КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 4 (28), 2025
ISSN 2663 - 4023
DOI 10.28925/2663-4023.2025.28.814
УДК 004.49

Терещенко Катерина Володимирівна
студентка
Державний університет "Київський авіаційний інститут" , Київ, Україна
ORCID ID: 0009-0008-8469-9854
katerina60411@gmail.com

Черниш Єлизавета Сергіївна
учениця
Ліцей № 124 "Спаський" Київ, Україна
ORCID ID: 0009-0000-6972-6027
ycool2680@gmail.com

Терещенко Тетяна Павлівна
старша наукова співробітниця
Військовий інститут телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна
ORCID ID: 0000-0002-9659-7897
tetiana.tereshchenko@viti.edu.ua

Черниш Юлія Олександрівна
старша наукова співробітниця
Військовий інститут телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна
ORCID ID: 0000-0002-6626-5656
yuliia.chernysch@viti.edu.ua

Самусь Олена Юріївна
наукова співробітниця
Військовий інститут телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна
ORCID ID: 0009-0000-7906-547X
OlenkaSamys357@gmail.com

АНАЛІЗ ПРОГРАМНОГО КОМПЛЕКСУ КРИПТОГРАФІЧНОГО ЗАХИСТУ ІНФОРМАЦІЇ

Анотація. Під обробкою інформації в системі розуміють виконання однієї або кількох операцій, зокрема: збирання, введення, записування, перетворення, зчитування, зберігання, знищення, реєстрації, приймання, отримання, передавання, які здійснюються в системі за допомогою технічних і програмних засобів. У цілому захисту потребує інформація з обмеженим доступом (критична інформація), а саме: конфіденційна, службова або таємна інформація (наприклад, державна таємниця, банківська таємниця або інша таємниця). Під час обробки інформації з обмеженим доступом повинен забезпечуватися її захист від несанкціонованого та неконтрольованого ознайомлення, модифікації, знищення, копіювання, поширення, тобто забезпечення основних її властивостей захищеності — конфіденційності, цілісності, доступності та спостереженості. Зокрема, передача службової й таємної інформації з однієї системи до іншої здійснюється в зашифрованому вигляді або захищеними каналами зв'язку за допомогою технічного та криптографічного захисту інформації. Інформація, що обробляється, є об'єктом захисту. Сучасні криптографічні системи забезпечують шифрування даних з метою приховування їх змісту, підтвердження справжності, цілісності та авторства. Симетричні та асиметричні методи шифрування лежать в основі побудови систем захисту даних і гарантують їх безпечне зберігання та передавання. Проводячи аналіз, можемо отримати знання й уміння, для успішної реалізації шифрування даних, налаштування політики безпеки, створення надійних механізмів захисту інформації в автоматизованих системах. Задля досягнення цього, необхідно здійснити впровадження програмного комплексу криптографічного захисту, який має забезпечити конфіденційність, цілісність і доступність даних. У процесі буде проведена всебічна перевірка функціональних можливостей обраного програмного забезпечення з криптографічного

захисту. Отримані в межах цього завдання знання й уміння дозволяють упевнено реалізовувати шифрування даних, налаштовувати політики безпеки, створювати надійні механізми захисту інформації в автоматизованих системах.

Ключові слова: криптографічний захист; програмне забезпечення; шифрування; VeraCrypt.

ВСТУП

На сьогодні первинним чинником, що впливає на політичну й економічну складові національної безпеки, є ступінь захищеності інформації й інформаційного середовища. Ось чому важливого значення набувають питання забезпечення захисту інформації в автоматизованих (інформаційних) і телекомунікаційних системах під час обробки у різних сферах діяльності [1]. Під обробкою інформації в системі розуміють виконання однієї або кількох операцій, зокрема: збирання, введення, записування, перетворення, зчитування, зберігання, знищення, реєстрації, приймання, отримання, передавання, які здійснюються в системі за допомогою технічних і програмних засобів. Такі програмні засоби (або програмне забезпечення), як і сама інформація, що обробляється, є об'єктами захисту. У цілому захисту потребує інформація з обмеженим доступом (критична інформація), а саме: конфіденційна, службова або таємна інформація (наприклад, державна таємниця, банківська таємниця або інша таємниця). Під час обробки інформації з обмеженим доступом повинен забезпечуватися її захист від несанкціонованого та неконтрольованого ознайомлення, модифікації, знищення, копіювання, поширення, тобто забезпечення основних її властивостей захищеності — конфіденційності, цілісності, доступності та спостереженості. Зокрема, передача службової й таємної інформації з однієї системи до іншої здійснюється в зашифрованому вигляді або захищеними каналами зв'язку за допомогою технічного та криптографічного захисту інформації. Одним із ключових методів захисту є криптографічний захист, який реалізується за допомогою програмних, апаратних і програмно-апаратних засобів. Сучасні криптографічні системи забезпечують шифрування даних з метою приховування їх змісту, підтвердження справжності, цілісності та авторства. Симетричні та асиметричні методи шифрування лежать в основі побудови систем захисту даних і гарантують їх безпечне зберігання та передавання [2].

Постановка проблеми. Інформаційна система, в якій здійснюється обробка, зберігання та передача даних, потребують захисту від несанкціонованого доступу, спотворення чи знищення. Тому предметом дослідження є: процес впровадження та функціонування програмного комплексу криптографічного захисту інформації, а також перевірка його ефективності в умовах експериментального полігону.

Аналіз останніх досліджень і публікацій. Актуальність нашого дослідження обумовлена стрімким розвитком інформаційних технологій та зростаючими вимогами до забезпечення надійного захисту інформації. У сучасному світі симетричні криптосистеми залишаються основними засобами захисту даних у різних сферах: електронній комерції, банківських транзакціях, захищених комунікаціях та зберіганні конфіденційної інформації. Сформульовано багато практичних рекомендацій для підвищення рівня безпеки при роботі з критичними даними, щодо використання каскадного шифрування з включенням щонайменше двох алгоритмів; максимального використання функціоналу ключових файлів, комбінацій їх з довгими паролями та високим PIM [2] — [5].

Проводячи аналіз, можемо отримати знання й уміння, для успішної реалізації шифрування даних, налаштування політики безпеки, створення надійних механізмів захисту інформації в автоматизованих системах.

Мета статті. Метою роботи є: забезпечення високого рівня захисту інформації, яка обробляється в межах інформаційної системи. Задля досягнення цього, необхідно

здійснити впровадження програмного комплексу криптографічного захисту, який має забезпечити конфіденційність, цілісність і доступність даних. У процесі буде проведена всебічна перевірка функціональних можливостей обраного програмного забезпечення з криптографічного захисту.

РЕЗУЛЬТАТ ДОСЛІДЖЕННЯ

Аналіз програмного комплексу криптографічного захисту інформації

У рамках цього етапу розглядається застосування утиліти VeraCrypt — сучасного інструменту для забезпечення криптографічного захисту даних. Програма реалізує функції симетричного шифрування, дозволяє створювати зашифровані томи й контейнери, підтримує багаторівневий захист, а також надає засоби для перевірки автентичності доступу. VeraCrypt також забезпечує управління ключами й паролями, підтримує роботу з різними алгоритмами шифрування (AES, Serpent, Twofish тощо) та сумісна з кількома операційними системами, зокрема Windows і Linux. Серед основних функціональних можливостей VeraCrypt варто відзначити здатність здійснювати як шифрування системного диска, так і створення окремих зашифрованих томів, у тому числі на зовнішніх накопичувачах. Користувач має можливість створити як звичайні, так і приховані томи, що дозволяє захищати особливо чутливу інформацію, застосовуючи концепцію заперечуваності доступу. У технічному плані VeraCrypt підтримує низку сучасних алгоритмів шифрування, зокрема AES, Serpent, Twofish, а також можливість комбінування цих алгоритмів для посилення криптостійкості. VeraCrypt використовує потужний і сучасний підхід до шифрування даних, реалізуючи режим XTS (XEX-based Tweaked CodeBook mode with ciphertext stealing). Цей метод, затверджений міжнародним стандартом NIST-IEEE P1619, розроблений спеціально для шифрування секторів на дисках і забезпечує високий рівень безпеки та ефективність обробки.

$$C = E_{K1}\left(P_j \text{^} (E_{K2}(n) \otimes a^i)\right) \wedge (E_{K2}(n) \otimes a^i)$$

Основна суть режиму XTS полягає в тому, що кожен блок відкритого тексту (P) перетворюється на шифротекст (C) за допомогою криптографічного ключа (K), що складається з двох незалежних частин: K1 — основного ключа шифрування (256 біт для кожного підтримуваного алгоритму) та K2 — вторинного ключа (також 256 біт). В процесі шифрування використовуються такі математичні операції, як побітова операція АБО (XOR) та модульне множення двох многочленів над бінарним полем Галуа по модулю $x128 + x7 + x2 + x + 1$, $GF(2^{128})$ з використанням спеціального примітивного елемента a, який відповідає поліному X (тобто 2). Це дозволяє створити унікальний шифротекст для кожного блоку, навіть якщо вхідні дані повторюються.

Розмір блоку даних завжди становить 512 байт, що є стандартним для секторів диска. VeraCrypt забезпечує конвеєрну (асинхронну) обробку даних: у той час, коли одна частина файлу читається або записується, паралельно виконується шифрування чи дешифрування іншої частини. Такий підхід дозволяє досягти високої швидкості роботи, співставної з незашифрованими носіями, і суттєво покращує користувацький досвід.

Ще однією важливою складовою шифрувальної архітектури VeraCrypt є вбудований генератор випадкових чисел (RNG), який використовується для створення головного і вторинного ключів, а також для генерації ключових файлів. RNG формує спеціальний басейн (pool) довжиною 320 байт, який наповнюється ентропією з кількох незалежних джерел:

- рухів миші;
- натискань клавіш;

– Windows CryptoAPI (із частотою 500 мс);
– мережевої статистики через NETAPI32;
– системних дескрипторів, часових змінних і лічильників Win32.

Усі отримані значення розбиваються на байти та додаються до басейну за допомогою операції додавання по модулю 28, без заміни попередніх значень. Після кожного запису курсор пересувається, а після кожних 16 записаних байтів автоматично виконується функція змішування, яка покращує рівень ентропії в басейні.

Такий комплексний підхід до шифрування, генерації ключів та обробки даних робить VeraCrypt одним з найнадійніших інструментів для захисту конфіденційної інформації на сучасному етапі.

Проте, як і будь-яке програмне забезпечення, VeraCrypt має як сильні сторони, так і обмеження, що варто враховувати при використанні в реальних умовах.

Серед основних переваг VeraCrypt слід відзначити відкритий вихідний код. Це забезпечує прозорість розробки, дозволяючи будь-кому аналізувати програму на наявність вразливостей, що підвищує довіру до її надійності. Крім того, користувачі з достатнім технічним рівнем можуть адаптувати код до власних потреб, розширюючи або змінюючи функціональність.

Ще однією важливою перевагою є підтримка широкого набору алгоритмів шифрування. Користувач може обрати найоптимальніший варіант або навіть поєднати декілька алгоритмів, що підвищує стійкість шифрування до криптоаналізу й ускладнює несанкціонований доступ до зашифрованих даних.

Окремої уваги заслуговують функції створення прихованих томів і операційних систем, які реалізують концепцію «відмови, що не заперечується». Це дозволяє зберігати найчутливішу інформацію в умовах, коли користувач може бути змушений надати пароль до системи. Навіть у разі розкриття основного тому, прихований залишиться невидимим та захищеним.

Універсальність VeraCrypt також виражається у підтримці різних операційних систем — Windows, macOS та Linux. Це робить програму придатною як для окремих користувачів, так і для організацій, що працюють у гетерогенному середовищі, забезпечуючи єдиний підхід до захисту інформації.

Втім, не обійшлося без недоліків. Основна складність полягає в необхідності достатнього рівня технічної обізнаності. Налаштування VeraCrypt може виявитися складним для новачків, а неправильне конфігурування здатне призвести до втрати даних або порушення безпеки.

Ще одним критичним аспектом є відсутність офіційної технічної підтримки, адже проєкт розвивається на добровільних засадах. Користувачі змушені звертатися до спільнот або форумів, а ризик припинення підтримки чи оновлення залишається.

Таким чином, VeraCrypt — це потужний і надійний інструмент для захисту даних, який ідеально підходить для користувачів із глибоким розумінням IT-безпеки. Його можливості значно перевищують середній рівень, але ефективне використання потребує часу, зусиль і технічної підготовки. Для тих, хто готовий до цього, VeraCrypt стане надійним захисником конфіденційної інформації [3].

Таблиця 1

Порівняльна характеристика програмного забезпечення, що реалізує функції шифрування

Критерій	VeraCrypt	Cryptic Disk	Secret Disk	Zecurion ZDisk
Операційна система	Windows, macOS, Linux	Windows	Windows	Windows
Ліцензія	Безкоштовна, з відкритим кодом (Apache License 2.0)	Комерційна	Комерційна	Комерційна

Підтримувані криптоалгоритми	AES, Serpent, Twofish, їх комбінації	AES, Serpent, Twofish, Blowfish	Відсутні	Відсутні
Можливість каскадного шифрування	Twofish-Serpent, Serpent-AES, Serpent-Twofish-AES тощо	AES, Twofish-Serpent, Blowfish-Serpent	Ні	Ні
Шифрування системного розділу	Так	Так	Так	Ні
Шифрування логічного розділу	Так	Так	Так	Ні
Створення прихованих томів/дисків	Так	Ні	Ні	Ні
Підтримка ключових файлів	Так	Так	Ні	Ні
Підтримка апаратного прискорення (AES-NI)	Так	Так	Ні	Ні
Автоматичне підключення при старті ОС	За бажанням користувача	Так	Так	Так
Можливість роботи з зовнішніми носіями	Так	Так	Обмежено	Ні
Можливість аварійного відключення/самознищення	Частково (при використанні скритих томів)	Так	Ні	Ні
Підтримка командного рядка	Так	Так	Ні	Ні
Можливість інтеграції в корпоративне середовище	Обмежено (через скрипти)	Так	Ні	Так
Оновлення та підтримка	Активне спільнотне оновлення	Регулярні комерційні оновлення	Періодичні оновлення	Комерційна підтримка

Створення тестового середовища

Під час дослідження функціональних можливостей програмного забезпечення VeraCrypt, було прийнято рішення застосувати віртуалізоване середовище. Такий підхід забезпечує ізоляцію дослідницької платформи від основної операційної системи, дозволяючи проводити експерименти без ризику впливу на стабільність основного середовища. У рамках даного дослідження в якості гіпервайзера другого типу було обрано VMware Workstation 17 Pro, а для операційної системи тестового середовища — Windows 10.

VMware Workstation 17 Pro є безкоштовним і потужним програмним забезпеченням для віртуалізації, що дозволяє створювати та керувати віртуальними машинами з різними операційними системами. Завдяки VMware можливо тонко регулювати апаратні ресурси віртуальної машини, підключати додаткові пристрої, знімати снапшоти поточного стану системи та швидко повертатись до стабільної конфігурації, що особливо корисно під час проведення експериментів та тестувань, які можуть призвести до непередбачуваних результатів.

Гостьова операційна система Windows 10 була обрана через свою сумісність із широким спектром програмного забезпечення, а також через офіційну підтримку VeraCrypt, що забезпечує стабільну роботу та повний доступ до всіх функцій утиліти.

Проводимо налаштування середовища для випробувань, забезпечивши можливість перевірки всіх основних функцій програмного комплексу — шифрування/дешифрування, перевірка автентичності та цілісності інформації, а також управління ключами.

ISSN 2663 - 4023

Створюємо завантажувальний носій з операційною системою, який надалі буде використано для встановлення середовища у віртуальній машині. Послідовність дій і сам процес відображено на рис. 1.

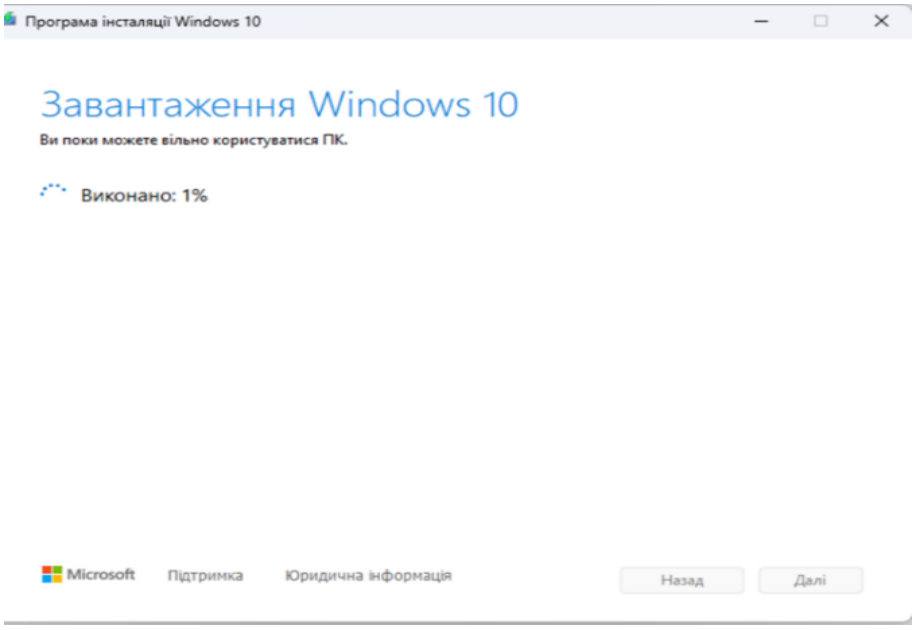

Рис. 1. Створення завантажувального носія

Після успішного створення завантажувального носія з операційною системою, підключаємо до віртуальної машини як джерело інсталяції. На рис. 2–4 детально відображено всі етапи налаштування параметрів віртуальної машини, необхідні для коректного підключення та запуску інсталяційного процесу.

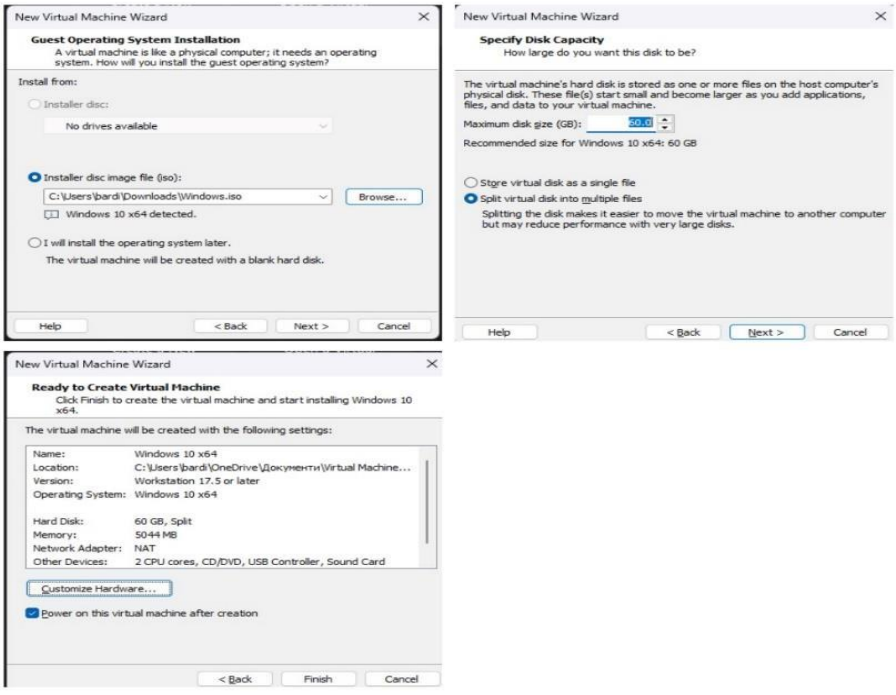

Рис. 2–4. Налаштування віртуальної машини у VMware Workstation 17 Pro

Основною метою тестування є оцінка ефективності виконання програмою процесів шифрування та дешифрування даних різного формату, а також перевірка коректності роботи механізмів автентифікації користувача. Дослідження передбачає створення умов, наближених до реального середовища, зокрема використання операційної системи Windows 10, яка є гостьовою у віртуальному середовищі, створеному за допомогою VMware Workstation 17 Pro.

Для тестування були підібрані файли різних типів: текстові документи, графічні зображення, а також архіви, що відображає типові об'єкти захисту у повсякденному користуванні. Обсяги тестових файлів варіювались від 5 МБ до 500 МБ, що дозволило оцінити поведінку програми як на малих, так і на великих об'ємах даних. Очікуваними результатами є відповідність програми заявленим характеристикам — зокрема, стабільна робота шифрувальних алгоритмів, відсутність втрати даних після дешифрування, а також достатній рівень продуктивності при роботі з великими обсягами інформації.

Методика проведення тестування включала використання вбудованих інструментів VeraCrypt для вимірювання швидкості шифрування та дешифрування даних з різними параметрами. Крім того, були створені криптоконтейнери з використанням різних алгоритмів шифрування (AES, Serpent, Twofish та їх комбінації), що дало змогу зіставити їхню ефективність у практичному використанні. Також здійснювались контрольовані спроби несанкціонованого доступу до зашифрованих даних, що дозволило перевірити надійність системи автентифікації та захисту ключів.

Для фіксації та подальшого аналізу результатів застосовувалися статистичні методи обробки інформації, серед яких — розрахунок середнього часу шифрування та дешифрування залежно від розміру файлів і типу алгоритму. Отримані значення дозволили зробити обґрунтовані висновки щодо оптимального використання Verarypt в умовах підвищених вимог до захисту інформації.

Проведення тестування

Основну увагу зосереджено на аналізі ефективності алгоритмів шифрування, оцінці механізмів автентифікації, а також дослідженні стійкості системи до потенційних загроз безпеці.

Першим етапом тестування став порівняльний аналіз швидкості обробки даних. Було використано файл обсягом 100 МБ, до якого послідовно застосовувалися різні алгоритми шифрування, зокрема AES, Twofish, Serpent, а також комбіновані варіанти, як-от AES–Twofish і AES–Twofish–Serpent. Згідно з отриманими результатами, найвищу швидкість шифрування та дешифрування продемонстрував алгоритм AES. Водночас комбіновані алгоритми, хоч і потребують більше часу на обробку, забезпечують посилений рівень захисту, що є суттєвим фактором для сценаріїв з високими вимогами до безпеки.

У межах наступного етапу було здійснено перевірку працездатності кожного із підтримуваних VeraCrypt алгоритмів. Усі вони успішно впоралися із завданням шифрування та подальшого відновлення даних без жодних втрат чи спотворень інформації. Надійність механізмів автентифікації тестувалась через використання різних комбінацій паролів, ключових файлів, а також значень PIM (Personal Iterations Multiplier), які перевищували 1000. Усі спроби несанкціонованого доступу при введенні некоректних даних були успішно заблоковані, що підтверджує стабільність захисного механізму програми.

Додатково було здійснено моделювання потенційних атак, зокрема шляхом підбору пароля та аналізу часових характеристик обробки даних. У результаті встановлено, що поєднання складного пароля, ключового файлу та високого значення PIM значно підвищує загальний рівень захищеності системи, роблячи brute-force-атаки надзвичайно тривалими та малоефективними.

Інсталяція програми VeraCrypt

Для реалізації криптографічного захисту даних одним із найважливіших кроків є встановлення програмного комплексу VeraCrypt. Слід уважно вибрати версію, що відповідає використовуваній операційній системі, у нашому випадку — Windows 10. Після завантаження файлу запускається інсталятор, який відкриває вікно вітання з пропозицією обрати режим встановлення.

На початковому етапі користувачу пропонується два варіанти: встановити програму стандартним способом або розпакувати портативну версію (без встановлення). Для повноцінної роботи рекомендовано обрати перший варіант — «Install», що дозволяє інтегрувати VeraCrypt у систему та отримати повний доступ до її функціональності.

Далі відкривається майстер інсталяції, в якому слід прийняти умови ліцензійної угоди. Після цього обирається директорія для встановлення програми — за замовчуванням це папка C:\Program Files\VeraCrypt, однак користувач може змінити її за власним бажанням (див. рис. 5). Після підтвердження вибраних параметрів натискається кнопка «Install», і розпочинається процес інсталяції. Він займає кілька хвилин, після чого з’являється повідомлення про успішне завершення встановлення (див. рис. 6).

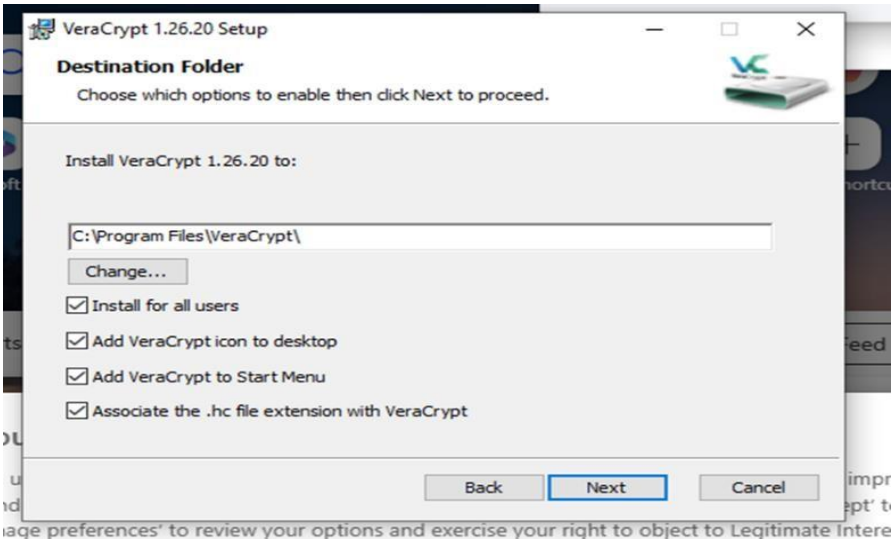

Рис. 5. Майстер інсталяції

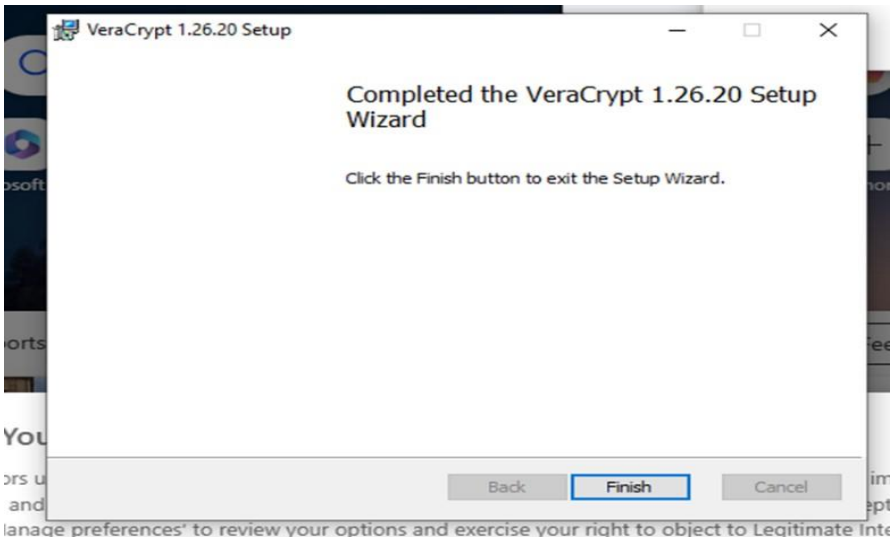

Рис. 6. Завершення інсталяції

ISSN 2663 - 4023

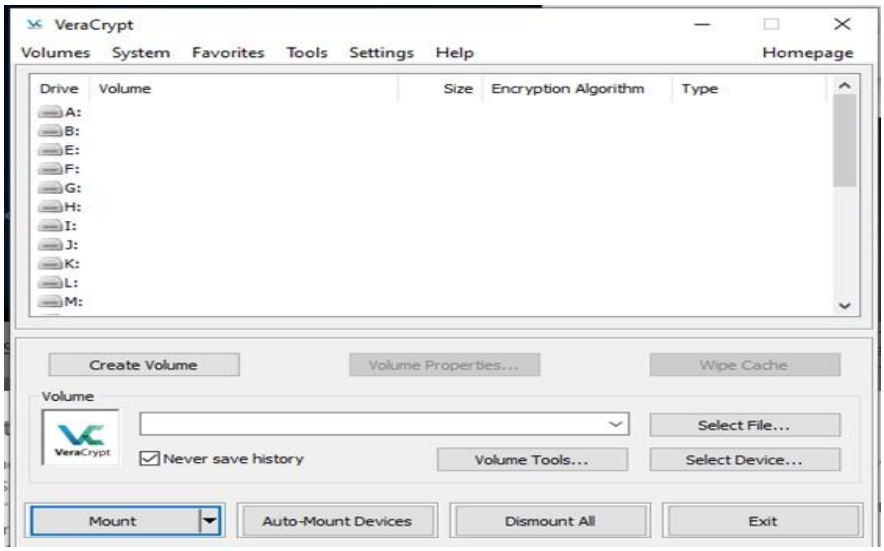

Рис. 7. Головне вікно програми VeraCrypt

Одним із базових і водночас найнадійніших способів забезпечення захисту даних в автоматизованих системах є створення віртуального зашифрованого диску, розміщеного у спеціальному файлі-контейнері. Такий диск імітує звичайний логічний том, однак фізично існує у вигляді одного файлу, що шифрується цілком, включаючи файлову структуру.

Процес створення зашифрованого контейнера в VeraCrypt складається з кількох логічних етапів:

Після запуску програми натискаємо кнопку «Створити том» (Create Volume), що відкриває майстер створення зашифрованих об'єктів (див. рис. 8). Обираємо опцію «Створити зашифрований файловий контейнер» (Create an encrypted file container) та переходимо далі (див. рис. 9, лівий верхній фрагмент). На наступному етапі користувач має обрати тип тома — звичайний або прихований (див. рис. 9, правий верхній фрагмент). Далі зазначається місце розташування файлу-контейнера, його назва та розширення (яке може бути довільним, наприклад .dat, .bin, або .hc) (див. рис. 9, лівий нижній фрагмент). Важливо вибрати надійне місце для зберігання контейнера, бажано на окремому логічному диску або зовнішньому носії. Після цього користувачеві пропонується обрати алгоритми шифрування і хешування. VeraCrypt підтримує AES, Serpent, Twofish, а також їх комбінації. Для більшої швидкодії підійде AES, а для підвищеної безпеки — комбінація AES–Twofish або Serpent–AES (див. рис. 8, правий нижній фрагмент).

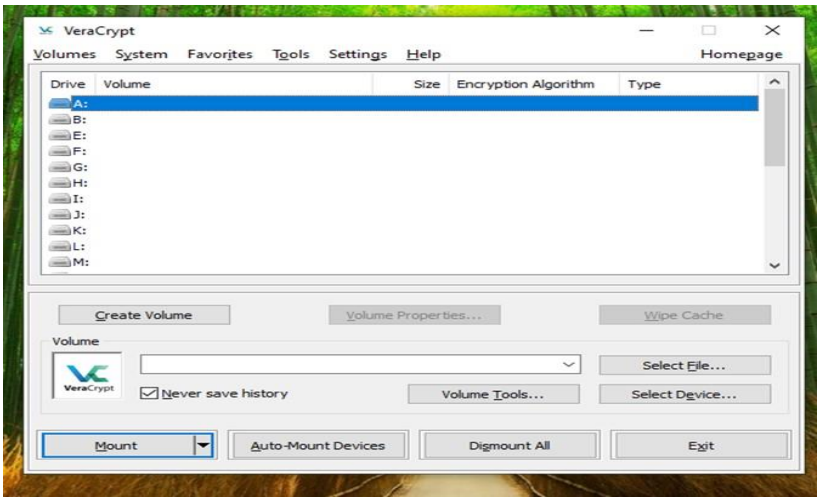

Рис. 8. Створення тому

№ 4 (28), 2025
ISSN 2663 - 4023

Рис. 9. Ключові етапи роботи з майстром створення з ашифрованого файлового контейнер: вибір типу тома, визначення його розташування, а також налаштування параметрів шифрування

Наступним кроком є вказівка розміру файлу-контейнера (див. рис. 10, лівий верхній фрагмент). Для тестових цілей зазвичай достатньо 100–500 МБ, однак у реальному використанні обсяг обирається з урахуванням обсягів конфіденційної інформації. Після цього створюється надійний пароль (див. рис. 10, правий верхній фрагмент). Його довжина повинна перевищувати 20 символів і містити комбінацію літер різних регістрів, цифр і спеціальних символів. За бажанням додається ключовий файл, який підвищує рівень безпеки і виступає додатковим елементом автентифікації. Перед фінальним створенням VeraCrypt запропонує форматувати контейнер та обрати файлову систему — FAT, exFAT, NTFS тощо (див. рис. 10, лівий нижній фрагмент). Для Windows 10 бажано використовувати NTFS, особливо якщо передбачається зберігання великих файлів.

У ході фінального етапу користувач активно рухає курсором у вікні програми — це необхідно для генерації випадкових значень, які забезпечують криптографічну стійкість. Після завершення створення файл-контейнер можна монтувати як окремий диск через головне вікно VeraCrypt: обравши вільну літеру диска, натискаємо «Mount» та вводимо пароль [3].

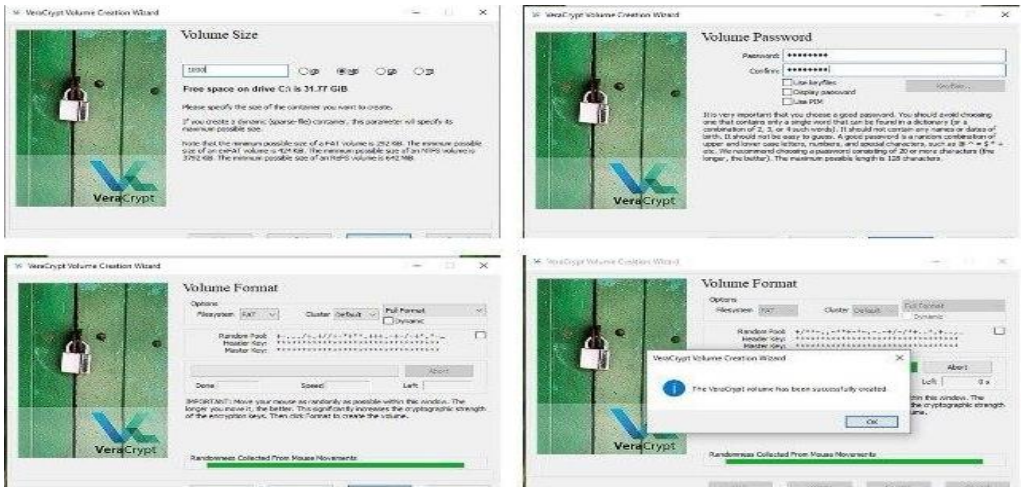

Рис. 10. Завершальні кроки створення зашифрованого віртуального диска у VeraCrypt: встановлення розміру тому, введення пароля, вибір файлової системи та форматування контейнера

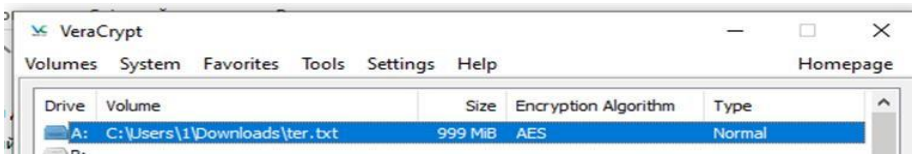

Рис. 11. Змонтований том у вікні програми VeraCrypt

Тепер ми маємо доступ до зашифрованого тому як до звичайного розділу (рис. 12).

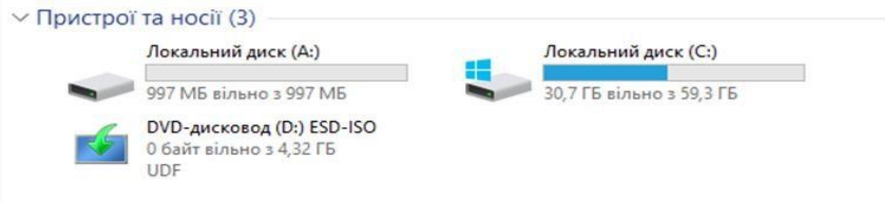

Рис. 12. Змонтований том програми VeraCrypt

Скопіюємо дані на том VeraCrypt та демонтуємо диск, натиснувши кнопку «Розмонтувати» (див. рис. 13).

Рис. 13. Розмонтування тому VeraCrypt

Шифрування системного розділу

Щоб розпочати шифрування системного розділу, необхідно відкрити програму VeraCrypt та обрати в головному меню пункт System → Encrypt System Partition/Drive. Запускається спеціальний майстер шифрування системного розділу, який поетапно проводить користувача через увесь процес. На першому етапі користувач обирає, що саме буде зашифровано: лише системний розділ (де встановлена операційна система) або весь фізичний диск, якщо на ньому розміщено декілька розділів, які також потрібно захистити (див. рис. 14, правий верхній фрагмент). В більшості випадків достатньо шифрування системного розділу, однак для підвищеної безпеки доцільно обрати повне шифрування диска. Далі користувачеві пропонується вибрати тип інсталяції: звичайне шифрування або створення прихованої операційної системи (див. рис. 14, лівий верхній фрагмент). Останнє є частиною концепції заперечуваності доступу, однак для стандартних завдань із захисту системи вибирається звичайне шифрування. Наступні кроки включають вибір алгоритмів шифрування та хешування. Рекомендовано використовувати AES або Serpent, оскільки вони мають високу продуктивність і доведену стійкість (див. рис. 14, правий нижній фрагмент). Після цього потрібно встановити надійний пароль для доступу до системи, а також, за бажанням, підключити ключовий файл, що підвищує рівень безпеки автентифікації (див. рис. 15).

№ 4 (28), 2025
ISSN 2663 - 4023

Рис. 14. Етапи налаштування параметрів шифрування системного розділу у VeraCrypt: обрано тип шифрування (Normal), визначено область для шифрування, зазначено конфігурацію з однією ОС (single-boot) та задано алгоритм AES і хеш-функція SHA-512

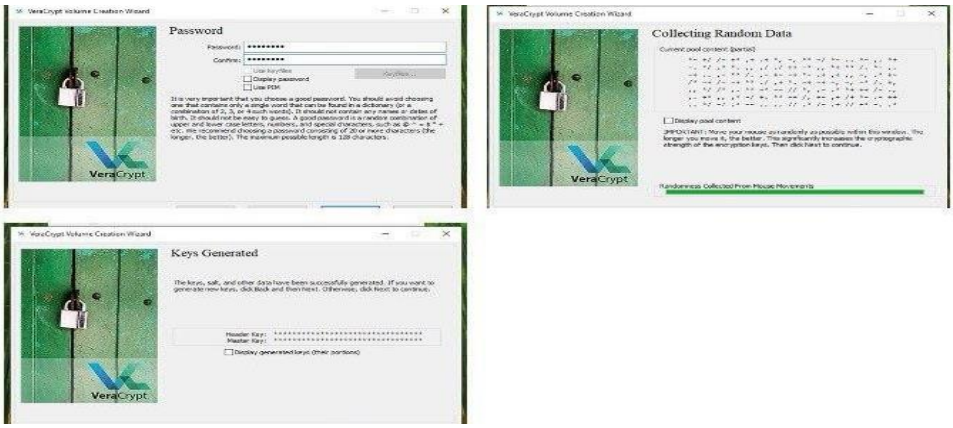

Рис. 15. Встановлення пароля та генерація ключів при створенні зашифрованого тому у VeraCrypt

Після завершення вибору параметрів шифрування, VeraCrypt пропонує створити рятувальний диск (Rescue Disk) — спеціальний ISO-образ, який необхідно записати на зовнішній носій (USB або диск) (див. рис. 16). Цей диск дозволяє відновити доступ до системи у разі пошкодження завантажувача або втрати деяких компонентів VeraCrypt.

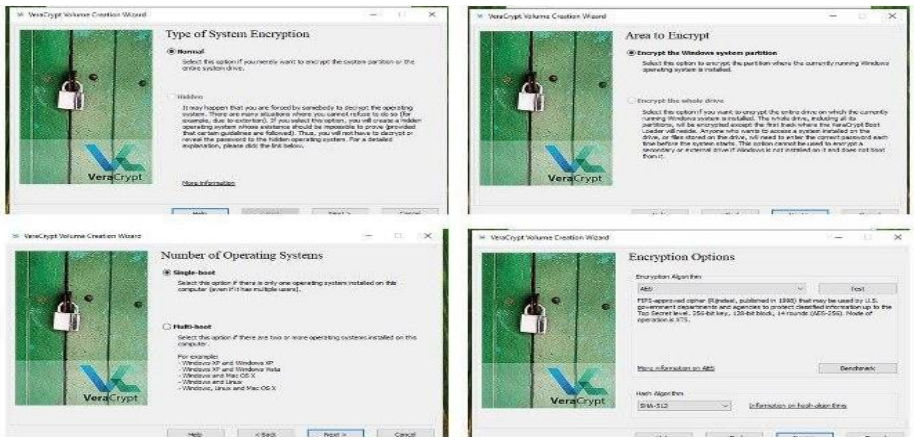

Рис. 16. Створення рятувального диска та вибір режиму стирання (Wipe Mode) у VeraCrypt, який визначає, скільки разів дані будуть перезаписані для запобігання їх відновленню

Перш ніж розпочати фактичне шифрування, програма виконує перевірку конфігурації

(тестовий перезапуск), під час якої перевіряється працездатність завантажувача VeraCrypt. Після успішного завершення цього етапу користувач підтверджує початок процесу шифрування. Залежно від розміру диска та продуктивності системи, процес може зайняти від кількох хвилин до кількох годин. Після завершення шифрування системний розділ стає повністю захищеним. Надалі, при кожному увімкненні комп'ютера, VeraCrypt відображатиме екран автентифікації ще до запуску операційної системи. Лише після введення правильного пароля (та, за потреби, ключового файлу) відбувається розблокування системи й завантаження Windows (див. рис. 17).

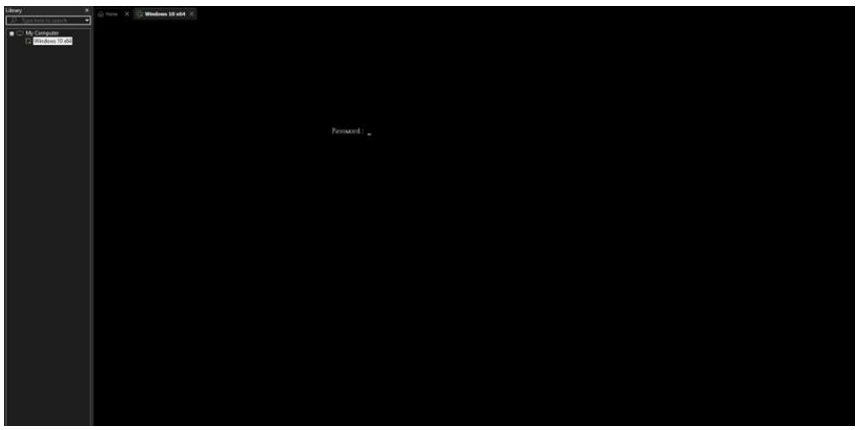

Рис. 17. Каскадне шифрування файлів

Каскадне шифрування — це процес, під час якого дані послідовно шифруються кількома криптографічними алгоритмами. Такий підхід значно підвищує стійкість до криптоаналітичних атак, адже навіть у разі компрометації одного з алгоритмів, інші залишаються активними бар'єрами захисту. Програма VeraCrypt підтримує каскадне шифрування «з коробки», надаючи користувачеві можливість обрати одну з кількох комбінацій алгоритмів шифрування ще на етапі створення контейнера або віртуального диска. Щоб реалізувати каскадне шифрування, насамперед слід запустити програму VeraCrypt та натиснути кнопку «Create Volume», яка активує майстер створення зашифрованого контейнера. У вікні майстра обираємо перший варіант — «Create an encrypted file container» і натискаємо «Next» (див. рис. 18, лівий верхній фрагмент). Далі визначається тип тома — звичайний VeraCrypt том або прихований том. У більшості випадків для реалізації каскадного шифрування достатньо стандартного тома. Після цього користувач задає шлях до файлу, де зберігатиметься майбутній зашифрований контейнер, і переходить до етапу налаштування параметрів шифрування (див. рис. 18, правий верхній фрагмент). На цьому етапі відкривається вікно «Encryption Options», де VeraCrypt пропонує список підтримуваних алгоритмів. Серед них — AES, Serpent, Twofish, а також їх каскадні комбінації: AES–Twofish; Serpent–AES; Twofish–Serpent; AES–Twofish–Serpent; Serpent–Twofish–AES .Користувач може обрати будь-яку з комбінацій залежно від пріоритету між продуктивністю та максимальною криптографічною стійкістю [4].

ISSN 2663 - 4023

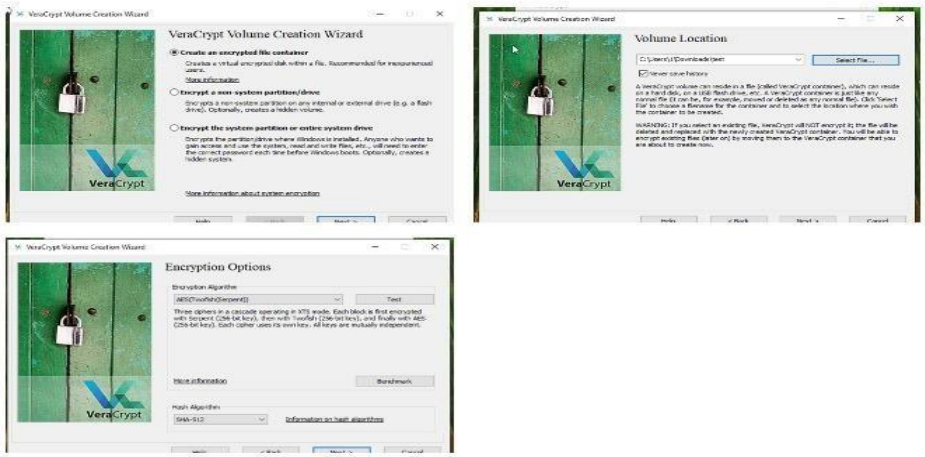

Рис. 18. Процес створення зашифрованого контейнера у програмі VeraCrypt: вибір типу тома, розташування файлу та параметрів шифрування

Після вибору алгоритму шифрування, вказується хеш-функція (наприклад, SHA-512 або Whirlpool), яка використовується для створення ключів на основі введеного пароля. Далі користувач задає розмір контейнера, створює надійний пароль, за потреби додає ключовий файл, та обирає тип файлової системи (NTFS, FAT тощо) для форматування контейнера (див. рис. 19). Фінальним етапом є генерація ентропії, де користувач рухає курсор миші, аби посилити криптостійкість ключів, після чого виконується створення та форматування контейнера. Після завершення цих кроків користувач отримує повноцінний віртуальний диск, зашифрований одразу кількома алгоритмами. Для роботи з ним необхідно монтувати його через VeraCrypt, вказавши пароль і (якщо використовувались) ключові файли. Усі файли, які записуються на цей диск, автоматично шифруються за обраною каскадною схемою.

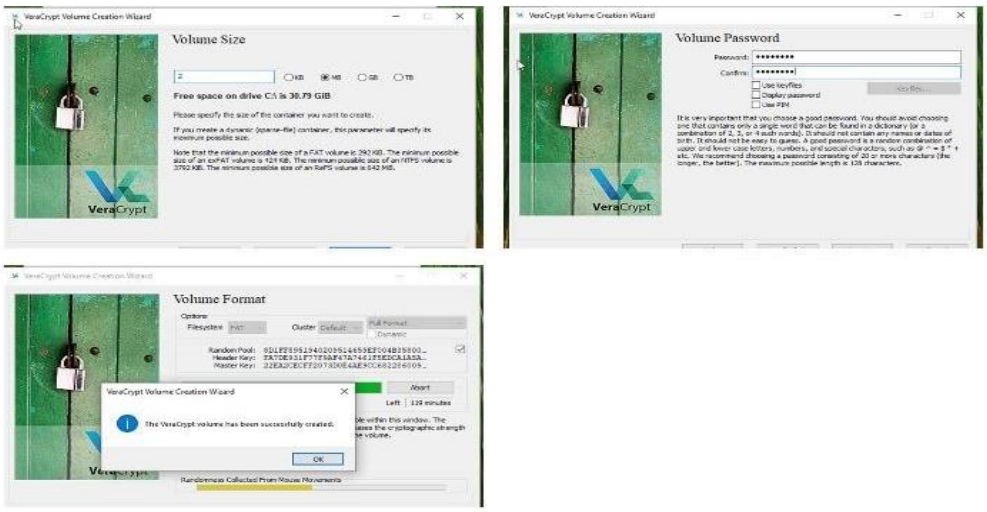

Рис.19. Процес налаштування параметрів шифрованого контейнера у VeraCrypt: вибір розміру, встановлення пароля та форматування тому

Після успішного створення зашифрованого контейнера виконаємо його монтування у системі як окремий логічний диск із призначенням букви «B». Процедура монтування включає вибір створеного файлу-контейнера, введення встановленого пароля доступу та підтвердження операції. У результаті контейнер стає доступним для перегляду й запису даних, аналогічно до звичайного диска. Усі етапи цього процесу, а також отримані результати, наочно зображені на рисунках 20–22.

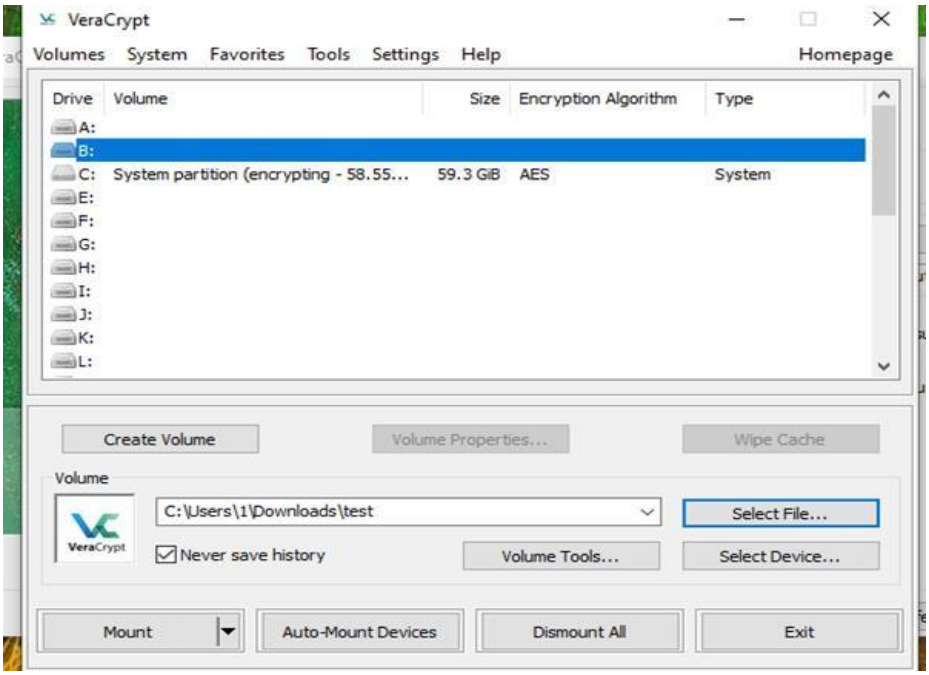

Рис. 20. Процедура монтування файлового контейнеру

Рис. 21. Вікно відображення файлового контейнеру

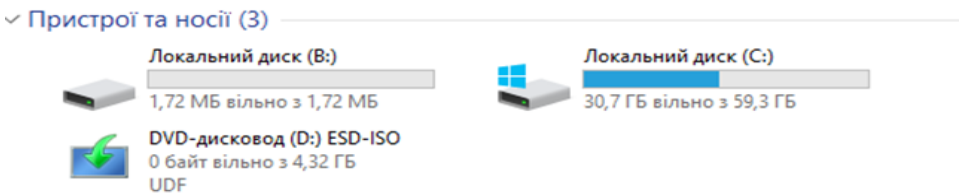

Рис. 22. Вікно відображення змонтованого зашифрованого контейнеру

Аналіз результатів

У межах проведеного дослідження було здійснено всебічне тестування програмного комплексу VeraCrypt, під час якого перевірялись функціональні можливості шифрування, автентифікації, стійкості до атак та відповідності сучасним вимогам інформаційної безпеки. На основі зібраних даних здійснено глибокий аналіз ефективності роботи утиліти та її потенціалу для використання в автоматизованих

системах. Під час тестування були оброблені дані різного типу та обсягів: текстові файли, зображення, архіви — від 5 МБ до 500 МБ. Результати показали, що VeraCrypt демонструє стабільну швидкість шифрування, яка повністю відповідає заявленим характеристикам розробника. Найшвидшим алгоритмом, згідно з тестами, виявився AES, що підтверджує його доцільність для використання в задачах, де важлива продуктивність. Водночас, каскадні алгоритми — такі як AES–Twofish–Serpent — забезпечили вищий рівень захисту, хоча і з меншими швидкісними показниками. Усі перевірені алгоритми успішно виконали шифрування та дешифрування без втрат або спотворення даних, що свідчить про надійність функціоналу. Автентифікаційні механізми також показали високу ефективність. Під час тестування було змодельовано декілька спроб несанкціонованого доступу шляхом введення неправильного пароля, перевірялась реакція системи на використання ключових файлів та параметра PIM. Усі захисні механізми спрацювали коректно: програма заблокувала доступ, що підтверджує стійкість до перебору паролів. Підвищене значення PIM у комбінації з ключовим файлом значно ускладнює brute-force-атаки, що є суттєвою перевагою в контексті сучасних вимог безпеки[5].

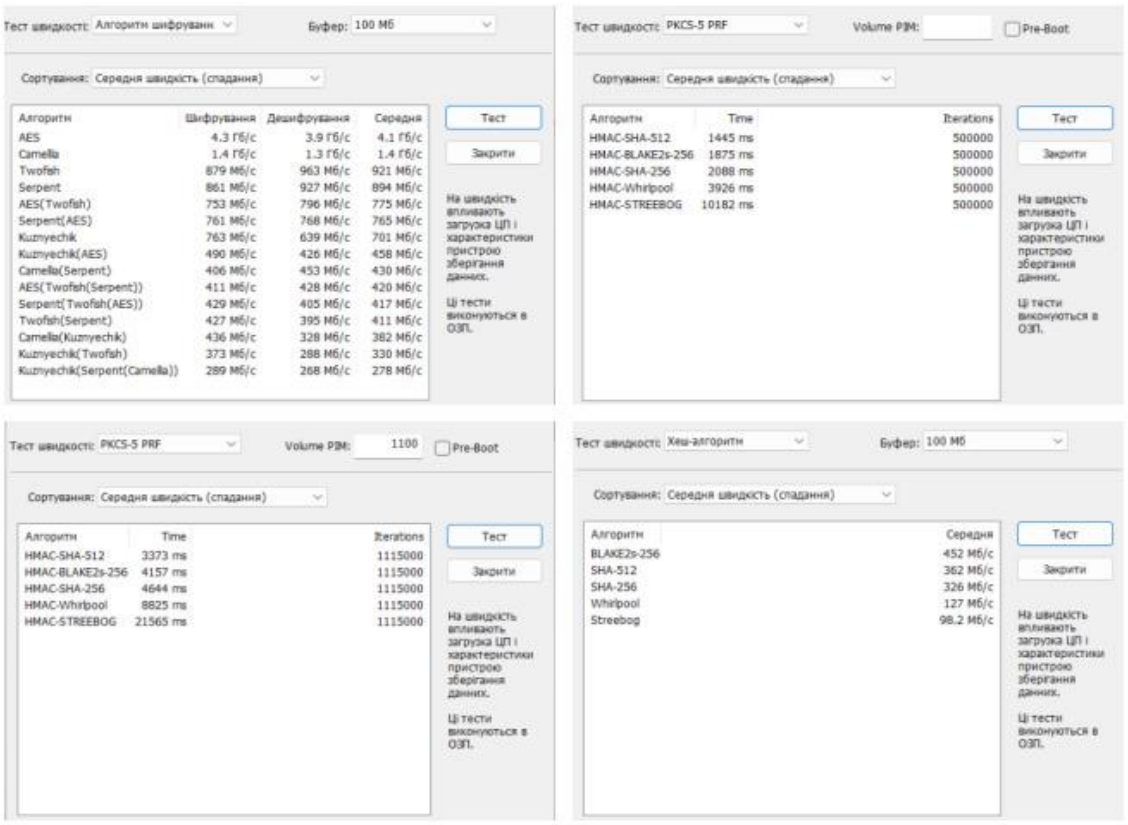

Рис. 23. Результати тестування швидкодії алгоритмів шифрування, хешування та функцій псевдовипадкового генерування в середовищі VeraCrypt

Щодо відповідності VeraCrypt чинним стандартам інформаційної безпеки, програма демонструє високу відповідність міжнародним рекомендаціям і стандартам, зокрема NIST SP 800-38E, FIPS 197, FIPS 140-2, ISO/IEC 10118-3:2004. Це робить її надійним інструментом для організацій і користувачів, які потребують сертифікованих засобів захисту даних.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У результаті дослідження ми отримали цілісне уявлення про можливості сучасних засобів шифрування та принципи їх налаштування в умовах реального інформаційного середовища. Було створено безпечне віртуальне середовище, ізольоване від основної операційної системи, що забезпечило безпечне тестування утиліти. Здійснено повний цикл дослідження: від встановлення програми, налаштування її параметрів, створення зашифрованих файлових контейнерів і віртуальних дисків, до реалізації шифрування системного розділу. Це дало змогу побачити, як працює механізм шифрування не лише окремих файлів, а й повноцінної операційної системи в цілому.

Поглиблений аналіз швидкодії алгоритмів шифрування показав, що AES є найпродуктивнішим з-поміж підтримуваних алгоритмів, що підтверджується результатами тестування на файлах розміром 100 МБ. Однак, комбіновані (каскадні) схеми шифрування, хоч і менш швидкі, забезпечують вищу криптостійкість. Також вивчено особливості роботи функцій HMAC і безключових хеш-функцій, їх залежність від параметрів, зокрема PIM, і ефективність у захисті від атак типу brute-force. Результати тестів засвідчили, що використання високих значень PIM у поєднанні з ключовими файлами значно підвищує рівень безпеки.

Окремо варто наголосити на важливості правильного налаштування системи автентифікації. Ми переконалися, що VeraCrypt успішно реалізує механізми запобігання несанкціонованому доступу, автоматично блокує доступ при помилковому введенні даних та підтримує концепцію багаторівневої безпеки.

У підсумку можна зробити висновок, що VeraCrypt є повноцінним інструментом криптографічного захисту, який повністю відповідає вимогам сучасних стандартів (зокрема NIST SP 800-38E, ISO/IEC 10118-3:2004, FIPS 140-2) і може бути ефективно застосований як в особистому, так і в корпоративному інформаційному просторі. Його гнучкість, підтримка каскадного шифрування, багатокомпонентної автентифікації та кросплатформенність роблять його придатним для широкого кола користувачів.

На основі отриманих результатів сформульовано низку практичних рекомендацій. По-перше, для підвищення рівня безпеки при роботі з критичними даними доцільно використовувати каскадне шифрування з включенням щонайменше двох алгоритмів. По-друге, слід максимально використовувати функціонал ключових файлів, комбінуючи їх з довгими паролями та високим PIM. По-третє, для новачків варто забезпечити доступ до детальної документації або покрокових інструкцій, адже налаштування VeraCrypt може бути складним без базових знань у сфері інформаційної безпеки. Таким чином, VeraCrypt підтвердив свою ефективність як багатофункціональний, безпечний і надійний засіб криптографічного захисту даних, придатний до використання в навчальних, комерційних і державних інформаційних системах.

Отримані в межах цього завдання знання й уміння дозволяють упевнено реалізовувати шифрування даних, налаштовувати політики безпеки, створювати надійні механізми захисту інформації в автоматизованих системах. Цей досвід закладає ґрунт для подальшого вивчення прикладної криптографії, а також формує системне мислення щодо побудови безпечного інформаційного середовища[6].

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. *ELARTU — Instytutsiinyi repozytarii TNTU imeni Ivana Puliuia: Metodyka bezpechnoho zberihannia informatsii na tsyfrovykh nosiiakh.* (n.d.). ELARTU — Instytutsiinyi repozytarii TNTU imeni Ivana Puliuia: Domivka. https://elartu.tntu.edu.ua/handle/lib/30605
2. *VeraCrypt chy BitLocker: shcho krashche dlia shyfruvannia?.* (n.d.). Nixj. https://nixj.ua/veracrypt-chi-bitlocker-scho-krasche-dlya-shifruvannya
3. Kozhukhar, O. (n.d.). *Shcho take VeraSrypt ta yak vona dopomahaie prykhovuvaty faily na kompiuteri?* https://cybercalm.org/novyny/shho-take-verasrypt-ta-yak-vona-dopomagaye-pryhovuvaty-fajly-na-komp-yuteri/
4. *CivSocIT. Shyfruvannia dyska za dopomohoiu VeraCrypt na Windows dlia nedosvidchenykh korystuvachiv.* (n.d.). Khabr. https://habr.com/ru/articles/558254/
5. Mieshkov, O. Yu. (2023). *Kryptohrafiia ta kryptoanaliz: metodychni vkazivky do vykonannia laboratornykh robit dlia zdobuvachiv vyshchoi osvity spetsialnosti 125 "Kiberbezpeka".* Kremenchuk: Kremenchutskyi natsionalnyi universytet im. M. Ostrohradskoho.
6. Zavadskyi, I. Ya. (2019). *Kryptolohiia: navchalnyi posibnyk*. Lutsk: Vezha-Druk.

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 4 (28), 2025
ISSN 2663 - 4023

Katerina Tereshchenko
Student
State University "Kyiv Aviation Institute", Kyiv, Ukraine
ORCID ID: 0009-0008-8469-9854
katerina60411@gmail.com

Elizaveta Chernish
Pupil
Lyceum №124 "Spassky", Kyiv, Ukraine
ORCID ID: 0009-0000-6972-6027
ycool2680@gmail.com

Tetiana Tereshchenko
Senior Research Fellow
Kruty Heroes Military Institute of Telecommunications
and Information Technology, Kyiv, Ukraine
ORCID ID: 0000-0002-9659-7897
tetiana.tereshchenko@viti.edu.ua

Yuliya Chernish
Senior Research Fellow
Kruty Heroes Military Institute of Telecommunications
and Information Technology, Kyiv, Ukraine
ORCID ID: 0000-0002-6626-5656
yuliia.chernysch@viti.edu.ua

Olena Samus
Research associate
Kruty Heroes Military Institute of Telecommunications
and Information Technology, Kyiv, Ukraine
ORCID ID: 0009-0000-7906-547X
OlenkaSamys357@gmail.com

ANALYSIS OF THE SOFTWARE COMPLEX OF CRYPTOGRAPHIC INFORMATION PROTECTION

Abstract. Information processing in the system means the performance of one or more operations, in particular: collection, input, recording, transformation, reading, storage, destruction, registration, acceptance, receipt, transmission, which are carried out in the system using hardware and software. In general, information with restricted access (critical information) requires protection, namely, confidential, proprietary or secret information (e.g., state secrets, banking secrets or other secrets). When processing restricted information, it must be protected from unauthorized and uncontrolled familiarization, modification, destruction, copying, dissemination, i.e., its basic security properties of confidentiality, integrity, accessibility and observability must be ensured. In particular, the transfer of proprietary and secret information from one system to another is carried out in encrypted form or via secure communication channels using technical and cryptographic information protection. The processed information is the object of protection. Modern cryptographic systems provide data encryption to conceal its content, confirm its authenticity, integrity and authorship. Symmetric and asymmetric encryption methods are the basis for building data protection systems and guarantee their secure storage and transmission. By conducting the analysis, we can gain the knowledge and skills to successfully implement data encryption, set up security policies, and create reliable mechanisms for protecting information in automated systems. To achieve this, it is necessary to implement a cryptographic protection software package that should ensure the confidentiality, integrity and availability of data. In the process, the functionality of the selected cryptographic protection software will be comprehensively tested. The knowledge and skills acquired in this task will allow you to confidently implement data encryption, set up security policies, and create reliable mechanisms for protecting information in automated systems.

Keywords: cryptographic protection; software; encryption; VeraCrypt.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. *ELARTU — Instytutsiinyi repozytarii TNTU imeni Ivana Puliuia: Metodyka bezpechnoho zberihannia informatsii na tsyfrovykh nosiiakh.* (n.d.). ELARTU — Instytutsiinyi repozytarii TNTU imeni Ivana Puliuia: Domivka. https://elartu.tntu.edu.ua/handle/lib/30605
2. *VeraCrypt chy BitLocker: shcho krashche dlia shyfruvannia?.* (n.d.). Nixj. https://nixj.ua/veracrypt-chi-bitlocker-scho-krasche-dlya-shifruvannya
3. Kozhukhar, O. (n.d.). *Shcho take VeraSrypt ta yak vona dopomahaie prykhovuvaty faily na kompiuteri?* https://cybercalm.org/novyny/shho-take-verasrypt-ta-yak-vona-dopomagaye-pryhovuvaty-fajly-na-kompyuteri/
4. *CivSocIT. Shyfruvannia dyska za dopomohoiu VeraCrypt na Windows dlia nedosvidchenykh korystuvachiv.* (n.d.). Khabr. https://habr.com/ru/articles/558254/
5. Mieshkov, O. Yu. (2023). *Kryptohrafiia ta kryptoanaliz: metodychni vkazivky do vykonannia laboratornykh robit dlia zdobuvachiv vyshchoi osvity spetsialnosti 125 "Kiberbezpeka".* Kremenchuk: Kremenchutskyi natsionalnyi universytet im. M. Ostrohradskoho.
6. Zavadskyi, I. Ya. (2019). *Kryptolohiia: navchalnyi posibnyk.* Lutsk: Vezha-Druk.