УДК 004.056.57 DOI 10.31673/2412-4338.2025.029875

Чернігівський Іван Андрійович
Київський столичний університет імені Бориса Грінченка, м. Київ, Україна
ORCID ID 0009-0003-4568-3212

Крючкова Лариса Петрівна
Київський столичний університет імені Бориса Грінченка, м. Київ, Україна
ORCID ID 0000-0002-8509-6659

ЕФЕКТИВНІ РІШЕННЯ ДЛЯ ШВИДКОГО ВИЯВЛЕННЯ СКОМПРОМЕТОВАНИХ ПК В ІНФОКОМУНІКАЦІЙНИХ МЕРЕЖАХ

Анотація. *Останнім часом все більш актуальною є необхідність вирішення завдань в умовах обмежених часових ресурсів. Це може бути наприклад мережева атака на корпоративні ресурси компанії, внаслідок якої невідома кількість ПК була скомпрометована, при повному ігноруванні з боку антивірусного захисту та системою запобігання вторгнень, що в свою чергу накладає значне обмеження за часом, оскільки потрібно швидко встановити «ступінь зараженості» кожного окремого ПК і ізолювати його від інших комп'ютерів інфокомунікаційної мережі. В такому випадку традиційний forensic-аналіз цифрових слідів, зібраний за допомогою Forensic Triage буде занадто довгим. Коли в звичайних умовах ці завдання виконували відомі програми і цього часу було достатньо, зараз постає питання, як пришвидшити "виконання" завдань, якщо конкурентних аналогів для програми не має? Тому виникає необхідність в додаткових можливостях, які б зменшили час виконання програми для вивантаження цифрових артефактів при збереженні достатньої ефективності. Або ж зменшили час роботи аналітика ІБ на аналіз одного ПК, що дасть йому змогу перевірити більше ПК за одиницю часу. Метою дослідження є обґрунтування ефективних рішень для зменшення часу аналітика інформаційної безпеки на виявлення зараженості конкретного ПК інфокомунікаційної мережі у якості «заражено/не заражено». 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. Це допомагає ІТ-аналітикам істотно заощадити час на виявлення зараженості конкретного ПК інфокомунікаційної мережі у якості «заражений/не заражений».*

Ключові слова: *інформація; кібербезпека; форензіка; цифрові сліди; Windows; MITRE; автозапуск; віруси.*

Chernihivskyi Ivan
Borys Grinchenko Kyiv Metropolitan University, Kyiv, Ukraine
ORCID ID 0009-0003-4568-3212

Kriuchkova Larysa
Borys Grinchenko Kyiv Metropolitan University, Kyiv, Ukraine
ORCID ID 0000-0002-8509-6659

EFFECTIVE SOLUTIONS FOR RAPID DETECTION OF COMMITTED PCS IN THE INFOCOMMUNICATION NETWORKS

Abstract. *Recently, the need to solve problems in conditions of limited time resources has become increasingly relevant. This could be, for example, a network attack on a company's corporate resources, as a result of which an unknown number of PCs have been compromised, while being completely ignored by AV and IPS, which in turn imposes a significant time constraint, since it is necessary to quickly establish the "degree of infection" of each individual PC and isolate it from other computers in the infocommunication*

network. In this case, the traditional forensic analysis of digital traces collected using Forensic Triage will be too long. When under normal conditions, these tasks were performed by well-known programs and this time was enough, now the question arises, how to speed up the "execution" of tasks if there are no competitive analogues for the program? Therefore, there is a need for additional capabilities that would reduce the program for extracting digital artifacts execution time while maintaining sufficient efficiency. Or reduce the time an IT analyst spends analyzing one PC, which will allow him to check more PCs per unit of time. The purpose of the study is to substantiate effective solutions to reduce the time an information security analyst spends on identifying a specific PC in the infocommunication network as infected/not infected. The work identifies a component/tactic without which modern computer viruses usually do not work. A list of programs for rapid virus detection and an optimization script using a relational table of artifacts are proposed, which allow reducing the number of elements required for further research by more than ten times. This helps IT analysts significantly save time on detecting the infection of a specific PC in the infocommunication network as "infected/not infected".

Keywords: *information; cybersecurity, computer forensics, computer digital artifacts, Windows, MITRE, autostart, viruses*

1. Постановка проблеми. Сьогодні, коли постійно зростає кількість кібератак на корпоративні ПК 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.

2. Аналіз останніх досліджень і публікацій. У працях науковців Diana Hintea, Robert Bird, Michael Green [1] Marcus K. Rogers, James Goldman, Rick Mislan, Timothy Wedge, Steve Debrota [2] та Jusas V., Birvinskas D., Gahramanov E. [3] Duc Tran Le, Truong Duy Dinh, Phuoc Hoang Tan Nguyen, Ammar Muthanna, Ahmed A. Abd El-Latif [4] пропонується перелік артефактів для Windows, загальні місця їх розташування в операційній системі (ОС), а також місця для автозапуску. Проте у деяких статтях відсутня інформація про конкретні місця розташування артефактів у ОС Windows, не враховуються ситуації, у яких неможливий збір повного спектру артефактів, а також не пропонується оптимізованих рішень для швидкого збору необхідних даних, що пришвидшують їх аналіз.

3. Мета і задачі дослідження

Метою дослідження є обґрунтування ефективних рішень для зменшення часу аналітика інформаційної безпеки на виявлення зараженості конкретного ПК інфокомунікаційної мережі у якості «заражено/не заражено».

Для досягнення поставленої мети необхідно вирішити такі 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 артефактів та скрипт оптимізації, які дозволяють скоротити кількість елементів, необхідних для подальших досліджень.

4. Результати дослідження

Типові проблеми інформаційно-комунікаційної мережі під час вивантаження артефактів розглянуто нами в [5]:

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

- різний час роботи співробітників, що не дозволяє аналітику інформаційної безпеки (ІБ) достовірно спрогнозувати наявність/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, 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, що перешкоджає отриманню інформації про стан ПК;

- залежність від користувачів, які вимикають свій ПК замість виходу з акаунту після переривання своєї роботи, 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 [5].

Для оптимізації наявних рішень 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.

Згідно з даними MITRE ATT&CK Matrix визначено, що 230 відомих вірусів (і APT, 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 [6,7].

Тож не має значення, який тип вірусу сховався на конкретному ПК: троян, вимагач, RAT або мережевий хробак, та які механізми приховування від користувача вони реалізовують, 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 [8], для автоматичного збору цифрових слідів ми будемо використовувати відомі програми, що дозволить знизити кількість помилок і зекономити час, такі як Autoruns [9,10].

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 [11].

Додатково, потрібно вивантажити інформацію про виключення Windows Defender & Windows Firewall, щоб отримати інформацію про ще невідомі вірусні компоненти (або їх місцерозташування) 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, згідно з таблицею Артефактів для ідентифікації стану ІКМ [5].

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 нашому випадку не потрібна [10,11,12,13]:

```
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.1. Консольна версія утиліти Autoruns

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, BHO, Winlogon, службах автозапуску та багато іншого (див. рис. 1).

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).

```
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. Утиліта ListDLLs

Netsh.exe – утиліта, яка дає можливість відобразити або змінити конфігурацію мережі віддаленого та локального комп'ютера за допомогою команд netsh (див. рис. 3).

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

Рис.3. Утиліта Netsh

PowerShell.exe – інструмент командного рядка, який запускає сеанс Windows PowerShell у вікні командного рядка (CLI), частина довідки показана на рис. 4.

```
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.4. 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 5.

```
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.5. Утиліта WMIC

Для вирішення згаданої вище проблеми неналежної оптимізації деяких програм розроблено і представлено цифрові сліди у вигляді реляційної таблиці для можливості більш ефективно ідентифікувати стан ПК (див. таблиці 1-3).

Таблиця 1

Таблиця мінімально необхідних артефактів [5]

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

Характеристики програм, якими можна знехтувати

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 непідписані бібліотеки	Підписані бібліотеки будуть легітимними тож їх можна не враховувати (наприклад, user32.dll)
Hide Microsoft entries	Для відкидання верифікованих легітимних програм

Таблиця 3

Реляційна таблиця артефактів

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	Hide Microsoft entries	No	No	No	Yes	No

Для реалізації представленої моделі автором розроблено скрипт .bat на основі наукових досліджень, наведених у літературі [14,15,16]. Скрипт необхідно запускати від імені адміністратора на досліджуваному комп'ютері. Сам код скрипта, який оптимізує вихід програм, наведено нижче:

```
@Echo off
cd %~dp0
echo %time:~0,-3%
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
echo %time:~0,-3%
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. 6).

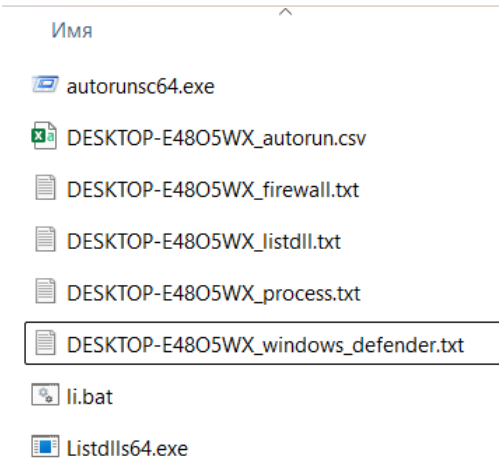

Рис.6. Результат роботи скрипта

Для визначення ефективності оптимізації програми розроблений скрипт, протестований 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 4.

Таблиця 4

Результат виконання програм у строках

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

Згідно з даними, наведеними в таблиці, бачимо, 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.

5. Висновки та перспективи подальших досліджень

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 інфокомунікаційної мережі:

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.

Список використаної літератури

1. Hintea 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 (date of access: 04.04.2025).

2. 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 (date of access: 04.04.2025).

3. 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 (date of access: 04.04.2025).

4. Cybersecurity management in education technologies / A. A. A. El-Latif et al. New York : CRC Press, 2023. URL: https://doi.org/10.1201/9781003369042 (date of access: 04.04.2025).

5. Chernihivskyi , I., Kriuchkova, L. (2025). Systemnyi pidkhid do vyrishennia zadachi zakhystu informatsii v infokomunikatsiinii merezhi vid vplyvu kompiuternykh virusiv. Elektronne fakhove naukove vydannia «Kiberbezpeka: osvita, nauka, tekhnika», 3(27), 572–590. https://doi.org/10.28925/2663-4023.2025.27.781 (date of access: 04.04.2025).

6. 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/ (date of access: 04.04.2025).

7. Scheduled Task/Job: Scheduled Task, Sub-technique T1053.005 - Enterprise | MITRE ATT&CK®. MITRE ATT&CK®. URL: https://attack.mitre.org/techniques/T1053/005 (date of access: 04.04.2025).

8. Daulaguphu S. 11 critical malware persistence mechanisms you must know. Tech Zealots. URL: https://tech-zealots.com/malware-analysis/malware-persistence-mechanisms/ (date of access: 04.04.2025).

9. Bencherchali N. Hunting malware with Windows Sysinternals – Autoruns. Medium. URL: https://nasbench.medium.com/hunting-malware-with-windows-sysinternals-autoruns-19cbfe4103c2 (date of access: 04.04.2025).

10. Autoruns - Sysinternals. Microsoft Learn: Build skills that open doors in your career. URL: https://learn.microsoft.com/en-us/sysinternals/downloads/autoruns (date of access: 04.04.2025).

11. ListDLLs - Sysinternals. Microsoft Learn: Build skills that open doors in your career. URL: https://learn.microsoft.com/en-us/sysinternals/downloads/listdlls (date of access: 04.04.2025).

12. 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 (date of access: 04.04.2025).

13. 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) (date of access: 04.04.2025).

14. Useful Windows commands. TAKSATI. URL: https://www.taksati.org/useful-windows-commands/ (date of access: 04.04.2025).

15. Best Practices for Windows Defender with PowerShell cmdlets. URL: https://documentation.arcserve.com/Arcserve-UDP/Available/9.0/ENU/Bookshelf_Files/HTML/ApplUG/Content/9k_best_practices_windows_defender_powershell_cmdlets.htm (date of access: 04.04.2025).

16. 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/ (date of access: 04.04.2025).