МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРИЗОВАНИХ СИСТЕМ ЗАХИСТУ ІНФОРМАЦІЇ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри
_____ С.В. Казмірчук
«_____»_____ 20__ р.

На правах рукопису
УДК 004.422.81

ДИПЛОМНА РОБОТА

ЗДОБУВАЧА ВИЩОЇ ОСВІТИ
ОСВІТНЬОГО СТУПЕНЯ «БАКАЛАВР»

Тема: Удосконалений програмний модуль для захисту інформаційних активів

Виконавець: М.М. Прус
Керівник: к.т.н., доцент Н.К. Гулак
Нормоконтролер: к.т.н., доцент Н.К. Гулак

Київ 2021

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет: Кібербезпеки, комп'ютерної та програмної інженерії

Кафедра: Комп'ютеризованих систем захисту інформації

Освітній ступінь: Бакалавр

Спеціальність: 125 «Кібербезпека»

Освітньо-професійна програма: «Безпека інформаційних і комунікаційних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ С.В. Казмірчук

«__» _____ 20__ р.

ЗАВДАННЯ

на виконання дипломної роботи

здобувача вищої освіти Пруса Миколи Миколайовича

1. Тема: *Удосконалений програмний модуль для захисту інформаційних активів* затверджена наказом ректора від «26» квітня 2021 р. № 652/ст.
2. Термін виконання: з 10.05.2021 р. по 20.06.2021 р.
3. Вихідні дані: проаналізувати загрози інформаційних активів на основі нормативно-правової бази України; побудувати модель порушника і надати класифікацію ризиків загроз для інформаційних активів; розробити програмний продукт на мові програмування Python в середовищі Visual Studio Code для захисту інформаційних активів.
4. Зміст пояснювальної записки: аналіз загроз інформаційних активів на основі нормативно-правової бази України; побудова моделі порушника і класифікація ризиків загроз для інформаційних активів; розробка програмного продукту на мові програмування Python в середовищі Visual Studio Code, для захисту інформаційних активів.

КАЛЕНДАРНИЙ ПЛАН

виконання дипломної роботи

№ п/п	Етапи виконання дипломної роботи	Термін виконання етапів	Примітка
1.	Уточнення постановки задачі	19.04.2021	*Виконано*
2.	Аналіз літературних джерел	20.04.2021	*Виконано*
3.	Обґрунтування вибору рішення	22.04.2021	*Виконано*
4.	Збір інформації	10.05.2021	*Виконано*
5.	Аналіз загроз інформаційних активів на основі нормативно-правової бази України	12.05.2021	*Виконано*
6.	Побудова моделі порушника і класифікація ризиків загроз для інформаційних активів	15.05.2021	*Виконано*
7.	Розробка програмного продукту на мові програмування Python в середовищі Visual Studio Code, для захисту інформаційних активів.	20.05.2021	*Виконано*
8.	Перевірка на антиплагіат	04.06.2021	*Виконано*
9.	Оформлення і друк пояснювальної записки	06.06.2021	*Виконано*
10.	Оформлення презентації	07.06.2021	*Виконано*
11.	Отримання рецензій від рецензента	08.06.2021	*Виконано*

Здобувач вищої освіти М. Прус

(підпис, дата)

Керівник дипломної роботи Н. Гулак

(підпис, дата)

РЕФЕРАТ

Дипломна робота складається зі вступу, трьох розділів, загальних висновків, списку використаних джерел і має 53 сторінки основного тексту, 39 рисунків. Список використаних джерел містить 22 найменування і займає 3 сторінки. Загальний обсяг роботи 63 сторінки.

Метою роботи є розробка програмного продукту для ефективного виявлення несанкціонованого доступу до інформаційних активів на мові програмування Python в середовищі Visual Studio Code і оперативного реагування на позаштатні події за рахунок додаткової програми повідомлення на мобільний телефон.

В роботі проаналізовано загрози інформаційних активів на основі нормативно-правової бази України, що дало можливість побудувати моделі порушника та загроз і визначити ризики для інформаційних активів.

В роботі побудовано модель порушника і надана класифікацію ризиків загроз для інформаційних активів, що дало змогу визначити відсутність в журналі подій відомостей про несанкціонований доступ зловмисників.

В роботі розроблено програмний продукт на мові програмування Python в середовищі Visual Studio Code для захисту інформаційних активів, що дало можливість визначати спроби проникнення в інформаційні активи, а також оперативно реагувати на позаштатні події за рахунок додаткової програми повідомлення на мобільний телефон.

Розроблене програмне забезпечення відноситься до галузі інформаційної безпеки і може бути використане для підвищення рівня захищеності.

Ключові слова: загроза, ризик, інформаційні активи, програмний модуль, аналіз загроз, захист інформації, несанкціонований доступ, модель порушника та модель загроз, журнал подій, програмний код, аналіз лог-файлів.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ПЗ – програмне забезпечення

ОС – операційна система

ПК – персональний комп’ютер

FTP (File Transfer Protocol) – протокол передачі файлів

ID (identity document) — унікальна ознака об'єкта, що дозволяє відрізняти його від інших об'єктів, тобто ідентифікувати.

API (application programming interface) - інтерфейс програмування веб-додатків

HTTP (HyperText Transfer Protocol) – протокол передачі даних в середовищі комп’ютерних мереж

ВСТУП

Актуальність. В реаліях сучасності третину світового ВВП займають саме інформаційні ресурси. Інформація перетворюється на товар та фактор впливу. В останні роки в Україні та у всьому світі зростає кількість кібератак та кіберзлочинів. Кожна компанія, кожна ланка суспільства володіє конфіденційною інформацією, витоки якої призводять до колосальних збитків та непоправних наслідків.

Існує також перелік інформаційних загроз зосереджений на знищенні, пошкодженні чи перехопленні різного роду даних, наслідками якого є зупинки робочого процесу, зниження цілісності мережевих структур, пошкодження комп'ютерної техніки та блокування доступності до зовнішніх і внутрішніх інформаційних ресурсів.

Для протидії кіберзлочинності великого значення набуває розробка методик та систем для захисту інформації. Кожна інформаційна система може бути захищеною, але з розвитком несанкціонованого доступу цей захист стає умовним і потребує удосконалення. Тому, тема дипломної роботи є досить актуальною на сьогодні.

Метою дипломної роботи є розробка програмного продукту для ефективного виявлення несанкціонованого доступу до інформаційних активів на мові програмування Python в середовищі Visual Studio Code і оперативного реагування на позаштатні події за рахунок додаткової програми повідомлення на мобільний телефон.

Досягнення мети потребує розв'язання таких **задач**:

- аналіз загроз інформаційних активів на основі нормативно-правової бази України;
- побудова моделі порушника і надання класифікації ризиків загроз для інформаційних активів;
- розробка програмного продукту на мові програмування Python в середовищі Visual Studio Code для захисту інформаційних активів.

Об’єкт дослідження: процес захисту інформаційних активів від несанкціонованого доступу.

Предмет дослідження: методи отримання доступу до інформаційних активів.

Практична цінність полягає в ефективному виявленні та оперативному реагуванні на спроби несанкціонованого доступу до інформаційних активів за рахунок розробки програмного продукту на мові програмування Python в середовищі Visual Studio Code й отримання сповіщення на мобільний телефон.

РОЗДІЛ 1. АНАЛІЗ ЗАГРОЗ ІНФОРМАЦІЙНИМ АКТИВАМ НА ОСНОВІ НОРМАТИВНО-ПРАВОВОЇ БАЗИ УКРАЇНИ

1.1. Поняття інформації та її захисту

Кожна термінологія містить безліч тлумачень та професійних найменувань, але відомості або перелік певних даних для подальшої оброки, зберігання та демонстрації завжди асоціюються з інформацією. В Україні діє Закон Про інформацію, що закріплює права громадян України на інформацію, закладає правові основи інформаційної діяльності. Грунтуючись на Декларації про державний суверенітет України та Акті проголошення її незалежності, Закон стверджує інформаційний суверенітет України і визначає правові форми міжнародного співробітництва в галузі інформації. [1]

Існування інформації є досить варіабельним питання, адже вона представлена у різних виглядах. До прикладу, це можуть бути паперові чи електронні документи, наукові чи інші види креслень, рисунки, радіо- чи звукові сигнали, імпульси або ж будь-які інші дані представлені у комп’ютерній сфері.

Інформація може піддаватися злочинним намірам: перехоплюватися, змінюватися та знищуватися. Для гарантування безпеки даних вводиться поняття захисту інформації.

Захист інформації представляє собою комплекс заходів, засобів та методик для підтримки доступності, конфіденційності та цілісності інформації в умовах впливів різних видів загроз, через які користувачі та власники інформації можуть зазнати збитків. Поняття загрози інформаційної безпеки означає сукупність умов і факторів, що створюють небезпеку життєво важливим інтересам особистості, суспільства і держави в інформаційній сфері. [2]

Здобування цілісності даних передбачає неспроможність видозмінення інформаційних активів особами, що не є власниками або користувачами певної сукупності інформації. Здобуття конфіденційності досягається при виконанні умови забезпечення обмеження користуванням даними для третіх осіб, що не мають прямого відношення до її створення та вживання.

Доступність здобувається за умов передбачених певною політикою безпеки для тієї чи іншої галузі або підприємства, коли доступ до користування, зміни та обробки даних має певна та визначена кількість осіб. Тому, захист інформації без дотримання цих аспектів є неможливим.

Захист інформації в Україні регулюється Законом України Про захист інформації в інформаційно-телекомунікаційних системах, який впорядковує відносини у сфері захисту інформації в інформаційних, телекомунікаційних та інформаційно-телекомунікаційних системах.[3]

Загрози, що виникають у державній сфері є досить глобальними і відбиваються не тільки на державних службовцях, а і на кожному громадянину країни. Вторгнення можуть викликати порушення конфіденційності стосовно державної таємниці, падіння статусу на міжнародній арені, а також колосальні економічні втрати. Тому, захист інформації є важливим поняттям, як для різного роду організацій, так і для всього суспільства.

Аналіз даних про інформацію та її захист зводиться до того, що удосконалення захищеності систем є нескінченним шляхом досягнення поняття безпеки. Кожної миті створюються нові засоби та методи забезпечення захисту ресурсів, але водночас із введенням новизни у сфері кібербезпеки, апетити злочинців також зростають. З’являються все нові і нові схеми проникнень та взломів, тому для протидії зловмисникам постійна модернізація систем оборони є неминучою.

1.2. Інформаційні активи та їх класифікація

Інформаційний актив – це об’єкт, який є інформацією або може містити у собі інформацію; слугує для зберігання, обробки чи передачі даних; має певну цінність для підприємства чи організації.

Як і будь яка складова інформаційного простору, інформаційні активи підлягають такій класифікації (рис. 1.1): первинні активи, активи підтримки та матеріальні активи.

Рис. 1.1. Класифікація інформаційних активів

До складу матеріального активу можна віднести всі складові, які мають певну цінність для підприємства чи організації та можуть завдати матеріальних збитків. Первинний актив характеризує собою поняття інформації, тобто це будь-яка сукупність даних та відомостей. Представляє собою персональну, стратегічну та життєво цінну інформацію.

Найдетальніше описується актив підтримки. До цієї ланки класифікації відноситься:

- Людський фактор (особа чи персонал як цілісний актив);
- Апаратне забезпечення (фізичне обладнання, що представляє ціннісний ресурс);
- Програмне забезпечення (актив, який описується цінністю додатків, застосунків та програмних засобів, що мають собівартість розробки, створення та обслуговування);

- Мережа (сукупність активів, що пов’язані з передачею інформації через локальну чи глобальну мережу та призводять до збитків внаслідок втрати даних або їх перехоплення).

Під час аналізу класифікації інформаційних активів не можна виділити будь-яку ланку як важливішу. Але у процесі розробки відповідній темі програмного модуля слід зосередити увагу саме на первинних активах та програмному забезпеченні, що належить до активу підтримки.

Важливість програмного фактору полягає не тільки у цінності певного продукту, а й у тому, що більшість застосунків працюють безпосередньо з інформацією і повсякчас звертаються до первинного активу – інформації. Більшість сучасних додатків мають у своїй складовій перелік персональних даних користувачів, бази даних клієнтів, постачальників, банківські активи та електронні гаманці, тощо.

Виходячи з викладеної інформації, програмні засоби мають бути ретельно захищені, адже вони є ключем до отримання великого обсягу відомостей, несанкціоноване привласнення яких, може привести до непоправних наслідків. Згідно формулювань чим більший обсяг взламаної інформації, тим збільшена є кількість збитків для власника чи користувача активів.

1.3. Ризики інформаційних активів, етапи управління ризиками

Для початку аналізу питання ризиків інформаційних активів слід висвітлити питання ризику. Поняття «ризик» трактується ймовірністю непередбачених збитків та втрат у певній складеній ситуації і розуміється як сумнівність результатів у можливій несприятливій ситуації.

У процесі вивченні питань: методів захисту інформації, ризиків та управління ризиками в нагоді стає світовий стандарт ISO /IEC 27005. Цим стандартом створюються певні апробації в питаннях безпеки інформації та

досяганні етапу захищеності, що згодом стають в нагоді при реалізації характеристик безпеки, які ґрунтуються на ключових елементах управління ризиками.

Поняття ризику в інформаційній безпеці ототожнюється можливістю ураження загрозою через певні вразливості активів, при цьому створюючи негативне явище – збитки. Вище згаданий стандарт містить чотири основні етапи процесу управління ризиками (рис. 1.2).

Рис. 1.2. Етапи процесу управління ризиками

1.3.1. Врегулювання обставин управління ризиками

Перший етап підпорядковує собою загальне зібрання обставин та аналізування всіх вхідних даних в подальшому дослідженні ситуації ризиків. При цьому обумовлюються основні цілі менеджменту небезпек. До них можна віднести:

- підтримування програми управління інформаційною безпекою;
- відповідність до правових норм;
- засвідчення відповідно приділеної уваги;
- формування сценарію неперервності роботи організації чи підприємства;
- формування сценарію відповідей на події загроз;

- підготовка та створення детального списку зобов’язань у питаннях досягнення інформаційної безпеки для певного програмного продукту чи роботи деякого механізму.

У підсумку першого етапу є визначеними аспекти критеріїв, які потрібні для перебігу управління ризиками інформаційної безпеки. Також отримано їхню галузь впливу, що містить перелік певних обмежень. Етап врегулювання обставин управління ризиками містить у собі низку критеріїв (рис. 1.3).

Рис. 1.3. Критерії врегулювання обставин управління ризиками

Критерій оцінки ризику розробляється задля визначення степені загрози інформаційної безпеки та базується на принципах критичності інформаційних активів та важливості дотримання конфіденційності, доступності й цілісності. Не менш важливим є використання даного критерію в питаннях обробки ризиків.

В основі критеріїв впливу знаходиться обрахунок відсотку збитковості та підсумовування виплат, що безпосередньо пов’язані з питанням інформаційної безпеки. Ці збитки та витрати можуть з’явитися у ситуаціях порушення конфіденційності, цілісності та доступності і все це залежить від рівня цінності активу. Також дані порушення можуть призвести до фінансових втрат, знижувати репутацію організації або й навіть призводити до адміністративної чи кримінальної відповідальності (в залежності від уставленої політики та зобов’язань). Отож, впливом можна вважати позитивний або негативний ефект, що реалізується на основі торгових та матеріальних наслідків.

Покладаючись на власні цілі та можливості організації, розробляється унікальний перелік критеріїв прийняття ризику. Затверджується певна шкала, що залежить від здатності підприємства брати виклики та степені можливості вистояти в непередбачуваних ситуаціях. Дана категорія може діяти на структуру загально, а також на окремі напрямки. До прикладу прийняття ризику може бути визначене у сфері фінансів, але не зачіпати правові чи технологічні аспекти. Може бути пов'язаний з тимчасовою діяльністю або певними діями організації чи бути довгостроковою ініціативою.

Підтримка організаційної структури є дуже важливим фактором в становленні системи управління ризиків інформаційними активами, які підпорядковують визначення шляхів прийняття рішень, зобов'язань та ролей. Кожна структурна форма має бути затверджена та схвалена відповідальним органом чи керівництвом.

Уся система виявлення та обрахунків ризиків має каскадоподібну форму або так званий вигляд дерева. Кожна класифікація має ряд нащадків і в свою чергу поділяється на фактори та складники. Для оцінювання ризиків активів організації існує розподіл на два типи. До них відносяться: критерії позбавлення цілісності, конфіденційності та доступності; критерії аналізу отриманих наслідків (рис. 1.4).

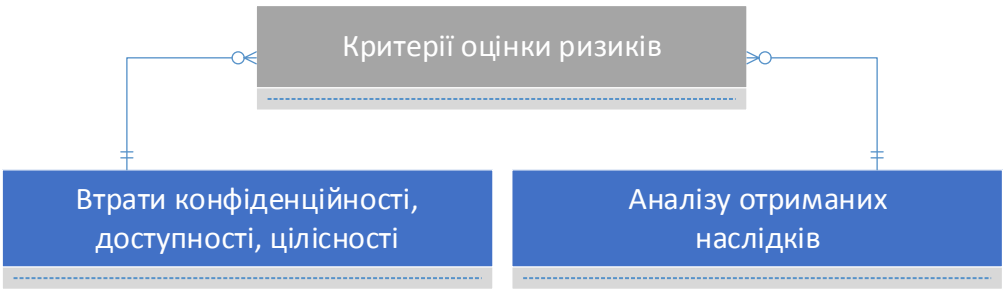

Рис. 1.4. Критерії оцінки ризиків

Перший тип критерію містить у своїй основі проблематику втрат через порушення законодавчих актів, падіння репутації, зниження рівня дії конфіденційних чинників, недотримання громадського порядку, безпеки осіб через посягання на їх приватний інформаційний простір, тощо. Окремими

факторами, що притаманні даному типу відмічається вплив на продуктивність бізнес-процесів, діяльності підприємства, шляхом фінансових втрат та потенційних погроз екологічної ситуації.

Другий тип критерію ґрунтується на обліку фінансових втрат та аудиті матеріальної збитковості, у проясненні кризових ситуацій та дії адміністративних проваджень або штрафів. Додатково враховується послаблення позиції на ринку та конкурентоспроможності, невиконання контрактних зобов'язань, звільненнях особового складу, а також повне припинення діяльності з подальшою втратою всіх активів.

Вибір типу критеріїв та їх факторів обираються кожною організацією в залежності від сфери діяльності та структури вимог, встановлених внутрішньою політикою чи рядом інструкцій. Для точного оцінювання активів цей етап вимагає детального вивчення та чіткого дотримання всіх аспектів в процесі реалізації.

Оцінка впливових характеристик теж містить пару класифікацій. До них відноситься безпосередньо прямий та непрямий впливи (рис. 1.5).

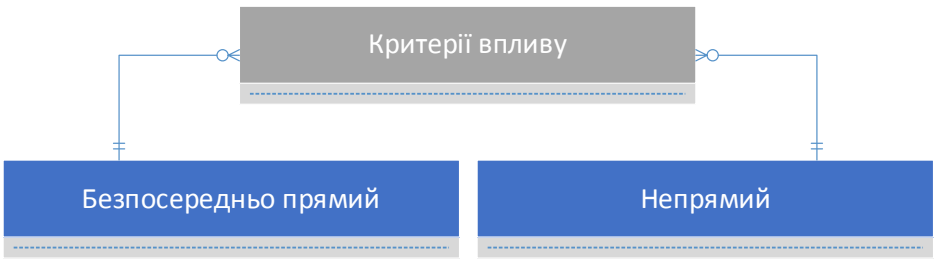

Рис. 1.5. Критерії впливу ризиків

Тип критерію, що найменується безпосередньо прямим, базується на аналізі матеріальних змін знищеного активу чи його частки. Це може спричиняти недотримання норм сфери інформаційної безпеки. В основі прямого впливу також знаходиться вартісне ведення аудиту закупівлі нових активів, зміни їх форми та завантаження резервних копій. Прораховується можливе відновлення послуг наданих активом через ліквідацію загрозливих інцидентів.

До непрямого впливу відноситься: користування хибною інформацією, що слідує через відхилення принципів захисту; недотримання етичних норм та духовності; ведення додаткових витрат на реформування активів та їхнє відновлення. Процес відновлення стосується активу, що був перерваний або не вірно використовуваний.

1.3.2. Оцінювання ризиків

Етап оцінювання ризиків є важливою ланкою процесу їх запобігання. Оцінювання ризику – це сегмент управління ризиком, що надає спроможність досягати структуризації цього перебігу, під час якого обумовлюються аспекти ефективного здобуття планових результатів.

Під час процесу оцінки, перед прийняттям будь-якого рішення є важливим висування аналізу шансів успішності та наслідків. У ході оцінювання ризиків охоплюються елементи, визначені етапом керування ризиком, а також наступні (рис. 1.6):

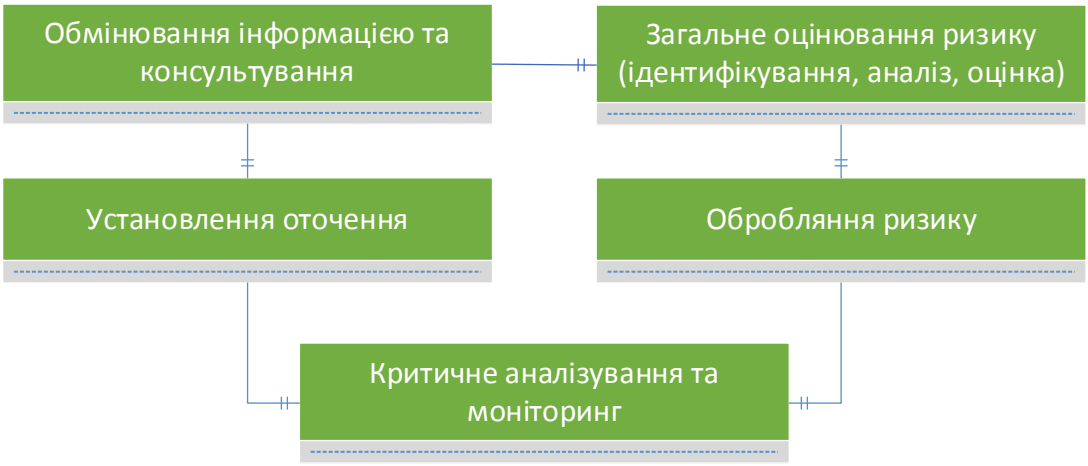

Рис. 1.6. Елементи оцінювання ризиків

Організація повинна виконувати оцінювання ризиків через заплановані інтервали або коли запропоновані чи відбуваються суттєві зміни з урахуванням критеріїв, визначених в пункті розподілу обов’язків. В ньому визначено, що заходами безпеки передбачено конфліктуючі обов’язки та сфери відповідальності, які мають бути розподілені для зменшення можливостей неавторизованої чи ненавмисної модифікації або неправильного використання

ресурсів системи управління інформаційною безпекою (СУІБ) організації. Не менш важливим елементом є утримання підсумків. Організація повинна зберігати задокументовану інформацію стосовно результатів оцінювання ризиків інформаційної безпеки. [4]

1.3.3. Розподіл на рівні пріоритетності

Для розподілу на рівні пріоритетності, кожна організація має проектувати, вводити й перевіряти нові процеси, потрібні у реалізації вимог сфери інформаційної безпеки. Не менш важливим є впроваджувати цілі для модернізації захищеності активів. Також потрібно впроваджувати плани для підвищення безпеки інформації, зберігати документовану інформацію в обсязі, необхідному для впевненості, що виконання процесу відбувається згідно плану. [5]

Організація повинна: контролювати заплановані зміни та переглядати наслідки непередбачених змін, застосовуючи дії для усунення будь-яких шкідливих дій, за потреби; гарантувати, що процеси, віддані на аутсорсинг, визначені й контрольовані.

В рамках планування ризиків діяльності потрібно спочатку ідентифікувати та оцінити ризики з усієї сукупності об’єктів аудиту, надалі – визначити потенційні для дослідження об’єкти аудиту, встановити пріоритетність їхнього дослідження за допомогою факторів відбору.

У процесі розподілу пріоритетності ризиків досить важливо дотримуватись плану поставлених задач, щоб найефективніше досягти очікуваних результатів.

Для аналізу важливості кожного об’єкта, додатково використовується набір «факторів відбору», який допомагає визначити пріоритетність (першочерговість) дослідження відповідного об’єкту ризику. Термін «фактори відбору» точно відображає сутність цього етапу – відбір тих об’єктів, дослідження яких є найбільш доцільним (рис. 1.7). [5]

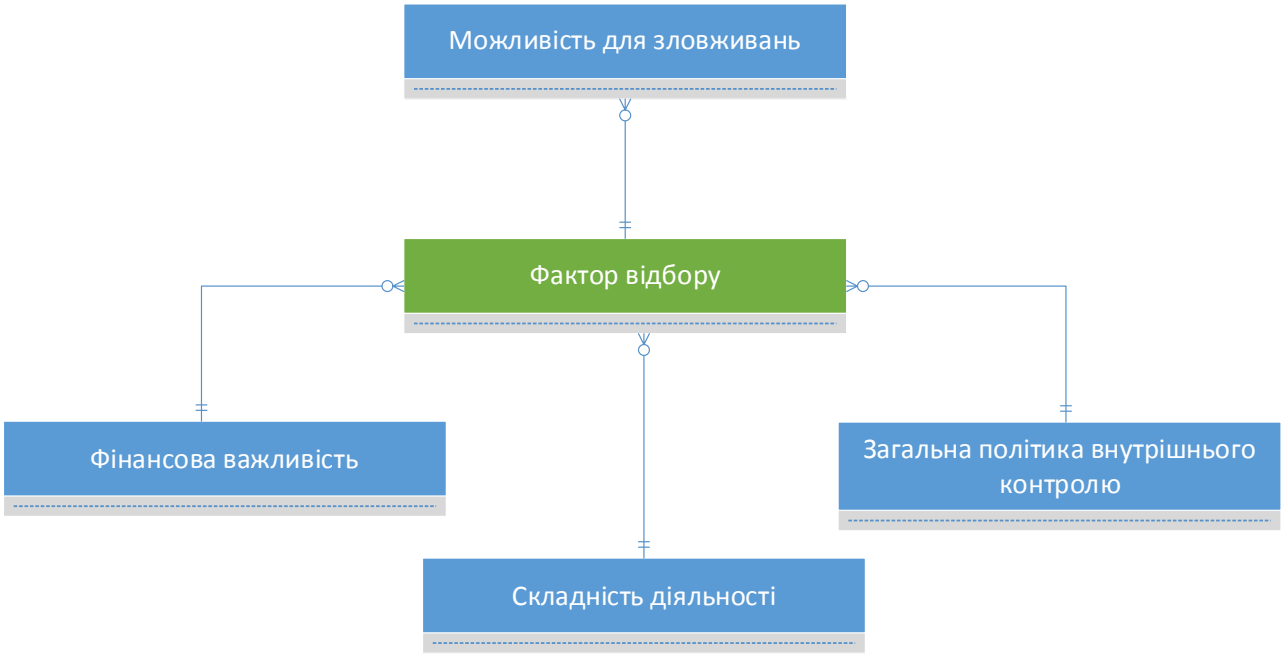

Рис. 1.7. Фактори відбору

1.3.4. Моніторинг

Для проведення моніторингу ризиків, кожна організація має аналізувати та оцінювати результативність інформаційної безпеки, а також ефективність системи управління інформаційною безпекою. Обираючи методики, що надають результати порівняння та відтворення, можна їх розглядати як обґрунтовані. Організація повинна визначати:

a) елементи, на які потрібно звернути увагу й проводити їхній моніторинг та вимірювання, включно процеси інформаційної безпеки та заходи забезпечення;

b) методи моніторингу, вимірювань, аналізу та оцінювання, які можуть бути застосовані для гарантії обґрунтованих результатів;

c) часові рамки проведення моніторингу та вимірювання;

d) виконавців моніторингу та вимірювання;

e) коли результати моніторингу та вимірювань потрібно аналізувати й оцінювати;

f) відповідальних за аналіз й оцінку результатів.

Організація повинна зберігати відповідну задокументовану інформацію як доказ результатів моніторингу та вимірювань. [6]

1.4. Висновки до першого розділу

В першому розділі було:

- проаналізовано поняття інформації та її захисту, що закріплено законодавчими актами (Закон України Про інформацію та Закон України Про захист інформації в інформаційно-телекомунікаційних системах). Судячи з цього визначено основні злочинні наміри та загрози;
- проаналізовано поняття інформаційних активів та їх класифікацію. Виходячи з даної класифікації встановлено активи підтримки, як одні з найуразливіших, так як охоплюють велику кількість факторів (в тому числі і програмне забезпечення), що підлягають злочинним намірам;
- проаналізовано сутність ризиків інформаційних активів, а також етапи управління ними на основі стандарту ISO /IEC 27005. Також надано інформацію про класифікації етапів контролю, основними з яких є врегулювання та оцінювання ризиків.

РОЗДІЛ 2. ЗАГРОЗИ ІНФОРМАЦІЙНИХ АКТИВІВ

2.1. Систематика загроз інформаційних активів

Згідно стандарту ISO/IEC 27005:2008 визначається класифікація загроз від випадкових безрезультатних до навмисних із втратами цілих груп інформаційних активів. Існує відповідний список, де кожен тип загроз вирізняється певним позначенням (рис. 2.1).

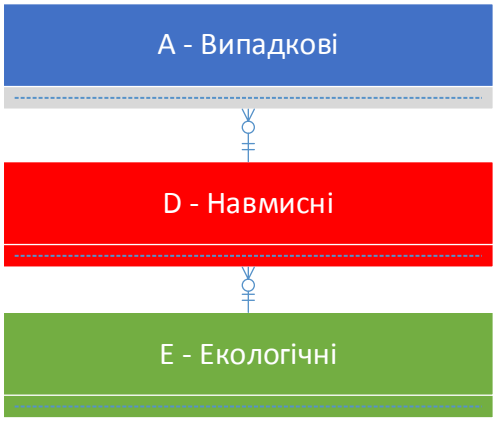

Рис. 2.1. Типи загроз

До випадкових небезпек з індексом «A» відносять перелік можливих необачних дій, що безпосередньо пов’язані з людським фактором. Людина може випадково завдати шкоди інформаційним активам. Це робить даний тип загрози притаманним для осіб, які мають доступ до певної групи даних та відомостей.

Навмисні загрози активів, що позначаються літерою «D» мають основну відмінність від випадкових, що проявляється спланованістю акцій на навмисне втручання для подальшого знищення, часткового або повного перехоплення інформації. Ці загрози активам свідомо створюються і містять злочинний контекст.

Екологічний тип загроз трактується інцидентами, що не стосуються людського впливу і відбуваються внаслідок катаклізмів чи природніх явищ, які можуть завдати шкоди інформаційним активам.

Класифікація загроз (рис. 2.2), яка включає їхні типи, є досить загальною та охоплює різні сфери захисту інформації від цілком відмінних аспектів їхньої появи. Для виявлення основи в подальших дослідженнях аналізується весь список небезпек та відкидається те, що не стосується затребуваних характеристик певного роду діяльності.

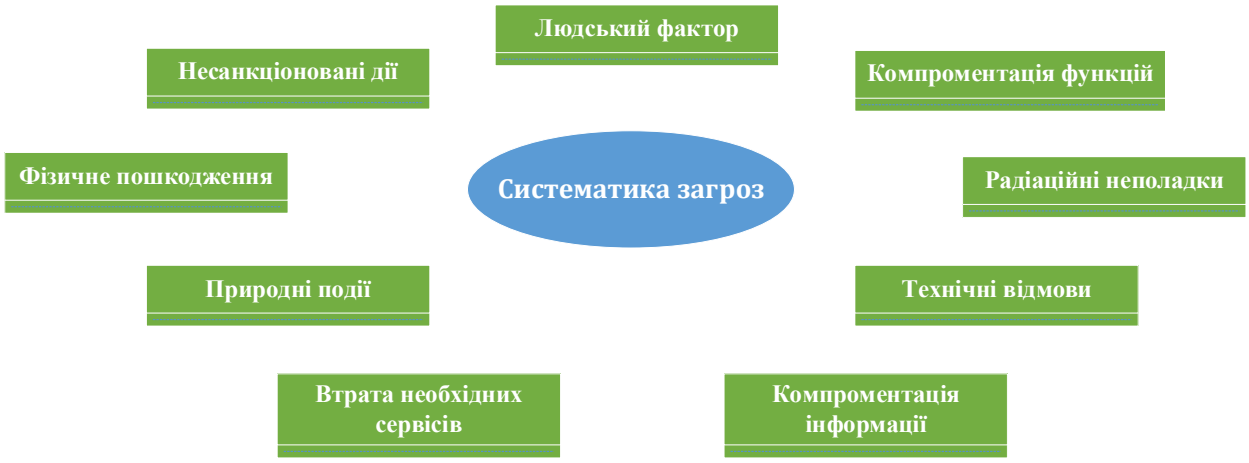

Рис. 2.2. Класифікація загроз

Аналізуючи систематику загроз у випадку протидії сферою інформаційної безпеки, слід виділити деякі елементи класифікації. Потрібно розглянути ті фактори, що безпосередньо відносяться до питання захисту активів, методами комп'ютерних інформаційних технології.

Пряме відношення до ІТ має компроментація інформації. До складу загроз, що входять до даного типу належать індексовані літерою «D» (прослуховування; відправка та перехоплення сигналу; віддалений шпіонаж; крадіжка носіїв чи обладнання; відновлення даних, що містяться на бракованих носіях; втручання до програмних та апаратних засобів) та одночасно літерами «A» і «D» (загрози виявлення та отримання даних з ненадійних джерел).

Також варто відмітити фактор технічних відмов. Характеризується він індексованими загрозами класу «A» (програмний чи апаратний збій; відмови

роботи обладнання) та «A», «D» (порушення ремонтопридатності та обмеження інформаційної системи). Програмна відмова є прямою загрозою ряду інформаційних активів, що мають відношення до комп'ютерних технологій.

Несанкціоновані дії є провідною ланкою класифікації загроз у відношенні до комунікаційних активів. Їх небезпека у більшості випадків є навмисною і описується діями: шахрайського копіювання та підробки програмних засобів; несанкціонованого користування обладнанням; незаконною обробкою та спотворенням інформації. Випадковими, тобто віднесеними до типу класифікації «A», можуть бути тільки у разі користування підробленого чи скопійованого програмного забезпечення.

Найнебезпечнішим та непередбачуваним є людський фактор класифікації загроз. Джерелами небезпеки в даній ситуації можуть бути хакери, комп'ютерні злочинці, терористи та шпигуни. Хакер, мотивований зазвичай власними переконаннями чи фінансовою вигодою, може створювати ряд негативних наслідків (рис. 2.3).

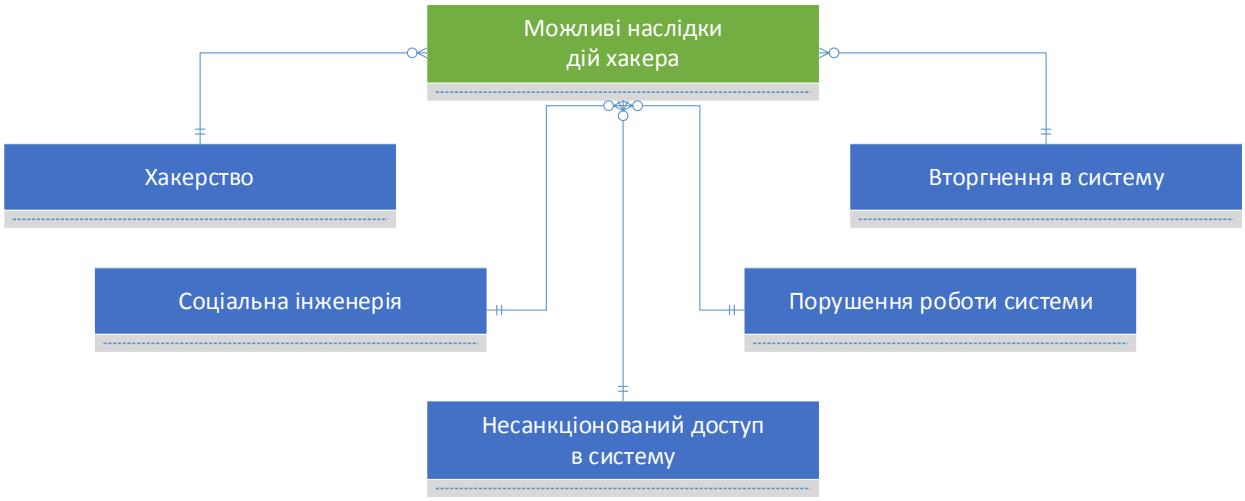

Рис. 2.3. Наслідки дій хакера

Комп'ютерний злочинець мотивований: руйнуванням інформації, стягненням фінансових активів та незаконним розкриттям конфіденційних чи інших видів даних. Схожість результатів діянь зловмисника з хакером проглядається, але існує індивідуальний перелік наслідків (рис. 2.4).

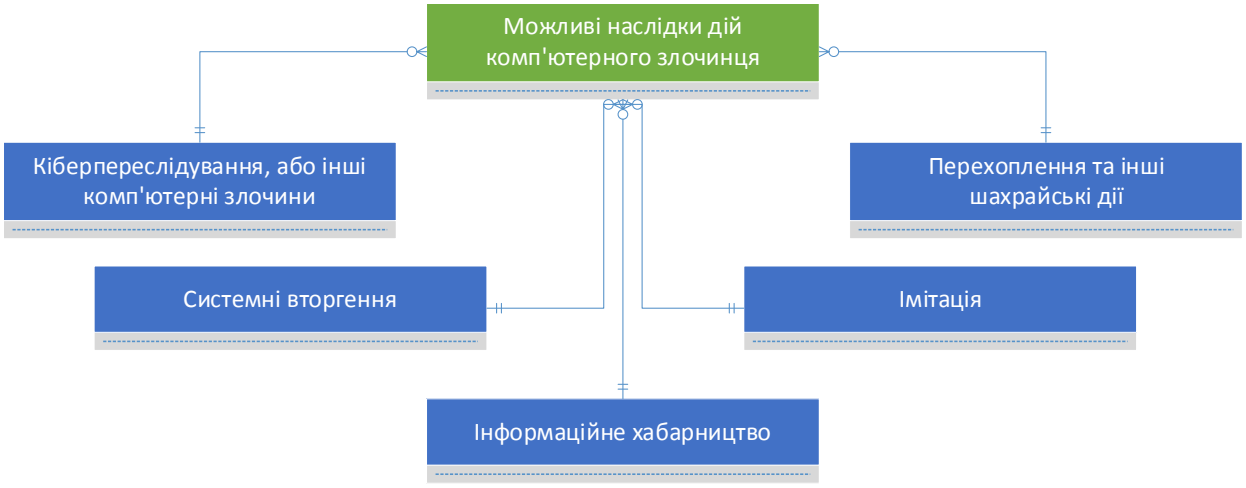

Рис. 2.4. Наслідки дій комп’ютерного злочинця

Терорист, як потенційна загроза, мотивується: політичними вигодами; шантажем та освітленням в пресі; бажанням помсти, руйнування або запланованою розвідкою. Підсумовується діяльність терориста наступними факторами (рис. 2.5).

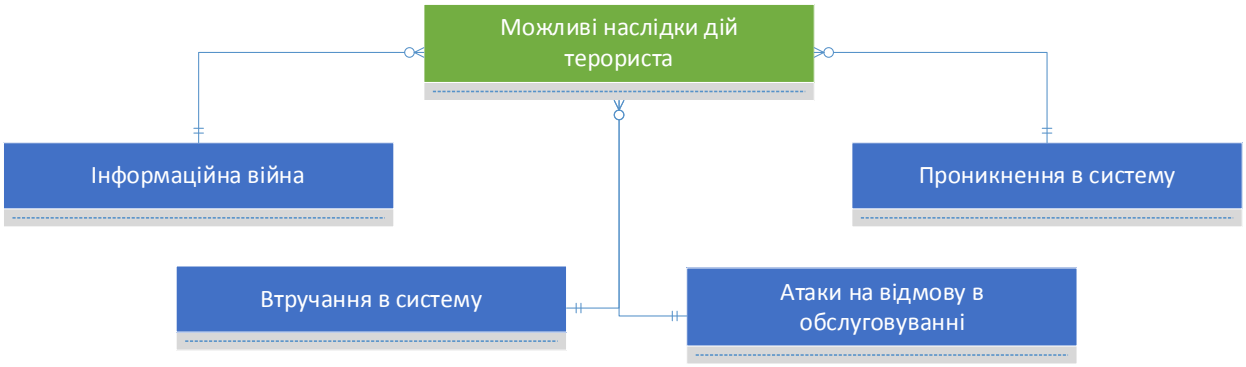

Рис. 2.5. Наслідки дій терориста

Фінальним джерелом загроз, що виходить із людського фактору, є шпигунство та дії інсайдерів. Дані складники умотивовані здобуттям конкурентної переваги та фінансово-грошової вигоди. До цієї групи класифікації відноситься також економічне шпигунство та розвідка, з метою здобуття переваги шляхом недобросовісної конкуренції, що веде до негативних наслідків постраждалої сторони (рис. 2.6).

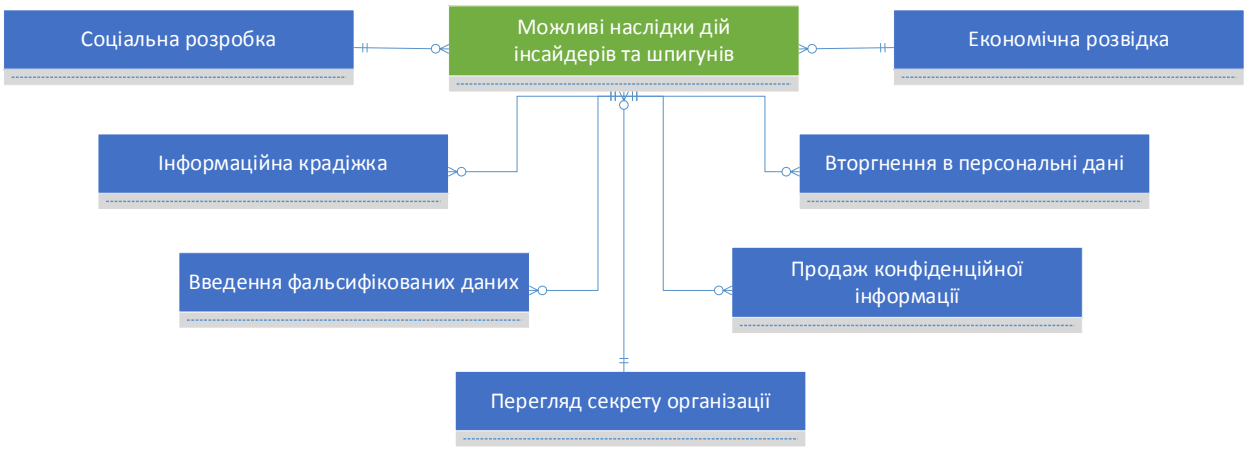

Рис. 2.6. Наслідки дій інсайдерів та шпигунів

2.2. Модель загроз

Модель загроз є підсумком аналізу потенційних загроз та представляє собою абстрактний структурований опис. Структура опису загрози рекомендується Нормативними документами України («Типове положення про службу захисту інформації в автоматизованій системі»). Загрози спрямовані на порушення властивостей інформації (рис. 2.7). [10]

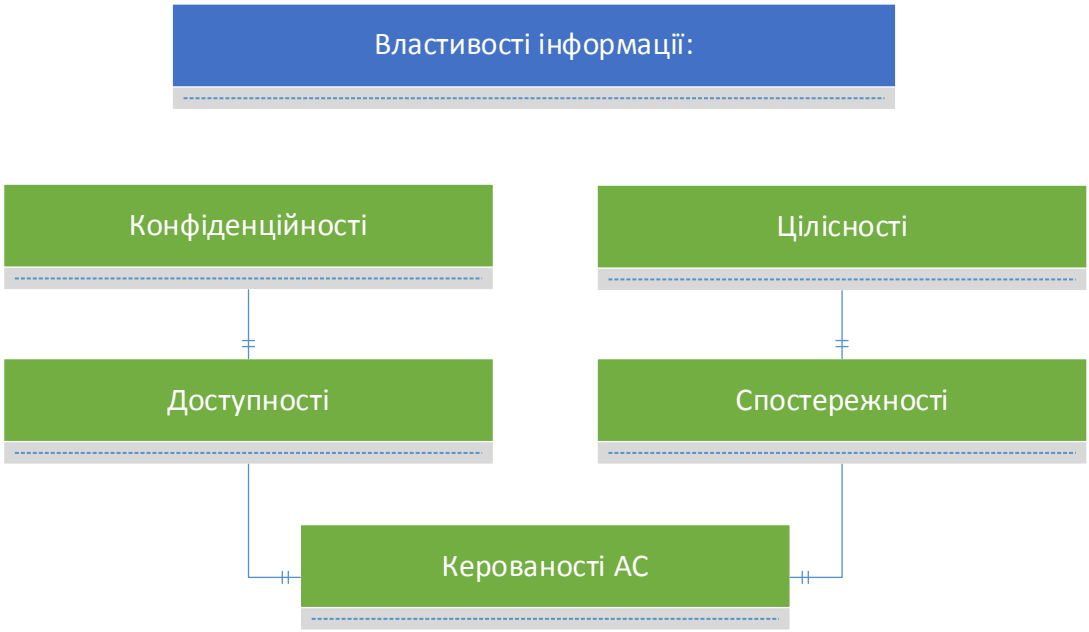

Рис. 2.7. Властивості інформації

Джерела виникнення загрози описують, які суб’єкти АС або суб’єкти, зовнішні по відношенню до неї, можуть ініціювати загрозу (це демонструється на модель порушника).

Загрози можуть здійснюватися наступними способами:

- технічними каналами, що включають до складу канали побічних електромагнітних випромінювань, акустичні, оптичні, радіо- та радіотехнічні, хімічні та інші канали;
- каналами особливого впливу, шляхом створення полів і сигналів, із метою руйнування системи захисту або порушення цілісності інформації;
- несанкціонованим доступом, шляхом підключення до апаратури та ліній зв’язку, маскування під зареєстрованого користувача, подолання заходів захисту, з метою використання інформації або нав’язування хибної інформації, застосування закладних пристроїв чи програм та вкорінення комп’ютерних вірусів. [4]

Дані способи можна розподіляти за принципом: перший та другий до фізичного доступу, третій – до логічного доступу.

Розуміння понять доступності досить просте. Кожна проведена дія людини з обчислювальною машиною (комп’ютером) проходить через фізичний доступ. До прикладу використання ноутбуків, для початку роботи з пристроєм, він вмикається внаслідок натискання кнопки живлення. До фізичного доступу відноситься також можливість ремонту, заміни комплектуючих чи чистка обладнання від пилу.

До логічної доступності підлягають всі нефізичні взаємодії людини з комп’ютером. Тобто це ситуації, коли можна працювати віддалено з обладнанням або виконувати налаштування паролів. Будь-яка робота з веб-ресурсами - це доступність до файлів, що зберігаються на серверах, які теж є комп’ютерами. Фізично користувачі сайтів не мають прямого зв’язку з серверним обладнанням, але за допомогою комп’ютерної логіки та мережевого існування, даний функціонал доступний.

При існуванні будь-якої системи з’являються виняткові ситуації. До прикладу, певна сторона зацікавлена у припиненні роботи організації-конкурента або має на меті завдати фінансових збитків. Тобто на полі зору з’являється потенційний порушник безпеки інформаційних активів.

Згідно проаналізованої інформації про спрямування загроз та способи їх реалізації, отримується дана модель (рис. 2.8).

Рис. 2.8. Модель загроз

2.3. Модель порушника

Модель порушника – це всебічна структурована характеристика порушника, яка використовується сумісно з моделлю загроз, для розробки політики безпеки інформації. [2] Україною є загальноприйнята нижче описана

структура моделі порушника. Він може належати до певної категорії осіб зображеної на моделі порушника (рис. 2.9).

Рис. 2.9. Модель порушника

Слідом за визначенням категорії, обумовлюється мета порушника. Основними цілями злочинця є здобування потрібної інформації. Отримання можливості вносити зміни в інформаційні потоки, у відповідності зі своїми намірами, може виконувати роль інсайдера чи шпигуна. Терористичні дії

можуть проглядатися у процесі нанесення збитків, шляхом знищення матеріальних та інформаційних цінностей.

Повноваження порушника в АС:

- запуск фіксованого набору задач (програм);
- створення і запуск власних програмних засобів;
- керування функціонуванням і внесення змін у конфігурацію системи;
- підключення чи зміна конфігурації апаратних засобів.

Порушник може бути оснащений технічними засобами відображеними на рисунку 2.10.

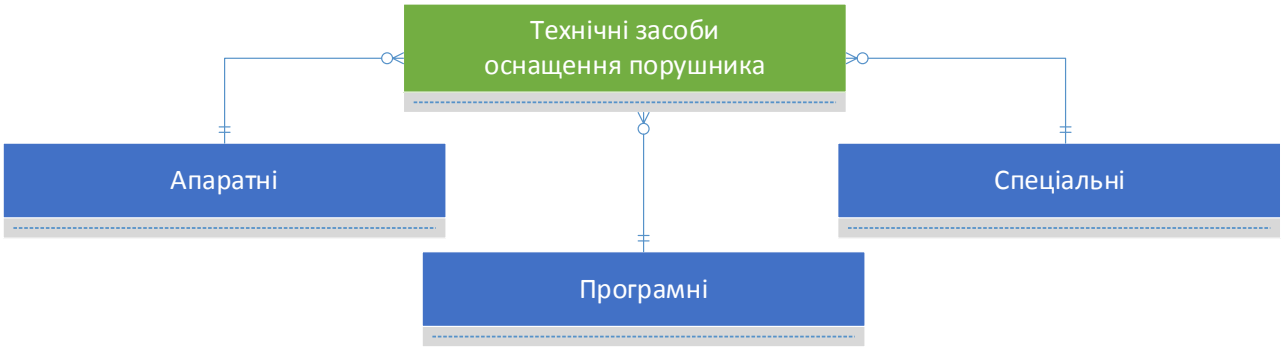

Рис. 2.10. Технічні засоби оснащення порушника

При аналізі загроз, кваліфікація порушника завжди визначається високою. В такому разі краще спрогнозувати потенційні небезпеки та підготувати інформаційну систему до гіршого. Базуються загрози порушника загалом на атаках. Атака (attack) — це спроба реалізації загрози. [10]

Тобто це дії кіберзловмисників або шкідливого програмного забезпечення, які спрямовані на перехоплення інформаційних даних віддаленого комп'ютера, отримання повного контролю над ресурсами комп'ютера, з метою виведення системи з ладу. [4]

Під атакою на інформаційну систему розуміють дії (процеси) або послідовність зв'язаних між собою дій порушника, які призводять до реалізації загроз інформаційним ресурсам, шляхом використання уразливості цієї інформаційної системи. [11]

Досить складним є процес передбачення кожного кроку злочинця. Для більш детального вивчення моделі порушника, розглядається запропонована Пітером Меллом (Peter Mell) класифікація:

- віддалене проникнення (remote penetration). Атаки, які дозволяють реалізувати віддалене керування комп'ютером через мережу;

- локальне проникнення (local penetration). Атака, що призводить до отримання несанкціонованого доступу до вузла, на якому вона ініційована;

- віддалена відмова в обслуговуванні (remote denial of service). Атаки, що дозволяють порушити функціонування системи або перезавантажити комп'ютер через мережу (в тому числі через Інтернет);

- локальна відмова в обслуговуванні (local denial of service). Атаки, що дозволяють порушити функціонування системи або перезавантажити комп'ютер, на якому вони ініційовані. Приклади атак цього типу: аплет, що перезавантажує процесор (наприклад, відкриттям великої кількості вікон великого розміру), що призводить до неможливості обробки запитів інших програм;

- сканування мережі (network scanning). Аналіз топології мережі і активних сервісів, що доступні для атаки. Атака може здійснюватись за допомогою службового програмного забезпечення;

- використання сканерів вразливостей (vulnerability scanning). Сканери вразливостей призначені для пошуку вразливостей на локальному або віддаленому комп'ютері. Вони в першу чергу призначені служити діагностичним інструментом системних адміністраторів, але можуть бути використані і як зброя для розвідки й атаки. Найвідоміші з таких програмних засобів: SATAN, SystemScanner, Xspider, Nessus;

- злом паролів (password cracking). Для цього використовуються програмні засоби, що підбирають паролі користувачів. В залежності від

надійності системи зберігання паролів, можуть використовуватись методи зламу або підбору пароля за словником;

- аналіз протоколів (sniffing - прослуховування трафіку). Пасивна атака, яка спрямована на розкриття конфіденційних даних, зокрема, ідентифікаторів і паролів доступу;

- підміна об’єкта (spoofing). Типові приклади: несправжній DNS-сервер, підміна IP-адреси джерела (IP spoofing), несправжній ARP-запит (ARP spoofing). [6]

Розглядаючи перший елемент класифікації, можна відзначити, що в час новітніх розробок існує безліч програмних застосунків для віддаленого керування комп’ютерами, які можуть піддаватися несанкціонованому захопленню.

Другий фактор, що описує локальні проникнення, актуальний через отримання доступу до персонального комп’ютера, що знаходиться в локальній мережі. Де-факто це може бути продовження першого елемента класифікації, адже ця доступність буває реалізована фізично (безпосередньо напряму, наприклад, підключення носія із шпигунським програмним забезпеченням) та логічно (використовуючи злом через мережу інтернет або порушуючи роботоспроможність програм для віддаленого керування).

Відмова у обслуговуванні, що виконується віддалено, є також сучасною проблемою, протистояти якій наразі буває досить складно. Адже дана процедура зазвичай відбувається атакою на порти персональних комп’ютерів через мережу інтернет. А так як сучасну організацію уявити без доступу до глобальної мережі не можливо, то і ризик появи даної небезпеки досить великий. Головним рушієм в даному випадку є поява зацікавленості конкурентів чи зловмисників в руйнуванні інформаційної системи.

Проблема локальної відмови в обслуговуванні описує вище згадану проблему, але базується на виведенні із ладу тільки певних комп’ютерів. До прикладу, це може бути комп’ютер адміністратора, після чого можуть

слідувати наступні атаки, адже адміністративні дії та аналіз системи професійним наглядом будуть заблоковані.

Мережеве сканування є досить глобальною з можливих атак. Виконавши аналіз мережі підприємства, зловмисник може визначити слабкі місця та програмне забезпечення, яким можна провести несанкціоновані дії. Щоб визначати вразливості у інформаційних системах, було розроблено сканери їхнього виявлення. Але це призвело не тільки до покращення проектування систем безпеки, а й на жаль, до використання даного програмного забезпечення в цілях зловмисників.

Злом паролів відбувається за допомогою спеціалізованого програмного забезпечення або вручну. Його сутність полягає у підборі паролів або зломі баз даних, що містять дані ідентифікації, з подальшим використанням «прозорості» входу до системи.

Підбор паролю називають брутфорс. Це метод пошуку всіх можливих розв’язків задачі, що полягає у підборі можливих кандидатів на розв’язок. Загалом, дана проблема може вирішитися добре спроектованою системою виявлення. Адже якщо зібрати статистичні дані і проаналізувати їх, то повторна невдала авторизація – сигнал потенційної загрози інформаційній системі.

Стосовно пасивних атак, то за допомогою прослуховування трафіку та їх протистоянню, варто відмітити можливість використання протоколів безпечного з’єднання, а також технології VPN. Дана технологія дозволяє створити віртуальну мережу безпосередньо поверх інших мереж, які є більш уразливішими.

Проаналізувавши дану класифікацію, можна прийти до висновку, що повна послідовність у ній не дотримується. Частина атак відокремлюється за кінцевим результатом чи метою реалізації, інша частина - за способом здійснення атак.

Важливо відмітити, що крім атак, порушник націлений на отримання несанкціонованого доступу до ресурсів та інформаційних активів, з подальшим використанням даних у власних, загальнопідступних намірах.

2.3.1. Аспекти несанкціонованого доступу

Несанкціонований доступ до інформації (unauthorized access to information) — доступ до інформації, здійснюваний з порушенням правил розмежування доступу, що регламентують норми процесів здійснюваних користувачами. [7]

Несанкціоновані дії можуть призводити до порушення порядку доступності певної інформації, що знаходиться в інформаційній системі. Головною проблемою, від якої залежить збитковість від проникнень, є невчасне виявлення порушника в системі.

Виходячи з цієї проблеми, обов’язковою стратегією розвитку безпеки організації стає розробка політики безпеки, проектування систем захисту інформації, а також введення моніторингу та тестування.

Створена політика безпеки може підпорядковувати собою різні правила. До прикладу, містити в своїй структурі обов’язкові перевірки журналів інцидентів в певний проміжок часу. Також якщо це розроблені системи захищеності програмного рівня – виконувати їхній періодичний запуск або запускати їх у фоновому режимі. Це дозволить на постійній основі контролювати потенційні проникнення і своєчасно реагувати на спроби взломів.

Під час несанкціонованих дій, порушник намагається авторизуватись на потрібних йому ресурсах різними методами, наприклад, підбором ідентифікаторів та паролів. Для повного розуміння, слід визначити поняття ідентифікації, авторизації та автентифікації. Ідентифікація - отримання суб’єктом первинного фінансового моніторингу від клієнта (представника клієнта) ідентифікаційних даних. [8]

Ідентифікація застосовується у процедурі отримання інформації суб'єкта мережі і базується на заданому ідентифікаторі при реєстрації чи

створенні системи. Дана процедура є первинною у процесі присвоєння доступності до мережі. Вона працює за механізмом аналізу та порівняння введених даних із відомостями про особу, що містяться на сервері.

Для захисту інформації, що зберігається на серверах, може використовуватись криптографічне перетворення. Тобто всі серверні дані можуть бути зашифровані для зменшення відсотку втрат через проникнення.

В такому разі з’являється посередня ланка процесу ідентифікації між введенням даних і їх порівнянням – процес дешифрування. При задовільних результатах співставлення відбувається допуск до наступного етапу процедур отримання доступності. Процедурною ланкою стає здійснення автентифікації та авторизації.

Автентифікація - підтвердження того факту, що певна особа має певну ідентичність та/або має право здійснювати певні види діяльності. Способом автентифікації в мережі чи будь-якій інформаційній системі є попередня ідентифікація, що базується на ідентифікаторі та паролі користувача.

Після отримання введених даних система проводить аналіз й порівняння з даними певних баз даних, які мають належати даному користувачу. У разі, коли результат порівняння є задовільним користувачу надається доступ до авторизації у системі. [9]

Авторизація — це процес надання повноважень; встановлення відповідності між повідомленням (пасивним об’єктом) і його джерелом (створеним його користувачем або процесом). Етап авторизації є фінальним, після якого користувач має доступ до певної системи та збережених в ній інформаційних активів. [10]

Для перевірки, чи був здійснений несанкціонований доступ до інформаційних активів, які знаходяться в певній системі, потрібно розробляти нові методики, що прилаштовані для роботи конкретних організації. Дані методи мають включати тестування захищеності, до якого входять аналіз інформації роботи системи та виявлення вразливості.

Важливими даними, що використовуються в цьому процесі, стає модель загроз та модель порушника. Згідно даних моделей, можливо прогнозувати потенційні небезпеки і своєчасно удосконалювати інформаційну систему. Також дані моделі дають змогу визначати відсутність подій про несанкціонований доступ в журналах інцидентів і стають рушійними елементами щодо модернізації систем безпеки.

Проаналізувавши викладений матеріал, рекомендацією для ефективного виявлення несанкціонованого доступу можна вважати удосконалення програмних модулів, відповідальних за безпеку та планове тестування загроз.

2.4. Висновки до другого розділу

Основою даного розділу було:

- виведення класифікації загроз інформаційних активів, серед яких визначено: основні загрози, що вирішуються методами комп'ютерних інформаційних технології; класифіковано дії небезпеки людського фактору, як головного порушника, що ініціює неправомірні процеси;
- проаналізовано поняття моделі загроз, структура якої описується Нормативними документами України («Типове положення про службу захисту інформації в автоматизованій системі»). Визначено, що порушення спрямовані на посягання уставлених норм конфіденційності, доступності та цілісності;
- проаналізовано інформацію стосовно моделі порушника;
- розглянуто класифікацію атак, притаманних даній моделі, та визначено основні аспекти несанкціонованого доступу злочинцями.

РОЗДІЛ 3. УДОСКОНАЛЕНИЙ ПРОГРАМНИЙ МОДУЛЬ ДЛЯ ЗАХИСТУ ІНФОРМАЦІЙ АКТИВІВ

3.1. Аналіз програмних продуктів виявлення проникнень і взломів

В наш час на кожну дію є протидія і, на жаль, на кожну систему безпеки є або швидко з'являється методика проникнення та взлому. Злом ПЗ є сукупністю подій, що є спрямованими на знищення захисту системи чи програмного застосунку. Злом використовується з наміром вивести з ладу інформаційну систему або отримати вигоди.

Функціональні можливості програми можуть бути обмежені у разі порушення їх доступності. Тому важливим аспектом стає протидія кіберзлочинцям. Для того, щоб розробляти завжди нові та досконалі системи захисту, потрібно аналізувати можливі загрози й вторгнення та розпізнавати їхню дію в системі. В реалізації даного питання найдоцільніше використовувати програмні продукти попередження вторгнень або їхнього виявлення.

Системи попередження вторгнень – це апаратні чи програмні засоби, призначення яких полягає у виявленні подій несанкціонованого доступу до мережі чи інформаційної комп'ютерної систем або несанкціоновані маніпуляції даними, здійснені через Всесвітню мережу.

Використання даних систем є актуальним в сьогоденні, адже кількість атак може бути неосяжною для людських можливостей. Тому для ефективного і оперативного реагування на події порушення безпеки використовуються системи викриття атак. Серед багатьох програмних засобів виявлення можна виділити такі програмні продукти, як: fail2ban, Snort та Suricata. Для розуміння їхньої ефективності та якості роботи, проведемо аналіз та порівняння працездатності.

Fail2Ban – це програмна реалізація захисту серверного обладнання та серверів в цілому від атак, методом вирішення криптографічних задач, перебором всіх можливих варіацій ключів. Написана дана програма мовою програмування Python і може використовуватись у системах, яким властиві інтерфейси контролювання пакетних даних чи міжмережевого екрану.

Принцип роботи даного застосунку полягає в скануванні файлів журналів (логів) і забороні IP-адрес, які потенційно активно проявляють шкідливий вплив на роботу серверу. Прикладом такої події є багаторазове намагання увійти в систему з невірними даними автентифікації. Специфічністю даного програмного продукту є робота тільки на операційних системах сімейства Unix.

Snort – це система, що виявляє та запобігає атакам, комбінуючи інспектування протоколу, використовує методику зіставлення зі сигнатурами та використовує механізм виявлення аномальних ситуацій. Це крос-платформова програма з відкритим кодом, що є вагомою перевагою. в зрівнянні з іншими подібними застосунками.

Suricata – система, що виявляє та попереджує про можливі і діючі мережеві вторгнення. Підтримує автоматичне виявлення протоколів та дозволяє здійснювати операції над ними без прив’язок до номерів портів. До прикладу блокує трафік протоколу на нестандартних портах. Даний продукт визначається високою продуктивністю роботи, обробляючи при стандартному апаратному забезпеченні потоки швидкістю до 1 та навіть 10 гігабіт на секунду. Є крос-платформовою програмою.

Із проаналізованих програмних продуктів можна виділити Suricata, адже це найоптимізованіше ПЗ, яке ефективно виконує завдання виявлення та запобігання на багатьох операційних системах. Слідом за цією системою можна зіставити Snort, адже універсальна платформовість завжди має вагому перевагу в аналізі роботи різних мереж. Вирізняється серед інших і Fail2Ban, якщо ОС сімейства Unix і є потреба у рішенні криптографічних задач, то обрати саме це програмне забезпечення буде найдоцільніше.

Як можна помітити, кожна система має ряд переваг та недоліків і розробка нових систем завжди має право бути. Варто відмітити, що тенденцією у розробці програмного забезпечення виявлення атак та забезпечення захисту, стає використання машинного навчання.

Головною причиною використання такого підходу є спроможність до автоматизації обробки вхідного трафіку, а також проведення детального та комплексного аналізу загроз. Адже комп'ютери та обчислювальні машини здатні повторювати одні й ті ж дії мільйони разів, в той час як людина, здатна втрачати ефективність своїх досліджень при великому навантаженні. Виходячи з цих намірів, саме удосконалення існуючих та створення нових систем – це вклад до успіху в сфері безпеки кожної інформаційної системи. Тому впродовж наступних пунктів буде розкриватися зміст теми дипломної роботи, щодо створення удосконаленого програмного модулю по захисту інформаційних активів.

3.2. Журнал подій ОС Windows

Журнал подій описується даними, які зберігаються в певному структурованому вигляді і містять дані про будь-які значні зміни стану операційної системи чи програмних продуктів, про які мають бути повідомлені користувачі. Розгортання програм, кліки комп'ютерної миші, натискання клавіш на клавіатурі – це все відноситься до подій. Фіксуються ж до журналу найбільш важливі події, до яких відносяться неполадки системи та застосунків.

В операційній системі Windows 10 журнал інцидентів знаходиться за таким шляхом «Панель керування/Елементи панелі керування/Адміністрування/ Перегляд подій» (рис. 3.1).

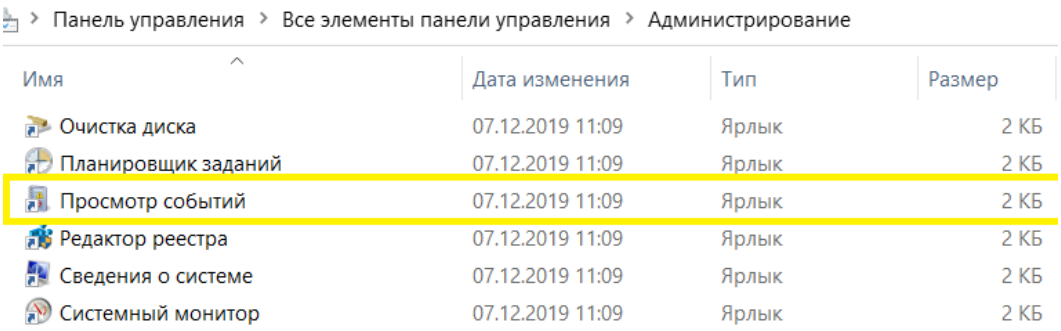

Рис. 3.1. Шлях знаходження перегляду подій ОС Windows

До основних елементів перегляду подій відносять «Журнали Windows», які містять директорії журналів: додатків, безпеки, налаштувань, системи та подій перенаправлення.

Серед переліку директорій в подальших пунктах роботи слід звертатися саме до журналів безпеки (рис. 3.2). Адже саме задля безпеки інформаційних активів, проводиться весь обсяг досліджень та реалізацій. В інцидентах безпеки зустрічаються повідомлення двох типів: аудит успіху та аудит відмов.

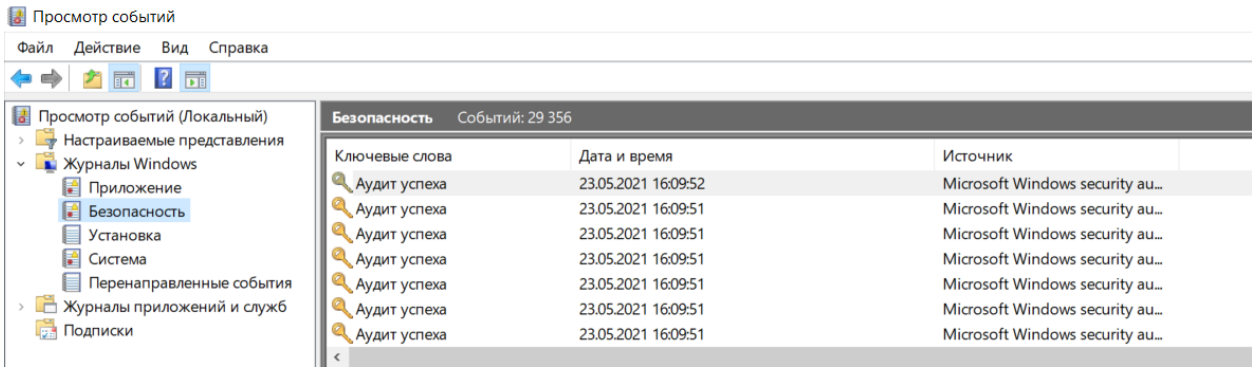

Рис. 3.2. Журнал інцидентів безпеки Windows

Аудит успіху – це подія, яка є символом успішного завершення дії, проведеної в структурі безпеки інформаційної системи. До таких подій відноситься успішна авторизація користувача.

Аудит відмов – це подія, що є символом невдалого виконання дії в структурі безпеки системи. До таких подій можна віднести невдалі спроби авторизуватися користувачеві в системі.

3.3. Аналіз використовуваного програмного забезпечення

Проаналізувавши програмні засобі виявлення вторгнень та розглянувши типовий аналог фіксації несанкціонованих дій системи Windows, можна перейти до використаних засобів виконання досліджень. У процесі роботи з реалізацією програмного модулю був використаний FTP-сервер та клієнт. FTP-сервер є стандартним протоколом прикладного рівня, призначений для пересилання від клієнта до сервера і навпаки. Даний серверний протокол, крім означення обміну файлами, підтримує функції відповідальні за безпеку.

У процесі автентифікації та захисту даних використовується побудований на основі протоколів SSL/TLS протокол присвоєний для даного типу сервера із назвою FTPS. Дане позначення містить розширення протоколу передачі файлів, що удосконалене системою захисту транспортного шару і використовує криптографічні протоколи.

Застосунки були обрані з урахуванням переваг, важливих у даній тематиці, а саме підтримці ведення власного журналу інцидентів. Програмування є процесом розробки, тестування та подальшої підтримки програмних засобів для персональних комп’ютерів чи інших технічних засобів. Так як цей процес містить у собі багато елементів та етапів, то він потребує допоміжних засобів. Вибір програмного середовища завжди є одним із початків будь-якого процесу створення програм. Адже саме в ньому розробник проводить найбільше часу на протязі всього життєвого циклу додатку.

Програмне середовище обралося згідно критеріїв чіткості та лаконічності. Visual Studio Code — є засобом розробки та редагування сучасних веб-додатків і програмних застосунків. Дане середовище є безкоштовним і доступним у використанні. Призначене для більшості операційних систем.

3.4. Розробка програмного коду для виявлення взлому

Процес розробки програмного коду для захисту інформаційних активів можна поділити на декілька етапів. Для точного і якісного дотримання програмним продуктом теми роботи кожна ланка проекту має бути досконало вивченою та реалізованою.

За основу для практичних дослідів були взяті логи. Детальніше можна розглянути питання логування в наступному підпункті.

3.4.1. Аналіз логів

Логування та логи – це вид спеціальних файлів для накопичення та збирання службових, а також статистичних даних про розгортання подій в системі чи іншому програмному продукті.

У операційних системах (найчастіше для серверного використання), а також у серверному програмному забепеченні проводиться розвинена система логування. Завдяки чому у файлах-логах реєструються де-факто всі події. Головною особливістю логів є їх розподіл. Тобто, для прикладу, логи веб-ресурсів містять інформацію гостів сайту (IP-адреси комп’ютерів, дату та час відвідування, тривалість сесії, переглянуту інформацію, тощо).

Всі дані, які можна побачити у лог-файлах, надалі використовують в аналізі роботи ресурсу: кількість авторизацій, помилки у роботі, технічні збої, тощо.

Завдяки логуванню збираються статистичні дані, на основі яких, створюються звіти, відбувається відстеження поведінки потенційних зловмисників та підозрілих користувачів.

Виходячи з вище викладеної інформації, для побудови програмного модуля захисту інформаційних активів варто обрати за основу саме аналіз логів. Аналогом логування можна вважати «Перегляд подій» у системі Windows. Але цей журнал безпекової інформації є надто слабким засобом відслідковування. Більшість подій, спроб несанкціонованого доступу чи інших

злодіянь, особливо в середовищі програмних засобів – «аудитор» логування Windows не помічає.

Слідуючи з цього, було проаналізовано різні програмні засоби, що мають потрібний функціонал. Для того щоб зімітувати роботу Windows, якнайкраще підходить взаємозв'язок типу «клієнт-сервер». Таким чином, вибір впав на безкоштовне програмне забезпечення «FileZilla». Ця програма представляє собою універсальний FTP, що містить дві версії «клієнтську» та «серверну».

FTP - це приклад мережевого протоколу призначеного для обміну файлами сторонами клієнта та сервера в локальній та глобальній комп'ютерних мережах. Повертаючись до суті, можна відмітити, що вагомою перевагою даного продукту є відкритий код, що дає змогу до повного аналізу роботи засобу та його удосконалення чи проведення змін зі сторони програмування.

Торкаючись піднятого питання логування, важливо відмітити ще одну з причин обирання програмного забезпечення «FileZilla». Даною причиною стало саме серверне логування. Тобто кожна дія, кожен здійснений крок із сервером записується у лог-файл (рис. 3.3).

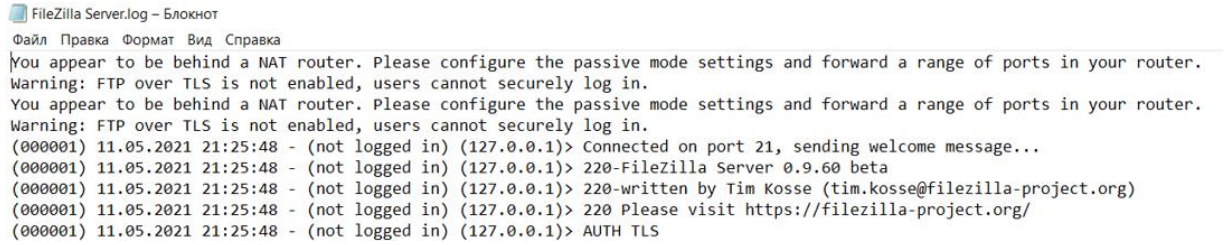

Рис.3.3. Приклад логування серверної сторони програми

На даному рисунку відображено приклад структури файлу. Можна помітити детальну інформацію: час, дату, статус (авторизований клієнт чи ні), IP-адресу, довідкову та іншу важливу інформацію. Це доволі короткий перелік даного аудиту, бо зрозумілим є те, що кожен крок клієнта може бути унікальний і все це зберігає система ведення логів даного програмного застосунку.

Отже, база на основі якої можна аналізувати, є гнучкою та чудовою для реалізації будь-якої прикладної задачі. Для більш наочного розуміння роботи застосунку «FileZilla» зімітуємо повну ситуацію втручання потенційного зловмисника у нашу систему «клієнт-сервер».

При першому запуску програми, файл логування виглядає таким чином (рис. 3.4):

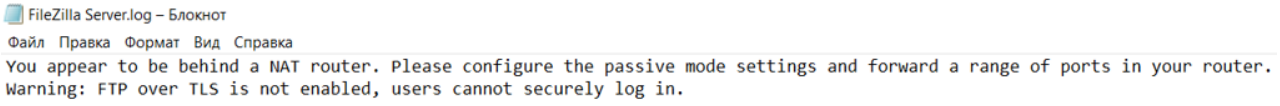

Рис. 3.4. Приклад файлу, який ще не містить логування

Жодних системних записів, крім повідомлення, що FTP-взаємодія не активна і користувачі не авторизувались на сервері, – не виявлено.

Наступним етапом запускаємо наш сервер, прописавши ім’я хоста, порт та пароль адміністратора (рис. 3.5).

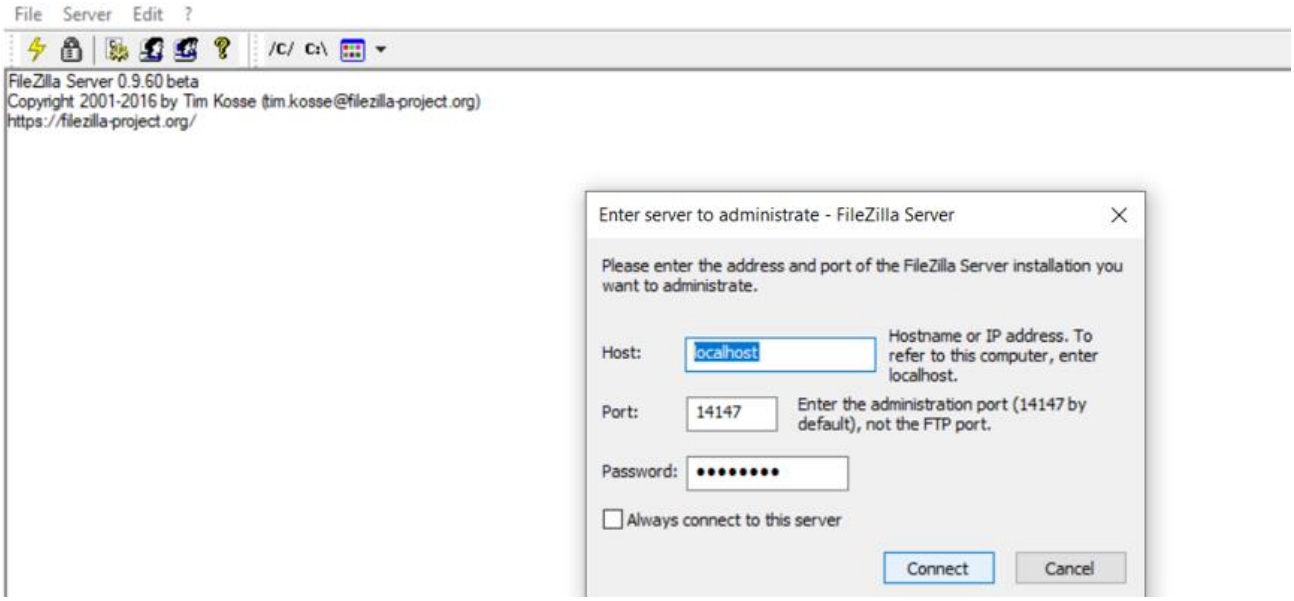

Рис. 3.5. Запуск сервера адміністратором

Важливою дією є точне дотримання правильності набору всіх даних авторизації. У верхній частині помітні вкладки налаштувань сервера,

редагування файлу та середовища. Нижче розміщені вкладки роботи з директоріями для визначення подальшої доступності клієнту. Наявне меню допомоги, що є вагомою перевагою у користуванні даним продуктом.

Слідом за цим вмикається клієнтська версія програмного забезпечення і відбувається спроба підключитись до серверу по наявному паролю та логіну користувача зареєстрованого на сервері. Обов'язково вказується ім'я хоста та порт підключення. До початку з'єднання відображається область файлової бібліотеки локального комп'ютера, а також зображено заблоковану область віддаленого підключення та вікно подій, яке не містить жодних записів (рис. 3.6).

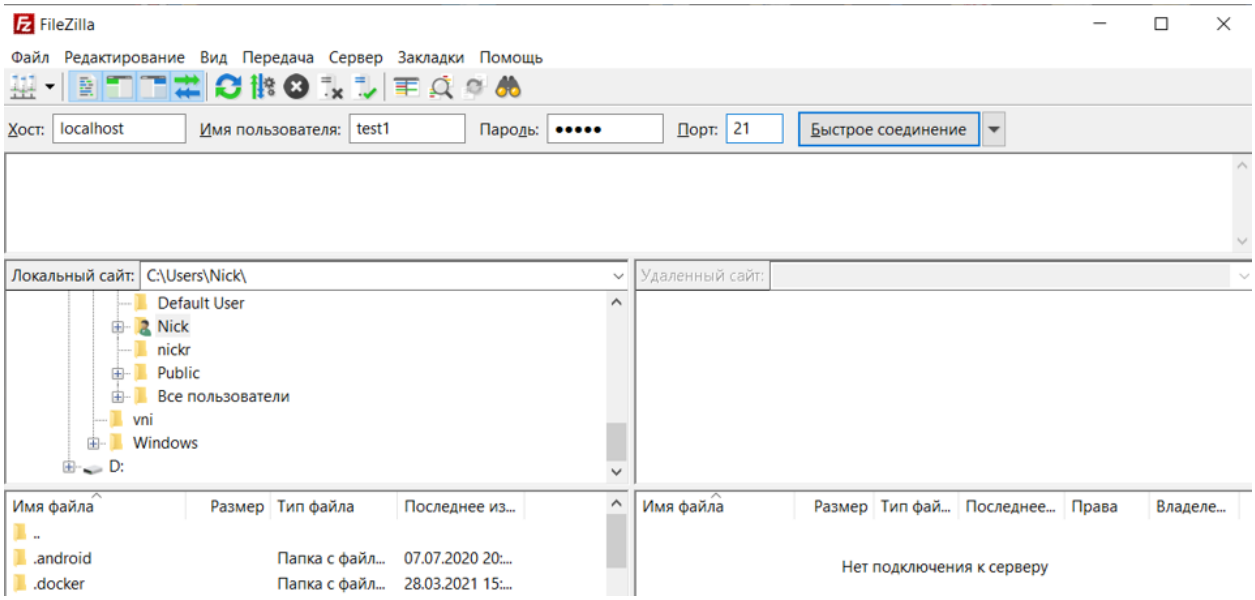

Рис. 3.6. З'єднання клієнтського ПЗ із сервером

За умови правильності введених даних, проводиться успішне підключення до серверу (рис. 3.7). Програмне забезпечення починає вести аудит подій. В списку можна помітити успішну авторизацію та отримання доступу до каталогу файлів серверних жорстких дисків.

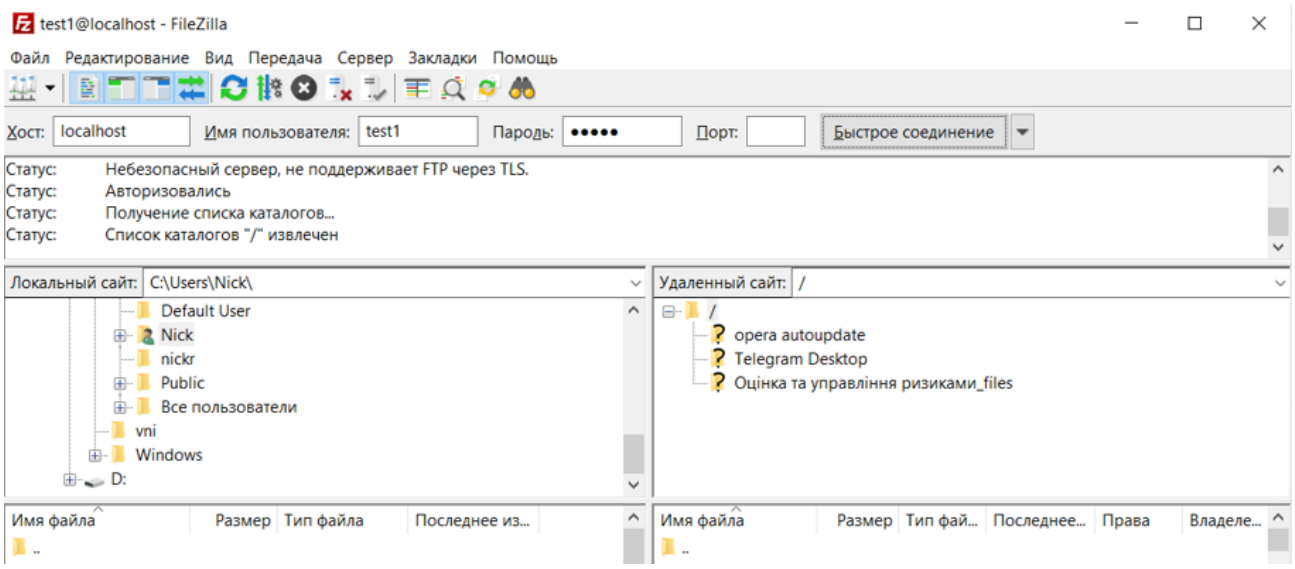

Рис. 3.7. Успішна авторизація на сервері

В той же час можна розглянути файл логування, запис якого проводить серверна частина програмного забезпечення. У даному файлі більш детально прописані всі дії, виконані під час підключення та роботи з сервером. Чітко відмічається дата, час, користувач, IP-адреса, а також та чи інша дія фіксована в роботі з обладнанням (рис. 3.8).

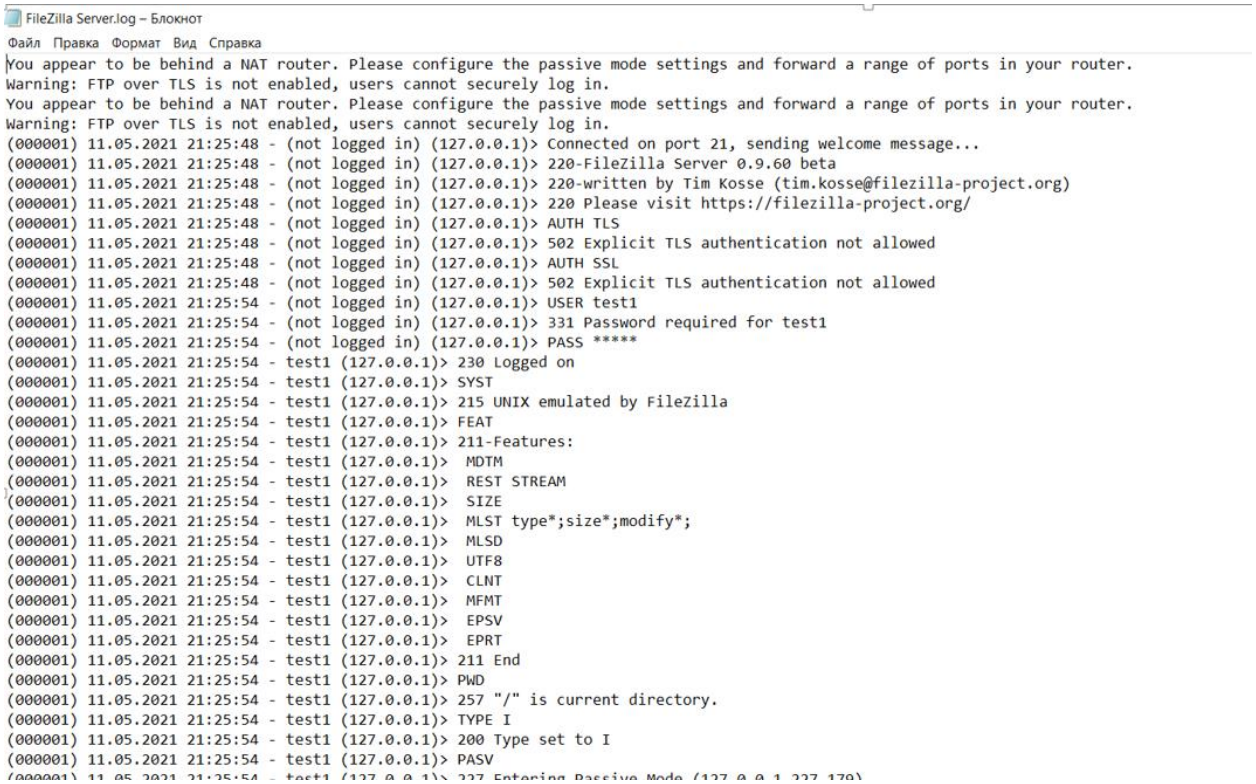

Рис. 3.8. Зображення виконання запису логів серверу

3.4.2. Імітація несанкціонованого доступу

Проаналізувавши роботу логування даним програмним забепеченням, було помічено, що вагома частка записів залишається поза людським оком та відправляється до файлу з логами. Працівник служби, відповідальної за безпеку, звісно може шукати самостійно потрібний йому запис, але серед мільйонів рядків – це стає дуже складною і іноді не можливою задачею.

Сам програмний засіб не відображає всіх логів, а записує їх у файл. В даному випадку можна зімітувати модель потенційного зловмисника. Яким чином це реалізується? Логічним фактом підозрілих дій стає неодноразова хибна авторизація на сервері. Це може стати сигналом, що хтось має на меті отримати доступ, ламаючи сервер підбором паролів.

Продемонструємо дану ситуацію на наступних рисунках. Першим ділом потрібно обов’язково від’єднатися від серверу, завершивши попередню сесію підключення. Для прикладу введемо спеціально невірне ім’я користувача «error» і спробуємо приєднатись до нашого сервера (рис. 3.9).

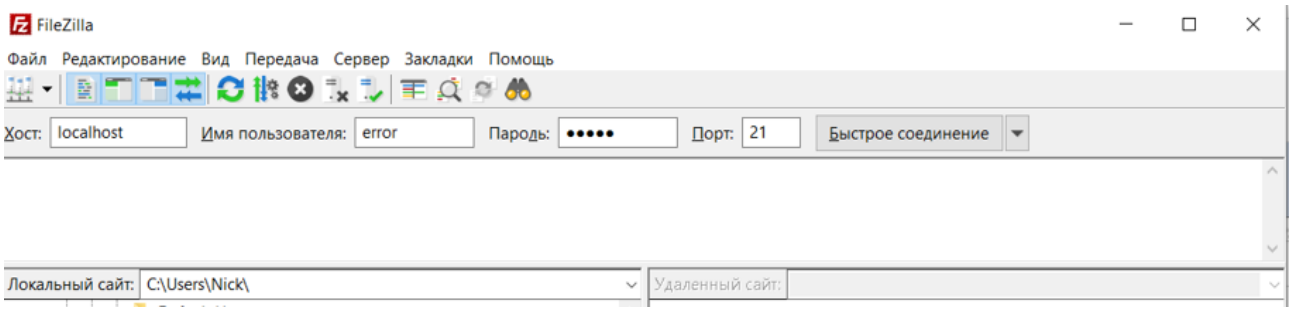

Рис.3.9. Введення невірного логіну користувача

Внаслідок невірно введених даних в інформаційному вікні, через помилку, помічаємо повідомлення (рис. 3.10). Але, на жаль, при великому потоці щосекундної спроби доступу або й успішної авторизації зловмисника, дане єдине повідомлення можна втратити в такому маленькому вікні.

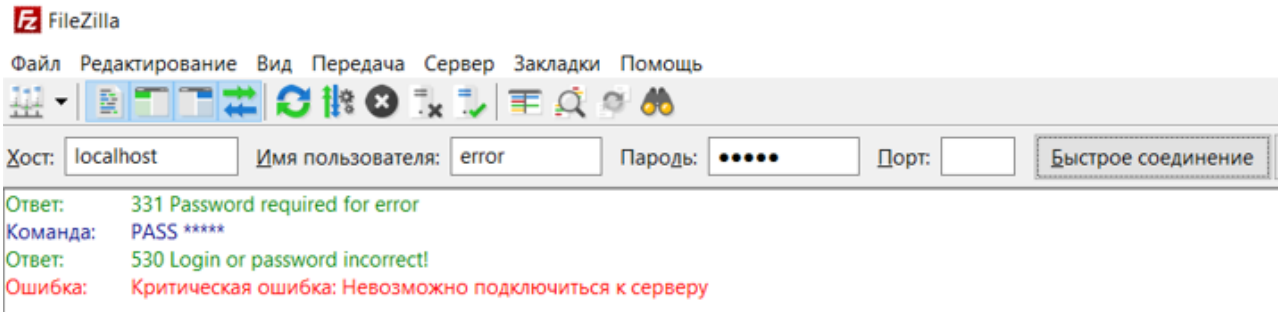

Рис. 3.10. Відображення помилки авторизації

Щоб перевірити всю складність ситуації, потрібно звернутися до логів записаних у файлі. Слідом проаналізувати всю ситуацію, що виникла у зв'язку потенційного вторгнення на сервер. Після чого можна зробити висновки, створити рекомендації та спробувати покращити роботу фіксації та аналізу даного журналу подій.

Серед великої кількості інформації знаходиться та, що цікавить нас (рис. 3.11). Помічаються зафіксовані помилки у роботі сервера та спроби авторизації, які не виявились успішними.

```
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> Connected on port 21, sending welcome message...
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> 220-FileZilla Server 0.9.60 beta
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> 220-written by Tim Kosse (tim.kosse@filezilla-project.org)
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> 220 Please visit https://filezilla-project.org/
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> AUTH TLS
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> 502 Explicit TLS authentication not allowed
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> AUTH SSL
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> 502 Explicit TLS authentication not allowed
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> USER error
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> 331 Password required for error
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> PASS *****
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> 530 Login or password incorrect!
(000002) 11.05.2021 21:28:31 - (not logged in) (127.0.0.1)> disconnected.
```

Рис. 3.11. Визначення запису про взлом сервера

3.4.3. Розробка програмного модулю захисту інформаційних активів

Роботу програмного застосунку серверної частини «FileZilla» можна вважати досить хорошою, адже у відмінності від деяких схожих програм, вона веде аудит і логування, а також записує всі результати до файлу. З цим можна вже працювати, і працівнику відповідальному за кібербезпеку підприємства можна знаходити «дірки» в системі та спроби взломів.

Але що вдіяти працівнику котрий стикається з нескінченним списком запитів та дій, які відбувалися на сервері? В такому випадку доводиться

витрачати великі проміжки часу тільки на пошук можливих помилок чи збоїв. Також людський фактор призводить до припускання помилок і пропусків певної інформації, що може потягнути за собою певні наслідки, іноді й фатальні для інформаційних систем.

Саме із подібних намірів, згідно обраної теми диплому, був розроблений програмний продукт на мові Python. Основними задачами продукту стали пошук та виявлення інформації потенційних загроз із файлу логування. Найпершим до чого була зведена ця інформація – аналіз великої кількості даних, з визначенням помилки авторизації користувача на сервері. Даний фактор для реалізації базувався на хибності логіну чи паролю введеного в систему доступу.

Детально вивчивши файл логування, було визначено дане сповіщення про помилку. На основі тексту повідомлення побудувалася логіка програмного продукту.

3.4.3.1. Блок схема алгоритму програмного застосунку

Перед процесом розробки будь-якого програмного продукту створюється блок-схема, що зображає алгоритм для проектування. До цієї схеми відноситься детальний аналіз кожного етапу роботи програми та варіант розгортання подій в тій чи іншій ситуації. Блок схема для розробки програмного модулю захисту інформаційних активів відображена на рисунку 3.12.

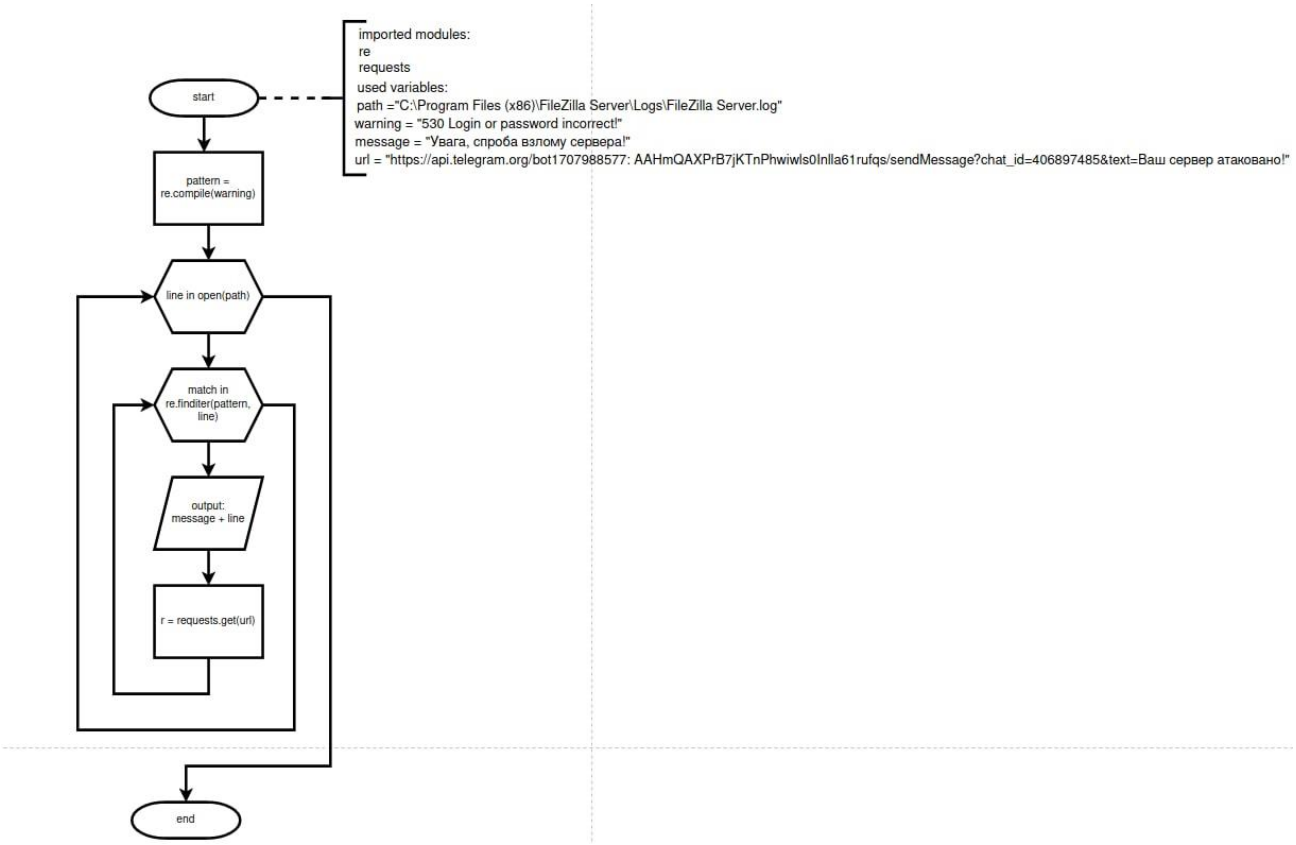

Рис. 3.12. Блок схема алгоритму програмного модулю

3.4.3.2. Програмний код модулю захисту інформаційних активів

Після створення алгоритму та визначення всіх аспектів роботи, розробляється програмний код. В даному випадку він виглядає наступним чином:

```
import re
import requests
pattern = re.compile("530 Login or password incorrect!")
for line in open("C:\Program Files (x86)\FileZilla Server\Logs\FileZilla Server.log"):
for match in re.finditer(pattern, line):
print("Увага, спроба взлому сервера!""\n' + line)
r = requests.get ("https://api.telegram.org/bot1707988577: AAHmQAXPrB7jKTnPhwiwls0Inlla61rufqs/sendMessage?chat_id=406897485&text=Ваш сервер атаковано!")
```

Завдяки використанню мови Python, код вийшов коротким, лаконічним і водночас здатним виконувати усі поставлені перед нами задачі. Суть даного програмного коду полягає в аналізі файлу з логами, який записує використаний у роботі FTP сервер. Виконується пошук і фіксація певного роду помилки в текстовому форматі.

Після знаходження сповіщення про невірний логін чи пароль (що є потенційною причиною взлому) у консолі програмного середовища, з'явиться запис про несанкціонований доступ до серверу. Також у сповіщенні буде прописано дату, час, запис помилки та інші дані про складену ситуацію.

На даному етапі можна було б завершити виконання даного програмного продукту. Але в такому разі можливості в роботі цього продукту будуть обмежені консоллю середовища. Згідно всіх міркувань доцільним стає створення глобальнішого сповіщення для покращеного виявлення загрози вторгнень.

Тому даний програмний модуль вмістив ще один рядок із важливим функціоналом. Було розроблено можливість отримувати сповіщення у месенджері Telegram через спеціального бота. Даний месенджер наразі є популярним у всьому світі, в тому числі, і в Україні. Це дозволить ефективно виявляти загрози, маючи відповідне повідомлення про уразливості. Даний сервіс можна налаштувати на будь-якого користувача, тобто адміністратор може тестувати систему в програмному середовищі і дублювати результати роботи як собі, так і наприклад керівнику відділу чи підприємства.

Ці дії точно можуть стати в нагоді для захисту інформаційних активів тієї чи іншої ланки, бо саме обізнаність про небезпеку стає приводом до готовності протистояти та захищати інформаційну систему. Можна детальніше розібрати, який функціонал містить створена програма та розкласти на множники всі структури даного програмного забезпечення.

import re – дана частина виконує імпортування до нашого середовища (в якому виконується створений нами продукт) регулярних виразів. Регулярні вирази використовуються в мові Python для:

- заміни, пошуку чи вилучення символів;
- швидкого виконання незвичайних операцій;
- розрізу та поділу рядків на підрядки;
- визначення необхідного формату, наприклад, email-адреси, телефонного номеру, тощо.

Найчастіше у мові Python використовують такі регулярні вирази як: re.match(), re.compile та re.search(), до одного з яких звертається створений програмний код.

import requests – даний рядок імпортує до середовища бібліотеку запитів. Дана бібліотека є стандартним інструментарієм мови Python при складанні запитів HTTP. Використання цього імпорту надало можливість отримати доступ до запиту GET, який став в нагоді для роботи з API. Таким чином, це полегшить трудомісткість процесу у створенні запиту та виконає взаємодію із певними службами та даними використаних у додатку.

pattern = re.compile("530 Login or password incorrect!") – цей рядок коду виконує функціонал збереження патерну (шаблону), компілюючи об’єкт регулярного виразу, для майбутнього використання. Тобто, створюється об’єкт із заданого тексту, який надалі буде використовуватися для пошуку слідів вторгнень до серверу. Помилка «530 логін чи пароль є некоректними».

for line in open("C:\Program Files (x86)\FileZilla Server\Logs\FileZilla Server.log"). Заданий рядок це відкриття циклу в програмі, який аналізуватиме кожну лінію (рядок), відкривши файл по заданому користувачем шляху. В даному випадку шлях вказує на розміщення файлу логування FTP-серверу.

for match in re.finditer(pattern, line). Це рядок вкладеного циклу до попереднього, який виконуватиме пошук в масиві всіх даних по кожному рядку, використовуючи створений вище патерн для порівняння.

Якщо заданий шаблон буде збігатися з якимось із рядків, то виконається тіло циклу і буде запущено даний рядок: print("Увага, спроба взлому сервера!""\n' + line). Він виводить сповіщення на консоль про потенційний

взлом сервера та вказує всі дані, які стосуються певної сесійної ланки на сервері.

3.4.4. Удосконалення програмного модулю

Не враховуючи уже очікувану удосконалену форму роботи програмного продукту, в зрівнянні зі стандартним засобом ОС Windows (журналом інцидентів безпеки), дана програма пройшла ще один етап модернізації. Крім рядка виведення сповіщення у консоль середовища з результатом виконання програмного модулю, виконується наступний рядок коду (рис. 3.13).

```
r = requests.get ("https://api.telegram.org/bot1707988577:AAHmQAXPrB7jKTnPhwiwls0Inlla61rufqs/sendMessage?chat_id=406897485&text=Ваш сервер атаковано!")
```

Рис. 3.13. Програмний код відповідальний за зв’язок із месенджером

Створюється змінна, аргументом якої є об’єкт, що належить бібліотеці запитів і виконується запит типу GET. Що з себе представляє цей запит в даному випадку? Його структура описує звернення до API месенджера Telegram – жовтий колір. А саме звертається до бота з ідентифікатором виділеним сірим кольором та його ключа, виділеного синім кольором.

Також в структурі запиту є частина виділена червоним кольором, що виконує відправлення повідомлення, яке буде нашим сповіщенням певному користувачу зі вказанням ідентифікатора отримувача та тексту повідомлення.

Що стосується налаштування та створення чат-боту для Telegram, то ця задача стає ефективною у виконанні. Адже по даному месенджеру є безліч корисної інформації та документації стосовно розробки нових можливостей. Можна проаналізувати в наступних рядках як був створений використаний в програмному коді чат-бот і який результат його виконання.

Для створення нового чат-боту в Telegram потрібно звернутися до BotFather (рис. 3.14).

Рис. 3.14. Звернення до BotFather в Telegram

Цей бот представляє собою предка у створенні будь-якої реалізації даного месенджера. Для початку виконується його запуск, після чого з'являється список команд доступних для виконання. Серед можливих дій обирається команда /newbot, яка результується підтвердженням про створення нового боту та запитом на присвоєння його назви (рис. 3.15).

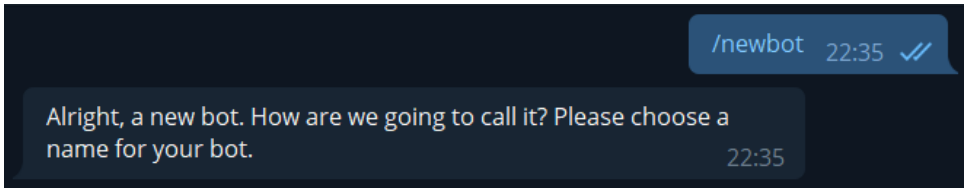

Рис. 3.15. Виконання команди для створення нового боту

Прописуючи назву, наступним кроком важливо зазначити ім'я користувача (логін) даної реалізації. Після успішного виконання заданих інструкцій, предок всіх чат-ботів відправляє повідомлення про успішне створення нащадку. Також надається ідентифікатор та ключ щойно створеного боту (рис. 3.16).

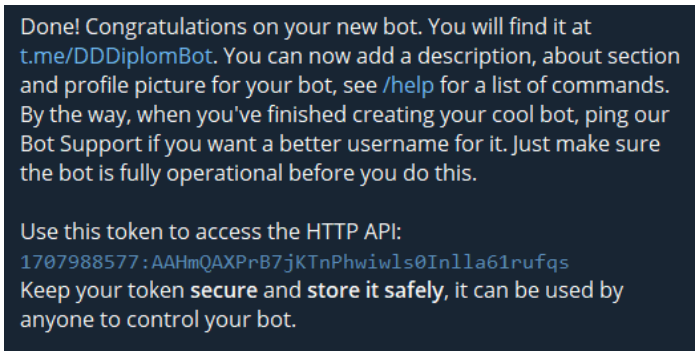

Рис. 3.16. Успішне створення нащадку та отримання його атрибутів

Наступним етапом є визначення, через спеціальний інструментарій месенджеру, ID користувача системи (рис. 3.17).

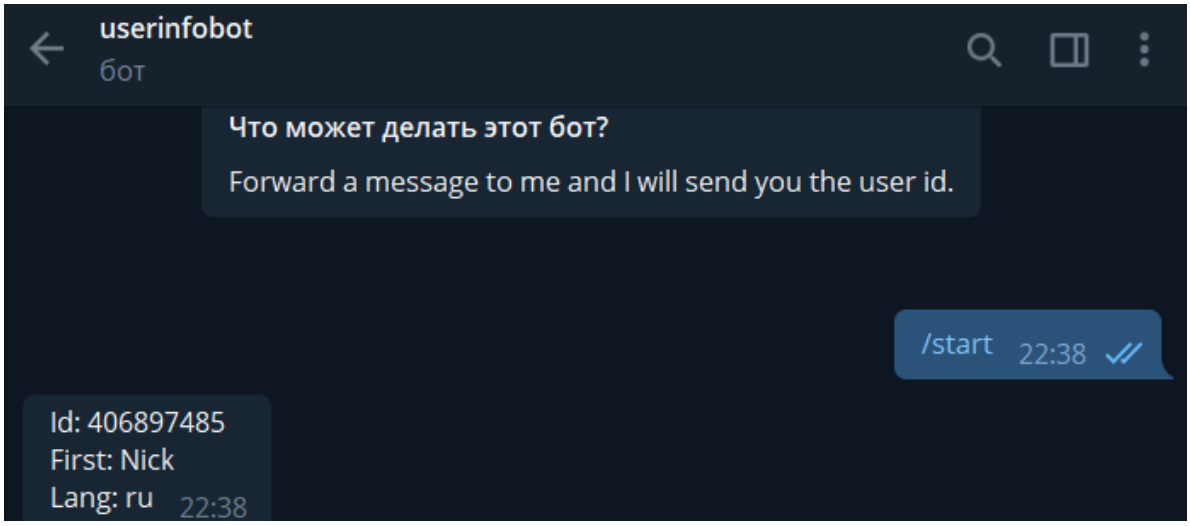

Рис. 3.17. Визначення ідентифікатора користувача

Використання отриманих даних відбувається у створеному програмному продукті для визначення отримувача сповіщення. Саме цей ідентифікатор користувача та ідентифікатор і ключ, отримані для боту (рис. 3.17), було використано при реалізації GET запиту. У підсумку виконаних дій, створений чат-бот готовий виконувати подальші вказівки програмного коду та має наступний вигляд (рис. 3.18).

Рис. 3.18. Вигляд створеного чат-боту

3.4.5. Підсумкові результати виконання програмного модулю

В попередніх частинах було розглянуто початковий етап розробки програмного модулю та описано всі основні моменти даного додатку. На даному етапі можна перейти до завершення аналізу зробленої роботи та демонстрації в дії можливостей розробленого програмного модулю.

На рисунку 3.11 відображений список логів, що повідомляють про потенційний несанкціонований доступ до серверу. Програмний код уже проаналізовано і визначено, що виконує кожен рядок та кожна ланка. Доцільно авторизувати обліковий запис користувача месенджера Telegram і виконати запуск програми (рис. 3.19).

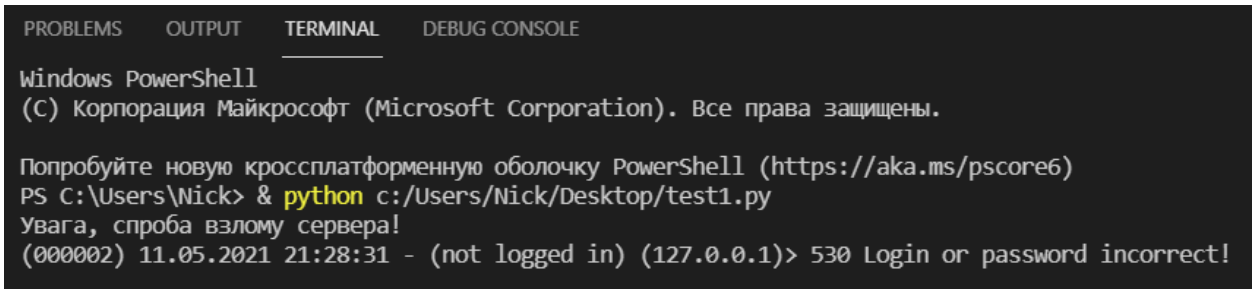

Рис. 3.19. Результат виконання програми у консолі програмного середовища

Одразу після запуску створеного модулю, в терміналі програмного середовища з’явилося відповідне повідомлення. У файлі логування сервера FTP виявлено некоректну спробу авторизації та повідомлено про можливу спробу взлому сервера. Також наочно видно всю потрібну інформацію для аналізу виникнення події та нейтралізації можливої загрози.

На підставі отриманих даних, визначивши потенційну загрозу атаки на сервер, розроблений чат-бот Telegram, який моментально із запуском програмного модулю, відсилає сповіщення вказаному при розробці користувачу (рис. 3.20).

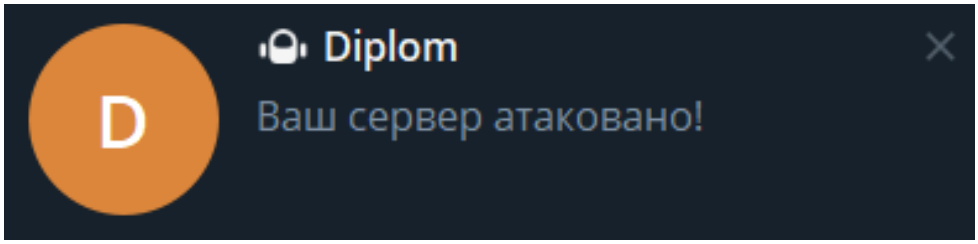

Рис. 3.20. Сповіщення месенджера на ПК

Так як телеграм використовується на різних пристроях, повідомлення буде отримано на кожному авторизованому гаджеті одночасно. Тому не помітити спробу несанкціонованого доступу буде в край складно (рис. 3.21).

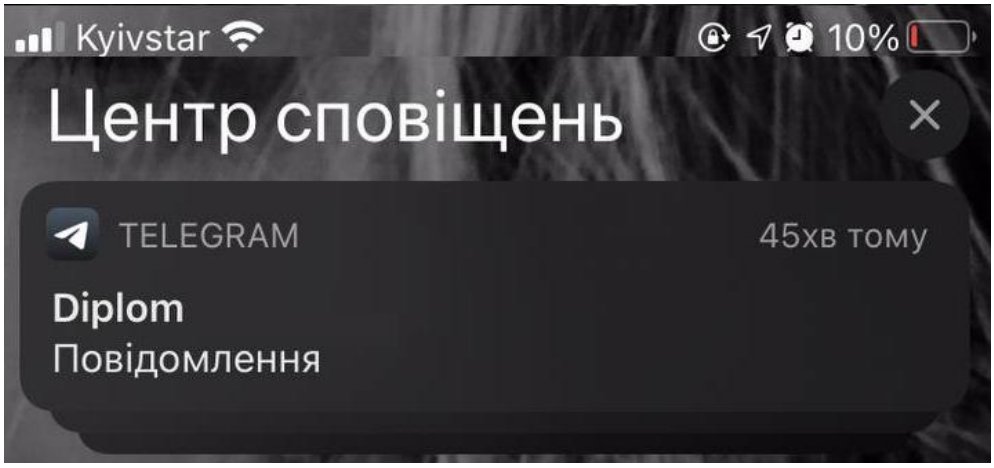

Рис. 3.21. Сповіщення месенджера на смартфоні

Перелік виконаних дій можна підсумувати, як вдалою реалізацією програмного модулю. Створений код має певні переваги над функціоналом стандартного журналу інцидентів операційної системи Windows. Подія потенційного взлому сервера відображена в логах файлу, а програмний модуль навчений відрізняти цей інцидент та не тільки виявити, а й сповістити в консоль та смс-повідомленням до заданого месенджеру. Даний випадок втручання не зафіксовано операційною системою у часових рамках проведення атак (рис. 3.22).

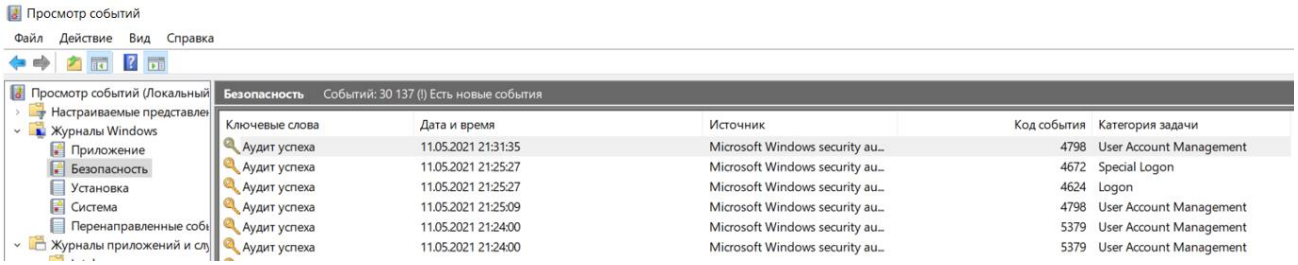

Рис. 3.22. Журнал подій безпеки і відсутність подій взлому

Тому створений програмний модуль можна вважати удосконаленим, враховуючи його додаткові медіатори виявлення несанкціонованого доступу, порівняно із системним журналом. Також даний код, удосконалений системою оповіщення користувача, що є вагомою перевагою для швидкого реагування на надзвичайні ситуації.

3.5. Висновки до третього розділу

В даному розділі було:

- проаналізовано програми виявлення проникнень і визначено, що саме використання, удосконалення та створення нових систем безпеки є вагомим аспектом захищеності інформаційних активів;
- проаналізовано роботу стандартного аудиту інцидентів безпеки, що входить до складу журналу подій ОС Windows, та виявлено недосконалу форму фіксації епізодів. Усіх підозрілих та потенційно небезпечних дій стандартний засіб не помічає, користувач може не дізнатися про активні та минулі проникнення;
- проаналізовано використовуване програмне забезпечення у процесі дослідження, проектування та розробки модулю захисту активів;
- розроблено програмний код для виявлення взлому.

Процес розробки базувався на аналізі інцидентів, записаних до серверного лог-файлу. Наступним етапом було імітовано спробу підбору ідентифікаційних даних для несанкціонованого доступу. Після чого,

визначивши всі аспекти даного процесу, були розроблені: блок-схема алгоритму дій та програмний код.

Даний модуль можна вважати удосконаленим, так як надається функціонал виявлення та сповіщення проникнень, чим не може відзначитись журнал подій ОС Windows. Також був проведений аналіз результатів виконання програмного модулю і виявлено, що розроблений тип сповіщення є універсальним і модернізованим під сучасне використання.

ВИСНОВКИ

В результаті дипломної роботи на тему: «Удосконалений програмний модуль для захисту інформаційних активів» було:

- проаналізовано загрози інформаційних активів на основі нормативно-правової бази України, що дало можливість побудувати моделі порушника та загроз і визначити ризики для інформаційних активів;
- побудувано модель порушника і надано класифікацію ризиків загроз для інформаційних активів, що дало змогу визначити відсутність в журналі подій відомостей про несанкціонований доступ зловмисників при стандартній програмі;
- розроблено програмний продукт на мові програмування Python в середовищі Visual Studio Code для захисту інформаційних активів, що дало можливість визначати більшу кількість спроб проникнення в інформаційні активи, а також оперативно реагувати на позаштатні події за рахунок додатково розробленої програми сповіщення на мобільний телефон.

Даний програмний модуль є удосконаленим в зрівнянні зі стандартним засобом операційної системи Windows, а також модернізований системою сповіщень через месенджер.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Закон України [Електронний ресурс]: Закон України «Про інформацію» – 2006. – Режим доступу: World Wide Web. – URL: http://alex-ua.com/lawdoc/zakon2.html.

2. Лекції.нет [Електронний ресурс]: Поняття про модель загроз та модель порушника – 2014. – Режим доступу: World Wide Web. – URL: https://lektsii.net/1-62960.html.

3. Закон України [Електронний ресурс]: Про захист інформації в інформаційно-телекомунікаційних системах – 1994. – Режим доступу: World Wide Web. – URL: https://zakon.rada.gov.ua/laws/show/80/94-вр#Text.

4. Інформаційна безпека [Електронний ресурс]: Класифікація загроз інформаційній безпеці – Режим доступу: World Wide Web. – URL: https://sites.google.com/site/informacijnabezpeka15/zagrozi-informacijnij-bezpeci/klasifikacia-zagroz-informacijnij-bezpeci.

5. Інформаційні технології [Електронний ресурс]: Методи захисту системи управління інформаційною безпекою – 2016. – Режим доступу: World Wide Web. – URL: https://www.assistem.kiev.ua/doc/dstu_ISO-IEC_27001_2015.pdf.

6. Лекції.нет [Електронний ресурс]: Приклади дій порушника та типові атаки на інформаційній ресурс – 2014. – Режим доступу: World Wide Web. – URL: https://lektsii.net/1-62961.html.

7. Нормативний документ системи технічного захисту інформації [Електронний ресурс]: Термінологія в галузі захисту інформації в комп’ютерних системах від несанкціонованого доступу – 1999. – Режим доступу: World Wide Web. – URL: https://tzi.com.ua/downloads/1.1-003-99.pdf.

8. Закон України [Електронний ресурс]: Про запобігання та протидію легалізації (відмиванню) доходів, одержаних злочинним шляхом, фінансуванню тероризму та фінансуванню розповсюдження зброї масового знищення: Закон України від 14.10.2014 № 1702-VII // Відомості Верховної Ради України. – 2014.

– № 50-51. – Ст. 2057 – Режим доступу: World Wide Web. – URL: http://finmonitoring.in.ua/wp-content/uploads/2018/12/terminologichnij-slovnik_finmonitoring.pdf.

9. Посібник з європейського права у сфері захисту персональних даних. – К.: К.І.С., 2015. – 216 с.

10. Нормативний документ системи технічного захисту інформації [Електронний ресурс]: Властивості інформації і загрози – 2014. – Режим доступу: World Wide Web. – URL: https://tzi.com.ua/downloads/1.1-003-99.pdf.

11. Системи виявлення НСД до інформаційних ресурсів [Електронний ресурс]: Атака на інформаційну систему – Режим доступу: World Wide Web. – URL: http://www.rusnauka.com/14_NPRT_2010/Informatica/66714.doc.htm.

12. Керування ризиком [Електронний ресурс]: Методи загального оцінювання ризику – Режим доступу: World Wide Web. – URL: https://khoda.gov.ua/image/catalog/files/dstu%2031010.pdf.

13. Оцінка інформаційних ризиків [Електронний ресурс]: Переваги та недоліки методів оцінки інформаційних ризиків – Режим доступу: World Wide Web. – URL: http://www.rusnauka.com/21_SEN_2014/Informatica/4_174674.doc.htm.

14. Політика інформаційної безпеки об’єкта [Електронний ресурс]: Оцінювання ризиків і загроз – Режим доступу: World Wide Web. – URL: https://ela.kpi.ua/bitstream/123456789/8581/1/24_p23.pdf.

15. Міжнародний стандарт [Електронний ресурс]: Методи і засоби забезпечення безпеки. ISO/IEC 27005 – Режим доступу: World Wide Web. – URL: https://exebit.files.wordpress.com/2013/11/iso-27005-2011-ru-v1.pdf.

16. Інформаційна безпека [Електронний ресурс]: Класифікація інформаційних активів – Режим доступу: World Wide Web. – URL: https://ipiskunov.blogspot.com/2016/07/blog-post_0.html.

17. Хорев П. Б. Методы и средства защиты информации в компьютерных системах, М., Издательский центр «Академия», 2005, 256 с.

18. Регулярные выражения в Python [Електронний ресурс]: Регулярные выражения в Python: теория и практика – Режим доступу: World Wide Web. – URL: https://tproger.ru/translations/regular-expression-python.

19. Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу. НД ТЗІ 2.5-004-99. Затверджено наказом Департаменту спеціальних телекомунікаційних систем та захисту інформації Служби безпеки України від 28.04.1999 № 22 із змінами згідно наказу Адміністрації Держспецзв'язку від 28.12.2012 № 806.

20. Романюков М.Г. Критерії оцінки ймовірності витоку інформації через технічні канали. — 2015. Том 5, №3. — С. 240-248.

21. Python и API [Електронний ресурс]: Python и API: превосходное комбо для автоматизации работы с публичными данными – Режим доступу: World Wide Web. – URL: https://proglib.io/p/python-i-api-prevoshodnoe-kombo-dlya-avtomatizacii-raboty-s-publichnymi-dannymi-2021-02-26.

22. BotCreators [Електронний ресурс]: Как создать своего бота в BotFather? – Режим доступу: World Wide Web. – URL: https://botcreators.ru/blog/kak-sozdat-svoego-bota-v-botfather.