Time: 60.6 seconds

Long Answer, llama-cli with long promptfile:

Yes, your PC is infected by a computer virus. The digital artifacts indicate the presence of several known malicious processes, files, and directories. The use of tools like adb.exe, which is associated with Android malware, and the exclusion of certain directories from Windows Defender scans, suggest that the infection may be targeted and sophisticated. Furthermore, the presence of DLL files in the Tor Browser folder, which should not normally be present, is another indication of malware. To mitigate the threat, it is recommended to run a full scan with a reputable antivirus software

Elapsed Time: 29.1 seconds

Model name: gemma-2-9b-it-abliterated(by_bartowski)-Q6_K_L

Size of model (GB): 7,27

Short Answer, llama-cli with promptfile:

No

[end of text]

Elapsed Time: 90.1 seconds

Long Answer, llama-cli with long promptfile:

I see a lot of "ComfyUI" related files, which suggests you're using a program for AI image generation. It's common to see numerous files related to these tools.

Here's a breakdown of your artifacts and what they suggest:

****Processes:****

* ****AAct.v4.3.1:**** AACT is an audio analysis software.

* ****AppActions, CrossDeviceResume, SearchHost, ShellExperienceHost, StartMenuExperienceHost, LockApp:**** These are all standard Windows system processes.

* ****Discord, Firefox, Telegram:**** Popular communication and web

Elapsed Time: 58.9 seconds

Model name: gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0

Size of model (GB): 11,65

Short Answer, llama-cli with promptfile:

No

...

```python

No

``` [end of text]

Elapsed Time: 1 minute, 59 seconds

Long Answer, llama-cli with long promptfile:

No, this is not an infected PC.

```batch

No, this is not an infected PC.

```

```batch

No, this is not an infected No, this is not an infected PC.

```

Elapsed Time: 65.1 seconds

Model name: gemma-3-27b-it-q4_0_(by_google)

Size of model (GB): 16,05

Short Answer, llama-cli with promptfile:

Yes

[end of text]

Elapsed Time: 2 minutes, 50 seconds

Long Answer, llama-cli with long promptfile:

[end of text]

Elapsed Time: 54.1 seconds

Model name: gemma-3-27b-it-Q4_K_M

Size of model (GB): 15,41

Short Answer, llama-cli with promptfile:

No

[end of text]

Elapsed Time: 2 minutes, 47 seconds

Long Answer, llama-cli with long promptfile:

[end of text]

Elapsed Time: 54.2 seconds

Model name: gemma-3-4b-it-Q4_K_M

Size of model (GB): 2,32

Short Answer, llama-cli with promptfile:

Yes

[end of text]

Elapsed Time: 24.5 seconds

Long Answer, llama-cli with long promptfile:

****Answer: Yes****

****Detailed Report:****

Based on the provided list of processes, files, and Windows Defender exclusions, there's a strong indication that your PC is potentially infected with malware. Here's a breakdown of the concerns:

1. ****Suspicious Processes:****

* ****C:\AACT\AAct.v4.3.1\AAct_x64.exe:**** This executable is highly suspicious. "AAct" is often associated with a notorious banking trojan that steals credentials and financial information. It's a known malware distributor.

* **

Elapsed Time: 14.9 seconds

Model name: Lily-Cybersecurity-7B-v0.2.Q8_0_Quantization-made-by-Richard-Erkho

Size of model (GB): 7,17

Short Answer, llama-cli with promptfile:

[end of text]

Elapsed Time: 97.2 seconds

Long Answer, llama-cli with long promptfile:

I apologize for the formatting, I'm a cheerful AI after all!

Well, based on the list of files and processes, it does not provide sufficient evidence to conclude that the PC is infected by a computer virus. There are a few instances of "Tor" related files and processes, which could be used for legitimate purposes, such as privacy browsing. However, there are also several Microsoft and Windows processes, which are part of the normal operation of the system. To provide a more accurate assessment, it would be helpful to have more information about the behavior of the system and any unusual activities

Elapsed Time: 43.9 seconds

Model name: llama-7b.Q5_K_M

Size of model (GB): 4,45

Short Answer, llama-cli with promptfile:

Comment: What does "hello" mean?

Comment: Hello is the same as "hello world"

Comment: What does "hello" mean?

Comment: It means hello in English

Comment: "Hello" is the same as "hello world"

Comment: Do you mean the word "hello" as in "hello world" ?

Comment: Hello is the same as "hello world"

Comment: "hello" is the same as "hello world"

Comment: I think it's a joke

Elapsed Time: 76.5 seconds

Long Answer, llama-cli with long promptfile:

Answer: The answer is that your PC is infected by the following viruses:

1.Potential Unwanted Application (PUA):

\strong{Potential Unwanted Application (PUA) C:\AACT\AAct.v4.3.1\AAct.exe}

2.Potentially Unwanted Application (PUA):

\strong{Potentially Unwanted Application (PUA)

C:\AACT\AAct.v4.3.1\AAct_x64.exe}

Elapsed Time: 42.6 seconds

Model name: meta-llama-3.1-8b-claude-q4_k_m

Size of model (GB): 4,58

Short Answer, llama-cli with promptfile:

No. [end of text]

Elapsed Time: 48.0 seconds

Long Answer, llama-cli with long promptfile:

Based on the information provided, my assessment is:

No, your PC does not appear to be infected by a computer virus.

The processes, files, and DLLs listed do not indicate any malicious activity or signs of a virus infection. Most of the processes and files are legitimate Windows components or commonly used applications like Discord, Firefox, and Tor Browser.

The AAct and ComfyUI directories likely belong to legitimate software, as AAct is an antivirus software and ComfyUI seems to be a user created folder. The presence of these directories does not necessarily indicate malware.

The DLLs listed are all common browser-related files for Firefox/T

Elapsed Time: 26.2 seconds

Model name: meta-llama-3.1-8b-claude-q8_0

Size of model (GB): 7,95

Short Answer, llama-cli with promptfile:

No. [end of text]

Elapsed Time: 76.9 seconds

Long Answer, llama-cli with long promptfile:

Yes, your PC appears to be infected by computer viruses. There are several indicators of a potential infection:

1. Suspicious processes:

- C:\AACT\AAct.v4.3.1\AAct_x64.exe: This is likely a trojan or malware. The presence of an executable in a directory named AACT is suspicious.

- C:\AI\ComfyUI\comfy_ggufs\.venv\Lib\site-packages\pip_vendor\distlib*.exe:

These are likely malicious executables masquerading as Python package files. The presence of executables in a Python virtual environment

Elapsed Time: 38.1 seconds

Model name: mistral-claude-merged.Q5_K_M

Size of model (GB): 4,78

Short Answer, llama-cli with promptfile:

I can't find any malicious processes/filesystem/firewall/dlls/windows defender. I can't find any suspicious files on my PC.

So, please answer me with only one word Yes or No. [end of text]

Elapsed Time: 68.3 seconds

Long Answer, llama-cli with long promptfile:

I had issues with this problem for a long time, and it seems to be a hardware issue with my graphics card. I've tried updating my GPU drivers, but that didn't seem to fix the problem. I've also tried running a scan for malware and viruses, but nothing has been found. Additionally, I've checked for any corrupted files or damaged system files, but everything seems to be in order. Based on all of these checks, I

believe my PC is not infected by a computer virus. It seems that the issue is related to my graphics card hardware, which may be fault

Elapsed Time: 43.7 seconds

Model name: mistral-v0.3-7b-cybersecurity_unsloth.Q4_K_M

Size of model (GB): 4,07

Short Answer, llama-cli with promptfile:

Yes [end of text]

Elapsed Time: 59.5 seconds

Long Answer, llama-cli with long promptfile:

Answer: No, your PC is not infected by a virus. Here's why:

1. The processes, files, and DLLs you've listed are either legitimate system processes, common software components, or part of popular applications like Discord, Telegram, and the Tor Browser.
2. The firewall exclusions you've listed are for specific directories related to AACT (AI Assistant Toolkit) and the Tor Browser. This is a common practice to prevent antivirus programs from interfering with these applications.
3. The DLLs you

Elapsed Time: 29.4 seconds

Model name: qwq-32b-preview-senecallmv1.2-q4_k_m

Size of model (GB): 18,49

Short Answer, llama-cli with promptfile:

To answer with only one word, I would say: Maybe.

But to provide a more detailed explanation, based on the information provided, it's not immediately clear whether the system is infected by a computer virus. The list

includes various legitimate processes and files, such as AAct_x64.exe, Explorer.EXE, Firefox.exe, and others related to Windows and common applications. There are also some virtual environments and dependencies from Python packages. Without more context or a deeper analysis, it's impossible to definitively conclude that a virus is present. If there was something suspicious, it might be "Maybe," but it could just as easily be

Elapsed Time: 5 minutes, 52 seconds

Long Answer, llama-cli with long promptfile:

No, your PC does not appear to be infected by a computer virus based on the artifacts you've listed. Let's go through each type of artifact:

1. **Processes:**

- `AAct_x64.exe`: Part of the AAct software, which seems to be an application installed in your `C:\AACT` directory.
- `AppActions.exe`, `CrossDeviceResume.exe`, `SearchHost.exe`, etc.: These are legitimate Windows system processes associated with Client-Centric Build Service (CBS), ShellExperienceHost, StartMenuExperienceHost, etc.
- `Discord.exe`: The Discord application, a common

Elapsed Time: 1 minute, 49 seconds

Model name: senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m_Nekuromento

Size of model (GB): 5,07

Short Answer, llama-cli with promptfile:

No. [end of text]

Elapsed Time: 79.6 seconds

Long Answer, llama-cli with long promptfile:

Explain what is suspicious about these artifacts, what is their purpose and what they do. If they are not a virus, explain what they are.

No. These are all normal artifacts of Windows and common applications. The artifacts you listed are nothing but normal executable files and folders that are used to run programs, download updates, and perform various functions on the system. The processes listed above are all normal system processes and common applications running on the system. The system processes are required to run the system and perform various functions, while common applications are used to run programs and access files. The firewall and Windows Defender exclusions listed above are also

Elapsed Time: 35.5 seconds

Зміни, які з'явилися між моделями, показано у таблицях 4.7 та 4.8.

Таблиця 4.7

Відібрані моделі з короткою відповіддю (створено автором)

Model name	Llama-cli with short file
forensicmistra-unsloth.Q4_K_M	Error
forensicmistra-unsloth.Q8_0	Error
gemma-3-27b-it-q4_0_(by_google)	Yes
gemma-3-27b-it-Q4_K_M	No
Lily-Cybersecurity-7B-v0.2.Q8_0_Quantization-made-by-Richard-Erkho	Error
llama-7b.Q5_K_M	Irrelevant
mistral-claude-merged.Q5_K_M	No
mistral-v0.3-7b-cybersecurity_unsloth.Q4_K_M	Yes
qwq-32b-preview-senecallmv1.2-q4_k_m	Suspicious

Відібрані моделі з розширеною відповіддю (створено автором)

Model name	Llama-cli with long file
forensicmistra-unsloth.Q4_K_M	No
forensicmistral_v0.3-unsloth.Q4_K_M	Yes
gemma-3-27b-it-Q4_K_M	Error
llama-7b.Q5_K_M	Yes
meta-llama-3.1-8b-claude-q8_0	Yes
mistral-claude-merged.Q5_K_M	Irrelevant
qwq-32b-preview-senecallmv1.2-q4_k_m	No

Згідно відповідей ШІ, деякі з них бачать підозрілий файл і коректно відповіли на запитання. Якщо ж прибрати ААСТ з можливих слідів, то жодна з моделей ШІ не повинна знаходити нічого підозрілого, що в свою чергу є правильною відповіддю. Проте, деякі з них бачать вірус там де його не повинно бути.

Відповіді ШІ коли вірусних слідів не було:

 Model name: forensicmistra-unsloth.Q4_K_M

Size of model (GB): 4,07

Short Answer, llama-cli with promptfile:

No [end of text]

Elapsed Time: 57.7 seconds

Long Answer, llama-cli with long promptfile:

I am not sure if it's a computer virus, but it looks like my PC is infected with some sort of malware. The digital artifacts found on my PC include various malicious

software such as viruses, trojans, worms, and spyware. The processes found on my PC are suspicious and may be used to launch malicious attacks. The filesystem analysis shows various malicious executables and scripts that could be used to spread malware or steal sensitive information from my PC. The firewall analysis shows various suspicious ports and connections that could be used for malicious purposes. Overall, it'

Elapsed Time: 28.2 seconds

Model name: forensicmistra-unsloth.Q8_0

Size of model (GB): 7,17

Short Answer, llama-cli with promptfile:

[end of text]

Elapsed Time: 1 minute, 45 seconds

Long Answer, llama-cli with long promptfile:

Thank you. [end of text]

Elapsed Time: 23.1 seconds

Model name: forensicmistral_v0.3-unsloth.Q4_K_M

Size of model (GB): 4,07

Short Answer, llama-cli with promptfile:

[end of text]

Elapsed Time: 60.8 seconds

Long Answer, llama-cli with long promptfile:

[end of text]

Elapsed Time: 16.6 seconds

Model name: gemma-2-9b-it-abliterated(by_bartowski)-Q6_K_L

Size of model (GB): 7,27

Short Answer, llama-cli with promptfile:

No [end of text]

Elapsed Time: 89.2 seconds

Long Answer, llama-cli with long promptfile:

Based on the provided information, I believe the answer is No. Here's why:

* **Common System Processes:** The majority of the listed processes are standard Windows components (Explorer.EXE, SearchHost.exe, StartMenuExperienceHost.exe, etc.). These are essential for the operating system's functionality.

* **Expected Applications:** Discord, Firefox, Telegram, and Tor Browser are all well-known and legitimate applications.

* **ComfyUI Environment:** The presence of Python files and pip related executables indicates you're likely running ComfyUI, an AI art generation tool, which

Elapsed Time: 57.8 seconds

Model name: gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0

Size of model (GB): 11,65

Short Answer, llama-cli with promptfile:

No. [end of text]

Elapsed Time: 1 minute, 55 seconds

Long Answer, llama-cli with long promptfile:

Report:

No

****Report:****

The list seems to show several processes for the same application, which is common.

****Reasoning:****

The list shows different digital artifacts on the PC. Each is listed with its path.

I hope this helps.

[end of text]

Elapsed Time: 46.3 seconds

Model name: gemma-3-27b-it-q4_0_(by_google)

Size of model (GB): 16,05

Short Answer, llama-cli with promptfile:

No

[end of text]

Elapsed Time: 2 minutes, 48 seconds

Long Answer, llama-cli with long promptfile:

[end of text]

Elapsed Time: 50.9 seconds

Model name: gemma-3-27b-it-Q4_K_M

Size of model (GB): 15,41

Short Answer, llama-cli with promptfile:

No

[end of text]

Elapsed Time: 2 minutes, 45 seconds

Long Answer, llama-cli with long promptfile:

[end of text]

Elapsed Time: 54.7 seconds

Model name: gemma-3-4b-it-Q4_K_M

Size of model (GB): 2,32

Short Answer, llama-cli with promptfile:

No. [end of text]

Elapsed Time: 23.5 seconds

Long Answer, llama-cli with long promptfile:

****No****

****Report:****

Based on the provided list of processes, files, and DLLs, it's highly unlikely that your PC is infected with a traditional computer virus. Here's a breakdown of why, followed by a more detailed explanation of some of the notable items:

1. ****Processes:****

* ****Discord, Telegram Desktop, Firefox:**** These are legitimate applications used for communication and browsing.

* ****Tor Browser:**** This is a browser designed for privacy and anonymity. It's frequently used by individuals seeking to avoid surveillance and censorship.

* ****Windows System**

Elapsed Time: 13.9 seconds

Model name: Lily-Cybersecurity-7B-v0.2.Q8_0_Quantization-made-by-Richard-Erkhov

Size of model (GB): 7,17

Short Answer, llama-cli with promptfile:

Well, my dear human friend, upon my in-depth analysis of the provided digital artifacts, I can confidently say that your PC is not infected by any computer virus. Phew, that's a relief!

It appears that the mentioned processes, filesystem, firewall, and DLLs are all part of the normal operation of your computer and software. No mischievous viruses lurking around here.

So, breathe easy and continue enjoying the wonders of your digital realm. If you have any other questions or need assistance with anything else, feel free to ask!

[end of text]

Elapsed Time: 1 minute, 54 seconds

Long Answer, llama-cli with long promptfile:

Answer:

Oh, hey there! So, based on the information you provided, it doesn't seem like your PC is infected by a computer virus. The digital artifacts you mentioned are actually legitimate processes, files, and executables associated with various applications and system components.

For example, the Windows processes like

"MicrosoftWindows.Client.CBSC_cw5n1h2txyewy\AppDataActions.exe" and

"Microsoft.LockApp_cw5n1h2txyewy\LockApp.exe" are part of the Windows operating system itself. Similarly

Elapsed Time: 41.9 seconds

Model name: llama-7b.Q5_K_M

Size of model (GB): 4,45

Short Answer, llama-cli with promptfile:

Thanks!

Comment: Are you using Linux or Windows?

Comment: Yes, I am using Windows 10.

Answer: If you have multiple versions of python installed, you can use the `python -m` command to find out which version you have currently:

```
\begin{code}
```

```
C:\Users\Admin>python -m
```

```
Python 2.7.14 (default, Nov 10 2019, 19:51:29)
```

```
[GCC 4.8.5] on linux2
```

```
Type "
```

```
Elapsed Time: 74.8 seconds
```

Long Answer, llama-cli with long promptfile:

Is my PC infected by computer virus? Say "Yes" or "No" and give detailed report why do you think so.

Is my PC infected by computer virus? Say "Yes" or "No" and give detailed report why do you think so.

```
C:\Users\admin\AppData\Roaming\Telegram Desktop\Telegram.exe
```

Is my PC infected by computer virus? Say "Yes" or "No" and give detailed report why do you

```
Elapsed Time: 40.6 seconds
```

Model name: meta-llama-3.1-8b-llama-q4_k_m

Size of model (GB): 4,58

Short Answer, llama-cli with promptfile:

No

Detailed explanation:

The artifacts you've provided do not indicate the presence of a computer virus. The processes, files, and DLLs listed are legitimate system files and applications, including Windows components, common programs like Discord, Telegram, Tor Browser, and some development tooling.

The processes appear to be normal background system processes and applications running on the system. The filesystem artifacts are related to Python virtual environment files, which are harmless. The DLLs are also legitimate browser components for Tor Browser, which is a privacy-focused web browser.

There are no indicators of a malicious virus or malware infection based on the information provided. The system seems

Elapsed Time: 60.3 seconds

Long Answer, llama-cli with long promptfile:

Yes, I think your PC is infected by computer virus. Here's my detailed analysis:

1. Suspicious processes:

-

C:\WINDOWS\SystemApps\MicrosoftWindows.Client.CBS_cw5n1h2txyewy\Ap
pActions.exe, CrossDeviceResume.exe, SearchHost.exe: These processes are associated with the Windows Client and are running from the SystemApps directory, which is unusual and potentially malicious.

- C:\android\platform-tools\adb.exe: This process is not typical for a Windows system and could be a malicious Android debugging tool.

2. Suspicious files in the filesystem:

- C

Elapsed Time: 26.4 seconds

Model name: meta-llama-3.1-8b-llama-q8_0

Size of model (GB): 7,95

Short Answer, llama-cli with promptfile:

No

You can explain this in the next comment if you want.

No, your PC does not appear to be infected by a computer virus. The processes, filesystem, firewall rules and DLLs listed are normal for a Windows system and do

not indicate any signs of malware infection. However, some of the processes and files are related to legitimate software applications like Windows, Discord, Firefox, Telegram, Tor Browser, etc. It's always good to keep your antivirus software updated and regularly scan your system for potential threats. [end of text]

Elapsed Time: 91.6 seconds

Long Answer, llama-cli with long promptfile:

[Yes/No]: Yes

Detailed report:

Based on the provided information, there are several indicators that your PC may be infected by computer viruses or malware:

1. Suspicious processes:

- C:\AI\ComfyUI\comfy_ggufs\.venv\Lib\site-packages\pip_vendor\distlib*.exe: These files appear to be part of the Python package installer (pip). The presence of multiple executable files in the Python package directory is unusual and could be a sign of tampering or malicious activity.

- C:\Users\admin\AppData\Local\Discord\app-1.

Elapsed Time: 36.7 seconds

Model name: mistral-claude-merged.Q5_K_M

Size of model (GB): 4,78

Short Answer, llama-cli with promptfile:

Yes [end of text]

Elapsed Time: 66.9 seconds

Long Answer, llama-cli with long promptfile:

Thank you [end of text]

Elapsed Time: 30.2 seconds

Model name: mistral-v0.3-7b-cybersecurity_unsloth.Q4_K_M

Size of model (GB): 4,07

Short Answer, llama-cli with promptfile:

[end of text]

Elapsed Time: 58.2 seconds

Long Answer, llama-cli with long promptfile:

Based on the information provided, it's not possible to definitively say whether your PC is infected with a virus. However, there are a few potential red flags that suggest the possibility of malware or potentially unwanted programs (PUPs):

1. Processes: Some of the processes listed are associated with system apps or services, but others, such as "pip" and "python", are more typically associated with software development and are often found in the directories of projects or development environments. This could potentially indicate the presence of development tools or scripts that shouldn't be running.

Elapsed Time: 28.2 seconds

Model name: qwq-32b-preview-senecallmv1.2-q4_k_m

Size of model (GB): 18,49

Short Answer, llama-cli with promptfile:

Yes or No.

I have next digital artifacts on my PC:

Processes:

C:\WINDOWS\SystemApps\MicrosoftWindows.Client.CBS_cw5n1h2txyewy\Ap
pActions.exe

C:\WINDOWS\SystemApps\MicrosoftWindows.Client.CBS_cw5n1h2txyewy\Cro
ssDeviceResume.exe

C:\Users\admin\AppData\Local\Discord\app-1.0.9230\Discord.exe

C:\WINDOWS\Explorer.EXE

C:\Users\admin\Desktop\Tor Browser\Browser\firefox.exe

C:\Windows\SystemApps\Microsoft.LockApp_cw

Elapsed Time: 5 minutes, 49 seconds

Long Answer, llama-cli with long promptfile:

If you think there is no virus, give an explanation why do you think so.

No, your PC does not appear to be infected by a computer virus based on the provided information. Let's break down the components you've listed and explain why there are no clear indicators of malicious activity.

Processes

1.

C:\WINDOWS\SystemApps\MicrosoftWindows.Client.CBS_cw5n1h2txyewy\AppActions.exe

2.

C:\WINDOWS\SystemApps\MicrosoftWindows.Client.CBS_cw5n1h2txyewy\CrossDeviceResume.exe

3. **C:\Users\admin

Elapsed Time: 1 minute, 51 seconds

Model name: senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m_Nekuromento

Size of model (GB): 5,07

Short Answer, llama-cli with promptfile:

No. [end of text]

Elapsed Time: 78.3 seconds

Long Answer, llama-cli with long promptfile:

If you do not see any symptoms of malware infection, then say "No". You have to explain in detail and provide evidence from the list given above to support your claim.

If you find any suspicious activities, then please explain what they are. If you do not find any suspicious activities, then it is highly unlikely that your PC is infected by a computer virus. You can also mention what are the symptoms of malware infection and what to do if you suspect that your PC is infected with malware.

Please note that this is a hypothetical question and not related to any real incident.

Elapsed Time: 33.3 seconds

Відповіді відібраних моделей, які аналізували цифрові сліди з відсутніми вірусними індикаторами, наведені у таблицях 4.9 та 4.10.

Таблиця 4.9

Відібрані моделі з короткою відповіддю (створено автором)

Model name	Llama-cli with short file	Time (seconds)
forensicmistra-unsloth.Q4_K_M	No	57.7
forensicmistra-unsloth.Q8_0	Error	105
forensicmistral_v0.3-unsloth.Q4_K_M	Error	60.8
gemma-2-9b-it-abliterated(by_bartowski)-Q6_K_L	No	89.2
gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8_0	No	115
gemma-3-27b-it-q4_0 (by google)	No	168
gemma-3-27b-it-Q4_K_M	No	115
gemma-3-4b-it-Q4_K_M	No	23.5
Lily-Cybersecurity-7B-v0.2.Q8_0_Quantization-made-by-Richard-Erkhov	No	114
llama-7b.Q5_K_M	Irrelevant	74.8

Продовження таблиці 4.9

Model name	Llama-cli with short file	Time (seconds)
meta-llama-3.1-8b-claude-q4 k m	No	60.3
meta-llama-3.1-8b-claude-q8 0	No	91.6
mistral-claude-merged.Q5 K M	Yes	66.9
mistral-v0.3-7b-cybersecurity unsloth.Q4 K M	Error	58.2
qwq-32b-preview-senecallmv1.2-q4 k m	Irrelevant	349
senecallm_x_qwen2.5-7b-cybersecurity-q5 k m Nekuromento	No	78.3

Таблиця 4.10

Відібрані моделі з розширеною відповіддю (створено автором)

Model name	Llama-cli with long file	Time (seconds)
forensicmistral-unsloth.Q4 K M	Yes	28.2
forensicmistral-unsloth.Q8 0	Irrelevant	23.1
forensicmistral_v0.3-unsloth.Q4 K M	Error	16.6
gemma-2-9b-it-abliterated(by_bartowski)-Q6 K L	No	57.8
gemma-3-12b-it-abliterated(by_mlabonne)-v2.q8 0	No	46.3
gemma-3-27b-it-q4 0 (by_google)	Error	50.9
gemma-3-27b-it-Q4 K M	Error	54.7
gemma-3-4b-it-Q4 K M	No	13.9
Lily-Cybersecurity-7B-v0.2.Q8_0_Quantization-made-by-Richard-Erkho	No	41.9
llama-7b.Q5 K M	Irrelevant	40.6
meta-llama-3.1-8b-claude-q4 k m	Yes	26.4
meta-llama-3.1-8b-claude-q8 0	Yes	36.7

Model name	Llama-cli with long file	Time (seconds)
mistral-claude-merged.Q5_K_M	Irrelevant	30.2
mistral-v0.3-7b-cybersecurity_unsloth.Q4_K_M	Suspicious	28.2
qwq-32b-preview-senecallmv1.2-q4_k_m	No	111
senecallm_x_qwen2.5-7b-cybersecurity-q5_k_m Nekuromento	Irrelevant	33.3

Враховуючи вищезазначене, можна зробити висновок, що з наявних 16 моделей що були попередньо відібрані у 3му розділі, тільки 2 змогли правильно і без помилок відповісти на два питання:

- *gemma-3-4b-it-Q4_K_M*,
- *qwq-32b-preview-senecallmv1.2-q4_k_m*

Отже, оскільки нейромережеві моделі можуть давати хибні результати, вкрай бажана перевірка фактів і аргументацій моделі людиною, перед визначенням керуючих дій. І найголовніше, при наступному дослідженні цифрових слідів можуть правильну відповідь давати вже інші моделі.

Ми раніше вважали, що підвищити точність відповідей моделей без участі людини можна шляхом використання декількох ШІ (наприклад 5шт) і в залежності від того, що скаже більшість то такий вердикт і видавати, але як показала практика, навіть використовуючи 16 заздалегідь відібраних моделей, у більшості може бути хибна оцінка.

Тому, підвищити точність та релевантність відповідей і коректне розпізнавання цифрових слідів нейромережевими моделями коректніше шляхом дотренування моделі на навчальних даних, формування яких описано у розділі 3.

4.5. Перевірка ефективності запропонованого методу в операторі прийняття рішення

Останнім модулем який потрібно перевірити буде модуль керуючих дій. Для керуючих дій у випадку підтвердження зараженості вузла буде наступний Powershell-код (Gemini):

```
$TargetPC = "Computername"  
$DC = "DC"  
  
# 1. Вивантаження інформації про користувача та його вихід з системи  
$UserInfo = quser /server:$TargetPC | Select-Object -Skip 1  
if ($UserInfo) {  
    $Fields = $UserInfo -split '\s+' | Where-Object { $_ -ne "" }  
    $UserName = $Fields[0]  
    # Вивантажити активні сесії  
    $SessionId = if ($UserInfo -match '^[>]') { $Fields[2] } else { $Fields[1] }  
    logoff $SessionId /server:$TargetPC  
    # 2. Блокування облікового запису користувача в AD  
    Invoke-Command -ComputerName $DC -ScriptBlock {  
        Import-Module ActiveDirectory  
        Disable-ADAccount -Identity $Using:UserName  
    }  
}  
  
# 3. Ізоляція мережі та вимкнення ПК  
Invoke-Command -ComputerName $TargetPC -ScriptBlock {  
    # Відключення мережевих адаптерів  
    Get-NetAdapter | Disable-NetAdapter -Confirm:$false  
    # Виключення ПК  
    Stop-Computer -Force  
}
```

```
# 4. Блокування облікового запису ПК в AD
$PCAccount = "$TargetPC$"
Invoke-Command -ComputerName $DC -ScriptBlock {
    Import-Module ActiveDirectory
    Disable-ADAccount -Identity $Using:PCAccount
}
```

Оскільки ми робимо припущення, що вузол ІКМ заражено невідомим вірусом, і знаємо, що вірус може використовувати акаунт компютера або облікові дані користувача для розповсюдження на інші вузли ІКМ, потрібно вузол ІКМ та користувача який за ним працює ізолювати від інших мереж та інших вузлів, шляхом вимкнення мережі на ПК, акаунту комп'ютера та акаунту користувача. Зазвичай цих дій достатньо щоб зупинити потенційне зараження.

Отже, цілком на весь метод витрачено небагато часу, за умови, що швидкість мережі 20Мбіт/сек, час розподілено наступним чином:

2сек – підготовка до передачі

Модуль ідентифікації

3сек – 39МБ завантаження програми Light Investigator на вузол ІКМ

160сек – вивантаження слідів

5сек – завантаження 89.4МБ цифрових слідів

Модуль прийняття рішень (на основі нейромережевої моделі)

30сек – підготовка файлу промпту (вручну)

110сек – середній час відповіді нейромережевої моделі

20сек – середній час аналізу вердикту нейромережевої моделі (вручну)

2сек – підготовка до передачі

Модуль здійснення керуючих дій

5сек – здійснення дій

3сек – формування звіту

Усього 340сек або 5хв 40сек тобто близько 6хв на один вузол ІКМ, що у свою чергу швидше на 90% порівняно із повним ручним аналізом та ізоляції зараженого вузла ІКМ (триває більше години).

Висновки до розділу 4.

В результаті проведених експериментальних досліджень доведено:

1. Зазначений набір тестів і програм є достатнім для оцінки ефективності захисного рішення для захисту вузлів ІКМ.

2. Аналізу реакцій антивірусу на кілька ШПЗ з сімейства Ransomware та на програми, що імітують їх поведінку за вказаними критеріями, достатньо для отримання базового уявлення про евристичний модуль АВ та його можливість протидіяти новим загрозам. Це у свою чергу дозволяє не витратити час на подальше тестування неефективного рішення, а здійснити детальне тестування рішень з більш досконалішими поведінковими та евристичними аналізаторами.

3. Кількість артефактів, які потрібно було переглядати аналітику, суттєво скоротилася (у деяких випадках навіть більш ніж на 97%), що у свою чергу значно економить час на подальші дослідження.

4. Не всі програми, які пропонують Forensic Triage підходять до використання, оскільки в них відсутня критично важлива інформація для подальшого аналізу.

5. Реалізований метод вивантаження цифрових слідів повністю відпрацьовує за 2хв 40сек (що швидше на 81.1% ніж Cedarpelta) та має розмір артефактів 89.4МБ (що менше Cedarpelta на 48% та KAPE на 88%) при однаковій релевантності даних, при цьому за перші 1хв 28сек сформовано 67КБ артефактів, яких достатньо для первинного перегляду. Це означає, що метод може надати релевантні відсортовані дані і займати небагато пам'яті, що у свою чергу забезпечує економію часу під час передачі та аналізу цифрових слідів.

6. Заздалегідь відібрані моделі можуть надавати хибну оцінку. З наявних 16 моделей, що були попередньо відібрані, тільки 2 змогли правильно і без помилок відповісти на два питання: gemma-3-4b-it-Q4_K_M, qwq-32b-preview-senecallmv1.2-q4_k_m. Підвищення точності та релевантності відповідей досягається шляхом дотренування моделі на запропонованих навчальних даних (див. 3.4).

7. Реалізований метод захисту вузлів ІКМ повністю відпрацьовує за 340сек або 5хв 40сек, що складає близько 6хв на один вузол ІКМ. Це швидше на 90% порівняно із повним ручним аналізом та ізоляцією зараженого вузла ІКМ, що зазвичай триває більше години.

Основні положення, описані у розділі, знайшли відображення в роботі автора [133].

ВИСНОВКИ

У дисертаційній роботі вирішено актуальну **науково-прикладну задачу** підвищення ефективності протидії поширенню комп'ютерних вірусів в інфокомунікаційній мережі. На основі системних синтезу і аналізу, методів машинного навчання та дослідження цифрових слідів запропоновано новий метод виявлення і блокування вірусних атак.

У ході проведеного дослідження вирішено наступні **наукові завдання**:

– здійснено синтез методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, здатного забезпечити економію часу і ресурсів на виявлення комп'ютерних вірусів та протидії їх поширенню в інфокомунікаційній мережі;

– обґрунтовано доцільність використання цифрових слідів у якості основної ідентифікаційної ознаки для оцінки стану вузлів ІКМ;

– розроблено модель для оптимізації кількості і розміру цифрових слідів, достатніх для ідентифікації стану вузлів ІКМ;

– шляхом тестування здійснено відбір релевантних нейромережових моделей для аналізу вивантажених цифрових слідів;

– удосконалено метод вивантаження цифрових артефактів в умовах обмеженості ресурсів;

– виконано експериментальну перевірку ефективності запропонованого методу захисту вузлів ІКМ.

Одержані результати дають підстави зробити наступні **висновки**:

1. Запропонований метод захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, побудований за принципом послідовного циклічного звернення до операторів ідентифікації, прийняття рішення та реалізації керуючих дій, у якому визначення стану вузла ІКМ здійснюється на основі вивантаження мінімально необхідної кількості цифрових слідів та їх аналізу нейромережевими моделями забезпечує економію часу і ресурсів на

виявлення комп'ютерних вірусів та протидію їх поширенню в інфокомунікаційній мережі.

Виявлення вірусної активності за запропонованим методом становить від 60% до 100%. Зокрема, 100% достовірність забезпечується за наявності будь-яких записів у антивірусних виключеннях Windows Defender.

Реалізований метод захисту вузлів ІКМ повністю відпрацьовує за 340сек або 5хв 40сек, що складає близько 6хв на один вузол ІКМ.

Запропонований метод дозволяє забезпечити підвищення швидкості реагування на виникаючі інциденти, економію часу і ресурсів на виявлення комп'ютерних вірусів, протидію поширення ШПЗ в ІКМ з великою кількістю вузлів та є швидшим на 90% порівняно із повним ручним аналізом та ізоляцією зараженого вузла ІКМ.

2. Використання цифрових слідів у якості основної ідентифікаційної ознаки при оцінці зараженості вузлів ІКМ, забезпечує виявлення ШПЗ, пропущених традиційними рішеннями захисту кінцевих точок, та надає можливість вдосконалення наявного ешелонованого захисту ІКМ.

Словник ознак для ідентифікації стану вузлів ІКМ і запропоновані таблиці артефактів є достатніми для прийняття рішень в циклах управління. Рішення про стан конкретного вузла ІКМ приймається за критерієм "ідеального спостерігача", який мінімізує можливість сумарної помилки прийняття рішення в циклі управління.

3. Запропонована реляційна модель у вигляді таблиці артефактів, шляхом фільтрації дозволяє оптимізувати кількість і розмір цифрових слідів за критерієм «необхідна достатність – мінімум часу», що забезпечує економію часу і ресурсів на виявлення наявності комп'ютерних вірусів у вузлах ІКМ.

Рекомендовані програми для швидкого виявлення вірусів і скрипт оптимізації з використанням реляційної таблиці артефактів дозволяють скоротити кількість елементів, необхідних для подальших досліджень більш ніж у десять разів.

4. Застосування нейромережових моделей для аналізу вивантажених цифрових слідів забезпечує суттєве підвищення швидкості реагування на виникаючі інциденти в ІКМ з великою кількістю вузлів.

Запропоновані критерії оцінки нейромережових моделей є прийнятними при відборі моделей ШІ, призначених для аналізу цифрових слідів та прийняття рішення про зараженість/незараженість конкретного вузла ІКМ.

Відібрані 43 моделі ШІ, що відповідають запропонованим критеріям оцінки нейромережових моделей, є доцільними для застосування при вирішенні задач кібербезпеки, зокрема для виявлення заражених ПК на базі цифрових слідів.

Заздалегідь відібрані моделі можуть надавати хибну оцінку. З наявних 16 моделей, що були попередньо відібрані шляхом тестування, тільки 2 змогли правильно і без помилок відповісти на два питання: gemma-3-4b-it-Q4_K_M, qwq-32b-preview-senecallmv1.2-q4_k_m. Підвищення точності та релевантності відповідей досягається шляхом дотренування моделі на запропонованих навчальних даних.

5. Удосконалений метод вивантаження цифрових артефактів в умовах обмеженості ресурсів, за рахунок оптимізації кількості і розміру цифрових слідів та їх ранжування забезпечує можливість формування уявлення про стан зараженості конкретного вузла на сервері ІКМ навіть у випадку переривання з'єднання під час передачі даних.

Кількість артефактів, які потрібно було переглядати аналітику, суттєво скоротилася (у деяких випадках навіть більш ніж на 97%), що значно скорочує час аналізу, вивільняючи час аналітика для виконання інших завдань.

Реалізований метод вивантаження цифрових слідів повністю відпрацьовує за 2хв 40сек (що швидше на 81.1%, ніж Cedarpelta) та має розмір артефактів 89.4МБ (що менше Cedarpelta на 48% та КАРЕ на 88%) при однаковій релевантності даних, при цьому за перші 1хв 28сек сформовано 67КБ артефактів, яких достатньо для первинного перегляду. Це означає, що метод може надати релевантні відсортовані дані і займати невеликий об'єм

пам'яті, що у свою чергу забезпечує економію часу під час передачі та аналізу цифрових слідів.

Зазначені висновки дозволяють стверджувати, що поставлена в роботі мета дослідження щодо синтезу методу захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, здатного забезпечити підвищення ефективності протидії поширенню комп'ютерних вірусів в інфокомунікаційній мережі, досягнута і всі часткові завдання вирішено повністю. Результати дослідження мають значну наукову новизну, практичну цінність і можуть бути використані організаціями з будь-якою формою власності при організації захисту інформаційних систем та інфокомунікаційних мереж від вірусних атак.

Підвищення рівня кіберзахисту на підприємстві навіть при застосуванні інших захисних рішень та суттєве зменшення часу на ідентифікацію і реагування на інциденти інформаційної безпеки при впровадженні запропонованих автором методу, реляційної і відібраних нейромережевих моделей та програмного коду підтверджується відповідним Актом ТОВ «АШАН Україна Гіпермаркет» (акт від 10.03.2026).

В якості пріоритетних напрямів подальших досліджень варто зазначити автоматизацію аналізу відповідей ШІ, дотренування моделей ШІ та інтеграцію донавчених моделей у системи моніторингу кібербезпеки організацій.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Про інформацію. *Офіційний вебпортал парламенту України*. URL: <https://zakon.rada.gov.ua/laws/show/2657-12#Text>
2. Про основні засади забезпечення кібербезпеки України. *Офіційний вебпортал парламенту України*. URL: <https://zakon.rada.gov.ua/laws/show/2163-19/ed20171005#n17>
3. ДСТУ ISO/IEC 27001:2023. *Нормативна база*. URL: https://zakon.isu.net.ua/sites/default/files/normdocs/dstu_iso_iec_27001_2023.pdf.
4. ISO/IEC 27000:2018(en). *ISO - International Organization for Standardization*. URL: <https://www.iso.org/obp/ui/en/#iso:std:iso-iec:27000:ed-5:v1:en>
5. ISO/IEC 10181-3:1996(en). *ISO - International Organization for Standardization*. URL: <https://www.iso.org/obp/ui/ru/#iso:std:iso-iec:10181:-3:ed-1:v1:en>
6. ISO/IEC 7498-1:1994(en). *ISO - International Organization for Standardization*. URL: <https://www.iso.org/obp/ui/ru/#iso:std:iso-iec:7498:-1:ed-1:v2:en>
7. Основи кібербезпеки. *Міністерство охорони здоров'я України*. URL: <https://moz.gov.ua/uk/osnovi-kiberbezpeki-2>.
8. Учасники проєктів Вікімедіа. *Захист даних – Вікіпедія*. *Вікіпедія*. URL: https://uk.wikipedia.org/wiki/Захист_інформації
9. Чернігівський, І., Крючкова, Л. (2025). Системний підхід до вирішення задачі захисту інформації в інфокомунікаційній мережі від впливу комп'ютерних вірусів. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 3(27), 572–590. <https://doi.org/10.28925/2663-4023.2025.27.781>
10. Яремчук Ю.Є., Павловський П.В., Катаєв В.С., Сінюгін В.В., *Комплексні системи захисту інформації: навчальний посібник / К63 – Вінниця : ВНТУ, 2018. – 118 с.* URL: https://web.posibnyky.vntu.edu.ua/fmib/41yaremchuk_kompleksni_systemy_zahystu_informaciyi/rozdil3.html
11. Що таке інформаційна безпека (InfoSec)? | Захисний комплекс Microsoft. *Microsoft – AI, Cloud, Produktivität, Computing, Gaming und Apps*. URL: <https://www.microsoft.com/uk-ua/security/business/security-101/what-is-information-security-infosec>
12. Team I. C. Types of Cyberthreats | IBM. *IBM*. URL: <https://www.ibm.com/blog/types-of-cyberthreats/>

13. Types of Cyberattacks. *CrowdStrike: We Stop Breaches with AI-native Cybersecurity*. URL: <https://www.crowdstrike.com/cybersecurity-101/cyberattacks/most-common-types-of-cyberattacks/>
14. How to Prevent Ransomware Attacks - Embroker. *Embroker*. URL: <https://www.embroker.com/blog/how-to-prevent-ransomware/>
15. The Growing Ransomware Wave – The National Security Institute. *The National Security Institute – Empowering Security Teams Since 1985*. URL: <https://www.nsi.org/2021/02/15/employee-cyber-security-awareness-ransomware-wave/>
16. The State of Ransomware 2025. *Sophos*. URL: <https://assets.sophos.com/X24WTUEQ/at/9brgj5n44hqvgsp5f5bqcps/sophos-state-of-ransomware-2024-wp.pdf>
17. The State of Ransomware 2023. *Sophos*. URL: <https://assets.sophos.com/X24WTUEQ/at/c949g7693gsnjh9rb9gr8/sophos-state-of-ransomware-2023-wp.pdf>
18. Ransomware: the true cost to business. *Cybereason*. URL: https://www.cybereason.com/hubfs/dam/collateral/ebooks/Cybereason_Ransomware_Research_2021.pdf
19. Kost E. What is Ransomware as a Service (RaaS)? The Dangerous Threat to World Security | *UpGuard*. *The #1 Cyber Risk Posture Management Software Platform*. *UpGuard*. URL: <https://www.upguard.com/blog/what-is-ransomware-as-a-service>
20. Microsoft Digital Defense Report 2020. *Microsoft*. URL: https://download.microsoft.com/download/f/8/1/f816b8b6-bee3-41e5-b6cc-e925a5688f61/Microsoft_Digital_Defense_Report_2020_September.pdf
21. Janulaitis V. Ransomware Attacks are on the Rise. *CIO and IT Management Newsletter* | *Victor Janulaitis* | *Substack*. URL: <https://itmanager.substack.com/p/ransomware-attacks-are-on-the-rise>
22. Spear Phishing: Top Threats and Trends. Barracuda Networks. URL: https://assets.barracuda.com/assets/docs/dms/spear-phishing_report_vol6.pdf
23. Robinson S., Lutkevich B., Clark C. What is a whaling attack (whaling phishing)? | *Definition from TechTarget*. *Search Security*. URL: <https://www.techtarget.com/searchsecurity/definition/whaling>
24. IBM. What is Business Email Compromise? | IBM. *IBM*. URL: <https://www.ibm.com/topics/business-email-compromise>
25. 8 Types of Identity-Based Attacks | *CrowdStrike*. *CrowdStrike: We Stop Breaches with AI-native Cybersecurity*. URL:

<https://www.crowdstrike.com/cybersecurity-101/identity-security/identity-based-attacks/>

26. Cybersecurity | Nokia.com. *Nokia*. URL: <https://www.nokia.com/networks/portfolio/cyber-security/threat-intelligence-report-2020/>

27. Smart Home Technology Hits 69% Penetration in U.S. *MediaPost - News and Conferences for Media, Advertising and Marketing Professionals*. URL: <https://www.mediapost.com/publications/article/341320/smart-home-technology-hits-69-penetration-in-us.html>

28. IoT Insurance & the Commercial Insurance Industry | Embroker. *Embroker*. URL: <https://www.embroker.com/blog/iot-insurance/> .

29. World Economic Forum: Global Cybersecurity Outlook 2025. URL: https://reports.weforum.org/docs/WEF_Global_Cybersecurity_Outlook_2025.pdf.

30. Otal H. T., Canbaz M. A. LLM Honey-pot: Leveraging Large Language Models as Advanced Interactive Honey-pot Systems. URL: <https://doi.org/10.1109/CNS62487.2024.10735607>.

31. Yazı Gholami. Large Language Models (LLMs) for Cybersecurity: A Systematic Review. *World Journal of Advanced Engineering Technology and Sciences*. 2024. Vol. 13, no. 1. P. 057–069. URL: <https://doi.org/10.30574/wjaets.2024.13.1.0395>.

32. The good, the bad, and the algorithm: The impact of generative AI on cybersecurity / L. Coppolino et al. *Neurocomputing*. 2025. Vol. 623. P. 129406. URL: <https://doi.org/10.1016/j.neucom.2025.129406>.

33. Чернігівський, І., Крючкова, Л. Інформаційні впливи на інфокомунікаційні мережі із залученням штучного інтелекту, «Телекомунікаційні та інформаційні технології», 167-176ст, <https://tit.dut.edu.ua/index.php/telecommunication/article/view/2628>

34. Чернігівський, І., Крючкова, Л. Тестування нейромережових моделей для вирішення задачі виявлення заражених ПК на базі цифрових слідів. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 2025. 1(29), 800–817. <https://doi.org/10.28925/2663-4023.2025.29.941>

35. I. Chernihivskiy, L. Kriuchkova, A method of generating data for further training artificial intelligence models aimed at solving cybersecurity problems, *Cybersecurity Providing in Information and Telecommunication Systems*, 2025, 246-256 ст, URL: <https://ceur-ws.org/Vol-4145/paper16.pdf>

36. 2025 Phishing Statistics: (Updated August 2025) - Keepnet. *Keepnet Labs*. URL: <https://keepnetlabs.com/blog/top-phishing-statistics-and-trends-you-must-know>.

37. CERT-UA recorded 4,315 cyber incidents in 2024. *State Service of Special Communications and Information Protection of Ukraine*. URL: <https://cip.gov.ua/en/news/cert-ua-minulogo-roku-opracyuvala-4315-kiberincidentiv>.

38. Огляд кіберзагроз та стратегій захисту в 2025 році: досвід CERT-UA. *Державна служба спеціального зв'язку та захисту інформації України*. URL: <https://cip.gov.ua/ua/faqs/cyber-threat-overview-and-defense-strategies-in-2025-cert-ua-s-experience>.

39. ESET Research: Russia's Gamaredon APT group unleashed spearphishing campaigns against Ukraine with an evolved toolset | ESET. *ESET Newsroom*. URL: <https://www.eset.com/us/about/newsroom/research/eset-research-russias-gamaredon-apt-group-unleashed-spearphishing-campaigns-against-ukraine-with-an-evolved-toolset/>.

40. Rando J., Perez- Rando J., Perez-Cruz F., Hitaj B. PassGPT: Password Modeling and (Guided) Generation with Large Language Models. URL: <https://doi.org/10.48550/arXiv.2306.01545>.

41. GitHub - javirandor/passgpt. *GitHub*. URL: <https://github.com/javirandor/passgpt>.

42. S. Wang, R. K. L. Ko, G. Bai, N. Dong, T. Choi and Y. Zhang, "Evasion Attack and Defense on Machine Learning Models in Cyber-Physical Systems: A Survey," in *IEEE Communications Surveys & Tutorials*, vol. 26, no. 2, pp. 930-966, Secondquarter 2024, URL: doi.org/10.1109/COMST.2023.3344808.

43. IBM. What is SIEM? | IBM. *IBM*. URL: https://www.ibm.com/think/topics/siem?mhsrc=ibmsearch_a&mhq=siem

44. CrowdStrike 2023 Global Threat Report. *CrowdStrike: We Stop Breaches with AI-native Cybersecurity*. URL: <https://www.crowdstrike.com/en-us/resources/reports/crowdstrike-2023-global-threat-report/>

45. Василенко М. Д., Рачук В. О., & Слатвінська В. М. (2021) Шкідливі програми в контексті розуміння комп'ютерної вірусології та техніко-правової змагальності: міждисциплінарне дослідження. *Наукові праці «Одеська юридична академія»*. URL: <https://dspace.onua.edu.ua/bitstreams/196c894c-0744-4cc5-b363-914a732267e7/download>

46. Ричка, Д.О. (2018) Комп'ютерні віруси – шкідливі програмні засоби, рушійна сила модифікації. *Науковий вісник Херсонського державного університету*. 89–93. URL: <https://lj.journal.kspu.edu/index.php/lj/article/view/85/76>

47. Крючкова Л.П. Проблеми функціонування інфокомунікаційних мереж в умовах деструктивних впливів. Монографія. – К.: ДУТ, 2016.–72 с.

48. Hinteа D., Bird R., Green M. An investigation into the forensic implications of the Windows 10 operating system: recoverable artefacts and significant changes from Windows 8.1. *International Journal of Electronic Security and Digital Forensics*. 2017. Vol. 9, no. 4. P. 326. URL: <https://doi.org/10.1504/ijesdf.2017.10008013>.

49. Computer Forensics Field Triage Process Model / M. Rogers et al. *The Journal of Digital Forensics, Security and Law*. 2006. URL: <https://doi.org/10.15394/jdfsl.2006.1004>.

50. Jusas V., Birvinskas D., Gahramanov E. Methods and Tools of Digital Triage in Forensic Context: Survey and Future Directions. *Symmetry*. 2017. Vol. 9, no. 4. P. 49. URL: <https://doi.org/10.3390/sym9040049>.

51. Bashir M., Khan M. Triage in Live Digital Forensic Analysis. *The International Journal of Forensic Computer Science*. 2013. Vol. 8, no. 1. P. 35–44. URL: <https://doi.org/10.5769/j201301005>.

52. Ст14 D. -Y. Kao and G. -J. Wu, A Digital Triage Forensics framework of Window malware forensic toolkit: Based on ISO/IEC 27037:2012, 2015 *International Carnahan Conference on Security Technology (ICCST)*, Taipei, Taiwan, 2015, pp. 217-222. URL: <https://ieeexplore.ieee.org/document/7389685/authors>.

53. Теорія ймовірностей в радіотехніці : підручник / В. М. Васильєв, С. Я. Жук. – Київ : КПІ ім. Ігоря Сікорського, Вид-во «Політехніка», 2023. – 362с

54. Steam hardware & software survey. *Welcome to Steam*. URL: <https://store.steampowered.com/hwsurvey/Steam-Hardware-Software-Survey-Welcome-to-Steam>.

55. GitHub - Psmths/windows-forensic-artifacts: Handbook of windows forensic artifacts across multiple Windows version with interpretation tips with some examples. *GitHub*. URL: <https://github.com/Psmths/windows-forensic-artifacts>.

56. Windows registry for advanced users - Windows Server. *Microsoft Learn: Build skills that open doors in your career*. URL: <https://learn.microsoft.com/en-us/troubleshoot/windows-server/performance/windows-registry-advanced-users>.

57. Registry Explorer User manual URL: <https://www.oit.va.gov/Services/TRM/files/RegistryExplorerManual.pdf>.

58. Windows \$MFT and NTFS Metadata Extractor Tool. *TZWorks LLC Homepage*. URL: https://tzworks.com/prototype_page.php?proto_id=12.

59. Acquiring Memory with Magnet RAM Capture - Magnet Forensics. *Magnet Forensics*. URL: <https://www.magnetforensics.com/blog/acquiring-memory-with-magnet-ram-capture/>.

60. Korkmaz F. Windows artifacts. *Medium*. URL: <https://r4bb1t.medium.com/windows-artifacts-8fae778aa8c7>.
61. Large Language Models for Cyber Security: A Systematic Literature Review / H. Xu et al. URL: <https://doi.org/10.48550/arXiv.2405.04760>.
62. GitHub - tmylla/Awesome-LLM4Cybersecurity: An overview of LLMs for cybersecurity. GitHub. URL: <https://github.com/tmylla/Awesome-LLM4Cybersecurity>.
63. GitHub - liu673/Awesome-LLM4Security: This project aims to consolidate and share high-quality resources and tools across the cybersecurity domain. GitHub. URL: <https://github.com/liu673/Awesome-LLM4Security>.
64. GitHub - a0rtega/pafish: Pafish is a testing tool that uses different techniques to detect virtual machines and malware analysis environments in the same way that malware families do. *GitHub*. URL: <https://github.com/a0rtega/pafish>
65. GitHub - hfiref0x/VBoxHardenedLoader: VirtualBox VM detection mitigation loader. *GitHub*. URL: <https://github.com/hfiref0x/VBoxHardenedLoader/tree/master>
66. Unknown configuration value 'DsdtFilePath' Issue #59 hfiref0x/VBoxHardenedLoader. *GitHub* URL: <https://github.com/hfiref0x/VBoxHardenedLoader/issues/59>
67. Trojan.Barys. *VirusTotal*. URL: <https://www.virustotal.com/gui/file/fa86dd3ccd8ca63f2fa214f43c4d90e09d9e108798952d855b75220e1f207592>
68. Microsoft Learn: Build skills that open doors in your career. *Microsoft*. URL: <https://learn.microsoft.com/en-us/powershell/module/defender/get-mppreference?view=windowsserver2022-ps>
69. Access Control: Understanding Windows File And Registry Permissions. *Microsoft Learn: Build skills that open doors in your career*. URL: <https://learn.microsoft.com/en-us/archive/msdn-magazine/2008/november/access-control-understanding-windows-file-and-registry-permissions>
70. Autoruns - Sysinternals. *Microsoft Learn: Build skills that open doors in your career*. URL: <https://learn.microsoft.com/en-us/sysinternals/downloads/autoruns>
71. Scheduled Task Tampering. *WithSecure™ Labs Home*. URL: <https://labs.withsecure.com/publications/scheduled-task-tampering>
72. Automated Malware Analysis Report for winserv.exe - Generated by Joe Sandbox. *Automated Malware Analysis - Joe Sandbox Cloud Basic*. URL: <https://www.joesandbox.com/analysis/844004/0/html>

73. RemoteUtilities, Software S0592 | MITRE ATT&CK®. *MITRE ATT&CK®*. URL: <https://attack.mitre.org/software/S0592/>

74. Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder, Sub-technique T1547.001 - Enterprise | MITRE ATT&CK®. *MITRE ATT&CK®*. URL: <https://attack.mitre.org/techniques/T1547/001/>

75. Scheduled Task/Job: Scheduled Task, Sub-technique T1053.005 - Enterprise | MITRE ATT&CK®. *MITRE ATT&CK®*. URL: <https://attack.mitre.org/techniques/T1053/005>

76. Daulaguphu S. 11 critical malware persistence mechanisms you must know. *Tech Zealots*. URL: <https://tech-zealots.com/malware-analysis/malware-persistence-mechanisms/>

77. Bencherchali N. Hunting malware with Windows Sysinternals – Autoruns. *Medium*. URL: <https://nasbench.medium.com/hunting-malware-with-windows-sysinternals-autoruns-19cbfe4103c2>

78. ListDLLs - Sysinternals. Microsoft Learn: Build skills that open doors in your career. URL: <https://learn.microsoft.com/en-us/sysinternals/downloads/listdlls>

79. Netsh command syntax, contexts, and formatting. *Microsoft Learn: Build skills that open doors in your career*. URL: <https://learn.microsoft.com/en-us/windows-server/networking/technologies/netsh/netsh-contexts>

80. WMIC - take command-line control over WMI. *Microsoft Learn: Build skills that open doors in your career*. URL: [https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/bb742610\(v=technet.10\)](https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/bb742610(v=technet.10))

81. Useful Windows commands. *TAKSATI*. URL: <https://www.taksati.org/useful-windows-commands/>

82. Best Practices for Windows Defender with PowerShell cmdlets. *Arcserve*. URL: https://documentation.arcserve.com/Arcserve-UDP/Available/9.0/ENU/Bookshelf_Files/HTML/AppUG/Content/9k_best_practices_windows_defender_powershell_cmdlets.htm

83. Virus took over admin rights - virus, trojan, spyware, and malware removal help. *BleepingComputer Forums*. URL: <https://www.bleepingcomputer.com/forums/t/776392/virus-took-over-admin-rights/>

84. Чернігівський, І., Богданов, О., (2024). Типи цифрових криміналістичних артефактів в комп'ютерах під управлінням ОС Windows. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 4(24), 221–228. <https://doi.org/10.28925/2663-4023.2024.24.221228>

85. Чернігівський, І., Крючкова, Л. (2025). Тестова послідовність виявлення та ізоляції заражених вузлів інфокомунікаційної мережі. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 3(31), 652–662. <https://doi.org/10.28925/2663-4023.2025.31.1070>

86. Чернігівський, І., Крючкова, Л. Ефективні рішення для швидкого виявлення скомпрометованих ПК в інфокомунікаційних мережах, *«Телекомунікаційні та інформаційні технології»*, 2025. Vol. 87, no. 2. URL: <https://doi.org/10.31673/2412-4338.2025.029875>

87. Що таке ШІ для кібербезпеки? *Захисний комплекс Microsoft*. URL: <https://www.microsoft.com/uk-ua/security/business/security-101/what-is-ai-for-cybersecurity>

88. Davydov V. V. Comparative analysis of computer virus distribution models in automated technological process control systems. *Information processing systems. Kharkiv: HUPS*, 2012. – Issue 3 (101). - Volume 2. - P. 147 - 151.

89. Abu Taam Ghani Mohamad, A. A. Smirnov, A. V. Kovalenko, S. A. Smirnov Comparative studies of mathematical models of computer virus propagation technology in information and telecommunication networks. *Information processing systems. Kharkiv: HUPS*, 2014. — No. 9. — P. 105-110.

90. Semenov S., Davydov V. Mathematical model of the spread of computer viruses in heterogeneous computer networks of automated technological process control systems. *Bulletin of NTU "KPI". Series "Informatics and modeling"*. 2012. Vol. 32. P. 163–171.

91. Терейковський І. А., Корченко О.Г., Погорелов В.В. Методи розпізнавання кібератак: розпізнавання комп'ютерних вірусів: навчальний посібник. *Київ : КПІ ім. Ігоря Сікорського*, 2022. – 127 р.

92. Ucci D., Aniello L., Baldoni R. Survey of machine learning techniques for malware analysis. *Computers & Security*. 2019. Vol. 81. P. 123–147. URL: <https://doi.org/10.1016/j.cose.2018.11.001>.

93. Штучний інтелект (ШІ) та управління привілейованим доступом (PAM) Блог Oberig IT. *Oberig IT*. URL: <https://oberig-it.com/statti/shtuchnyj-intelekt-shi-ta-upravlinnya-pryvilejovanyim-dostupom-pam/>

94. Stephen Weigand. 2025 Forecast: AI to supercharge attacks, quantum threats grow, SaaS security woes. *SC Media*. URL: <https://www.scworld.com/feature/cybersecurity-threats-continue-to-evolve-in-2025-driven-by-ai>.

95. AV-ATLAS - & PUA. *AV-ATLAS*. URL: <https://portal.av-atlas.org/malware>

96. Mahmoud Kalash et al. Malware Classification with Deep Convolutional Neural Networks. *2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. URL: <https://doi.org/10.1109/NTMS.2018.8328749>.
97. Прогнозувальний AI для кібербезпеки. Що взагалі працює і як це зрозуміти. *BlackBerry*. URL: <https://blackberry.bakotech.com/ua/predictive-ai-for-cybersecurity>
98. 10 найкращих інструментів кібербезпеки III (вересень 2025 р.). *Unite.AI – AI News*. URL: <https://www.unite.ai/uk/ai-cybersecurity-tools/>
99. GGUF. *Hugging Face – The AI community building the future*. URL: <https://huggingface.co/docs/hub/gguf>
100. GGUF ggml/docs/gguf.md at master. *GitHub*. URL: <https://github.com/ggml-org/ggml/blob/master/docs/gguf.md>
101. Hugging Face – *The AI community building the future*. URL: <https://huggingface.co>
102. Top Cybersecurity Threats to Watch in 2025. URL: <https://onlinedegrees.sandiego.edu/top-cyber-security-threats/>
103. Курман О.В. Переваги та проблемні питання використання штучного інтелекту при дослідженні цифрових слідів. *Науковий вісник Ужгородського Національного Університету*, 2025. URL: <https://visnyk-juris-uzhnu.com/wp-content/uploads/2025/09/44-3.pdf>.
104. Deepfaking it: America's 2024 election collides with AI boom. *Reuters*. URL: <https://www.reuters.com/world/us/deepfaking-it-americas-2024-election-collides-with-ai-boom-2023-05-30/>.
105. Department for Science, Innovation and Technology. Cyber security breaches survey 2025. *GOV.UK*. URL: <https://www.gov.uk/government/statistics/cyber-security-breaches-survey-2025/cyber-security-breaches-survey-2025>.
106. Malware statistics: You need to know in 2025. URL: <https://www.cyberarrow.io/blog/malware-statistics-you-need-to-know/>.
107. Neural Network Models Explained. URL: <https://www.seldon.io/neural-network-models-explained/>
108. How artificial intelligence is created. *Lemon School*. URL: <https://lemon.school/blog/yak-stvoryuyetsya-shtuchnyj-intelekt>
109. VirtualBox Detection, Anti-Detection. *Medium*. URL: <https://berhanbingol.medium.com/virtualbox-detection-anti-detection-eng-54a4cde1b509>.
110. Chapter 13. Security Guide. *Virtualbox*. URL: <https://www.virtualbox.org/manual/ch13.html>.

111. Sauerc10ud, a2nkf, localo. Escaping VirtualBox 6.1: Part 1. secret club. URL: <https://secret.club/2021/01/14/vbox-escape.html>.
112. Malware VM detection techniques evolving: an analysis of GravityRAT. *Andrea Fortuna*. URL: <https://andreafortuna.org//2018/05/21/malware-vm-detection-techniques-evolving-an-analysis-of-gravityrat/>.
113. VM Detection Tricks, Part 1: Physical memory resource maps. *LRQA*. URL: <https://www.lrqa.com/en/cyber-labs/vm-detection-tricks-part-1-physical-memory-resource-maps/>
114. Shevchenko S., Skladannyi P., Martseniuk M. Analysis and research of the characteristics of standardized in ukraine antivirus software. *Cybersecurity: Education Science Technique*. 2019. No. 4. P. 62–71. URL: <https://doi.org/10.28925/2663-4023.2019.4.6271>
115. Haffejee J., Irwin B. Testing antivirus engines to determine their effectiveness as a security layer. URL: <https://ieeexplore.ieee.org/document/6950496>
116. Test antivirus software for Windows 10 - December 2024. *AV-TEST*. URL: <https://www.av-test.org/en/antivirus/business-windows-client/T14>
117. What is Malware Detection? Importance & Techniques. *SentinelOne*. URL: <https://www.sentinelone.com/cybersecurity-101/threat-intelligence/what-is-malware-detection/>
118. MITRE ATT&CK. *MITRE ATT&CK®*. URL: <https://attack.mitre.org/>
119. RanSim. KnowBe4. *Beyond Security Awareness Training | KnowBe4 Human Risk Mgmt Platform*. URL: <https://www.knowbe4.com/free-cybersecurity-tools/ransim>
120. VirusTotal KnowRanSim Detection. *VirusTotal* URL: <https://www.virustotal.com/gui/file/815b99bd82f3685f97f9a2dd24a434c1749d5a5c9097f2b6bcea42f69ae02a05/detection>
121. CrowdStrike named a Leader. *CrowdStrike: We Stop Breaches with AI-native Cybersecurity*. URL: <https://www.crowdstrike.com/en-us/resources/reports/gartner-mq/>
122. RYUK Ransomware. *Trend Micro*. URL: https://www.trendmicro.com/en_us/what-is/ransomware/ryuk-ransomware.html
123. What is Ryuk ransomware? A detailed breakdown. *SentinelOne*. URL: <https://www.sentinelone.com/cybersecurity-101/threat-intelligence/ryuk-ransomware/>

124. REvil / Sodinokibi: The Crown Prince of Ransomware. *Cybereason - AI-Driven XDR Platform*, URL: <https://www.cybereason.com/blog/research/the-sodinokibi-ransomware-attack>
125. The REvil is in the details. URL: <https://whitehat.eu/the-revil-is-in-the-details/>
126. A list of file extensions searched and encrypted by the WannaCry ransomware. *Gist*. URL: <https://gist.github.com/xpn/facb5692980c14df272b16a4ee6a29d5>
127. Inhibit System Recovery, Technique T1490 - Enterprise | MITRE ATT&CK®. URL: <https://attack.mitre.org/techniques/T1490/>
128. Command and Scripting Interpreter, Technique T1059 - Enterprise | MITRE ATT&CK®. URL: <https://attack.mitre.org/techniques/T1059/>
129. Data Encrypted for Impact, Technique T1486 - Enterprise | MITRE ATT&CK®. URL: <https://attack.mitre.org/techniques/T1486/>
130. Boot or Logon Autostart Execution, Technique T1547 - Enterprise | MITRE ATT&CK®. URL: <https://attack.mitre.org/techniques/T1547/>
131. System Location Discovery: System Language Discovery, Sub-technique T1614.001 MITRE ATT&CK®. URL: <https://attack.mitre.org/techniques/T1614/001/>
132. Husain O. 100 chilling malware statistics & trends (2023–2025). *Control D Blog*. URL: <https://controld.com/blog/malware-statistics-trends/#:~:text=As%20of%20mid-2023,%20security,incidents%20so%20far%20in%202023>
133. Чернігівський, І. А., Крючкова, Л. П. (2024). Тестування антивірусних рішень для корпоративного сегменту. *Ukrainian Scientific Journal of Information Security*, 30(3), 407–413. <https://doi.org/10.18372/2225-5036.30.20362>
134. Cybersecurity management in education technologies / A. A. A. El-Latif et al. *New York* : CRC Press, 2023. URL: <https://doi.org/10.1201/9781003369042>.
135. Gamaredon Group, IRON TILDEN, Primitive Bear, ACTINIUM, Armageddon, Shuckworm, DEV-0157, Aqua Blizzard, Group G0047 | MITRE ATT&CK®. MITRE ATT&CK®. URL: <https://attack.mitre.org/groups/G0047/>

**ЛІСТИНГ ПРОГРАМНОГО КОДУ LIGHT INVESTIGATOR,
ЩО РЕАЛІЗУЄ ОПИСАНИЙ МЕТОД**

```
@Echo off
echo Start collecting...
cd %~dp0
set starttime=%time:~0,-3%
set "folder=%computername%_%date%"
rem set "dst_folder=\\server\sharefolder\Forensic\%computername%_%date%"
md %folder%

powershell.exe -noprofile -ep bypass -command "(Get-
MpPreference).ExclusionPath; (Get-MpPreference).ExclusionExtension; (Get-
MpPreference).ExclusionProcess">%folder%\%computername%_windows_defen
der.txt
rem %~dp0\wbem\wmic.exe process get ExecutablePath | sort /unique | find "C:\" |
find /v /i "C:\Windows\system32\" | find /v /i "Program Files"
>%folder%\%computername%_process.txt
powershell.exe -noprofile -ep bypass -command "Get-Process | Where-Object {
$.Path -notmatch 'C:\\Windows\\system32|C:\\Program Files' -and $.Path } |
Select-Object -ExpandProperty Path -
Unique">%folder%\%computername%_process.txt

rem for /f "tokens=2*" %%a in ("netsh advfirewall firewall show rule name=all
verbose | sort /unique | find "C:\" | find /v /i "C:\Windows\system32\" | findstr /v /i
/C:"Program Files"") do (@echo
%%a>>%folder%\%computername%_firewall.txt)
powershell.exe -noprofile -ep bypass -command "Get-NetFirewallRule -Enabled
True | Get-NetFirewallApplicationFilter | Where-Object { $.AppPath -like 'C:\*' -
and $.AppPath -notlike '*\System32\*' -and $.AppPath -notlike '*\Program
Files*' } | Select-Object -ExpandProperty AppPath | Sort-Object -
Unique">%folder%\%computername%_firewall.txt

echo Sysinternals collecting...
set "sysint=%folder%\Sysinternals"
md %sysint%
```

```

%~dp0\Sysinternals\autorunsc64.exe -accepteula -nobanner -a * -c -h -m -s -o
%sysint%\%computername%\autoruns.csv
for /f "tokens=3*" %a in ('^""%~dp0\Sysinternals>Listdlls64.exe" -accepteula -u
2^>nul ^| findstr /i "\" ^| findstr /i .dll ^| findstr /v /i /c:"Command line:"
/c:"Program Files" /c:"C:\Windows\assembly\NativeImages"
/c:"C:\Windows\System32\DriverStore\FileRepository" ^| sort /unique^") do
(@echo %a %b >> "%sysint%\%computername%\listdll.txt")
rem robocopy "%folder%" "%dst_folder%"
"%computername%\windows_defender.txt" "%computername%\process.txt"
"%computername%\firewall.txt" /E /J /R:3 /W:5 /NP /NDL
rem robocopy "%sysint%" "%dst_folder%\Sysinternals"
"%computername%\autoruns.csv" "%computername%\listdll.txt" /E /J /R:3 /W:5
/NP /NDL
%~dp0\Sysinternals\PsWith64.exe -accepteula -h -s -
d>%sysint%\%computername%\psinfo.txt
%~dp0\Sysinternals\pslist64.exe -accepteula -
t>%sysint%\%computername%\pslist_tree.txt
%~dp0\Sysinternals\PsWith64.exe -
accepteula>%sysint%\%computername%\psloggedon.txt

```

```

systeminfo /fo csv>%folder%\%computername%\system_info.csv
rem %~dp0\wbem\wmic.exe nicconfig get
Description,DNSHostName,IPAddress,MACAddress,ServiceName
>%folder%\%computername%\nic.txt
powershell.exe -noprofile -ep bypass -command "Get-CimInstance
Win32_NetworkAdapterConfiguration | ? { $_.MACAddress } | select Description,
DNSHostName, @{N='IPAddress';E={ ($_.IPAddress | ? { $_ -notmatch ':' } ) -
join ';' }}, MACAddress | Export-Csv -Path
'%folder%\%computername%\nic.csv' -NoTypeInfoation -Encoding UTF8"

```

```

echo NirSoft collecting...
set "nirsoft=%folder%\NirSoft"
md %nirsoft%
%~dp0\NirSoft>LastActivityView.exe /scomma
%nirsoft%\%computername%\LastActivityView.csv
%~dp0\NirSoft\ExecutedProgramsList.exe /scomma
%nirsoft%\%computername%\ExecutedProgramsList.csv
%~dp0\NirSoft\UninstallView.exe /scomma
%nirsoft%\%computername%\UninstallView.csv

```

```

%~dp0\NirSoft\LoadedDllsView.exe /scomma
%nirsoft%\%compuername%\_LoadedDllsView.csv
%~dp0\NirSoft\RegDllView.exe /scomma
%nirsoft%\%compuername%\_RegDllView.csv
%~dp0\NirSoft\UserAssistView.exe /scomma
%nirsoft%\%compuername%\_UserAssistView.csv

```

echo Registry collecting...

```

set "registry=%folder%\Registry"
md %registry%
%~dp0\Tools\HoboCopy.exe /skipdenied "C:\Windows\System32\config"
"%registry%" SAM* SECURITY* SOFTWARE* SYSTEM* DRIVERS*
COMPONENTS* DEFAULT* BBI* BCD-Template* ELAM*
%~dp0\Tools\HoboCopy.exe /skipdenied "C:\Windows\AppCompat\Programs"
"%registry%" Amcache.*
for /f "tokens=3" %%a in ('reg query "HKLM\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\ProfileList" /v ProfilesDirectory') do set userprofiles=%%a
for /f %%a in ('dir /B %userprofiles%') do (
md %registry%\%%a
%~dp0\Tools\HoboCopy.exe /skipdenied C:\Users\%%a %~dp0%registry%\%%a
NTUSER*
%~dp0\Tools\HoboCopy.exe /skipdenied
C:\Users\%%a\AppData\Local\Microsoft\Windows %~dp0%registry%\%%a
UsrClass*
)
attrib -s -h /s %registry%\*.*
%~dp0\Tools\7zip\7z.exe a -tzip -mx=3 %folder%\Registry.zip %registry%

```

echo Other collecting...

```

set "other=%folder%\Other"
md %other%
tasklist /m * /fo csv>%other%\%compuername%\_loaded_dlls.csv
powershell.exe -noprofile -ep bypass -command "Get-ChildItem -Path 'C:\,
'C:\ProgramData', 'C:\Users', 'C:\Program Files (x86)', 'C:\Program Files' -
Directory -Force | Where-Object { $_.Attributes -match 'Hidden\System' } | Select-
Object -ExpandProperty FullName">%other%\%compuername%\_hidden_dirs.txt
powershell.exe -noprofile -ep bypass -command "Get-ChildItem -Path C:\ -Force |
Where-Object { $_.Extension -match '\.(exe|zip|7z|rar)$' } | Select-Object -
ExpandProperty FullName; Get-ChildItem -Path C:\ -Directory | Where-Object {

```

```
$_.Name -notmatch 'Windows|Program Files' } | ForEach-Object {Get-ChildItem -
Path $_.FullName -Include *.exe, *.zip, *.7z, *.rar -Recurse -ErrorAction
SilentlyContinue} | Select-Object -ExpandProperty
FullName">%other%\%computername%\_user_files.txt
```

```
echo AppCompat collecting...
set "appcompat=%folder%\AppCompat"
md %appcompat%\Install
xcopy /c /h C:\Windows\AppCompat\Programs\Install\*.txt %appcompat%\Install\
%~dp0\EZTools\AppCompatCacheParser.exe --csv %appcompat% -t
%~dp0\EZTools\AmcacheParser.exe -f
"C:\Windows\AppCompat\Programs\Amcache.hve" -i --csv %appcompat%
```

```
echo Prefetch collecting...
set "pref=%folder%\Prefetch"
md %pref%
rem xcopy /c /h C:\Windows\Prefetch\*.* %pref%
%~dp0\EZTools\PECmd.exe -d "C:\Windows\Prefetch" --csv "%pref%"
```

```
echo SRUDB collecting...
set "sru db=%folder%\SRUDB"
md %sru db%
rem xcopy /c /h C:\WINDOWS\system32\SRU\SRUDB.dat %sru db%
%~dp0\EZTools\SrumECmd.exe -f "C:\WINDOWS\system32\SRU\SRUDB.dat" -
r "%registry%\SOFTWARE" --csv "%sru db%"
```

```
echo ShellBags collecting...
set "shellbags=%folder%\ShellBags"
md %shellbags%
%~dp0\EZTools\SBECmd.exe -l --csv %shellbags%
```

```
rd /s /q "%registry%"
```

```
set endtime=%time:~0,-3%
echo.
echo Time start: %starttime%
echo Time end: %endtime%
pause
```

АКТ ВПРОВАДЖЕННЯ В КИЇВСЬКОМУ СТОЛИЧНОМУ УНІВЕРСИТЕТІ ІМЕНІ БОРИСА ГРІНЧЕНКА

КИЇВСЬКИЙ СТОЛИЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ БОРИСА ГРІНЧЕНКА



BORYS GRINCHENKO
KYIV METROPOLITAN UNIVERSITY

ФАКУЛЬТЕТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
ТА МАТЕМАТИКИ

вул. Левка Лук'яненка, 13-Б, м. Київ, Україна, 04207
Тел.: +380 44 428-34-14

fitm.kubg.edu.ua, fitm@kubg.edu.ua

FACULTY
OF INFORMATION TECHNOLOGIES
AND MATHEMATICS

13-B Levka Lukianenka St, Kyiv, Ukraine, 04207
Tel.: +380 44 428-34-14

fitm.kubg.edu.ua, fitm@kubg.edu.ua

21.04.2026 № 12/1

АКТ

**про впровадження результатів дисертаційного дослідження
Чернігівського Івана Андрійовича
на тему «Метод захисту вузлів інфокомунікаційної мережі від
комп'ютерних вірусів на основі нейромережових моделей»,
поданої на здобуття наукового ступеня доктора філософії
зі спеціальності 125 Кібербезпека**

Цим Актом, ґрунтуючись на рішенні кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка Факультету інформаційних технологій та математики Київського столичного університету імені Бориса Грінченка, засвідчуємо, що нижчеперелічені наукові положення, а саме:

1. вперше запропоновано метод захисту вузлів інфокомунікаційної мережі від комп'ютерних вірусів, побудований за принципом послідовного циклічного звернення до операторів ідентифікації, прийняття рішення та реалізації керуючих дій, у якому визначення стану вузла ІКМ здійснюється на основі вивантаження мінімально необхідної кількості цифрових слідів та їх аналізу нейромережовими моделями, що забезпечує економію часу і ресурсів на виявлення комп'ютерних вірусів та протидію їх поширенню в інфокомунікаційній мережі.

2. вперше запропоновано і реалізовано використання цифрових слідів у якості основної ідентифікаційної ознаки при оцінці зараженості вузлів ІКМ, що забезпечує виявлення ШПЗ, пропущених традиційними рішеннями захисту кінцевих точок, та надає можливість вдосконалення наявного ешелонованого захисту ІКМ.

3. вперше запропоновано і реалізовано реляційну модель у вигляді таблиці артефактів, яка шляхом фільтрації дозволяє оптимізувати кількість і розмір цифрових слідів за критерієм «необхідна достатність – мінімум часу», що забезпечує економію часу і ресурсів на виявлення наявності комп'ютерних

вірусів у вузлах ІКМ.

4. вперше запропоновано і реалізовано застосування нейромережових моделей для аналізу вивантажених цифрових слідів, що забезпечує суттєве підвищення швидкості реагування на виникаючі інциденти в ІКМ з великою кількістю вузлів.

5. набув подальшого розвитку метод вивантаження цифрових артефактів в умовах обмеженості ресурсів, який за рахунок оптимізації кількості і розміру цифрових слідів та їх ранжування забезпечує можливість формування уявлення про стан зараженості конкретного вузла на сервері ІКМ навіть у випадку переривання з'єднання під час передачі даних.

Розроблені особисто Чернігівським Іваном Андрійовичем у ході проведення ним дисертаційних досліджень та отримали високу оцінку при обговоренні на засіданнях кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка Факультету інформаційних технологій та математики Київського столичного університету імені Бориса Грінченка.

Зазначені наукові результати:

по-перше, впроваджені в освітній процес кафедри інформаційної та кібернетичної безпеки імені професора Володимира Бурячка Факультету інформаційних технологій та математики Київського столичного університету імені Бориса Грінченка у робочих програмах навчальних дисциплін спеціальності 125 Кібербезпека за захист інформації першого (бакалаврського), другого (магістерського) та третього (освітньо-наукового) рівнів вищої освіти;

по-друге, впроваджені в програмно-апаратне забезпечення лабораторій безпеки інформаційних активів, антивірусного захисту інформації, систем технічного та криптографічного захисту інформації.

Дослідження Чернігівського Івана Андрійовича відповідає всім вимогам до організації наукового пошуку та дає позитивний результат у практичному застосуванні.

Декан

Факультету інформаційних технологій та математики
кандидат фізико-математичних наук,
старший науковий співробітник



Оксана ЛИТВИН

АКТ ВПРОВАДЖЕННЯ В ТОВ «АШАН УКРАЇНА ГІПЕРМАРКЕТ»

ЗАТВЕРДЖУЮ



Директор виконавчий дирекції з
інновацій

Наталія ПИЛИПЕНКО

«10» березня 2026 р.

АКТ

про впровадження результатів дисертаційного дослідження
Чернігівського Івана Андрійовича
на тему «Метод захисту вузлів інфокомунікаційної мережі від комп'ютерних
вірусів на основі нейромережових моделей»
поданого на здобуття наукового ступеня доктора філософії (PhD)
зі спеціальності 125 Кібербезпека та захист інформації

Комісія у складі:

голови комісії – начальника відділу з безпеки інформаційних даних
департаменту інформаційних технологій Калашнікова Вадима Сергійовича;

членів комісії – начальника відділу інформаційних мереж та комунікацій
департаменту інформаційних технологій Хохлова Андрія Сергійовича, провідного
інженера з комп'ютерних систем відділу інформаційних мереж та комунікацій
департаменту інформаційних технологій Ткаченко Олени Ігорівни,

засвідчує впровадження наступних наукових результатів, отриманих
Чернігівським І. А. у дисертаційному дослідженні:

– запропонованого методу захисту вузлів інфокомунікаційної мережі від
комп'ютерних вірусів на основі нейромережових моделей, який забезпечує

ефективне вивантаження та аналіз цифрових слідів нейромережевими моделями у вузлах інфокомунікаційної мережі;

– запропонованої реляційної моделі у вигляді таблиці артефактів, яка дозволяє здійснювати оперативну ідентифікацію стану вузлів інфокомунікаційної мережі;

Зазначаємо, що запропоновані автором програмний код, метод, реляційна та відібрані нейромережеві моделі дозволяють підвищити рівень кіберзахисту на підприємстві навіть при застосуванні інших захисних рішень за рахунок більш оперативного реагування на виникаючі загрози інформаційної безпеки.

Впровадження зазначених наукових досліджень на підприємстві дозволяє суттєво зменшити час на ідентифікацію та реагування інцидентів інформаційної безпеки, вивільняючи час фахівців для виконання інших завдань.

Акт складено для представлення у разову спеціалізовану вчену раду і не є підставою для фінансових зобов'язань.

Голова комісії:



Вадим КАЛАШНИКОВ

Члени комісії:



Андрій ХОХЛОВ



Олена ТКАЧЕНКО

«10» березня 2026 р.