Міністерство освіти і науки України
Департамент освіти і науки виконавчого органу Київської міської ради
(Київської міської державної адміністрації)
Комунальний позашкільний навчальний заклад «Київська Мала академія наук учнівської молоді»

Відділення: комп’ютерні науки
Секція: безпека інформаційних та телекомунікаційних систем

СТВОРЕННЯ АНТИВІРУСНОЇ ПРОГРАМИ ДЛЯ ПРОТИДІЇ ВІРУСАМ-ШИФРУВАЛЬНИКАМ

Роботу виконав:
Жайворонок Дмитро Вячеславович,
учень 10-а класу Ліцею інформаційних технологій № 79 Печерського району м. Києва

Науковий керівник:
Китайцев Олег Миколайович,
завідувач кафедри інформаційних технологій Ліцею інформаційних технологій № 79 Печерського району м. Києва

Київ – 2021

Комунальний позашкільний навчальний заклад
«Київська Мала академія наук учнівської молоді»

Анотація

Жайворонок Дмитро Вячеславович,
учень 10 класу,
ЛІТ №79 м.Києва

Науковий керівник: Китайцев Олег Миколайович, завідувач кафедри інформаційних технологій ЛІТ № 79

СТВОРЕННЯ АНТИВІРУСНОЇ ПРОГРАМИ ДЛЯ ПРОТИДІЇ ВІРУСАМ-ШИФРУВАЛЬНИКАМ

Дослідницьку роботу присвячено розробці та детальному аналізу методів захисту даних та інформації, а саме створенню антивірусної програми для виявлення та знешкодження вірусів-шифрувальників.

Досліджено історію виникнення вірусів-шифрувальників, їх будову, принципи роботи та поширення.

Проаналізовано існуючі методи захисту від вірусів-шифрувальників.

Досліджено можливості мови програмування Python для створення антивірусних програм. Створено прототип антивірусної програми для виявлення та знешкодження вірусів-шифрувальників. Створено власну модель віруса-шифрувальника для експериментальних досліджень.

Проведено експериментальне дослідження ефективності використання створеного прототипу антивірусної програми.

Проаналізовано та узагальнено алгоритми роботи антивірусної програми та результати експериментальних досліджень.

Ключові слова: антивірусна програма, вірус, вірус-шифрувальник, віртуальна машина, Avast, BadUSB, Fishing-розсилки, Python, Torent-трекери, WindowsDefender.

ЗМІСТ

ВСТУП

Сьогодні, в еру цифрових обчислювальних машин, коли більшість інформації зберігається у кіберпросторі дуже гостро постала проблема її захисту, адже заволодівши конфіденційними даними можна не лише зашкодити людині або компаній, а навіть поставити під загрозу її життя. Інтернет – своєрідне відкрите сховище цієї інформації, й саме це сприяло дуже швидкому поширенню комп'ютерних вірусів. Зловмисники використовують їх для пошкодження даних з зловмисним наміром, або наміром отримання власної матеріальної вигоди. Одними з найпоширеніших на сьогоднішній момент є віруси-шифрувальники, вони шифрують всю дані на жорсткому диску користувача й для повернення доступу до них вимагають кошти. За даними інтерактивної карти кіберзагроз (CYBERTHREAT REAL-TIME MAP, https://cybermap.kaspersky.com/en) у світі щосекунди відбувається близько 10 000 атак з використанням вірусів-шифрувальників. За даними ЗМІ за останній рік через атаки вірусів-шифрувальників постраждали майже всі країни світу, збитки становлять ~ ₴100 млрд, а в Україні за останній рік зафіксовано близько 500 000 випадків кібератак через віруси-шифрувальники. Такі показники доводять, що ризик зараження та втрати персональних даних дуже високий та постійно зростає. Саме це доводить, що створення методів протидії вірусам-шифрувальникам дуже актуальна тема сьогодні. Тому тема дослідницької роботи наразі є актуальною і доцільною для дослідження.

Об'єкт дослідження – комп'ютерні системи.

Предмет дослідження – віруси-шифрувальники та антивірусні програми.

Мета дослідницької роботи: створення антивірусного програмного продукту для ефективної боротьби з вірусами-шифрувальниками на основі проведення аналізу існуючих методів захисту даних та інформації та дослідженні вірусів-шифрувальників, їх будови, принципів роботи та поширення;

Відповідно до мети визначено такі **завдання** проведення дослідження:

- проаналізувати науково-методичну літературу, пов'язану з темою вірусів-шифрувальників;

- теоретично проаналізувати та практично перевірити методи захисту інформації від зараження вірусами-шифрувальниками;
- розробити антивірусну програму на основі проведених досліджень для виявлення та знешкодження вірусів-шифрувальників;
- провести апробацію розробленого прототипу антивірусної програми.

Використані методи досліджень: аналіз науково-методичної літератури з зазначеної проблематики, експериментальна перевірка розроблених алгоритмів захисту, які використовуються у розробленому прототипі антивірусного програмного забезпечення.

Практична значущість полягає у створенні антивірусної програми, що здатна ідентифікувати та знешкоджувати до 90% файлів, заражених вірусами-шифрувальниками. Створено програму, яка моделює роботу віруса-шифрувальника та може використовуватись для проведення досліджень, вдосконалення методів захисту та антивірусних програм.

РОЗДІЛ 1
ІСТОРІЯ РОЗВИТКУ ВІРУСІВ-ШИФРУВАЛЬНИКІВ

1.1. Перший вірус-вимагач (шифрувальник)

Віруси-шифрувальники відомі ще з середини 1980-х років. Спочатку вони були націлені на програмне забезпечення та інформацію, розміщену на комп'ютерах, які працювали під управлінням операційних систем MS-DOS і Windows.

Зараз же вони загрожують користувачам по всьому світу. Вірусів-шифрувальників з'являється все більше й вони стають все небезпечнішими, а починалося все зі звичайного бажання помститися за піратство.

Вимогу викупу містив перший же вірус, що викликав глобальну епідемію серед персональних комп'ютерів. Його історія почалася в 1986 році в невеликому комп'ютерному магазині в пакистанському Лахорі.

Керівники магазину брати Фарук Алві , 17-річний Басить і 24-річний Амджад - розробили програму для відстеження стану серцево-судинної системи. Програму відразу вкрали пірати, в результаті брати недоодержували прибуток.

Тоді брати створили Brain («Мозок») – для захисту інтелектуальної власності, як пояснили вони через два роки журналу Time. Цей вірус мав атакувати комп'ютери зі встановленою нелегальною копією їхньої медичної програми.

Вірус сповільнював роботу жорсткого диску й (за деякими джерелами) шифрував частину даних користувача будь-якого комп'ютера, куди вставляли заражені дискети. Він швидко вийшов з-під контролю і атакував університети і компанії, де ніколи не чули про програму братів.

Своє дітище брати присвятили «мільйонам вірусів, яких з нами більше немає», що нагадувало пророцтво – комп'ютерні віруси ще не були настільки масовим явищем.

Вірус «пропонував» зателефонувати в той самий магазинчик в Пакистані, щоб отримати «вакцину». Брати почали отримувати дзвінки з усього світу.

Вони вимкнули телефони і, не отримавши жодного покарання, продовжили займатися бізнесом – вже легальним: зараз Фаруки Алві володіють великим пакистанським інтернет-провайдером Brain Net.[10]

1.2. Розвиток вірусів-шифрувальників

З моменту створення першого вірусу-шифрувальника пройшло багато часу й змінилося майже все, від будови до принципів роботи.

Спочатку зловмисники не задумувалися над маскуванням та швидкістю роботи вірусу, тому що на той момент ще не існувало достатньо досконалої системи для протидії вірусам-шифрувальникам. Але з часом почали з'являтися програми, які були здатні захистити користувача від атак вірусів подібного типу, й зловмисникам довелося задуматися над тим, як зробити так, щоб їхня розробка обходила методи захисту та працювала як їм завгодно.

Саме через це більшість вірусів-шифрувальників почали проходити процедуру обфускації – видозміни програмного коду, що допомогло обійти захист існуючих на той момент антивірусних систем, але це продовжувалося не довго. Вже через декілька місяців розробники антивірусних програм навчили свої розробки ідентифікувати навіть обфусковані віруси-шифрувальники. Але це не зупинило хакерів, вони почали переписувати віруси на інші мови програмування, зокрема на C++, але це не дало бажаного результату, тоді вони почали використовувати різні формати кодування, що принесло бажаний результат, але ненадовго.

У середині 90-х років минулого століття після появи таких мов програмування, як Python та Java перед розробниками антивірусних програм постала нова проблема, їхні програми не могли розпізнавати віруси-шифрувальники написані цими мовами, але на щастя інтернет ще не був масово поширений і вони встигли вдосконалити свої розробки й вберегти користувачів від масової кіберепідемії.

1.3. Віруси-шифрувальники сьогодні

На сьогоднішній день у світі є дуже велика кількість вірусів-шифрувальників, від звичайних «іграшок» створених заради втіхи, до серйозних, які здатні завдати великої шкоди користувачам. Кожну секунду у світі з'являється близько 5 нових вірусів-шифрувальників та проходить більше 10 000 атак з їх використанням (CYBERTHREAT REAL-TIME MAP, https://cybermap.kaspersky.com/en). В порівнянні зі своїми попередниками, сучасні віруси становлять високу загрозу, тому що здатні

зашифрувати всі дані жорсткого диску, а також й взагалі знищити їх, наприклад при відмові сплатити кошти.

На відміну від вірусів-шифрувальників, які тільки починали шлях вірусів даного типу, новітні розробки важко ідентифікуються антивірусними системами, тому що зловмисники дуже добре навчилися їх маскувати, а саме: піддають їх декільком етапам обфускації, декілька разів проводять операцію шифрування різними стандартами шифрування.

Найпопулярнішими вірусами даного типу є:

- WannaCry – більше 500 000 заражених комп’ютерів.
- Petya – 300 000 ~ 500 000 заражених комп’ютерів.
- Bad Rabbit – 50 000 ~ 450 000 заражених комп’ютерів.

РОЗДІЛ 2
БУДОВА ТА МЕТОДИ ПОШИРЕННЯ ВІРУСІВ-ШИФРУВАЛЬНИКІВ

2.1. Принцип роботи вірусів-шифрувальників

Вірус-шифрувальник – вірус, який шифрує дані на жорсткому диску комп'ютера-жертви, а також перезаписує і шифрує головний завантажувальний запис (MBR) – дані необхідні для завантаження операційної системи. В результаті файли, що зберігаються на комп'ютері стають недоступними.

Після зараження комп'ютера вірус скачує з інтернету шифрувальник й намагається пошкодити ним частину жорсткого диску з даними, якщо все вдається, то вірус виводить «Синій екран смерті». Після перезавантаження ПК випливає повідомлення про перевірку жорсткого диску з проханням не вимикати живлення. Таким чином вірус видає себе за програму перевірки жорсткого диску, а насправді шифрує файли з певним розширенням. В кінці процесу з'являється повідомлення про блокування комп'ютера й інформація про те, як повернути дані.

2.2. Відмінність роботи звичайних вірусів від вірусів-шифрувальників

Зазвичай всі віруси працюють за однаковим принципом, спочатку потрапляння до системи, потім виконання головного файлу, який у свою чергу виконує другорядні.

Але віруси-шифрувальники працюють за видозміненим принципом, спочатку вони потрапляють до системи, потім запускається головний файл, який скачує сам шифрувальник з віддаленого серверу (може бути передбачений варіант коли сам шифрувальник вже знаходиться у програмному коді вірусу), та запускає його, а другорядні файли слідкують за процесом виконання та у випадку виникнення якоїсь проблеми можуть або самознищити вірус, або знищити всю інформацію на накопичувачах жертви, а також пошкодити інформацію необхідну для роботи системи, що призводить до неможливості використовувати комп'ютер до моменту перевстановлення системи або відновлення головного завантажувального запису.

2.3. Будова вірусів-шифрувальників

Віруси-шифрувальники складаються з головного та другорядних файлів.

Після потрапляння в вірусу до системи на виконання ставиться його **головний файл**, він може мати різні назви, але зазвичай – .text. Його роль полягає в об'єднанні та виконанні всіх другорядних файлів, які є допоміжними. У головному файлі зазвичай знаходяться всі алгоритми, які відповідають за роботу та розповсюдження вірусу, й їх пошкодження може повністю нейтралізувати вірус.

Після виконання головним файлом **другорядних**, кожен з них починає виконувати свою роль. Роль другорядних файлів може бути будь-якою, від звичайного блокування клавіатури або вимкнення комп'ютеру до шифрування даних.

2.4. Загальні методи поширення

На сьогоднішній день віруси-шифрувальники стає все важче й важче поширювати, особливо у великих компаніях. Зазвичай працівники в таких закладів проходять ознайомлення з базовою кібергігієною декілька разів на рік, тому зловмисникам стає все важче передати та поширити вірусне ПЗ. На сьогоднішній момент є декілька найпопулярніших методів поширення вірусів: Fishing-розсилки, Torent-трекери та атаки типу BadUSB[9].

Fishing-розсилки є дуже популярною практикою у світі вірусного ПЗ. Для прикладу візьмемо звичайну пошту. Вам надходить повідомлення з проханням перейти за посиланням й пройти соціальне опитування. Після переходу користувач не помічає нічого підозрілого, але його комп'ютер вже знаходиться під владою вірусу. Через певний час відбувається виконання головного файлу й шифрування даних, а далі на вибір користувача, або плата за відновлення даних, або їх знищення.

Один із найпопулярніших методів поширення вірусів-шифрувальників з моменту утворення **Torent-трекерів**. Атака відбувається завдяки об'єднанню вірусного файлу зі звичайною грою, фільмом або іншим матеріалом, який публікується на подібних ресурсах. Після завантаження файлу користувач може не помітити нічого підозрілого й покористуватися встановленим матеріалом й навіть видалити його та забути, але саме вірус спочатку завантажився до системи, а потім вже файл, й просто вичікував слушного моменту для запуску.

Атаки типу **BadUSB** не є настільки поширеними, але не менш небезпечними. Для проведення атаки даного типу потрібен спеціальний пристрій, який вільно продається в таких великих інтернет-магазинах, як: AliExpress, Joom та Ebay. Зазвичай даний пристрій маскується під звичайний дріт передачі даних (MicroUSB – USB чи ін.), або флеш накопичувач. Головним елементом такого пристрою є мікроконтролер ATmega32u4, на нього записується програмний код (payload), який буде виконано після під'єднання до комп'ютера-жертви. Після виконання програмного коду, який завантажить та запустить вірус, дані будуть зашифровані. Приклад подібної атаки найліпше показаний у серіалі Mr.Robot, де сестра головного героя розсипала подібні пристрої біля відділку поліції задля отримання доступу до системи.

РОЗДІЛ 3
СТВОРЕННЯ ПРОТОТИПУ АНТИВІРУСНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1. Антивірусне ПЗ. Основні завдання та принципи роботи

Антивірусне програмне забезпечення – це програмний код, який «піклується» безпечним користуванням персональними комп'ютерами. Таке програмне забезпечення використовується на тисячах комп'ютерів та покликане виявляти комп'ютерні віруси та знешкоджувати їх.

На сьогоднішній день існує дві найпоширеніші методики створення антивірусного ПЗ:

- Створення й порівняння хеш-коду програми або її фрагментів з існуючим у базі даних.
- Сканування й відслідковування дій програм для виявлення шкідливого ПЗ.

Створення й порівняння хеш-коду програми або її фрагментів з існуючим у базі даних – більш популярний та практичний метод, полягає у створенні хеш-коду програми за допомогою спеціальних засобів та порівнянням його фрагментів з існуючим у базі даних, якщо фрагмент збігається на 90% - файл містить вірус.

Сканування й відслідковування дій програм для виявлення шкідливого ПЗ – менш популярний й більш складний метод, полягає у відтворенні взаємодії програми з іншими ресурсами та системою, при виявленні підозрілих поєднань проводиться більш детальний аналіз й перевірка з можливими взаємодіями взятими з бази даних.

3.2. Розробка кросплатформеного прототипу антивірусної програми

За основу створеного прототипу було взято метод «створення й порівняння хеш-коду програми або її фрагментів з існуючим у базі даних».

За допомогою бібліотек доступних у мові програмування Python[1,2,3,11] розроблена антивірусна програма створює хеш-код файлів у директорії, яка сканується та порівнює їх з існуючими у базі даних. При виявленні ідентичних хеш-кодів вона сигналізує користувачу про проблемний файл та надає йому вибір: видалити файл, або проігнорувати та додати до виключення «Додаток А».

Для розробки прототипу антивірусного ПЗ використовувалися технології: мова програмування Python 3.9[1,2,3], технологія UI (user interface – з англ. – користувацький інтерфейс), а також бази даних SQL[6,7] та SQlite 3.

Спочатку було створено консольний (працюючий без UI, лише завдяки використанню інтерфейсу CMD (командного рядку)) прототип, й після цього почалося створення бази даних з хеш кодом, для цього використовувалася спеціально розроблена програма, яка була додана до віртуальної машини задля забезпечення захищеності від вірусів основного комп'ютеру. Після вдалих тестувань та перевірок почалася розробка дизайну UI у програмах Figma та Adobe Photoshop. Після створення дизайну, було створено повністю працюючий UI опираючись на створені ескізи за допомогою відкритих бібліотек Tkinter та Pygame[3].

Після розробки антивірусне ПЗ було протестоване на можливість взаємодії з різними операційними системами, а саме: Windows, Linux, та Ubuntu. Завдяки використанню мови програмування Python, яка є кросплатформеною, антивірусне програмне забезпечення змогло взаємодіяти з усіма операційними системами, хоча результат й відрізнявся. Результати тестувань наведені у «Таблиця 1».

3.3. Створення програми, яка імітує роботу віруса-шифрувальника

Перед створенням програми, яка імітує роботу віруса-шифрувальника, для проведення тестувань розробленого антивірусного ПЗ було проаналізовано мови програмування для вибору найкращого варіанту. Серед варіантів розглядалися такі мови програмування, як: Python, C++, C# та Java. Після аналізу та порівняння найкращим варіантом виявилася мова програмування Python. «Таблиця 2»

Віруси-шифрувальники шифрують дані в певній директорії або диску, тож створена програма не буде виключенням, вона буде шифрувати дані у певній директорії за допомогою ключа, який буде також зашифрованим за допомогою декількох форматів шифрування, а саме: Шифр Цезаря та Base64.

Завдяки використанню мови програмування Python, розробка моделі буде значно полегшена тому, що є можливість використання великої кількості відкритих бібліотек, які значно полегшують роботу та беруть на себе такі завдання, як:

шифрування ключа, перебір файлів у директорії, а також полегшення взаємодії з файлами різних типів.

Після відтворення та тестування розробка показала високу швидкість шифрування даних (3 Гб за 1 хв), а також «прихованість» від антивірусів таких, як: Avast та WindowsDefender. Жоден з представлених антивірусів не зміг виявити розроблену програму, через відсутність схожих аналогів у базі даних антивірусу та проведення операції з накладанням різних типів шифрування по декілька разів. Для покращення «прихованості» було проведено обфускацію (приведення початкового коду або виконуваного програмного коду до вигляду, який зберігає його функціональність, але ускладнює аналіз, розуміння алгоритму роботи і модифікації при декомпіляції[8]), а потім зашифрована за допомогою Base64 «Додаток Б».

Створена програма, яка імітує роботу віруса-шифрувальника була протестована на таких операційних системах, як: Windows[5], Linux[4], а також Ubuntu. Найкращий результат шифрування даних був звичайно на ОС Windows, це пов’язано з недосконалістю ядра операційної системи, на Linux та Ubuntu швидкість шифрування погіршилася на 5%, це пов’язане через інакшою структурою ядра операційних систем, тому моделі вірусу-шифрувальника було складніше зашифрувати дані, але завдяки тому, що вона була написана за допомогою мови програмування Python, яка є кросплатформеною (може працювати на будь-якій операційній системі), все вдалося та дані були зашифровані.

3.4. Тестування та апробація антивірусного ПЗ

Антивірусна програма була протестована на предмет можливості виявляти віруси-шифрувальники. Тестування проводилося на віртуальній машині (віртуальному комп’ютері) зі встановленою операційною системою Windows. Під час перевірки вдалося виявити 80% заражених файлів. Результат був прекрасним порівняно з безкоштовними антивірусними програмами такими, як: Avast та WindowsDefender, але було прийняте рішення модернізувати базу даних з хеш-кодом, завдяки чому результат покращився до 90% та вдалося розпізнати велику

кількість вірусів, прихованих від таких популярних безкоштовних антивірусних програм, як: Avast та WindowsDefender.

Для більш точних результатів було проведене декілька перетестувань, але перевага залишалася на боці розробленого прототипу. Пізніше було визначено час сканування 40 файлів серед таких антивірусних систем, як: розроблений прототип, Avast та WindowsDefender «Додаток В».

ВИСНОВКИ

При виконанні дослідницької роботи «Створення антивірусної програми для протидії вірусам-шифрувальникам» розглянуто, вивчено та проаналізовано науково-методичну літературу з теми захисту інформації.

У ході виконання дослідницької роботи детально проаналізовано методи захисту інформації, розроблено антивірусну програму для виявлення та знешкодження вірусів-шифрувальників.

Досліджено історію виникнення вірусів-шифрувальників, їх будову, принципи роботи та поширення.

Проаналізовано існуючі методи захисту інформації від вірусів-шифрувальників.

Досліджено можливості мови програмування Python для створення антивірусних програм.

Розроблено прототип антивірусної програми для виявлення та знешкодження вірусів-шифрувальників.

Проведено експериментальне дослідження ефективності використання створеного прототипу антивірусної програми.

Розроблений прототип антивірусної програми здатен ідентифікувати до 90% файлів, заражених вірусами-шифрувальниками.

Проаналізовано та узагальнено алгоритми роботи антивірусної програми та результати експериментальних досліджень.

ВИКОРИСТАНІ ДЖЕРЕЛА

Список використаної літератури

1. Swaroop C.H., A Byte Of Python, ebshelf Inc, 2013. 110 с.
2. Бріґґс Джейсон Р., Python для дітей. Львів: Видавництво Старого Лева, 2017. 400 с.
3. Дж. Вандер Плас. Python для складних задач. Наука про дані та машинне навчання,. Київ:Питер, 2017. 576 с.
4. Рафаэль Херцог , Джим О'Горман , Мати Ахарони. Kali Linux від розробників. Київ: Питер, 2018. 320 с.
5. Энди Ратбон. Windows 10 для чайників. Київ:Діалектика. 2020. 480 с.
6. Vivian Siahaan, Rismon Hasiholan Sianipar. Learn SQLite with Python: Building Database-Driven Desktop. United States:SPARTA PUBLISHING. 2019. 895 с.
7. Anthony Molinaro. SQL Cookbook: Query Solutions and Techniques for Database Developers. O'Reilly Media. 2005. 877 с.
8. Вільна енциклопедія «Вікіпедія»: Обфускація URL:https://uk.wikipedia.org/wiki/Обфускація
9. Телеграм канали: Форум Кодебай, PentestersHome, HackSpace
10. Андрій Сошников. Історія вірусів-вимагачів Bbc News | Україна URL:https://www.bbc.com/ukrainian/features-40466315
11. YouTube канал Python Today URL:https://www.youtube.com/channel/UCrWWcscvUWaqdQJLQQGO6BA

ДОДАТКИ

Таблиця 1

Операційна система	Час с	Об'єм файлів для сканування Гб	Виявлені загрози (навмисно додані 2)
Windows	20	3	2
Linux	10	3	2
Ubuntu	10	3	2

Таблиця 2

Мова програмування	Велике ком'юніті	Легкість написання скриптів	Кросплатформеність	Open Source	Велика кількість відкритих бібліотек
Python	+	+	+	+	+
C++	+	-	+/-	-	+/-
C#	+	+/-	+/-	-	+/-
Java	-	-	+	+	+/-

Додаток А

1. Фрагмент програмного коду прототипу антивірусної програми

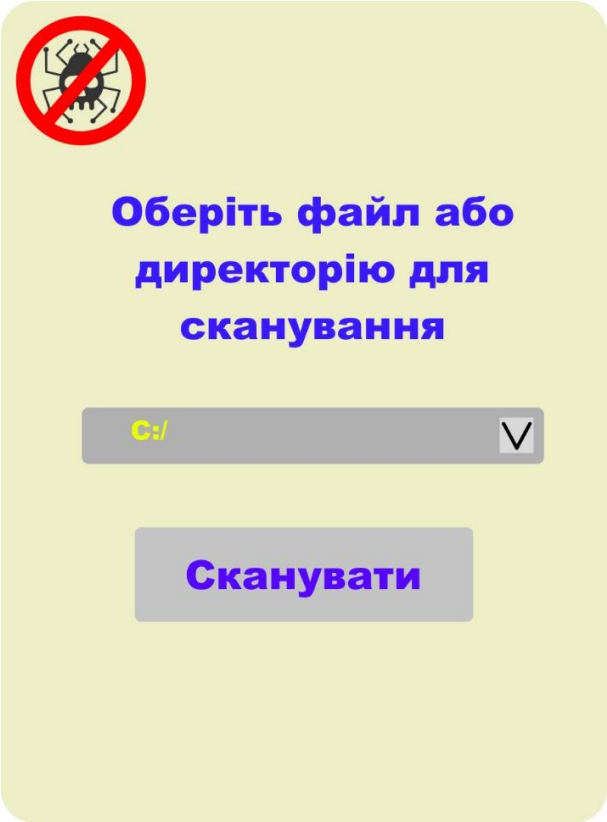
*/home/kali/Desktop/Antivirus-Python--master/AntiVirus-GUI_ENG.py - Mousepad

File Edit Search View Document Help

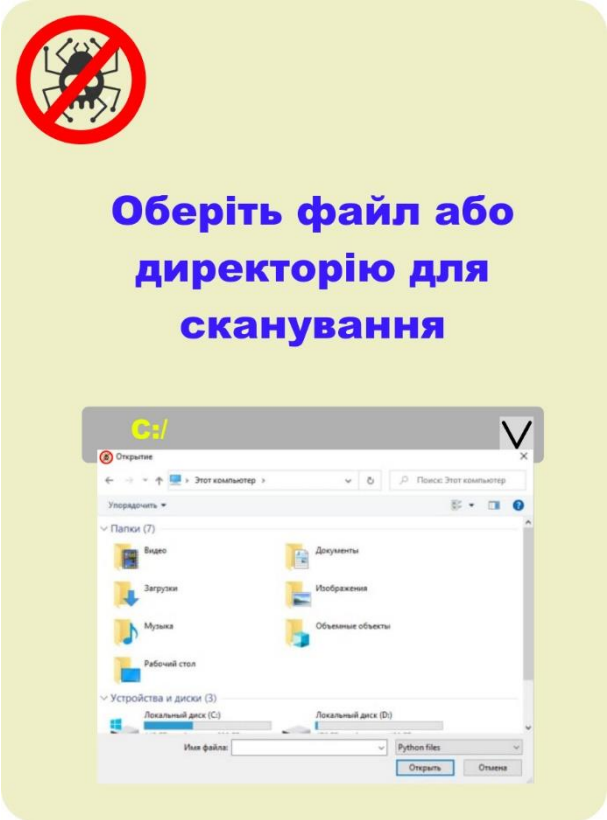
```
from threading import *
from tkinter import *
from tkinter.filedialog import askopenfilename
from tkinter.messagebox import showerror
import tkinter, tkinter.scrolledtext
import threading
import os
import sys
import urllib.request
import glob
import time
import hashlib
import socket
import subprocess
import quarantaene
import SystemFileScanner

os_name = sys.platform
verzeichnisse = []
files = []
partitionen = []
terminations = []


if "win" in os_name:
    if not os.path.exists("AntiVirus\\Quarantine\\"):
        os.makedirs("AntiVirus\\Quarantine\\")
    if not os.path.exists("AntiVirus\\sf\\"):
        os.makedirs("AntiVirus\\sf\\")
    if not os.path.exists("AntiVirus\\Large_Update_File\\"):
        os.makedirs("AntiVirus\\Large_Update_File")
    quarantine_folder = "AntiVirus\\Quarantine\\*"
    file_to_quarantine = "AntiVirus\\Quarantine\\"
    partitionen_folder = "AntiVirus\\sf\\sf.txt"
    links_current = "AntiVirus\\Large_Update_File\\links_current.txt"
    links_downloaded = "AntiVirus\\Large_Update_File\\links_downloaded.txt"
    large_signatures = "AntiVirus\\Large_Update_File\\signatures.txt"
    f = open(partitionen_folder, "a")
    f.close()
    f = open(links_current, "a")
    f.close()
    f = open(links_downloaded, "a")
    f.close()
    f = open(large_signatures, "a")
    f.close()
else:
```

2. UI розробленого прототипу

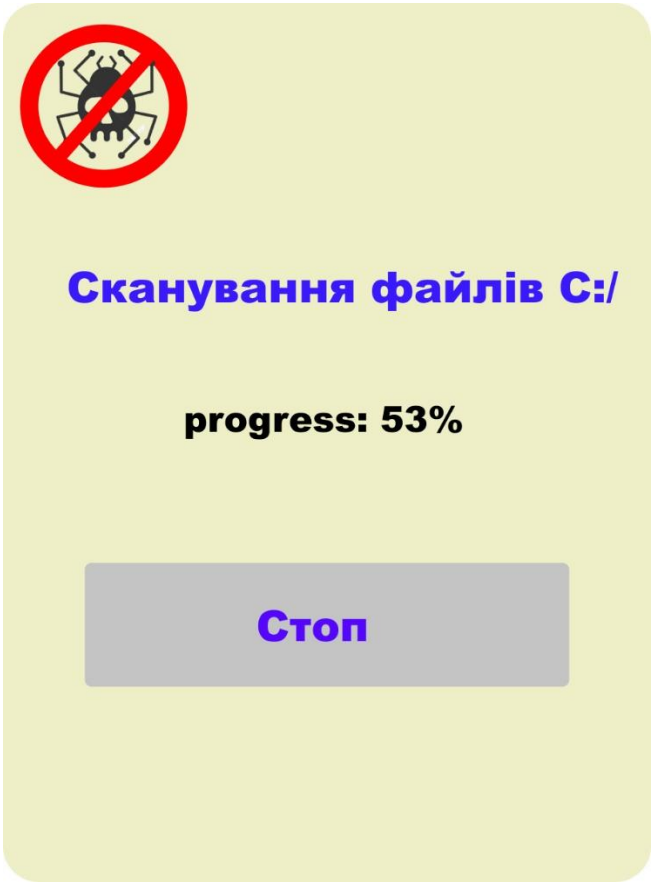
3. Початкове вікно прототипу



4. Вибір директорії для сканування



5. Процес сканування файлів

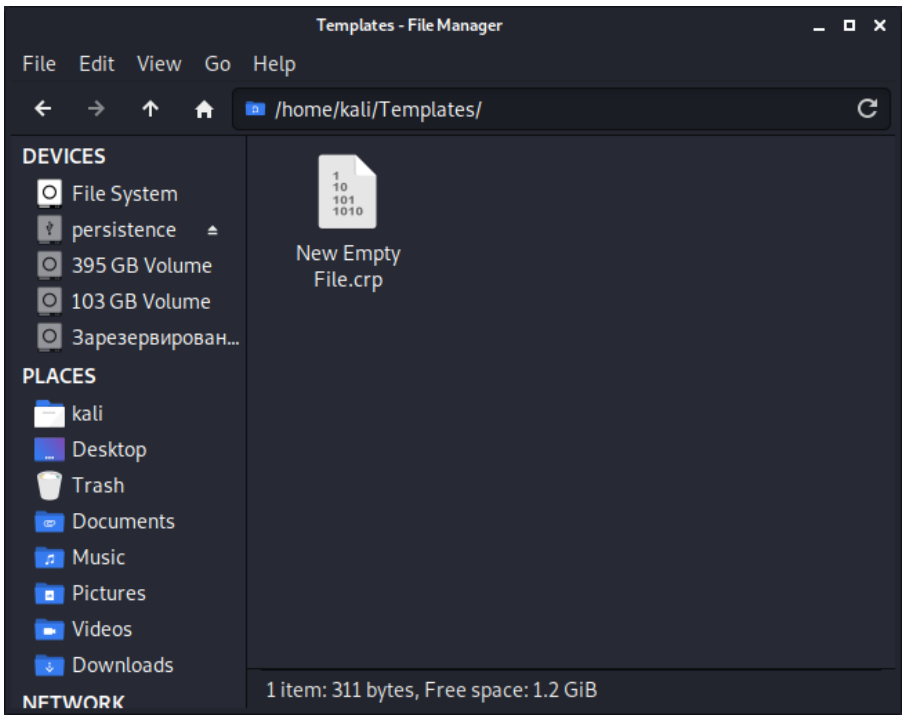


6. Повідомлення про загрозу

Додаток Б

1. Код програми, яка імітує роботу віруса-шифрувальника

2. Приклад роботи програми (тест на ОС Kali Linux)



3. Програмний код дешифрувальника

/home/kali/Documents/second var/1.py - Mousepad

File Edit Search View Document Help

```
import pyAesCrypt
import os
import sys

#decryptor

pas = "qwerty"

def decryption(file, password):

    buffer_size = 512 * 1024

    pyAesCrypt.decryptFile(
        str(file),
        str(os.path.splitext(file)[0]),
        password,
        buffer_size
    )

    print("[Файл '" + str(os.path.splitext(file)[0]) + "' дешифрован]")

    os.remove(file)

def walking_by_dirs(dir, password):

    for name in os.listdir(dir):
        path = os.path.join(dir, name)

        if os.path.isfile(path):
            try:
                decryption(path, password)
            except Exception as ex:
                print(ex)
        else:
            walking_by_dirs(path, password)


walking_by_dirs("/home/kali/Templates/", pas)
```

Додаток В

Порівняльна діаграма сканування 40 файлів