Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет *інфокомунікацій*
(повна назва)
Кафедра *інформаційно-мережної інженерії*
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Методи захисту веб-ресурсів від розвинутих стійких загроз на базі використання прихованого контенту
(тема)

Виконав:
студент *2* курсу, групи *ІМІм-21-1*
Курапов А.С.
(прізвище, ініціали)

Спеціальність *172 Телекомунікації та радіотехніка*
(код і повна назва спеціальності)
Тип програми освітньо-професійна
Освітня програма *Інформаційно-мережна інженерія*
(повна назва освітньої програми)

Керівник *ст.викл. Твердохліб В.В.*
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ *Безрук В.М.*
(підпис) (прізвище, ініціали)

2022 р.

Не містить відомостей, заборонених до відкритого публікування

Студент _____

Керівник_____

Харківський національний університет радіоелектроніки

Факультет *інфокомунікацій*

Кафедра *інформаційно-мережної інженерії*

Рівень вищої освіти перший (бакалаврський)

Спеціальність *172. Телекомунікації та радіотехніка*
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма *Інформаційно-мережна інженерія*
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)
«____»_____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові *Курапову Андрію Сергійовичу*
(прізвище, ім'я, по батькові)

1. Тема роботи *Методи захисту веб-ресурсів від розвинутих стійких загроз на базі використання прихованого контенту*

затверджена наказом університету від *21 жовтня* 2022 р. № *1376 Ст*

2. Термін подання студентом роботи до екзаменаційної комісії *27 грудня* 2022 р.

3. Вихідні дані до роботи *Виконати огляд найбільш поширених типів кібератак. Дослідити сутність та особливості здійснення APT. Обгрунтувати, що застосування механізмів маскування даних набуло застосування як один зі складників APT. Дослідити загальні підходи до побудови системи стегоаналізу трафіку у реальному часі. Довести доцільність та дослідити принципи функціонування алгоритмів стегоаналізу, уніфікованих та також таких, що орієнтовані на контейнери графічного типу*

4. Перелік питань, що потрібно опрацювати в роботі
Вступ.
1. *Найбільш поширені типи кібератак*
2. *Розвинуті стійкі загрози*
3. *Загальні підходи до побудови заходів зі стегоаналізу*
4. *Підходи до побудови алгоритмів виявлення стегоконтейнерів*

Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) _____
Слайди у форматі Power Point (назва та мета роботи, найбільш поширені типи кібератак, розвинуті стійкі загрози, загальні підходи до побуви заходів зі стегоаналізу, вибірка пакетів вхідного трафіку для аналізу, підходи до побудови алгоритмів виявлення стегоконтейнерів

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Вступ		
2	Найбільш поширені типи кібератак		
3	Розвинуті стійкі загрози		
4	Загальні підходи до побудови заходів зі стегоаналізу		
5	Підходи до побудови алгоритмів виявлення стегоконтейнерів		
6	Висновки		

Дата видачі завдання _____ _____20__ р.

Студент _____
(підпис)

Керівник роботи _____ *ст.викл. Твердохліб В.В.*
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 83 с., 20 рис., 22 джерела, 1 додаток

СТЕГОАНАЛІЗ, АРТ, БІТОВИЙ ЗРІЗ, МЕТОД ХІ КВАДРАТ, НЗБ, СИГНАТУРНІ МЕТОДИ

Об'єкт дослідження – алгоритми та заходи зі стегоаналізу, спрямовані на виявлення у реальному часі зловмисних маскованих каналів.

Мета роботи – дослідити особливості побудови та функціонування алгоритмів стеганографічного аналізу як складової частини комплексної системи кіберзахисту.

Розглядаються базові класи кібератак. Рокривається сутність застосування методів стеганографії у ході здійснення АРТ. Досліджується схема здійснення аналізу даних на предмет виявлення маскованих сегментів у реальному часі. Досліджуються найбільш поширені сьогодні алгоритми стегоаналізу.

THE ABSTRACT

Explanatory note: 83 p., 20 fig., 22 sources, 1 app.

STEGOANALYSIS, APT, BITTLE CUT, XI SQUARE METHOD, LSB, SIGNATORY METHODS

The object of research - algorithms and measures for stegoanalysis, aimed at detecting real-time malicious masked channels.

The purpose of the work is to investigate the peculiarities of construction and functioning of steganographic analysis algorithms as a component of a complex cyber defense system.

Basic classes of cyberattacks are considered. The essence of the application of steganography methods in the implementation of APT is revealed. The scheme of data analysis for the detection of masked segments in real time is investigated. The most common algorithms of stegoanalysis are studied today.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ

APT – (Advanced Persistent Threat) – розвинута стійка загроза;
ПЗ – програмне забезпечення;
НЗБ – найменш значимий біт – молодший біт двійкового опису;
HTTP – (HyperText Transfer Protocol) — протокол передачі даних прикладного рівня.

ВСТУП

Поняття «інформаційне середовище» у його сьогоднішньому значенні існує не більш, ніж 10-15 років. Проте, тим паче, навіть за такий короткий термін дане поняття у деяких аспектах кардинально змінилося. Зокрема, змінилися масштаби самого інформаційного середовища, його структура, об’єми інформаційних потоків, а також ряд інших характеристик.

Так, попри те, що інформаційне середовище є надзвичайно складною структурою як за топологією, так і з точки зору інформаційних потоків, з позиції їх наповненості та діапазонів бітових швидкостей окремих складових, одним з характериних показників інформаційного середовища є високий рівень його агресивності.

Для цього існує ряд причин, а саме:

- більшість традиційних сфер діяльності, у т.ч. з галузі комерційної діяльності, тобто, спрямованих на отримання прибутку, зараз отримали продовження в онлайн-середовищі;

- конкуренція у більшості ніш діяльності, стосовно веб-простору, останні кілька років суттєвим чином постійно загострюється;

- розвиток інформаційних технологій, зростання продуктивності клієнтських терміналів а також часткове удосконалення хмарних платформ надає зловмиснику у розпорядження ряд засобів, що можуть розглядатися як базис для реалізації різнорідних зловмисних впливів, що стосується, зокрема, сценаріїв недобросовісної конкуренції..

Що однією знаковою характеристику інформаційного простору за сучасних умов є постійне та невпинне зростання частки даних, для яких відсутня можливість спрощеного доступу, що надається для будь-якого стороннього клієнта. Така залежність, у свою чергу, пояснюється тим, що відсоток даних, які підлягають захисту від неавторизованого доступу на базі застосування криптографічних інструментів та інструментів стеганографії, постійно зростає.

Це, у свою чергу, зумовлюється тим, що:

- весь час зростає обсяг сервісів на базі інфокомунікаціної мережі, які, зокрема, передбачають трансляцію великої кількості даних конфіденційного типу у напрямку клієнт-сервер;

- нераціональністю або взагалі неможливістю реалізації підходів щодо збільшення рівня безпеки мережевих даних шляхом виокремлення потоків різного рівня доступу в окремі фізичні сегменти;

- утіленням великої кількості провідних технологічних концептів, які передбачають передавання значного відсотку даних закритого типу (роботизовані системи, проект «Безпечне місто», Smart City тощо).

У зазначених умовах отримують розвиток технології шифрування та приховування даних.

При цьому, частішають випадки застосування зазначених технологій зловмисником як самостійно, так і у складі комплексних заходів. Це стосується наступного:

- злам систем захисту з викраданням конфіденційних даних або інформації, що є об'єктом комерційної або державної таємниці;
- проникнення усередину інформаційно-комунікаційної системи з метою отримання управління її вузлами;
- виведення з ладу мережевої інфраструктури та ін.

Однією з ключових відмінностей дій зловмисників за останні роки є все частіша реалізація заходів, що отримали назву розвиненої стійкої загрози, або APT (APT – advanced persistent threat) [1]. При цьому, як свідчать дослідження, зростає частка використання методів маскування даних у ході проведення APT, що багаторазово зменшує ймовірність виявлення факту зловмисних дій на базі традиційних підходів до побудови системи кіберзахитсу, та, відповідно, збільшує шанси зловмисника.

У свою чергу, суттєво збільшити вірогідність виявлення APT, та значно зменшити його ефективність здатне виявлення прихованих каналів, що можливе з використанням методів стегоаналізу. Таким чином, дослідження, розробка та застосування алгоритмів та технологій стегоаналізу стогодні є гостро актуальними питаннями.

1 СЦЕНАРІЇ ЗЛОВМИСНОГО ВПЛИВУ НА КЛІЄНТСЬКІ ТА ВЕБ-ВУЗЛИ

1.1 Об’єкти зловмисного впливу

Будь-який зловмисний вплив на мережеву інфраструктуру зазвичай спрямований на одержання його авторами чи іншими зацікавленими особами матеріальних та/або моральних зисків за рахунок:

1. Прямого отримання фінансових активів з боку сторони, що є об’єктом атаки у наслідок:

- одержання доступу до її банківських рахунків;
- шантажу;
- крадіжки та наступного продажу інформаційних ресурсів (база клієнтів, тощо) третім особам.

2. Опосередкованого отримання фінансової вигоди за рахунок:

- тимчасового, або на постійній основі, усунення конкуренту у ніші;
- використання доробок компанії-конкурента, одержаної у наслідок крадіжки і т.д.

3. Компрометації об’єкту атаки (компанії, організації чи установи, та/або пов’язаних з нею осіб).

Обєктами зловмисного впливу різного роду найчастіше становляться:

- комерційні структури;
- фінансові організації;
- державні установи;
- приватні особи, що потенційно можуть представляти інтерес для зловмисників (публічна особа, власник бізнесу, топ-медежер фінансової структури, державний діяч і т.ін.).

У будь-якому випадку, незалежно від того який саме об’єкт атаки, для реалізації зловмисного впливу сьогодні існує велика кількість технік та сценаріїв.

Спочатку розглянемо ситуацію, коли об’єктом атаки є веб-вузол. Дана ситуація має місце тоді, коли мета зловмисника полягає у:

1. Отриманні управління самим веб-вузолом.
2. Розгляді веб-вузла, як проміжного засобу для реалізації атаки відносно мережевої інфраструктури організації, що є об’єктом атаки.

У першому випадку веб-вузол, як об’єкт контролю зловмисника, може бути використано для:

- отримання відомостей щодо клієнтської бази;
- отримання відомостей щодо комерційної політики компанії;
- застосування вузла, як проміжного етапу загального процесу інфікування ресурсів мережі-об’єкту атаки.

Для першого випадку загальний сценарій атаки може бути проілюстровано рисунком 1.1.

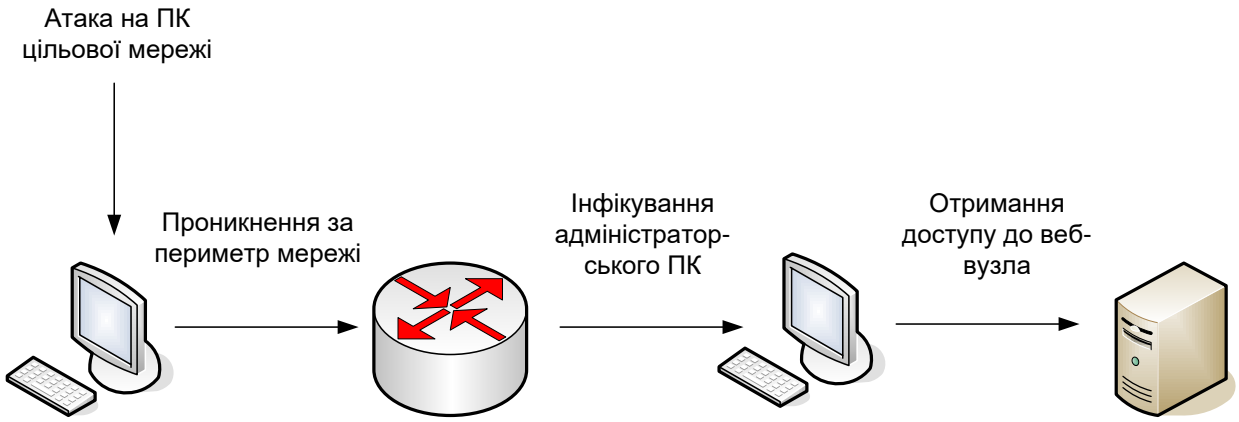


Рисунок 1.1 – Типовий сценарій атаки мережевої інфраструктури для отримання доступу до веб-вузлу компанії

У цьому випадку першим об’єктом атаки є один з ПК усередині мережі. Для його первинного інфікування найчастіше може бути використано, наприклад:

- фішинг;
- атаки на ланцюжки поставок програмного забезпечення тощо.

В іншому випадку, коли кінцевим об’єктом атаки є ті чи інші ресурси мережі (рис.1.2), наприклад – локальне файлосховище, атакуватися може веб-ресурс, використовуючи при цьому:

- попереднє інфікування мережевих ресурсів, з якою відбувається взаємодія (джерела оновлення плагінів та зовнішніх посилань, мережевих шрифтів тощо);
- php та/або sql-ін’єкції;
- атаки типу brootforce на адміністративні модулі ресурсу;
- імітація доступу з адміністративної робочої станції і т.ін.

1.2 Базові засоби атаки на вузли мережевої інфраструктури

1.2.1 Фішинг

Актуальність фішингу, як одного з поширених видів зловмисних впливів, пояснюється його орієнтованістю на існування людського фактору та його вагомість незалежно від того, яку архітектуру захисту корпоративної мережі використовується.

Відносно викока ефективність фішингу, як елементу соціальної інженерії, першочергово зумовлюється експлуатацією людських слабкостей – страху, допитливості, жадібності тощо.

Атаки, ключовою складовою яких є фішинг, зазвичай реалізуються на базі:

- фішинг-сайтів, що мають ознаки справжніх ресурсів;
- кампаній розсилання фітингових повідомлень у соціальних мережах, або поштові розсилки, чи СМС-повідомлення; в усіх випадках випадках повідомлення містять посилання на ресурси, створені для фітингу;
- розсилка ПЗ, що містить зловмисний код.

При цьому, частіше за все атаки на корпоративну мережеву інфраструктуру реалізуються відповідно до третього способу серед наведених вище.

Так зловмисний код, що міститься у файлі, який розсилається, може являти собою кіберзагрозу фактично будь-якої категорії. Зокрема, це може бути Trojan, здатний забезпечити доступ зловмиснику у середину периметру мережі, або, вірус-шифрувальник, що потрапляє до усіх робочих станцій мережі та шифрує інформацію, яка на них міститься. У підсумку, далі мережа може бути атакована зсередини, результатом чого частіш за все є крадіжка фінансових активів компанії, цінних даних або отримання контролю над мережею.

Разогм з тим, стандартний перелік засобів кіберзахисту, шо використовується для більшості корпоративних мереж, не може гарантувати безпеки. Це саме, очевидно, стосується веб-вузлів. Одна з причин – те, що антивірусні засоби не здатні виявити та надалі блокувати усю масу надходячих підозрілих файлів [2]. Виходячи з цього компанії, для яких інформаційна безпека є запорукою успішної життєдіяльності, застосовують додаткові інструменти протидії кібератакам.

На сьогодні одними з найбільш результативних серед них є т.з. «пісочниці» - синтезовані цифрові середовища, до яких надсилаються файли, що можуть бути потенційно небезпечними для подальшого аналізу та перевірки, як від «поводить себе».

Іншою важливою складовою системи кіберзахисту є періодичне та регулярне оновлення ПЗ робочих станцій, так як, зокрема, зловмисники створюють віруси, беручи до уваги виявлені уразливості програмних засобів.

Серед випадків фішингових атак значний відсоток належить веб-фішингу, що застосовується як інструмент зліму корпоративних інформаційних систем. Наприклад, зловмисники можуть отримати відомості щодо типу систему електронного документообігу, яких застосовується цільовою організацією, за результатами чого створити ресурс, що повністю імітує стартову сторінку такої системи. За рахунок цього реалізуються крадіжки облікових даних співробітників. Загальна схема такого механізму зводиться до:

- розміщення у мережі ресурсу, що майже повністю імітує реальний – сайт корпоративного сервісу, сайт компанії чи супутній ресурс тощо;
- надсилання співробітникам компанії посилань на даний ресурс;
- зчитування облікових даних, які співробітник може ввести, випадково звернувшись до даного ресурсу

При цьому, переважна більшість фішингових сайтів мають схожі ознаки, серед яких [1]:

1. Ім'я домену, максимально схоже на реальне, але при цьому таке, що відрізняється від нього. Зокрема, це може бути, наприклад, «online.bank.com» замість «onlinebank.com». При цьому, нерідко автори фішинг-атаки мають у своєму активі сайт у піддомені - «bank.site.com».
2. Відсутність SSL-сертифіката. Сьогодні адреси більшості інтерне-ресурсів виаокристовують протокол https. Тобто, у випадку, коли адреса сайту має префікс «http: //», цей факт можна розглядати як можливу ознаку інтернет-ресурсу, який було побудовано зловмисником.
3. Аномалії оформлення, наприклад:

- дизайн, web 1.0;
- граматичні і орфографічні помилки на сторінкай сайту;
- некоректна верстка;
- елементи дизайну, що не відповідають стилістиці ресурсу.

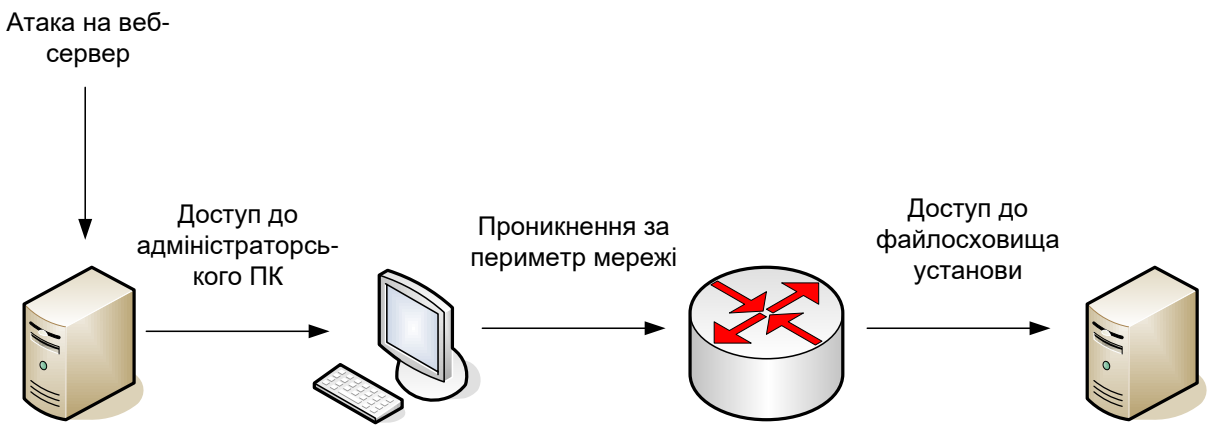


Рисунок 1.2 – Один з можливих сценаріїв атаки на мережеву інфраструктуру з виходом на управління файлосховищем, з розглядом веб-вузлу проміжним об’єктом доступу

1.2.2 Атаки на ланцюжки поставок ПЗ

У загальному випадку шлях поставок ПЗ розглядається як процес, у структурі якого може бути локалізовано не менше, ніж 4 загальні етапи розробки, серед яких:

- сама розробка коду продукту;
- staging, або технологічний етап тестування створеного засобу у межах середовища, яке для цього створюється штучно;
- тестування створеного ПЗ;
- публікація релізів продукту.

Водночас, зловмисник може інтегрувати в оригінальний засіб модулі, які містять шкідливий код, у ході будь-якого перелічених вище пакетів. Також зловмисний код може бути внесено до пакетів оновлень, що відбувається частіше.

У решті решт користувачі програмного засобу, які у ході довгої співпраці звикли довіряти розробникам, фактично інтегрують зловмисних агентів у свою мережу самостійно.

При цьому, за участю засобів фішинга зловмисники можуть одержати доступ до системи електронного листування вендору ПЗ та модифікувати

програмний продукт, вносячи до коду ту чи іншу кількість додаткових рядків. Далі, якщо компанія-розробник користується хмарними платформами у ролі файлосховищ, доступ до них може бути отримано зловмисником, може бути інфікуваним весь доступний перелік ПЗ.

Ще одним поширеним прикладом атаки зловмисників на ланцюжки поставок ПЗ є підміни сервера, який, у свою чергу, задіяно у процесі оновлення ПЗ.

Так, відповідно до даного сценарію, було реалізовано атаку, коли зловмисники у 2017 році одержали можливість доступу до національного інтернет-ресурсу, що використовувався для оновленнями M.E.Doc - бухгалтерського пакету програмного забезпечення вітчизняних розробників [2].

Наслідком атаки на сервер M.E.Doc було масове ураження кінцевих пристроїв та мереж, що налічувало десятки тисяч епізодів. Це було спричинено результатами впливу віруса-шифрувальника NotPetya.

При цьому слід також зазначити, що ймовірність інфікування «великих» розробників програмних засобів, у загальному випадку відносно невелика. Це, у свою чергу, є результатом грунтовного контролю ланцюжка поставок ПЗ на кожному з технологічних етапів.

Разом з тим, використання інструментарію EDR - Endpoint Detection and Response - сприяє значному зменшенню ймовірності ураження ПЗ, що у деяких випадках знижується майже до нуля.

Тут EDR – перелік інструментів, що використовуються для виявлення та подальшого аналізу потенційно підозрілих активностей у межах робочих станцій. У цьому випадлку засоби EDR мають використовуватися спільно з антивірусними пакетами.

1.3 Поширені типи атак відносно веб-вузлів

1.3.1 PHP-ін’єкції як засіб зловмисного впливу

Для прикладу виконаємо аналіз скрипту, шо наведений нижче:

```
<?
...
$module = $_GET['module'];
include ($module.'.php');
```

```
...
?>
```

Наведений вище приклад скрипта є уразливим як наслідок того, що до наведеної змінної `$module` дописано конструкцію «.php». Відповідно далі за даним шляхом теоретично може бути підключено файлу будь-якого змісту.

Паралельно з цим, у межах підконтрольному зловмисникам сайту розміщується файл, у якому знаходиться код PHP, наприклад, такий, як `https://unsecuresite.com/inc.php`).

Таким чином, результатом виконання коду, наприклад може бути перехід на сайт за посиланням вигляду:

```
http://mysite.com/index.php?module=http://
unsecuresite.com/inc,
```

при цьому теоретично тут може бути задано виконання яких завгодно команд PHP.

1.3.2 Brutforce-злам

У нашому випадку, фактично, Brutforce-злам – це атака, що базується на пошуку паролю до адміністративної пателі сайти чи C-panel хостингу або FTP на базі простого перебору. На поточний момент часу brutforce займає одну з ведучих позицій як за кількістю атак, що були виконані протягом року, так і за рівнем їх результативності. Інакше кажучи, даний тип зловмисних впливів сьогодні є одним з найбільш поширених та продуктивних у арсеналі зловмисників серед тих що застосовуються для зламу веб-вузлів.

Така поширеність застосування зловмисниками brutforce-атак пояснюється тим, що:

- ймовірність виявлення brutforce-атак мінімальна, так як втручання реалізується у фоновому режимі, відтак навантаження, що чиниться у ході цього відносно сайту, мінімальне, а отже - виявити присутність зловмисного впливу опосередковано практично неможливо;

- практично усі стандартизовані системи кіберзахисту, поширені зараз, які базуються на евристичних та/або евристичних методах, не можуть виявити даний тип зловмисного впливу на стадії реалізації;

- суттєвий відсоток результативних атак – це «атаки за словником», тобто, частіше за все пароль та логін для доступу до ресурсу зловмисник знаходить з переліку найбільш застосовуваних.

У свою чергу, базовий сценарій реалізації brutforce-атаки є таким, як показано на рис. 1.3.

Водночас, для того, щоб суттєвим чином знизити ймовірність виявлення факту спроб вторгнення на базі brutforce-атак, нерідко може застосовуватися попередньо сформований botnet, так як втручання з одного вузла є ризикованим. При цьому, у будь-якому випадку звернення до сторінки login.php буде відбуватися методом GET.

У ході такого втручання поточний вузол, або bot, одержавши відгук від серверу, у відповідь надсилає послідовність символів p(i), як варіант паролю, відповідно до методу POST.

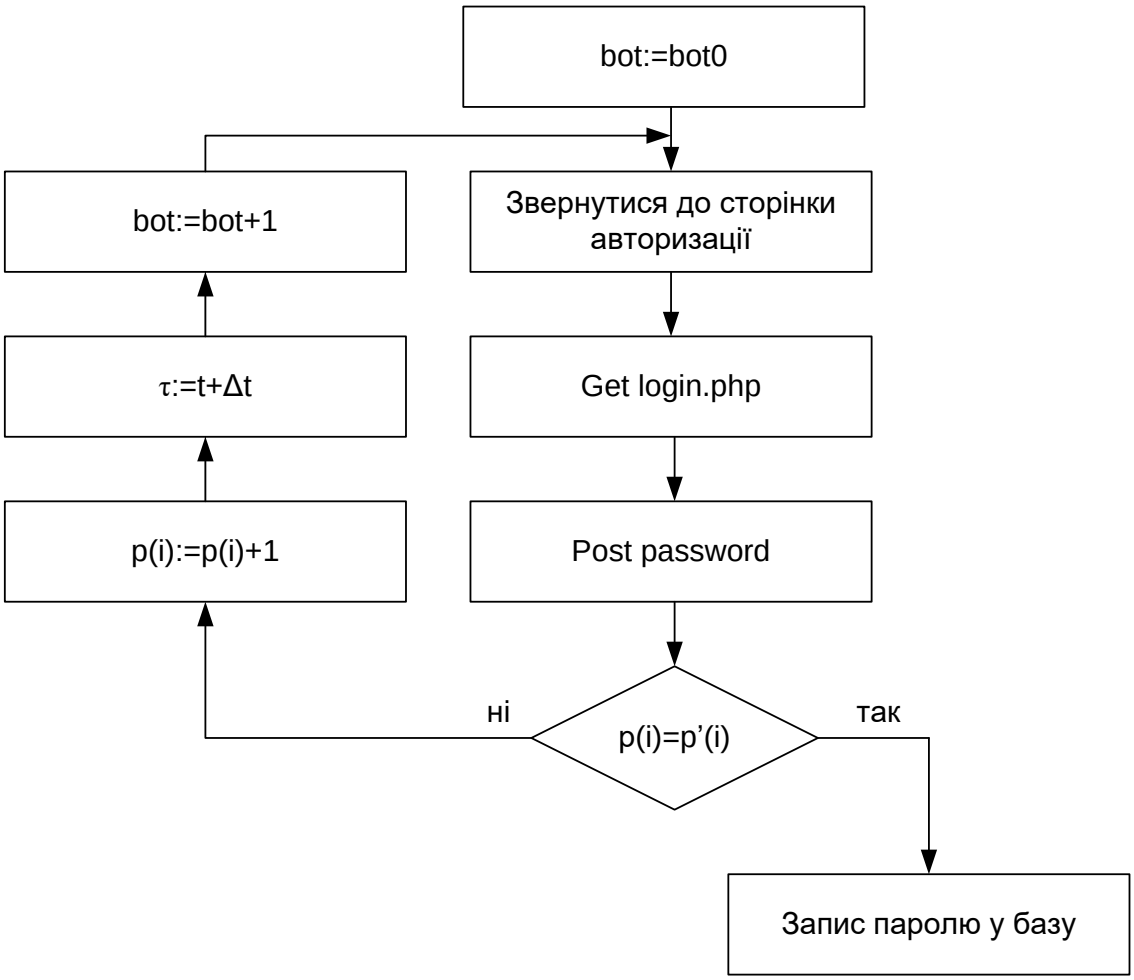


Рисунок 1.3 – Узагальнений принцип підбору паролей, що має місце під час реалізації brutforce-атаки

Далі припустимо, що пароль p(i), згенерований та надісданий ботом, не буде відповідати реальному p’(i). На цей випадок далі буде здійснено такі технологічні операції, як :

- вибір нової послідовності p(i) для реалізації послідуючого кроку підбору паролей, у ході чого $p(i) := p(i) + 1$, зводиться до змін ідентифікаторів «i» у переліку паролей з з бази найбільш ймовірних для умов, коли необхідно реалізовувати атаку за словником, чи до зміни символів у послідовності на випадок, коли необхідно виконувати повний перебір варіантів;

- прийняття рішення відносно часу τ послідуючої атаки; у даному випадку, у принципі, до фактичного часу t буде додано певну величину Δt, обчислення якої виконується за принципом:

$$\Delta t = \text{rand}[t_1; t_2], \tag{1.1}$$

де t_1 та t_2 - раніше задані максимальний та мінімальний інтернвали часу, що є між атаками;

- зміна боту для реалізації атаки на подальшому кроці з передаванням йому необхідного масиву даних для визначення можливої результативної послідовності даних p(i).

При цьому якщо на одному з етапів масив p(i) буде аналогічною паролю p’(i), далі зловмисники можуть одержати цілковитий доступ до панелі управління сайтом чи адміністративної панелі хостингу.

Також слід зазначити про те, що результативна brutforce-атака потребує виконання ряду передуючих дій, серед яких [17]:

1. Одержання доступу до переліку веб-вузлів, що характеризуються достатньо широкою користувацькою аудиторією; дужа часто такі відомості є відкритими;

2. Визначення типу CMS, на базі якої функціонує ресурс; Для цього зазвичай застосовуються спеціалізовані скрипти;

3. Збір переліку відомостей щодо типової належності адмінпанелей сайтів та їх адреси, користуючись даними, отриманими відносно систем управління контентом. Тут береться до уваги інформація, одержана відносно CMS на кроці, що передує поточному. Отримані таким чином дані, у свою чергу, надсилаються до brutforce-сервер. Далі на рівні серверу атаки активується скрипт, логіку роботи якого посянює алгоритмом, розміщений

рис. 1.3. У схемі загального сценарію реалізації brutforce-атаки це відповідає етапу 1 рис. 1.4.

4. Відправка успішно підібраних паролей до адмінпанелей до іншого brutforce-серверу, що відповідає етапу 2 на рис.1.4.

Отже, застосовуючи зібрані паролі, далі інший скрипт реалізує вхід до панелі керування CMS, за результатом чого файли на веб-сервері інфікуються зловмисним кодом, на рис.1.4 це етапи з 3 по n відповідно.

5. Внесення коду, що реалізує перенаправлення користувачів, до інфікованих файлів. Далі користувачі, які відвідали інфікований сайт, будуть спрямовуватися, до системи розподілу трафіку (TDS) зловмисника - етап n+1 на схемі, поданій рис. 3.2.

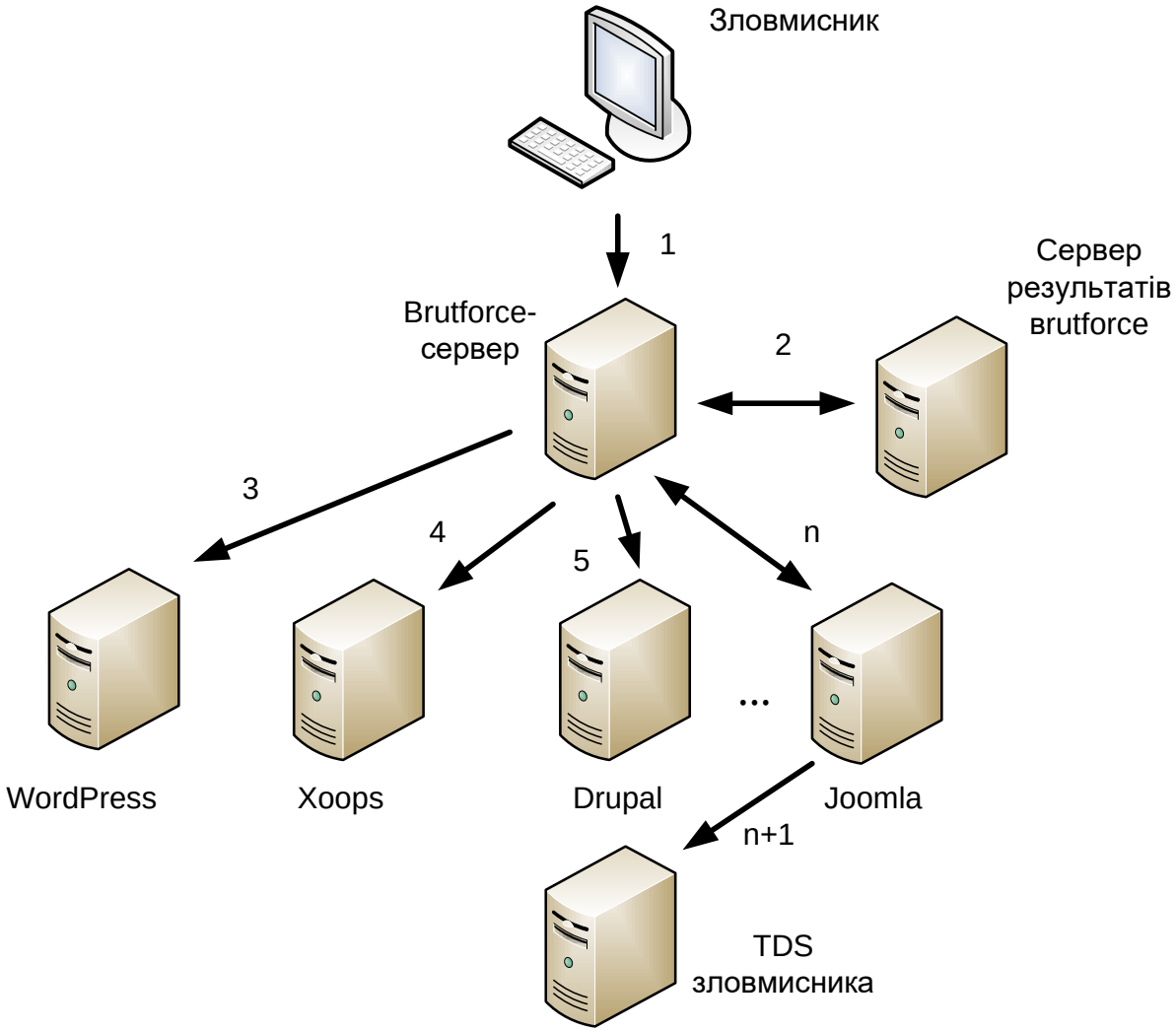


Рисунок 1.4 - Технологічні етапи проведення brutforce-атаки за типовим сценарієм

Поширеними також є випадки використання зламаної CMS сайту як однієї з компонент TDS зловмисника. При цьому, зловмисника взагалі може не цікавити доступ до змісту веб-сховища, або питання дискредитації ресурсу. Найчастіше сформований таким чином трафік далі зловмисником монетизується.

Перебіг подальшого розповсюдження зловмисного ПЗ мережею та, як наслідок, монетизації сформованого TDS трафіку зазвичай містить у собі ряд ключових кроків:

1. Незалежно від способу входу відвідувача до інфікованого ресурсу, як то – органічний, прямий, реферальний чи з соціальних медіа – спрацьовує перенаправлення відвідувача до серверу зловмисника. Головним елементом цієї процедури є TDS.
2. TDS-система зловмисника, реалізує редирект відвідувачів до серверів, які, у свою чергу, розповсюджують зловмисне ПЗ.

Важливим тут є те, що, власне, окрім виконання редиректів за заданою схемою, TDS здатна визначати типоналежність поточних відвідувачів, що значно підвищує її життєздатність. Даний функціонал убезпечує сформовану систему від викриття пошуковими системами. При цьому, якщо відвідувач розпізнається як краулер, TDS спрямовує його до серверу, де шкідливий контент відсутній.

3. Взаємодія зі зловмисними партнерськими програмами. Для реалізації даного етапу, як одного з ключових у процесі монетизації трафіку, попередньо виконується реєстрація у певній кількості шкідливих партнерських програм.

1.4 Попередні висновки

Методи виконання атак на робочу станцію у межах переметру мережі та на веб-вузли мають ряд спільних рис, та реалізуються з використанням нерідко одних і тих же інструментів.

При цьому, для протидії більшості типам атак буває достатньо дотримуватися мережевого регламенту, застосовувати стандартизований перелік засобів кіберзахисту та регулярно виконувати оновлення ПЗ, тим самим блокуючи його уразливості, які може бути використано зловмисником.

Разом з тим, зазначених засобів може бути недостатньо для виявлення та протидії комплексних цільових кібератак. До них, у першу чергу, відноситься APT.

2. РОЗВИНУТІ СТІЙКІ ЗАГРОЗИ, ЯК НЕТРИВІАЛЬНИЙ ТИП АТАК НА ІНФОРМАЦІЙНУ СИСТЕМУ ТА ЇЇ КОМПОНЕНТИ

2.1 Відмінності APT від інших типів загроз загрози

Термін APT, або Advanced Persistent Threat, який означає буквально «розвинуту стійку загрозу» використовується з початку 2000-х років.

На сьогодні класичним випадком розвинутої стійкої загрози є складні кібератаки з багальта рівнями реалізації, спрямовані зазвичай на конкретну мережеву інфраструктуру, веб-сервер чи базу даних [1, 4, 5]. При цьому, частіше за все APT реалізується відносно:

- об’єктів комерційного характеру;

- об’єктів, що прямо чи опосередковано відносяться до т.з. критичної інфраструктури. При цьому, у будь-якому випадку мова йде про стратегічно важливих об’єкти. Це, наприклад, можуть бути адміністративні об’єкти, військові об’єкти чи об’єкти, важливі на галузевому рівні.

Розвинута стійка загроза, при цьому, характеризується:

- сценаріями реалізації атаки, що мають у своєму складі ряд етапів та ряд альтернативних реалізацій;

- великою кількістю засобів, які беруть участь у реалізації атаки;

- значним рівнем професійності у галузі кеіберзахисту та кіберзагроз осіб, які здійснюються планування та реалізацію APT [5].

У той же час, як свідчать висновки фахівців з кіберзахисту Sophos [6], зараз єдині та одностойні ознаки, що вказують на те, що та чи інша атака, у сутності, є APT, фактично відсутні.

При цьому необхідно зазначити, що ряд потенційно успішних атаки може бути реалізовано з використанням переліку експлойнів що на сьогодні може вважатися класичним. Наприклад, саме з цієї причини уразливість нульового дня (0-day) у загальному випадку обов’язковим атрибутом розвинутої стійкої загрози зараз не вважаються.

Попри все вищезазначене, одним з критеріїв, який може вважатися непрямою ознакою APT, є розсилання фішингових листів, спрямованих на компрометацію облікового запису одного одного чи кілької співробітників компанії.

У майбутньому така розсилка фітинг-листів використовується, по-перше, як точка входу через периметр мережі, що являє собою об’єкт атаки.

По-друге, це є, у сутності, проміжний плацдарм, на базі якого далі реалізується доступ до критичних складових мережі, а саме - ПК топ-менеджерів та серверів компанії.

Зазначимо також, що зараз методи соціального інжинірингу є одними з найбільш застосовуваних у зловмисників. Виходячи з цього, розглядати їх у якості необхідної та достатньої умову APT не варто.

2.2 Типові ознаки розвинутої стійкої загрози

Аналізуючи та узагальнюючи існуючі сьогодні звіти багатьох фахівців у галузі кібербезпеки, можемо визначити типовий перелік характерних рис присутності APT, як показано рис.2.1, а саме [6]:

1. У бідь-якому випадку атака є цілеспрямованою. При цьому, її об’єктом зазвичай є не та чи інша людина чи, наприклад, компанія, а найчастіше значно ширший сегмент (наприклад, банки, установи, що належаті якійсь конкретній галузі тощо), чи певна група осіб, поєднаних спільною ознакою (клієнти однієї банківської установи, громадяни, що придбали один і той же туристичний тур, клієнти страхової компанії).

2. Довготривалість атаки. При цьому, зловмисний вплив може здійснюватися протягом значного часового відрізу (кілька місяців або кілька років), при цьому, продовжується або до досягнення встановленої мети, або до втрати доцільності.

3. APT-атака готується та утілюється на тлі дуже ґрунтовної фінансової підтримки. Це є очевидним, так як навіть у випадку звичайних атак DDoS-типу їх реалізація є досить витратною процедурою, надто тоді, коли вона триває довгий проміжок часу.

4. Реалізація розвинутої стійкої загрози передбачає багатостадійність здійснення впливу на атакований об’єкт. При цьому, під час перебігу APT зловмисники послідовно застосовують кілька векторів дій та ряд різних технік, що і забезпечує дієвість атаки. Хоча слід зазначити, що окремо взяті типові складники APT нерідко можуть бути примітивними та взагалі низько ефективними, відтак результативність атаки забазпечується саме їх поєднання. Типовим прикладом тут може бути надсилання одному зі співробітників компанії (що є об’єктом атаки) фішингових листів, щоб скомпрометувати його корпоративний обліковий запис. Далі, використовуючи цей обліковий запис (що у рамках мережі компанії є

довіреним), надіслади документ, який буде містити у собі зловмисний код, на ПК або керівника, або іншої особи, що має високі привілеї з доступу до інформаційних ресурсів компанії.

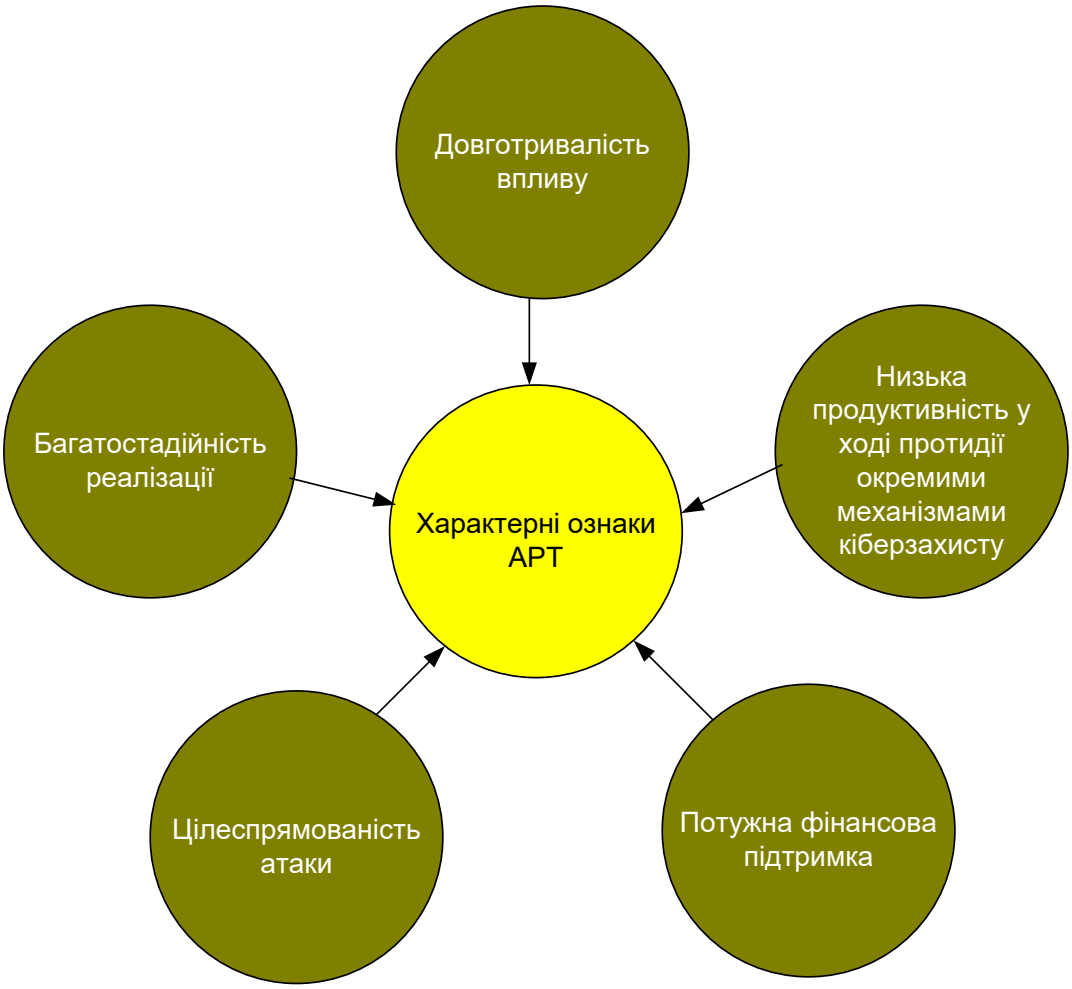


Рисунок 2.1 – Типові ознаки APT

5. Низька (нерідко - нульова) продуктивність протидії APT, використовуючи окремі модулі кіберзахисту. Зокрема, блокувати розвинуту стійку загрозу, користуючись окремими інструменти безпеки, такими, як антивірусне ПЗ, спам-фільтри, базові SIEM та IDS-системи неможливо. При цьому, попри застосування перелічених інструментів захисту, сам факт атаки може не бути поміченим довгий проміжок часу. У найкращому випадку, атака (на тлі її успішного перебігу) буде мати ознаки окремих типових інцидентів, що не є глобально критичними.

Тут слід відзначити те, що нетипова, а у деяких випадках – взагалі аномальна поведінка мережевих процесів, мережі у цілому, її окремих сегментів чи окремих пристроїв усередині периметру може зберігається, водночас планові перевірки з використанням вище перелічених засобів кіберзахисту при цьому не виявлять нічого підозрілого.

6. APT реалізується на базі провідних технік, які, зокрема, здатні досить ефективно виконувати маскування окремих складових атаки від більшості типових систем захисту. Це, зокрема, reverse shell, що використовується для обходу MCE.

Наприклад, згідно з наявними даними, наданими Sophos [7], нерідко у ході реалізації APT зловмисниками використовуються такі техніки, як (рис.2.2):

- фішинг;
- social engineering;
- DdoS;
- botnet;
- уразливості «нульового дня»;
- використання пристроїв, що попередньо були скомпроментовані;
- атаки інсайдерів;
- атаки, які реалізуються на рівні додатків.

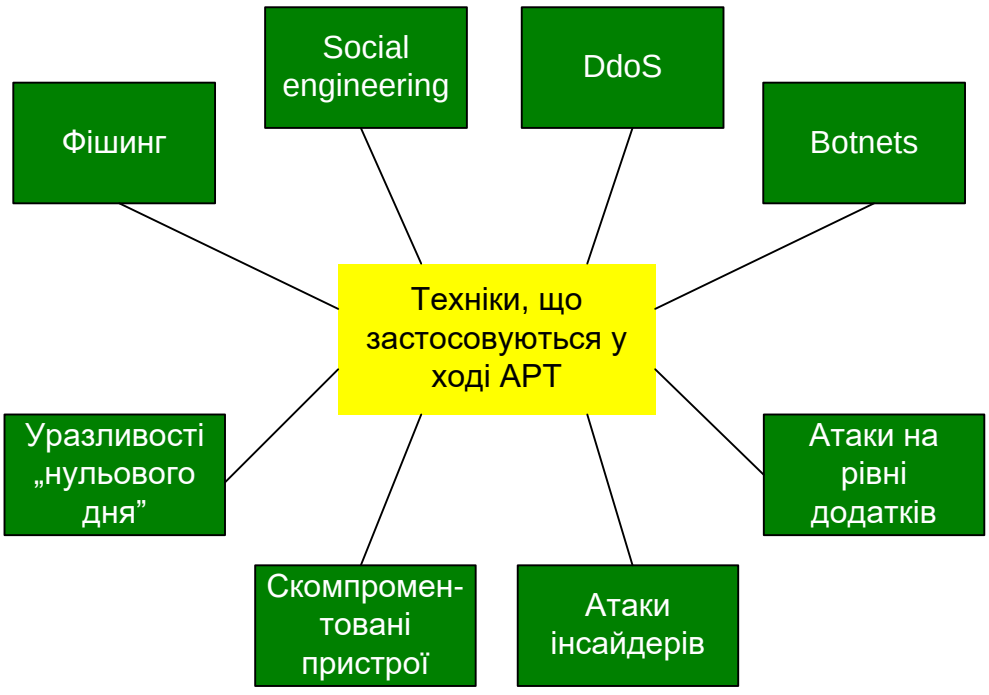


Рисунок 2.2 – Основні групи технік, що використовуються у ході APT

2.3 Етапи реалізації розвинутих стійких загроз

Незалежно від конкретної архітектури атаки APT-типу, її проведення у загальному випадку передбачає послідовну реалізацію ряду ключових етапві впливу (рис.2.3), серед яких [6, 8, 9]:

- отримання інформації відносно об’єкту атаки у пасивному режимі (виявлення та встановлення цілей, керуючись даними, отриманими з відкритих джерел);

- первинне інфікування (комплекс дій, спрямованих на спонукання користувача вузла усередині цільової мережі відвідати фітинговий сайт/сайти сайти, надсилання інфікованих документів);

- імплементація т.з. «бойового навантаження» (найчастіше тут мова йде відносно drive-by-завантаження, а також використання уразливостей браузеру та плагінів, інтегрованих у нього);

- активний етап (збільнешшя привілеїв у системі, обхід систем захисту для тогог, щоб одержати додаткову інформацію щодо системи; інтеграція у систему ключових зловмисних компонентів);

- одержання зловминсиком повного (часткового) віддаленого контролю (застосування кейлоггерів, бекдорів а також встановлення reverse shell).

- обмін даними з зовнішніми адміністративними вузлами в очікуванні наступних команд (для цього нерідко можуть використовуватися месенджери різних типів, поширені мережеві API та клієнти соціальних мереж; за рахунок цього здійснюється обхід наявниху системі firewall);

- реалізація встановленої мети (викрадення інформації, проведення злочинних фінансових транзакцій, побудова ботнету, одержання повноги чи часткового контролю відносно АСУ ТП тощо).

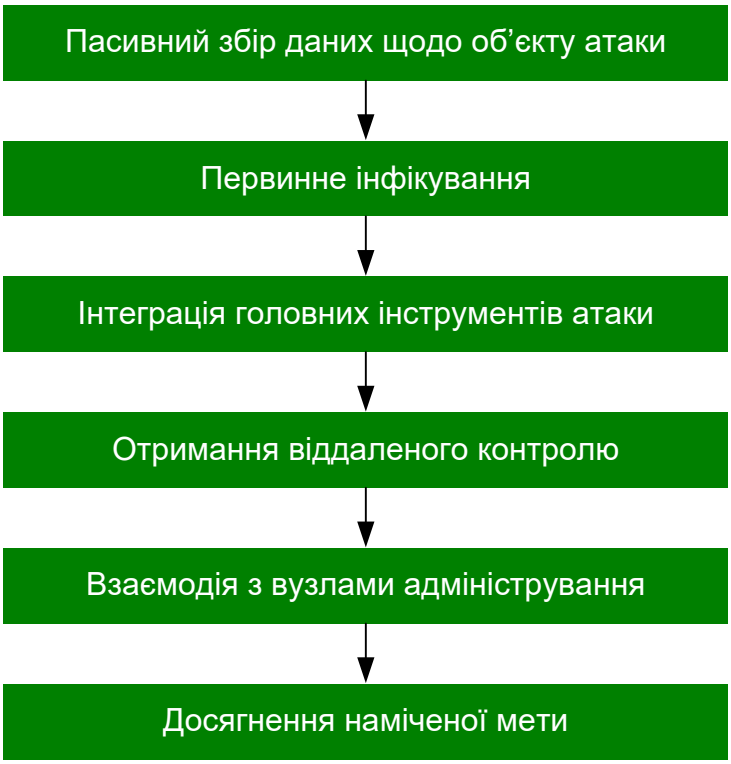


Рисунок 2.3 – Загальні етапи реалізації APT

2.4 Місце та роль методів стеганографії у процесі розробки та утілення розвинутих стійких загроз

У більшості випадків успішних APT стегонографічні алгоритми зловмисником було використано для того, щоб мати можливість обійти інструменти захисту від кібератак тід час реалізації таких технологічних етапів атаки, як попереднє інфікування, а також імплементація зловмисних модулів усередину периметру мережі чи до кінцевого пристрою, яким у загальному випадку, може бути сервер або робочий ПК [8,9].

На наступних кроках зловмисного втручання стеганграфічні алгоритми використовуються для формування прихованого каналу передавання даних, який, у свою чергу, використовується для підтримки взаємодії з віддаленими адміністративними вузлами APT.

Далі виконаємо огляд відомого на сьогодні сценарію реалізації APT, у ході якого використовуються стеганографічні алгоритми (рис. 2.4). Такий сценарій містить у собі наступні кроки:

1. До мережі або вузлу, які є об’єктом атаки, надсилається файл, що виконує функції збірника. Такий файл, при цьому, позбавлений будь-яких

формальних ознак такого, що він може являти собою потенційну небезпеку. Такий файл, найчастіше, може бути надіслано з використанням інструментів фішингу, чи за участю розсилки з використанням e-mail тощо.

2. За участю файла-збірника здійснюється завантаження файлу (файлів), усередині яких на базі стеганографічних алгоритмів вбудовано у неявному вигляді масковане зловмисне навантаження. Важливим є те, що такі файли не викликають підозри у систем кіберзахисту, ні за зовнішніми ознаками (у переважній більшості випадків це файли jpeg, gif або bmp-типів, що мають невисоку роздільну здатность), так як їм не властиві типові ознаки присутності зловмисних модулів.

3. Обробка попередньо звантажених файлів певним чином за участю файла-збірника. Результатом такої обробки є вилучення вбудованих сегментів даних.

4. Збірка зловмисних модулів з використанням попередньо вилучених прихованих даних.

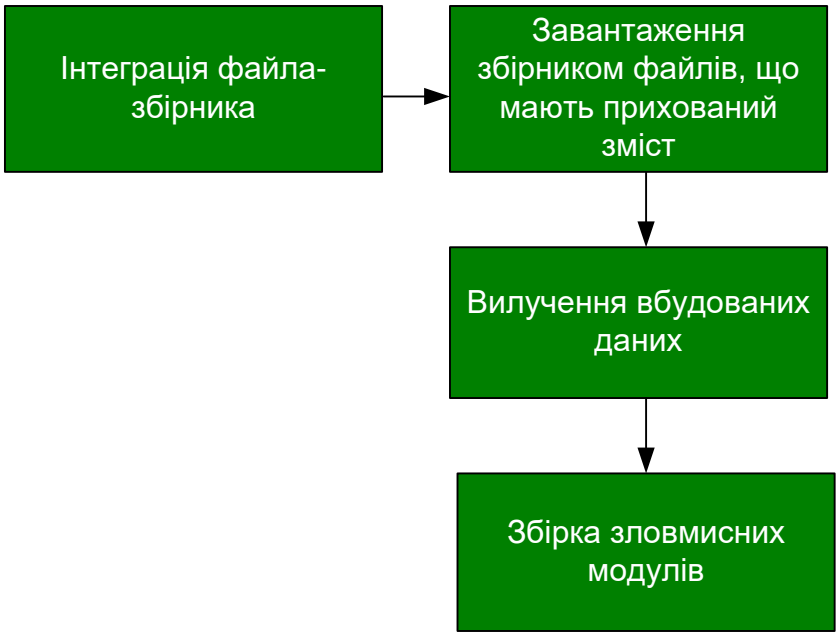


Рисунок 2.4 – Відомий сценарій APT з використанням стеганографічних алгоритмів

Так, на базі означеного сценарію отримав реалізацію широкий перелік успішних засобів кібершпигунства, серед яких слід зазначити наступні:

- NetTraveler;
- ZeusVM;

- six little monkeys;
- KinS;
- Enfal;
- Shamoon;
- Zberp;
- Fibbit та ін.

3. МЕХАНІЗМИ ВИЯВЛЕННЯ ЙМОВІРНИХ КОНТЕЙНЕРІВ У МЕЖАХ ТРАФІКУ, ЩО НАДХОДИТЬ ДО ВЕБ-РЕСУРСУ НА ЗАСАДАХ СТЕГОАНАЛІЗУ

3.1 Застосування пошукових шаблонів для виявлення ймовірних контейнерів

Процес виявлення даних, інкапсульованих з використанням тих чи інших контейнерів, розглянемо знайгіршого випадку. Тобто, процес повинен будуватися виходячи з того, що:

- алгоритм вбудовування, застосований зловмисником, апріорі є невідомим;
- тип контейнеру, використаний зловмисником, також є невідомим;
- зміст контейнеру є випадковию величиною.

За таких умов відсутня можливість використати такі типи атак на аналізований трафік, як, приміром, атака за відомим контейнером.

У такому випадку найбільш доцільним є використання т.з. «пошукових стеганографічних шаблонів».

Визначення. Пошуковим стеганографічним шаблоном у рамках системи стеганографічного аналізу є окремий аналітичний алгоритм, спрямований на виявлення інкапсуляцій, здійснених на базі одного певного алгоритму.

При цьому, на сьогодні існують пошукові шаблони 2 типів, а саме:

- спеціалізовані шаблони;
- уніфіковані шаблони.

Тут *спеціалізовані шаблони* застосовуються для пошуку ознак контейнеру, сформованого на базі певного алгоритму, тоді як *уніфіковані шаблони* є універсальними, тобто, такими, що потенційно здатні виявляи факт присутності контейнеру незалежно від алгоритму його заповнення, керуючись при цьому результатами виявлення аномалій статистичних характеристик.

Водночас, застосування пошукових шаблонів виключно одного типу не є ефективним, так як:

- спеціалізовані шаблони є функціонально обмеженими, оскільки кожен з них орієнтований виключно на певний стегоалгоритм;

- уніфіковані шаблони мають обмежене «вікно ефективності»; зокрема, більшість таких шаблонів здатні виявляти факт наявності контейнеру за умови, що його наповненість є не меншою, ніж 60%.

Виходячи з зазначеного вище, найбільш доцільним є стільне використання шаблонів обох груп.

Водночас, процедура виявлення контейнерів на рівні вхідного рафіку веб-ресурсу у загальному випадку зводиться до сканування прийнятого масиву даних з використанням пошукових шаблонів.

З урахуванням спільного застосування шаблонів обох типів така процедура може бути подана наступною схемою, як ілюструє рис.3.1 [11].

Далі виконаємо огляд загального принципу функціонування алгоритму стегоаналізу безвідносно тих чи інших типів контейнерів.

На на наведеній схемі рис 3.1 до даних ***потенційно небезпечних типів*** належать такі, які першочергово може бути використано для інкапсулювання прихованого змісту. Тобто, це дані тих типів, що з найбільшою ймовірністю можуть бути контейнами.

Водночас, до ***потенційно небезпечних каналів*** у загальному повадку відносяться такі, якими до вузла надсилаються дані, що є потенційно небезпечними.

При цьому, очевидно, що початково до множини D потенційно небезпечних належать усі канали надходження пакетів, як наслідок того, що:

- апріорі невідомо, пакети даних яких типів надходитимуть до веб-вузла;

- виходячи з реалій сучасного складу трафіку, можна констатувати, що дані потенційно небезпечних типів можуть рівноймовірно бути присутніми у більшій частині вхідних пакетів.

З самого початку роботи алгоритму потік надходячих до вузла даних інтерпретується як сукупність L_p пакетів, яка, у свою чергу, утворюється поєднанням пакетів різних типів [11, 12].

При цьому, множина L_p у загальному випадку розуміється як така, що не має чітких часових лімітів, як показано рис. 3.2.

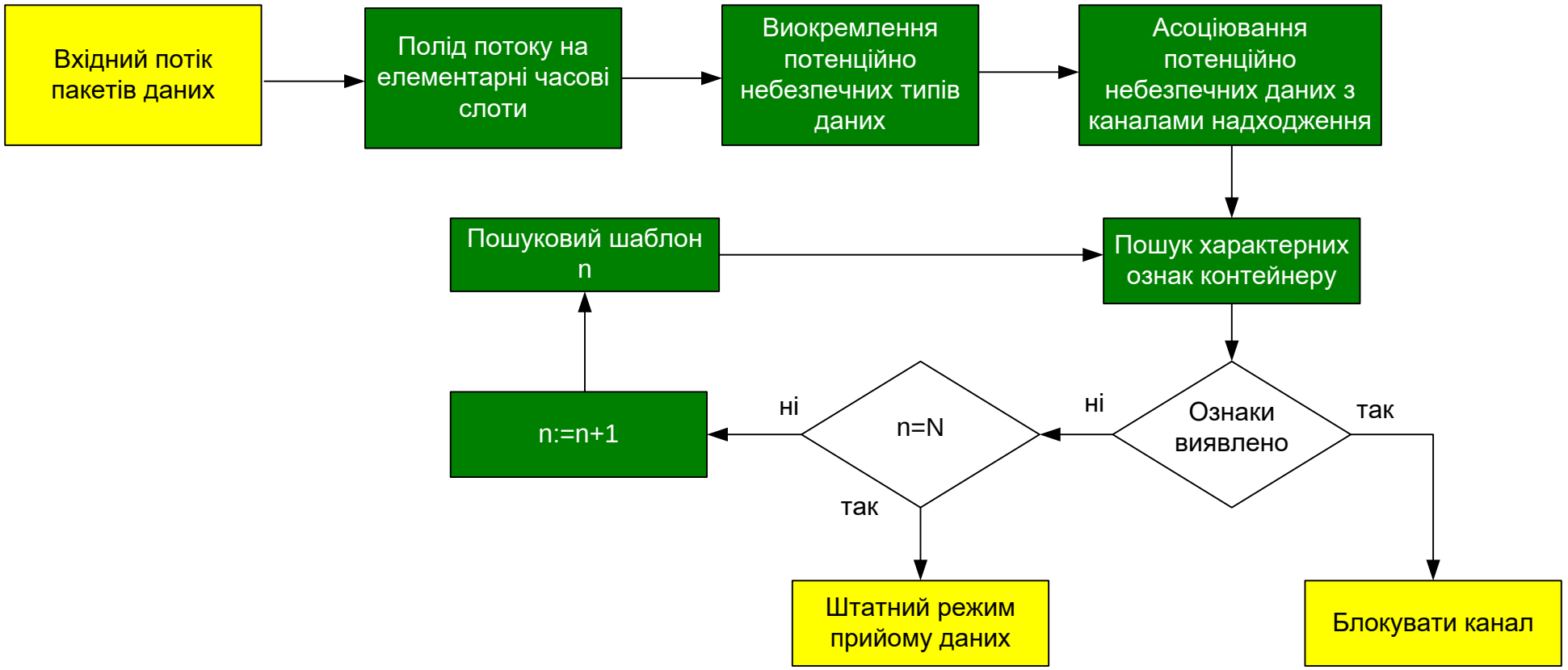


Рисунок 3.1 – Схема реалізації процесу аналізу вхідного трафіка з використанням спеціалізованих та уніфікованих пошукових шаблонів

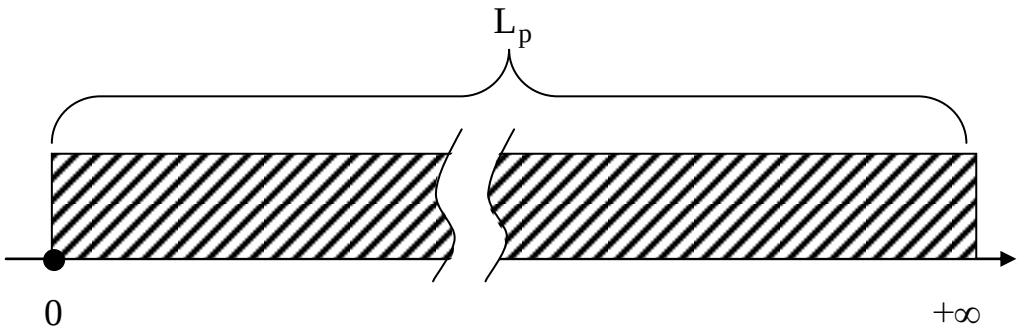


Рисунок 3.2 – Умовний часовий діапазон існування надходячої множини пакетів

Далі у складі множини L_p виокремлюються окремі часові слоти $L_p(t)$, що являють собою її елементарні локальні ділянки у часі на проміжку $t \in [0;+\infty]$.

Виходячи з того, що, як вже було зазначено, поява даних будь-якого типу (у т.ч. потенційно-небезпечних), у складі вхідних пакетів є рівноймовірною. Це дає ґрунтовну підставу стверджувати, що фактично у будь-який довільний фрагмент часу t слот $L_p(t)$, що йому належить, може бути наближено описаний наступним виразом:

$$L_p(t) = \{P_{tx}\} \bigcup \{P_a\} \bigcup \{P_{gr}\} \bigcup \{P_v\} \bigcup \{P_{ex}\} \bigcup \{P_{lb}\} \bigcup \{P_{et}\}, \qquad (3.1)$$

де $\{P_{tx}\}$ - сукупність пакетів даних, що містять інформацію текстового типу, включаючи як структуровані так і неструктуровані тексти, тобто це txt, rtf, htm, css, doc тощо;

$\{P_a\}$ - сукупність пакетів даних, до складу яких входить аудіо інформація. Це, зазвичай, VoIP, аудіо MP3 потокового типу, звукове супроводження відеотрансляції та ін.);

$\{P_{gr}\}$ - множина пакетів, які несуть у собі файли та фрагменти файлів графічних типів, серед яких у мережі найчастіше зустрічаються webp, jpeg, png, gif, bmp, pcx, pic тощо);

$\{P_v\}$ - сукупність даних відео, з урахуванням сервісів потокового та інтерактивного типів. Це може бути, наприклад, VoD, відеозв'язок, сервіси

на кшталт AR та VR і т.ін.; такі дані з найвищою йовірністю представлені у вигляді файлів mp4, webm, avi, чи flv. Водночас, переважний відсоток з них при цьому створено у базисі ідеології mpeg. Суттєвоменшу частку зазвичай складають файли, утворені на базі кодеків фрактального типу, алгоритмів mjpeg або mjpeg-2000;

$\{P_{ex}\}$ - exe- файли – файли додатків (виконувані файли);

$\{P_{lb}\}$ - файли типів lib та dll а також супутні їм, які є службовими по відношенню до файлів exe, та застосовуються для забезпечення функціонування файлів даного типу;

$\{P_{et}\}$ - файли, що належать іншим типам, беручи до уваги як досить поширені, проте використовувані відносно нечасто, так файли специфічних типів, зокрема такі, які розробник ПЗ використовую виключно у межах власних продуктів.

Отже, тоді структурне наповнення довільного часового слоту $L_p(t)$ вхідного потоку даних, з урахуванням виразу (3.1) у момент деякого гіпотетичного часового відліку t у загальних рисах відповідатиме моделі, що наводиться рис.3.3 [13].

Рисунок 3.3 – Узагальнена модель структурного наповнення часового слоту $L_p(t)$ вхідного потоку

Відтак, перший технологічний крок роботи алгоритму полягає у виокремленні з множини L_p часових слотів $L_p(t)$ з наступним їх послідовним зчитуванням, починаючи з першого, що надійшов у буфер.

Відповідно, далі у слоті $L_p(t)$ серед усих пакетів виявляються ті з них, що понетційно можуть виступати у ролі контейнерів для розміщення зловмисних модулів.

Разом з тим, прийняти на себе роль стегоконтейнеру може досить обмежений перелік типів файлів. Така закономірність, у свою чергу, розповсюджується також на пакети, що дані файли утворюють.

Далі виконаємо детальний розгляд файлів ряду найбільш поширених типів з точки зору ймовірності та, узагалі, можливості їх застосування як стеганографічних контейнерів.

3.1.1 Огляд файлів різних типів з точки зору можливості їх використання у ролі стегоконтейнерів для інкуапсулювання приховуваних даних

У загальному випадку умови, що визначають можливість застосування файлу того чи іншого типу для інкапсулювання даних, є наступними:

1. Максимально високий рівень розповсюдженості файлів даного типу у мережі та на рівні локальних файлосховищах а також кінцевих пристроїв.
2. Максимально високий показник надлишковості η представлення, що напряму впливає на рівень захищеності даних та ймовірну ємність контейнеру. Така залежність у загальному випадку може бути проілюстрованв наступним співвідношенням:

$$\eta\uparrow \rightarrow S, V\uparrow, \tag{3.2}$$

де S - умовний рівень забезпеченої захищеності вбудованих даних, що, за великим рахунком, визначає ефективність стеганографічного алгоритму у цілому;

V - кількість біт, що інкапсулюються до контейнеру того чи іншого типу.

Також обов’язковим до урахування є існуючий взаємозв’язок між показником рівня захищеності вбудованої інформації та об’ємом самої інкапсуляції. Такий взаємозв’язок може бути проілюстровано наступною залежністю:

$$S \sim \frac{1}{V}. \qquad (3.3)$$

Тобто, між згаданими величинами існує обернено пропорційна залежність. Інакше кажучи, з ростом числа V інкапсульованих у контейнер біт спостерігатиметься синхронне зменшення ступеню їх захищеності.

3. Робастність контейнеру. Іншими словами, файл, який планується застосовувати у якості стеганографічного контейнеру повинен цілком зберегти функціональність та цілісність.

4. Розповсюдження файлів даного типу мережею має бути цілком органічним явищем та не повинно викликати підозр.

Результати аналізу перелічених вищзе типів файлів з позиції відповідності вимозі щодо поширеності, наводяться у табл. 3.1

Таблиця 3.1 – Порівняння рівня розповсюдженості файлів різних типів

Тип файлу	Розповсюдженість
Текстові файли, гіпертекст	Висока. Файли присутні у складі усіх файлосховищ на всіх кінцевих вузлах
Файли додатків	Обмежена. Завантаження файлів даних типів з мережевих джерел обмежується або блокується рядом політик безпеки
Файли бібліотек	Обмежена. Пояснюється тим, що даний тип є супутній файлам додатків і окремо завантажується з мережі виключно у ході оновлення ПЗ або за необхідності розширення його функціоналу.
Файли зображень	Висока. Файли присутні у складі усіх файлосховищ на всіх кінцевих вузлах
Аудіо	Висока. Проте гарантовано файли можуть бути присутні у складі файлосховищ та на кінцевих вузлах за умови наявності ПЗ та апаратних пристроїв обробки та відтворення даних файлів
Відео	Висока. Проте гарантовано файли можуть бути присутні у складі файлосховищ та на кінцевих вузлах за умови наявності ПЗ та апаратних пристроїв обробки та відтворення даних файлів
Файли інших типів	Не гарантується. Разом з тим, деякі файли (наприклад, архіви) можуть бути достатньо розповсюдженими

Таким чином, аналізуючи дані, розміщені у таблиці 3.1, можемо зробити висновок про те, що на сьогодні найвищий рівень розповсюдженості зараз відповідає файлам графічного змісту, а також текстові файли різних форматів. Окрім цього, достатньо високий рівень розповсюдженості у мережі властивий також аудіо та відео файлам. Попри це, такі файли гарантовано будуть присутніми у мережі, а також на клієнтських терміналах на той випадок, якщо існуючий мережевий регламент, і окремо - регламент роботи її кінцевих вузлів передбачає ймовірність робота з файлами означених типів.

Далі виконаємо аналіз відповідності файлів вищезгаданих типів другій умові, яка стосується необхідності високого ступеню надмірності η опису, як зазначається табл.3.2.

Таблиця 3.2 – Порівняльний аналіз файлів, що належать різним типам, з позиції середнього рівня надмірності представлення

Тип файлу	Рівень надмірності представлення
Текстові файли, гіпертекст	+
Файли додатків	-
Файли бібліотек	-
Файли зображень	+++
Аудіо	++
Відео	++++
Файли інших типів	+/-

Отже, як можемо бачити за результатом аналізу табл. 3.2, максимальні рівні надлишковості властиві відеоданим, графічним даним та файлам аудіо.

У свою чергу, для файлів текстового типу також характерним є існування певної надмірності, одночас її рівень у загальному випадку значно поступається рівню надмірності файлам попередньо зазначених типів.

У підсумку це дає підставу вважати, що за критерієм надмірності найбільш ефективними для використання у ролі стеганографічних контейнерів є файли зображень, файли відео а також аудіо.

Далі виконаємо порівняльний аналіз файлів різних типів з позиції робастності, тобто, можливості після вбудовування даних зберігати цілісність та вихідний рівень функціональності (таблиця 3.3).

Таблиця 3.3 – Порівняння різних типів фалів з позиції можливості збереження функціональності та цілісності після процедури інкапсулювання приховуваних даних

Тип файлу	Збереження цілісності та функціональності
Текстові файли, гіпертекст	Певною мірою зберігається
Файли додатків	-
Файли бібліотек	-
Файли зображень	+++
Аудіо	+++
Відео	+++
Файли інших типів	+/-

Отже, беручи до уваги дані, розміщені у табл. 3.3, можна констатувати той факт, що цілісність та функціональність після процедури інкапсуляції у повній мірі зберігають графічні, відео та аудіо файли на випадок використання їх у ролі контейнерів.

Таким чином, як свідчить аналіз інформації з табл. 3.1-3.3, вимогам щодо надмірності, робастності та розповсюдженості найповніше відповідають графічні, аудіо та відеофайли.

З урахуванням вищезазначеного, далі необхідно окремо проаналізувати файли, що відносяться до перелічених типів. Тут необхідно визначити, які сами типи файлів задовольняють вимозі 4, тобто, які файли є потенційно підозрілими у разі їх розповсюдження/зберігання.

Розглядається кілька прикладів типів файлів звукових, графічних та аудіоданих (таблиця 3.4).

Таблиця 3.4 – Аналіз файлів з позиції відповідності вимозі стосовно відсутності підозри у ході зберігання та надсилання

Тип даних	Формат файлу	Підозра	Примітка
Звук	wav	+++	Розмір файлу у 5-15 разів перевищує розмір mp3 файлу такого ж змісту та довжини
	ogg	+/-	Даний тип не має досить широкої розповсюдженості у мережі
	Mp3	-	-

Продовження таблиці 3.4 – Аналіз файлів з позиції відповідності вимозі стосовно відсутності підозри у ході зберігання та надсилання

Тип даних	Формат файлу	Підозра	Примітка
Графічні дані	Jpeg, jpg	-	-
	gif	-	Формат опису характеризується низьким рівнем надмірності, що знижує цінність даного типу як контейнеру
	bmp	+++	Розмір файлу у 10-20 разів перевищує розмір jpeg файлу такого ж змісту та візуальної якості
	png	+	Має досить вузьку нішу застосування, що не дозволяє розглядати цей тип як універсальний
Відео	mp4	-	-
	mpg	+++	Розмір файлу до 8-10 разів може перевищувати mp4
	mjpeg	+	Розмір файлу до 5-8 разів може перевищувати mp4
	webm	-	Має відносно обмежену нішу застосовуваності, представлену VoD на базі HTTP

У підсумку бачимо, що вимозі 4 найбільш повною мірою відповідають:
- mp4 з відеофайлів;
- mp3 серед множини аудіоформатів файлів;
- jpeg серед файлів зображень.

3.1.2 Ключові концептуальні моделі стегоаналізу на базі диференційної обробки файлів за рівнем ймовірності застосування у ролі контейнеру

Попередньо виконаний аналіз типів даних на форматів файлів з точки зору відповідності вимогам до стегоконтейнерів свідчить про те, що сукупність типів файлів та їх специфічних форматів, здатних потенційно забезпечити максимальну захищеність стегоканалу зловмисника, є

обмеженою. На рис 3.1 вони їм надано загальну назву - потенційно небезпечні типи даних [11-13].

Отже, далі у випадках, коли усередині деякого довільного часового слоту $L_p(t)$ буде ідентифіковано хоча б один файл чи його фрагмент, що належить до перелічених типів, тоді канал, яким їх було надіслано до вузла, а також – їх джерело, попередньо вносяться до реєстру потенційно небезпечних.

Далі, у ході наступного технологічного кроку обробки вхідного потоку даних, виконується пошук прямих чи опосередкованих ознак існування контейнеру. Такі ознаки, у наслідок впливу ряду чинників, можуть різнитися між собою. На це, зокрема, впливає:

- належність контейнеру до певного типу даних;
- обраний зловмисником формат файлу-контейнеру;
- конкретний алгоритм стеганографічного вбудовування приховуваної інформації, обраний зловмисником.

Відповідно, виходячи з того, який саме тип даних та формат файлу підлягає аналізу, виявлення типових ознак заповненого стегоконтейнеру буде здійснюватися вже для певних умов. Такі умови, за великим рахунком, залежать майже повністю від застосованого зловмисниками стегоалгоритму інкапсулювання даних.

Зазначимо також, що у рамках даної концепції файли, які за результатами аналізу, виконаного у п.3.1.1 визнано потенційно неприйнятними до застосування у якості контейнерів, в умовах браку обчислювальних ресурсів, можуть не підлягати перевірці.

Разом з тим, найбільший рівень поширеності сьогодні відповідає декільком сімействам стеганографічних алгоритмів, що застосовуються для маскування інформації у тих чи інших специфічних умовах, що залежать, у свою чергу, від типу контейнеру на формату файлу.

Отже, виходячи з цього, актуальним є механізми аналізу потенційно небезпечних файлів, що поєднують у собі ряд технологічних етапів перевірки. У ході кожного такого етапу здійснюється пошук ознак контейнеру, утвореного за одним з алгоритмів. Зміна етапів перевірки здійснюється при цьому послідовно.

Застосування зазначеного механізму перевірки зумовлюється впливом таких чинників, як:

- недостатня ефективність універсальних, тобто, тих, що не орієтовані на виявлення ознак інкапсулювання за певним методом, алгоритмів стегоаналізу; найбільшою мірою це актуально в умовах низької заповненості стегоконтейнеру;

- низька ймовірність виявлення ознак контейнеру за однією з прямих ознак, як наслідок високої розповсюдженості нестандартизованих стегоалгоритмів; для усунення даної проблеми рішення про існування/відсутність контейнеру повинно прийматися, керуючись сукупністю виявлених опосередкованих ознак.

Отже далі за тієї умови, що у ході виконаного аналізу було виявлено контейнер, що має ознаку, яка відповідає застосуванню хоча б єдиного стегоалгоритму, поточний пакет даних поміщується з буферу до захищеного сховища для проведення поглибленого аналізу. Це також стосується файлу, до якого відноситься пакет а також файлів, що мають з ним певний логічний зв’язок.

Далі, канал, яким було надіслано ці дані, вноситься у реєстр активно-небезпечних, після цього прийом трафік від нього підлягає повному блокуванню [8 , 9, 11].

Водночас, попри незмінність загального механізму пошуку типових ознак стегоконтейнерів, його реалізація може здійснюватися щонайменше за 2 сценаріями, а саме:

1. Потенційно небезпечні файли та джерела їх надсилання апріорі підлягають блокуванню без обов’язкового виконання попередньої перевірки.

У рамках даного сценарію реалізується:

- вилучення файлів, що викликають підозру, з надходячого потоку даних з наступним поглибленим дослідженням, якому передує їх поміщення у захищене середовище;

- обов’язкове блокуванняя усієї множини пакетів, від джерела, з якого було одержано підозрілий файл;

- прийом рішення щодо необхідності подальшого блокування каналу/джерела, виходячи з результатів глибокої перевірки файлу; при цьому, за відсутності виявлених ознак контейнеру, на деякий обмежений час Δt обмін даними з джерелом поновлюється.

При цьому виконується моніторинг як підозрілих файлів серед потенційно небезпечних типів, так і таких, що не викликають підозри.

2. Дослідження надходячих файлів, що не є підозрілими, але належать до потенційно небезпечних.

Зазвичай саме такі файли зловмисник застосовує для того, щоб розміщувати усередині них приховувані дані, як і було показано таблицею 3.4.

Відтак, з будь-якого довільного часового слоту $L_p(t)$ вилучаються, а далі – досліджуються на предмет наявності інкапсулювання пакети, що містять цілі файли, або окремі фрагменти таких файлів.

Для даного сценарію типовий перелік технологічних етапів обробки вхідних пакетів наступний:

- прийняти до вхідного буферу фрагмент потоку даних, які уміщує у собі часовий слот $L_p(t)$;
- у межах поточного слоту $L_p(t)$ виявити пакети файлів, що відповідають вимогам 1-4;
- виконати аналіз змісту зазначених файлів на наявність ознак заповнених контейнерів;
- виконати блокування каналу у випадку ідентифікування ознак контейнерів.

Згідно з фундаментальними засадами стеганографічного аналізу передбачається, що достатньою підставою блокування каналу є факт виявлення заповненого контейнеру, чи обґрунтована підозра щодо його існування. Такий підхід, у сутності, забезпечує потенційно високий рівень захищеності веб-вузла та і будь-якого кінцевого пристрою.

Інакше кажучи, для того, щоб прихований канал передавання даних, створений на базі стеганографічних алгоритмів, можна було вважати зламаним, немає необхідності розшифрування та реконструкції змісту прихованих повідомлень, так як на цей випадок [8-12]:

- наявність самого факту існування потенційно небезпечного каналу, який при цьому апріорі невідомий для одержувача даних з нього, може вказувати на досить високу ймовірність майбутнього (чи вже триваючого) вторгнення; відтак, після ідентифікації такий каналів, надходження даних від них у першу чергу блокуються; після цього виявлені приховані дані підлягають глибокому аналізу;

- завдання, що полягає у дешифруванні та реконструюванні попередньо прихованих зловмисником даних, розміщених у стегоконтейнерах, відноситься до категорії нетривіальних.

Відтак, для його рішення може потребуватися значний обсяг ресурсів, як обчислювальних, так і часових.

Отже, у межах існуючих технологічних можливостей знаходження такого рішення у режимі реального часу жодним чином не може гарантуватися.

Також необхідно зазначити, що сама процедура стегоаналізу майже завжди потребує значних обчислювальних ресурсів, це є наслідком впливу наступних чинників:

- постійний ріст обсягу трафіку, що надсилається мережею, разом з постійною зміною структурних особливостей його складових, суттєвим чином утруднює процес виокремлення з подальшою збіркою файлів для подальшого аналізу;
- протягом стислого часу τ зазвичай необхідно проаналізувати досить значний обсяг B даних.

Неважко зрозуміти, що за таких умов буде існувати масштабне протиріччя між об’ємами даних, що постійно зростають, а також постійним ускладення їхньої структури з одного боку, а з іншого боку – суттєво лімітованою продуктивністю методів стеганоаналізу та, що є визначальним, обчислювальною спроможністю апаратних платформ, на основі яких забезпечується їх функціонування (рис.3.3).

Іншими словами, на сьогоднішній час питання забезпечення реалізації процесу стеганографічного аналізу відносно усіх даних, що надходять до приймача, які прицьому потенційно можуть використовуватися у ролі стеганоконтейнерів, неможливо вирішити.

Таким чином виходячи з існуючих умов, можемо констатувати, що найбільш актуальним може бути селективний підхід до формування та реалізації заходів зі стегоаналізу.

У наслідок цього, маємо протиріччя, як зазначає рис. 3.4.

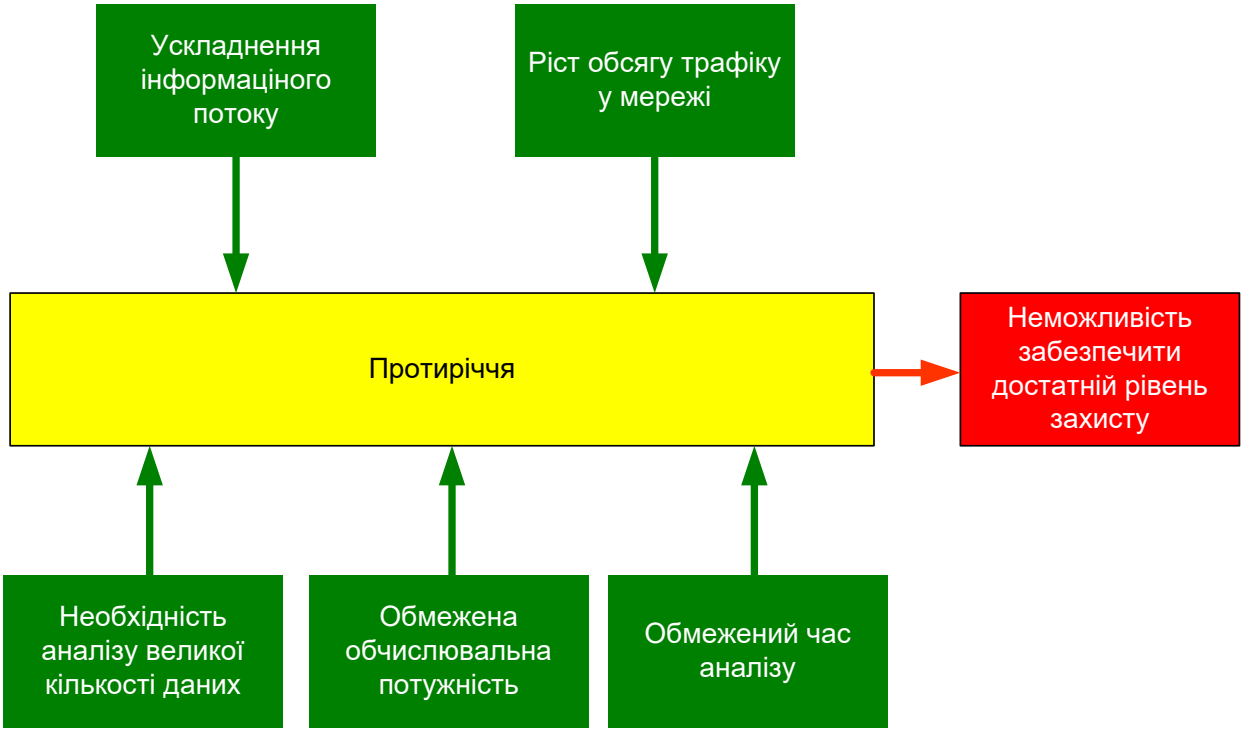


Рисунок 3.4 – Протиріччя, яке має місце у процесі забезпечення захисту веб-вузлів на основі виявлення прихованих стегоканалів

3.2 Побудова методу дослідження вхідного трафіку веб-вузла на базі селективного прицину

У загальному випадку необхідно спільно вирішити такі наступні завдання, як:

- створення умов, за яких можливо реалізувати дослідження вхідного трафіку для виявлення ймовірних маскованих каналів, створених зловмисником зловмисника, у реальному часі;

- забезпечити ефективний аналіз вхідного трафіку за умови, що ресурс доступної обчислювальної потужності можу є обмеженим; іншими словами, алгоритми стегоаналоізу повинні функціонувати з мінімальною часовою затримкою та з максимально незначним навантаженням на апаратну складову вузла. Зазначене вище, у свою чергу, зумовлює актуальність побудови механізмів стеганоаналізу з використанням селективного принципу реалізації [14]. Такий принцип, на відміну від випадків класичної реалізації (коли аналізу підлягає 100% вхідного трафіку), передбачає наступне:

1. З усієї умовної множини J пакетів, які надходять до вузла протягом деякого часу $\Delta t'$, аналізується лише певна частка даних потенційно

небезпечних типів, для цього використуються механізми формування вибірки часових слотів $L_p(t)$.

2. Дослідження пакетів вхідного трафіку може включати у себе 2 технологічні етапи обробки, зокрема такі, як:

- дослідження ділянок даних сі складу часових слотів $L_p(t)$, що можуть бути потенційно небезпечними, оскільки містять файли, та/або їх фрагменти, які можуть бути використані у ролі контейнерів;
- дослідження ділянок підозрілих даних (якщо такі присутні у вхідному потоці).

У рамках реалізації селективного принципу, головним механізмом його може вважатися механізми формування вибірки даних**.**

Далі разглянемо, на яких саме засадах функціонує даний механізм.

Припустимо, що на протязі деякого часового інтервалу $T = \bigcup_{j=1}^{J} t_j$ вхідний буфер веб-вузла прийнає J часових слотів $L_p(t)$, які, у свою чергу, у загальному випадку можуть містити у своєму складі ту чи іншу кількість файлів (фрагментів файлів) яких завгодно типів з тих, що згадувалися раніше (рис. 3.5).

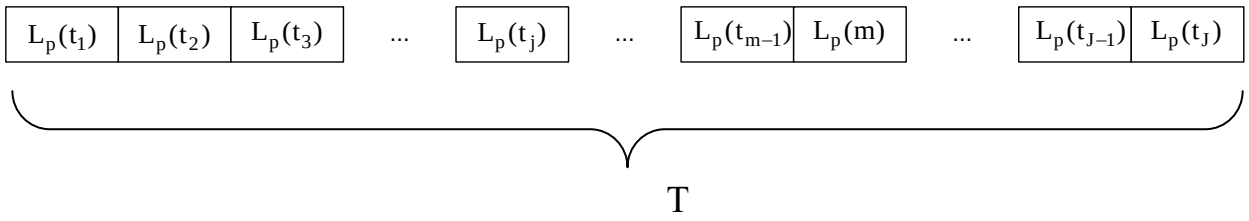


Рисунок 3.5 – Розгляд даних, що надходять до вхідного буферу веб-вузла, у вигляді множини J часових слотів $L_p(t)$

Важливим є те, що з загального часового відрізку T, кіди входить J часових слотів, аналізується деякаий обмежений обсяг Θ з усієї кількості J слотів $L_p(t)$. У цьому випадку найпростіший та тривіальний спосіб утворення множини Θ зводиться до формування вибірки ідентифікаторів t часових слотів, які далі підлягатимуть перевірці. В узагальненому вигляді механізм формування вибірок може бути побудовано за участю таких технологічних кроків, як:

1. Прийняття рішення щодо необхідної розмірності $\ell(\Theta)$, як $\ell(\Theta) = \text{rand}\{J\}$.

2. Знаходження сукупності $\{\Theta\}$ усіх ідентифікаторів t часових слотів $L_p(t)$ керуючись таким виразом:

$$\{\Theta\} = \bigcup_{i=1}^{\ell(\Theta)} \text{rand}\ \Theta_i \qquad (3.4)$$

3. Дослідження масиву часових слотів $\ell(\Theta)$, з яких складається множина $\{\Theta\}$. Очевидно, що такий спосіб формування вибірки часових слотів характеризується простотою реалізації.

Водночас, йому властивий ряд недоліків, що суттєвим чином можуть обмежувати його використання на практиці.

Так, результатом знаходження множини $\{\Theta\}$ ідентифікаторів t слотів $L_p(t)$ та її розміру $\ell(\Theta)$ можуть бути ситуації, за яких:

- розмірність $\ell(\Theta)$ буде замалою; тобто, включатиме у себе невеликий обсяг ідентифікаторів t а отже – зменшується ймовірність виявлення можливих контейнерів;

- розподіл ідентифікаторів t у межах $\{\Theta\}$ матиме нерівномірний характер.

При цьому, як перша, так і друга ситуації можуть бути присутні одночасно. А відтак, ефективний (рівно ймовірний) аналіз даних, що знаходяться у складі певної кількості слотів $L_p(t)$ з часового інтервалу T не може бути гарантовано (рис. 3.5).

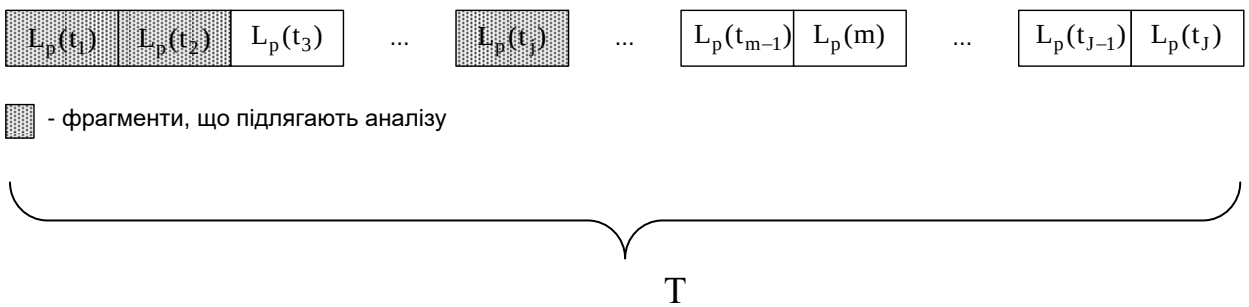


Рисунок 3.5 – Множина $\{\Theta\}$ часових слотів вхідного потоку до вузла, що підлягають аналізу, за відрізок T

Таким чином, як можемо бачити з рисунку 3.5, підхід, що базується на виразі (3.4), жодним чином не забезпечує рівномірного розподілу ідентифікаторів часових слотів для подальшого дослідження, спрямовного на виявлення стеганографічних контейнерів у масиві вхідних пакетів [14, 15].

Зрозуміло, що при цьому завдання забезпечення ефективного дослідження вхідного трафіку з використанням алгоритмів стегоаналізу не може бути вирішено.

Отже, важливим є забезпечити рівно ймовірну вибірку ідентифікаторів t з загального переліку J часових слотів $L_p(t)$ за відрізок часу T. Тоді принцип побудови множини Θ (принцип вибірки часових слотів) пропонується реалізувати на базі модифікованого підходу з використанням рандомізації, як раніше було показано виразом (3.4). Проте зараз знаходження масиву ідентифікаторів $t \in [t_1; t_J]$ часових слотів $L_p(t)$ пропонується виконувати на базі наступної системи виразів:

$$\begin{cases} t = \varepsilon + (t-1), \\ \varepsilon = \mathrm{rand}(J); \end{cases} \tag{3.5}$$

при цьому можливо розглядати початковий індекс (t-1) або фіксованою величиною, яка, приміром, починається з 1, або встановлювати його як випадкову величину, що належить до діапазону J.

У виразі (3.5) змінна ε має назву кроку вибірки. Вона, у свою чергу, повинна відповідати таким вимогам:

$$\varepsilon_{\min} \le \varepsilon \le \varepsilon_{\max}, \tag{3.6}$$

де $\varepsilon_{\max}$ - гранично можливий поріг кроку вибірки, який обчислюється з використанням параметричної величини $\xi = \overline{3;5}$, як:

$$\varepsilon_{\max} = \mathrm{ceil}\left(\frac{J}{\xi}\right), \tag{3.7}$$

де ε_{min} - найменше допустиме значення кроку вибірки, що частыше за все выдноситься до діапазону $\varepsilon_{min} = \overline{1;2}$.

У свою чергу, з виразів (3.5)-(3.7) бачимо, що коли величина J=20, тоді величина кроку вибірки ε належатиме діапазонові величин, що зверзху обмежений значенням 7, а знизу, відповідно, 1-2. При цьому також зрозумілим є те, що величина ξ є обернено пропорційною до верхньої межі діапазону. Інакше кажучи, чим більшим буде значення ξ, тим вужчою буде верхня межу діапазону.

Далі після того, як крок вибірки ε встановлено, тобто, створено умови для визначення ідентифікаторів слотів $L_p(t)$ у межах відрізку T на часовій вісі, здійснюється його скснування, у ході якого дані фрагментів, які увійшли до множини $\{\Theta\}$, зчитуються.

При цьому, коли одержане значення ε не відповідає умовам (3.6), виконується її повторне формування, керуючись виразом (3.5).

Так, для множини $\{\Theta\}$, якій відповідають значення $\varepsilon_{min} = 1$ та $\varepsilon_{max} = 7$ при J=20 приклад розподілу вибірок наводиться рис. 3.6.

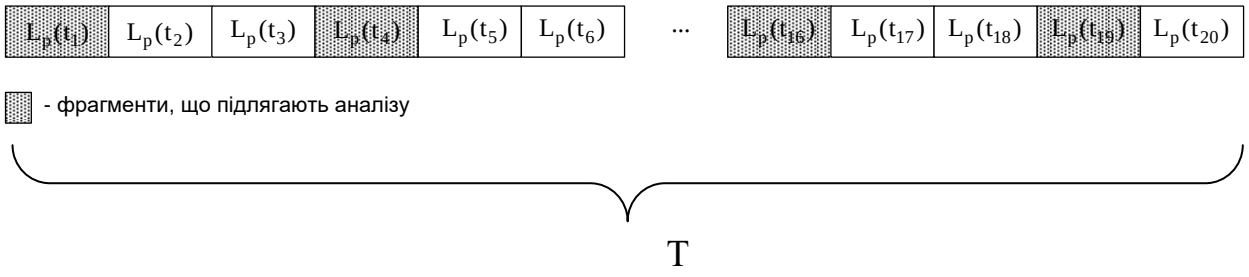


Рисунок 3.6 – Сукупність часових слотів $\{\Theta\}$, утворена для ε=3 та J=20

Таким чином, при цьому створюються умови для можливості рівно ймовірного до усіх часових слотів, що належать відрізку часу T у межах вхідного потоку.

4. ПОШУКОВІ ШАБЛОНИ ДЛЯ ПОСЛІДОВНОГО АНАЛІЗУ ЗМІСТУ ПОТЕНЦІЙНИХ СТЕГОКОНТЕЙНЕРІВ

4.1 Сутність поняття та принципи застосування пошукових шаблонів прихованого змісту у контексті аналізу вхідного потоку даних для захисту кінцевого вузла

Як було виявлено попередньо за результатами огляду файлів, що належать абсолютно різним форматам та типам даних, за найбільшою кількістю критеріїв найбільш прийнятним для застосування як елемент прихованого каналу є деякі файли, що містять графічну інформацію. Дані висновки повністю відповідають результатам ряду досліджень [6, 7].

Приймаючи до уваги зазначене вище, далі будемо розглядати пошукові шаблони, що спрямовані на обробку, у першу чергу, саме графічні файли.

Тут ***пошуковим шаблоном*** може вважатися окремий механізм виявлення стеганографічного вбудовування за одним з алгоритмів інкапсулювання приховуваних даних.

При цьому зазначимо, що, незважаючи на зростаючий попит щодо розробки та впровадження алгоритмів стеганоаналізу (та пошукових шаблонів на їх базі), аналітична галузь деякою мірою знаходиться на стадії розвитку. Це є наслідком, по-перше, відсутності уніфікованих концептуальних засад щодо їх архітектури та можливості подальшої станаризації. По-друге, така ситуація зумовлена досить широким обсягом підходів до реалізації механізмів стегоаналізу як у розрізі їх математичної реалізації, так і стосовно застосовуваності відносно тоги чи іншого типу повідомлень, що обробляються за їх участю [12, 13].

Незважаючи на це, серед існуючого переліку механізмів стеганоналізу існують такі групи, як:

1. Механізми, які фокусаються на той чи інший стеганографічний алгоритм, що відомий апріорі.
2. Уніфіковані механізми, які може бути використано теоретично відносно будь-якого алгоритму вбудовування приховуваних даних.

Для механізмів другої групи справедливо наступне:

- дослідження змісту контейнерів на придмет виявлення ознак стеганографічних модифікацій не потребує відомостей про використаний алгоритм вбудовування даних, криптографічний механізм (якщо

застосовано), застосовані алгоритми кодування, секретні стего- та криптографічні ключі та будь-які відомості щодо прихованого повідомлення;

- для забезпечення ефективності аналітичних механізмів даної групи попередньо необхідна реалізація процесу «навчання», у ході якого виконується ряд обробок масивів заповнених контейнерів та зображень, що не є контейнерами.

Разом з тим, аналітичні механізми обох груп ураховують той факт, що початковий файл (незаповнений контейнер), у який інкапсулюються біти приховуваного повідомлення, є недоступним для аналізу.

Також до першої групу належать ***сигнатурні*** та ***схемні*** алгоритми стеганоаналізу[16, 17].

Так, аналітичні алгоритми сигнатурного типу фокусуються на виконанні синтаксичного аналізі інформації, яку отримує приймача. Така інформація надходить у вигляді послідовності символів, які, у свою чергу, можуть формувати потенційний стеганоконтейнер.

У випадку виявлення належності надходячого на вхід приймача ланцюжка термінальних символів мові, що відповідає певній стегосистемі, робиться висновок про те, що файл, до якого відносяться перевірені дані, являє собою контейнер. Відповідно джерело, що генерує такі, дані, підлягає далі блокуванню. У даному разі термінальні символи - це певна частина, чи повний перелік символів ASCII, що включає у себе символи латинського алфавіту, цифри а також спецсимволи.

Головними перевагами механізмів даної групи є одностайне виявлення факту використання механізмів стеганографічного вбудовування, орієнтованих на приховування секретних даних. При цьому, слід відмітити, що ключовий недолік, який обмежує застосовуваність таких методів, є поріг заповненості контейнеру, нижче якого уніфіковані алгоритми не можуть ефективно застосовуватися. У загальному впадку такий поріг – це 60% від потенційного обсягу контейнеру.

При цьому, відносно схемних методів слід сказати про те[16], що головним сенсом їх застосування є дослідження гіпотези відносно ймовірності існування інкапсуляції прихованих даних з апріорно відомої стегосистеми.

Найчастіше до схемних методів, на базі яких формуються пошукові шаблони, належать:

- метод χ-квадрат;

- методи візуальної оцінки;
- методи виявлення статистичних закономірностей.

Так пошуковий шаблон на базі методу ***χ-квадрату*** найчастіше застосовується для підтвердження чи спростування гіпотези щодо присутності у межах прийнятої послідовності даних ознак контейнерів, які могли бути сформовані за участю програмних пакетів, таких, як OutGuess, Jsteg, Jpeg Hide & Seek та схожого [17].

У цьому випадку береться до уваги інформація щодо закономірностей розподілу статистичних даних у рамках ймовірних контейнерів. При цьому, такі закономірності є характерні для випадків використання програмних засобів ПЗ, переліченого вище, та подібного.

Методи, що належать даному класові, характеризуються такими перевагами, як:

- низька ймовірність того, що у процесі дослідження та за його результатами можуть бути отримані помилкові дані;
- можливість вилучення (з досить високою ймовірністю) прихованих даних за умови, що окрім викриття факту присутності заповненого контейнеру також розпізнається додатково сама стегосистема, хоча вона при цьому не залишає сигнатур у межах контейнеру.

Водночас, аналітичні алгоритми, що базуються на ***візуальній оцінці*** досліджуваних потенційних контейнерів, у класичному варіанті реалізації експлуатують можливості зорової системи людини аналізувати візуальний ряд над ходячої інформації та виявляти схожості та помітні розбіжності у зображеннях, що порівнюються [14, 18].

4.1.1 Метод візуальної оцінки

Метод візуальної оцінки відзначаються найпростішою реалізацією серед інших механізмів дослідження контейнерів графічного типу, так як процес аналізу при цьому зводиться до візуального аналізу експертом попередньо перехопленого та певним чином обробленого зображення.

Зазначимо також, що даний метод аналізу дозволяє встановити певні обмеження на обсяг даних, що можуть інкапсулюватися у контейнер.

Зокрема, якщо контейнером є повноколірне зображення палітри RGB у форматі bmp, непомітними для зорового апарату людини будуть викривлення у діапазоні від 0 до 3% обсягу усіх наявних візуальних даних [19].

З іншого боку, якщо використовуються контейнери на базі файлів jpeg, розпізнати факт інкапсуляції даних, у сутності, не є неможливим. Це є наслідком того, що часткові викривлення, які виникають у результаті інкапсуляції, можуть інтерпретуватися як наслідки некоректної реалізації процедури кодування.

Тобто, очевидним ключовий недоліком даного методу, разом з выдсутністю можливості візуального аналізу jpeg, є цілковита неможливість експерту вести перегляд даних у режимі реального часу [19, 20].

Дана обмеженість методів візуального аналізу є менш суттєвою для методу, що формально належить даній групі, а саме - ***методу візуального дослідження бітових площин контейнеру***.

Ідея даного методу зводиться до [15, 18, 20]:

- декомпозиції контейнеру до рівня бітових площин β_μ (де μ - індекс розряду біт $b(\mu)_{x,y}$) з наступним співставленням окремих площин нижчих порядків з початковим зображенням;

- локалізації аномалій структурного характеру, виявлених на рівні одного чи ряду бітових площин зображення - ймовірного контейнеру.

Тобто, процедура стеганографічного дослідження потенційного контейнеру з використанням візуальної оцінки бітових площин зводиться до поступового перегляду їх змісту на предмет присутності аномалій, не властивих зображенню, що не є контейнером.

При цьому, опис пікселя як компонентній формі, так і початково на рівні RGB за замовчуванням потребує 8 біт на один канал. Відтак, у обох випадках необхідно досліджувати 8 бітових площин.

У цьому випадку, для будь-якого RGB-каналу, або для яскравісних чи колірно-різницевих компонент перша бітова площина β_μ є зображенням, що утворюється сукупністю біт $b(0)_{x,y}$, другий зріз – побудоване на базі сукупності біт $b(1)_{x,y}$ другого розряду і т.д.

Далі кожен з образів бітової пощини β_μ аналізуються шляхом порівняння з досліджуваним зображенням (ймовірним контейнером). Якщо на рівні хоча б однієї площини β_μ фіксується розбіжність її візуалізованого змісту з фактичним зображенням, це дає підставу вважати, що досліджуваний файл є контейнером.

Зазначимо, що метод дослідження бітових площин орієнтується на виявлення вбудовувань, реалізованих на базі методу НЗБ (найменш значного біту).

У класичному варіанті реалізації метод НЗБ є нестійким до даного аналітичного алгоритму. Водночас, ефективність методу візуального дослідження суттєвим чином залежить від того, яким алгоритмом для заповнення контейнеру у межах НЗБ користується зловмисник. Наприклад, у разі інкапсуляції біт приховуваного повідомлення у біти контейнеру, розміщені поряд, чи, навпаки, коли інкапсулювання має рівномірний характер розподілу серед біт контейнеру (зокреме, на базі генератору псевдовипадкових величин), ймовірність виявлення ознаки заповненого контенеру може бути досить високою.

Важливим також є те, що ймовірнісні характеристики приховуваного масиву біт, який необхідно інкапсулювати, у загальному випадку не відповідає статистичним характеристиками молодших біт контейнеру, що не має заповнення.

З цього виходить, що за результатами аналізу бітової площини β_μ, яка міститиме у собі приховувані дані, розбіжність між незаповненою та заповненою ділянками контейнеру буде суттєвою [21].

При цьому, статистичні методи стеганографічного дослідження базуються на понятті т.з. «природного» контейнера.

Базис, на якому побудовано дані методи - це оцінки ймовірності присутності вбудованих секретних даних, виходячи з критерію оцінки близькості контейнера, що підлягає аналізу, до контейнеру «природного» типу.

Статистині методи, у сутності, не обмежені конкретною областю застосування, що, безумовно, є їх значною перевагою. Це є важливим з позиції можливості перевірки гіпотези щодо існуваня стегоконтейнера, який побудовано з використанням стеганографічного алгоритму, що є апріорі невідомим, а також для побудови схемних методів стеганографічного дослідження.

При цьому, слід зазначити, що головним обмеженням усіх статистичних методів якраз і є припущення щодо можливості існування т.з. "природного" контейнера.

4.2 Алгоритми стеганографічного дослідження, що базуються на аналізі статистичних характеристик досліджуваних контейнерів

4.2.1 Алгоритм стегоаналізу на базі розрахунку кількості бінарних переходів значень біт молодших розрядів між сусідніми елементами ймовірного контейнеру

Основу цього алгоритму складає закономірність, відповідно до якої множина молодших біт $b(0)_{x,y}$, що відносяться до сусідніх пікселів $p_{k,\ell}$ або біт $b(0)_{x,y}^{(ch)}$ (де ch – канал) компоненти $\phi_{x,y}$, а аткож усі інші біти т.з. «контейнерів природного типу» поєднані між собою кореляційними зв'язками [14-16].

Так, у тому випадку, коли аналізу підлягають контейнери на базі файлів bmp, елементами масиву, відносно якої необхідно виконувати аналіз, у першу чергу є сукупності біт НЗБ $\{b_{a,b}^{(c)}\} = \bigcup_{a=x\pm1}^{x\pm\nu} \bigcup_{b=y\pm1}^{y\pm\lambda} b_{a,b}^{(c)}$ каналів колірності сусідніх пікселів, які формують зображення.

Водночас, у процесі аналізу потенційних контейнерів на базі файлій jpeg підлягає дослідженню множина $\{b(ch)_{x,y}\} = \bigcup_{a=x\pm1}^{x\pm\nu} \bigcup_{b=y\pm1}^{y\pm\lambda} b(ch)_{a,b}$ НЗБ, розташованих поблизу компонент, величини яких відрізняються від 0 та 1, дані компоненти утворені за результатом виконання ДКП.

При цьому, існуюча залежність між бітами у відповідних розрядах елементів, що належать потенційному контейнеру, може бути описана марківским розподілом.

Окрім цього, більш детальні параметри залежності будуть визначаються індексом розряду.

У даному разі під переходом будемо розуміти зміну величини і-го елемента масиву в значення і+1-го елементу х, де i = 1,2, ..., n -1, n - довжина послідовності.

Так як при цьому масив є двійковим, аналізу підлягають 4 типи переходів, серед яких:

- 0 до 0;
- 0 до 1;
- 1 до 0;

- 1 до 1.

Після цього, на базі одержаних результатів, будується гістаграма. У цьому випадку перший стовпець діаграми выдображає кількість переходів з 0 до 0, у свою чергу, другий - з 0 в 1, далі третій стовпець, відповідно - з 1 до 0 і, у підсумку, четвертий - з 1 в 1, як показано рисунком рис.4.1.

Таким чином досліджуються окремо усі сукупності біт кожного з розрядів (бітові площини β_μ) ймовірного контейнеру.

При цьому, якщо, з одного боку, розглядається незаповнений контейнер (звичайного зображення) а з іншого боку - зображення, куди попередньо було вбудовано дані, тоді кількість переходів (у першу чергу – на рівні НЗБ) для цих двох випадків суттєво різнитиметься.

Наприклад, розподіл біт НЗБ для існуючого стеганографічного контейнера у загальному випадку має випадковий характер. Тобто, загальна кількість переходів F у межах множини НЗБ стосовно усіх зазначених вище станів (типів переходу) умовно буде однаковою. Це, у свою чергу, не є властивим для звичайного зображення що не має вбудовувань [14].

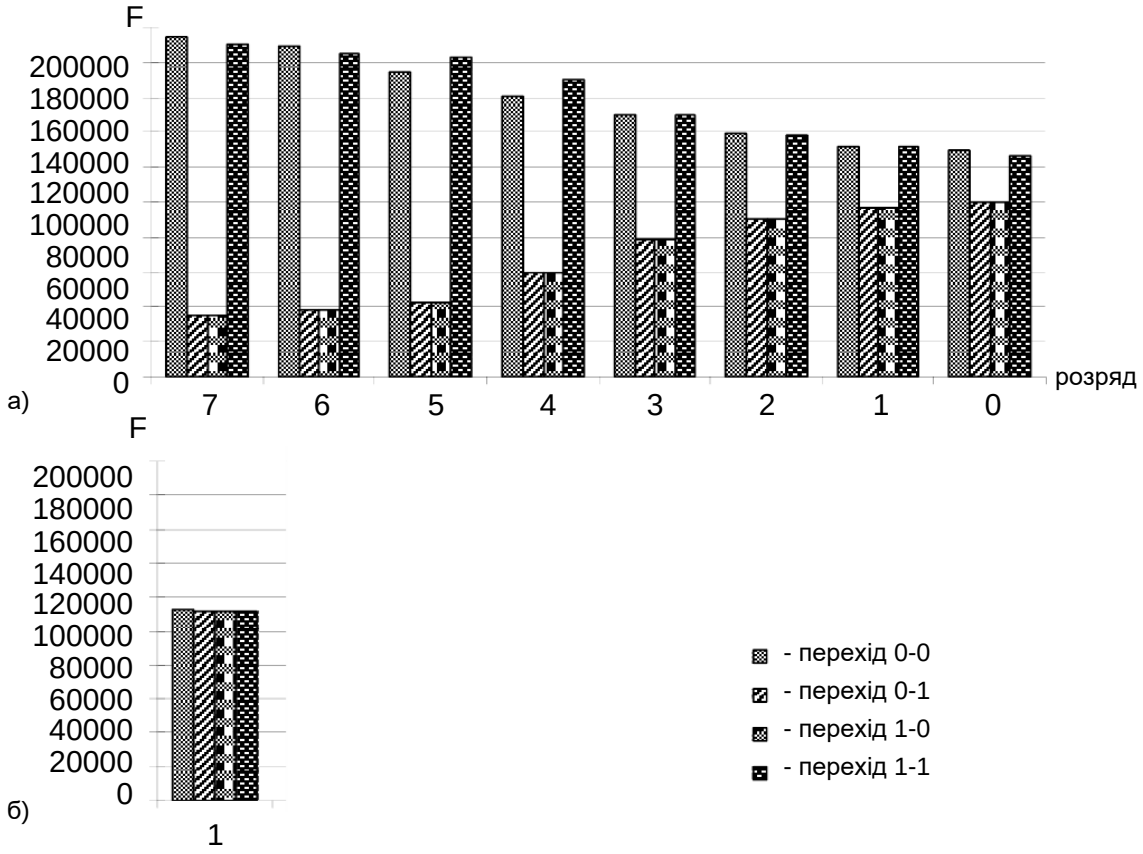


Рисунок 4.1 – Гістограма обсягу переходів бітових величин для звичайного зображення а) для заповненого контейнера б)

4.2.2 Аналітичний алгоритм на базі оцінки потоку НЗБ змісту контейнера з розрахунком частот появи k-бітових серій у його межах

В основу зазначеного алгоритму покладено достілження наявного характеру розподілу елементів, що локалізовані у просторі НЗБ, у ході якого виконується аналіз частоти появи окремих двійкових елементів та серій, що мають у своєму складі у собі k біт [15].

У даному разі, на рівні бітового представлення послідовності θ, що підлягає перевірці, фіксується статистика виявлення нульових та одиничних біт $(k=1)$, серій, що містять 2 елементи $(00, 01, 10, 11 : k=2)$, серій, сформованих трьома елементами $(000, 001, 010, 011, 100, 101, 110, 111 : k=3)$ і т.д. Після того, як статистичні дані зазначеного характеру у межах досліджуваного масиву бінарних даних отримано, далі здійснюється побудова гістаграм.

У даному випадку для побудови гістограми, що ілюструє закономірності виявлених k-бітових серій для потенційних контейнерів формату jpeg, застосовуються частоту появи серій бінарних елементів НЗБ-рівня, що належать компонентам ДКП-перетворення ожного з каналів. При цьому, беруться до уваги компоненти, що є відмінними від значень 1 та 0.

Ключова ідея даного алгоритму полягає у тому, що зображенням як jpeg, так і bmp за умови, що вони не є контейнерами, не властива ситуація наближеної рівності частот появи усіх серій, що наочно демонструє рисунок 4.2 а).

З іншого боку, після виконання процедури вбудовування біт повідомлення, яке приховується, частотність появи серій зазнає змін, приблизно вирівнюючись у значеннях, як показано рисунком 4. 2 б).

Отже, справедливість умов, що зображуються на рис. 4.2.б), є підставою попередньо вважати графічних об’єкт, що підлягає аналізу, заповненим стегоконтейнером.

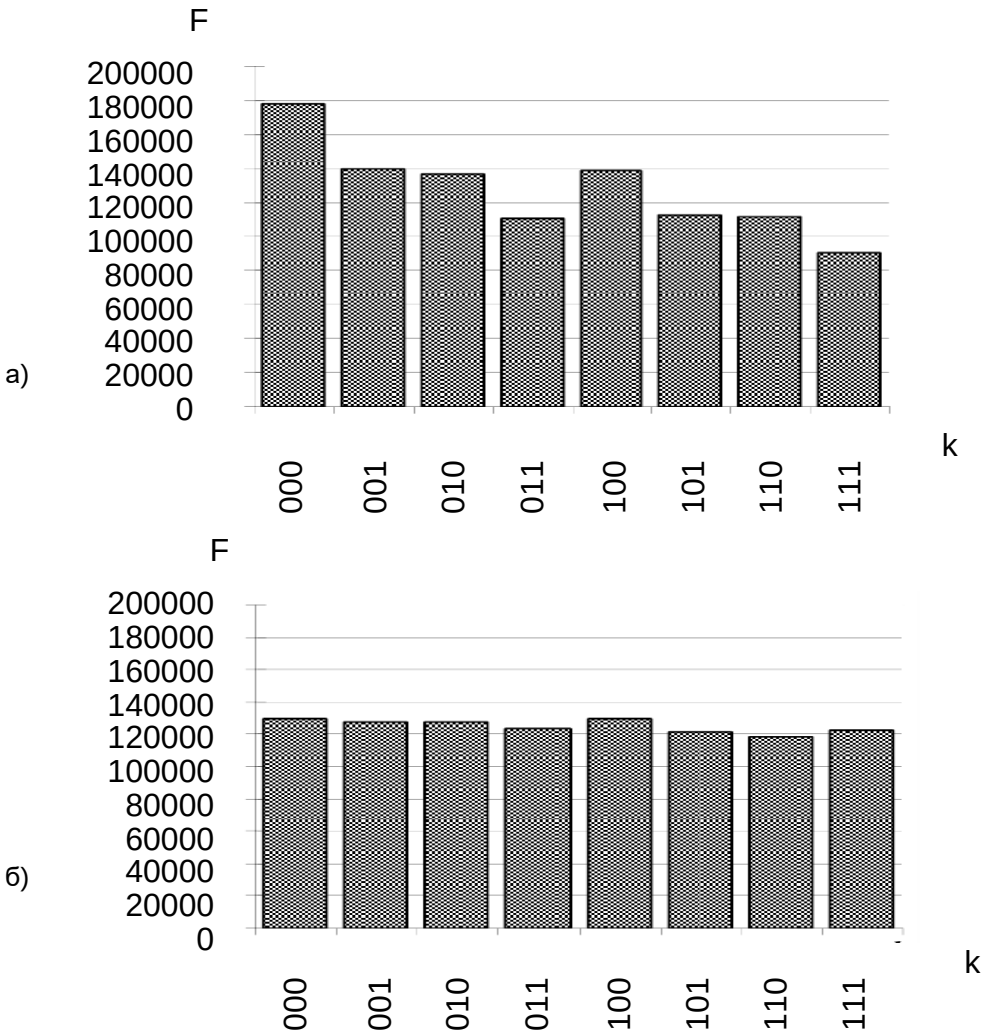


Рисунок 4.2 – Гістограма частот появи серії з трьох елементної на рівні сукупності НЗБ звичайного зображення а) для заповненого контейнеру на базі графічного файлу jpeg б)

Даний алгоритм демонструє високу ефективність в умовах, коли заповненість контейнеру є більшою, ніж на 60 відсотків, тобто, коли виконуються наступні умови:

$$V \geq 60\%. \tag{4.1}$$

У цьому випадку одностайно виявляється факт присутності контейнеру незалежно від того, який саме алгоритм було використано зловмисником для його побудови.

Водночас, якщо умови, зазначені виразом (4.1), не виконуються, результативність роботи алогритму суттєво зменшується та не може

гарантуватися. Це, у свою чергу, і є головним недоліком даного алгоритму. У свою чергу, для збільнешя ефективності загального каскаду застосовуваних пошукових шаблонів, принципи їх функціонування мають суттєвим чином різнитися між собою.

Отже, у відповідності до цього твердження, одним з алгоритмів стегоанаізу, що лежить в основі побудови пошукових шаблонів, може розглядатися алгоритм, що орієнтований на пошук особливостей у розміщенні елементів потенційного контейнері на двовимірній площині.

4.2.3 Алгоритм виявення закономірностей розподілу елементів ймовірного стеганоконтейнеру у двовимірному просторі

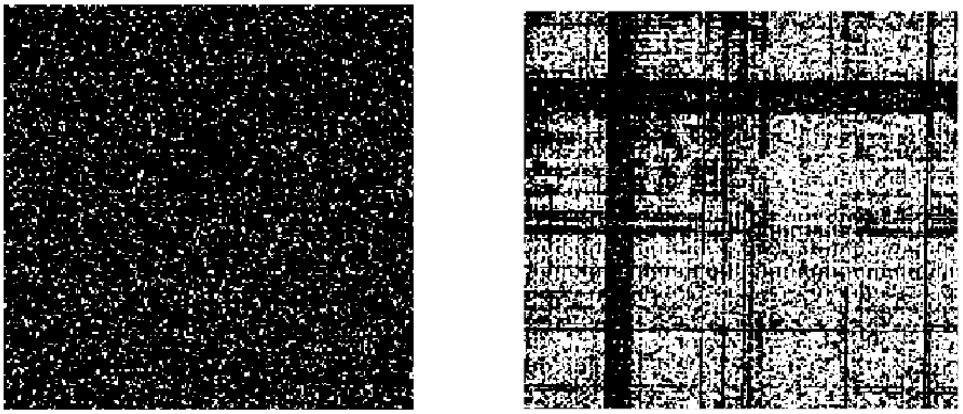
Базис, на якому грунтується даний алгоритм – це виявлення можливих залежностей, які поєднують елементами ймовірного стеганоконтейнеру, що досліджується. Такими елементами, у свою чергу, можуть бути біти та/або компоненти контейнеру [17-19].

При цьому, сам алгоритм передбачає, що у двовимірний масив розмірністю $(2^R - 1) \times (2^R - 1)$, де R - розрядність елементу контейнеру що підлягає перевірці, вносяться елементи з координатами $(x_i; x_{i+1})$. У даному випадку x_i яляють собою елементи аналізованої послідовності.

Водночас, $x, i = 1,2, ..., n-1, n$ - довжина послідовності, що аналізується. Після того, як побудовано розподіл елементів, далі виконується безпосередньо цього аналіз.

При цьому, якщо виявлений характер розподілу точок у межах отриманого поля буде рівномірним, це, к свою чергу, може вважатися ознакою того, що у межах послідовності, що підлягає аналізу, міжелементні залежності фактично є відсутніми. Тобто, у загальному випадку, це може вважатися однією з ключових ознак присутності заповненого контейнера (рис.4.3 а).

В іншщому випадку, за умови, що у структурі поля елементів аналізованої послідовності локалізуватимуться області скупчення елементів, що здатні формувати ті чи іншф геометричні фігури, це можна сприймати як ознаку відсутності інкапсульованої інформації на базі стеганографічних алгоритмів (рис.4.3 б) [15].



а) б)

Рисунок 4.3 – Розподіл елементів у межах площині в умовах присутності стегоконтейнеру а) в умовах, коли контейнер відсутній б)

4.3 Методи візуального дослідження

Як зазначалося попередньо, класичним методам стегоаналізу, що базуються на засадах візуального дослідження, відповідає занизька продуктивність.

Це, у свою чергу, не дозволяє їх застосовувати у чистому вигляді у режими дослідження надходячих пакетів у реальному часі.

Попри зазначене, загальна концепція візуального дослідження є достатньо перспективною у випадках, коли розвиток її реалізації буде здійснюватися за напрямками, що передбачають [20-23]:

- застосування концепції для подальшого розвитку підходів до аналізу НЗБ потенційних контейнерів;
- використання загальних принципів візуального аналізу спільно з методами дослідження контурної інформації у межах ймовірних стегоконтейнерів.

4.3.1 Сутність алгоритму аналізу НЗБ потенційного контейнеру

Підхід, що реалізується у рамках даного алгоритму, за великим рахунком являє собою подальший розвиток традиційних візуальних алгоритмів, удосконалених шляхом додавання механізмів обробки та інтерпретації даних.

За рахунок цього отримується можливість застосовувати алгоритм аналізу НЗБ у реальному масштабі часу.

При цьому, найбільш ефективно алгоритм проявив себе у ході рішення завдання щодо виявлення контейнерів, сформованих з використанням побітового способу інкапсуляції приклади якого показанона рисунках 4.4 та 4.5.

а) б)

Рисунок 4.4 – НЗБ зображення без внесення змін а) та НЗБ контейнеру з використанням побітового способу інкапсуляції б)

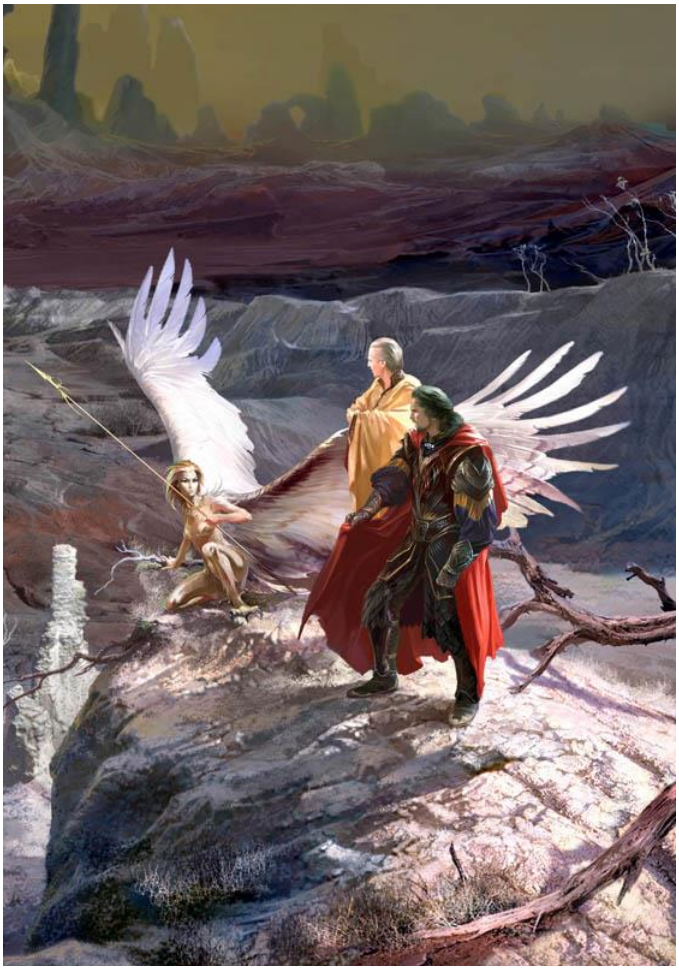


Рисунок 4.5 – Початковий вигляд зображення-контейеру

Таким чином, як бачимо з аналізу змісту рис. 4.4 б, за умови реалізації побітового способу інкапсуляції у множину НЗБ графічного контейнеру, внеситься модифікація характеру розподілу двійкових елементів у нульовому розряді двійкового опису компонент $\phi(ch)_{x,y}$ [19].

4.3.2 Математичний опис алгоритму виявлення ознак інкапсуляцій на рівні НЗБ

Отже для того, щоб мати змогу виявлення фактів вбудовування даних для таких умов без участі експерта, тобто у автоматичному режимі, попередньо необхідно виокремити НЗБ-зміст контейнеру. Для цього спочатку десятковий форма представлення компонент $\phi(ch)_{x,y}$ змінюється до двійкового. Якщо при цьому розглядається jpeg контейнер, то процедура конвертації формату представлення виконується за принципом, що ілюструється виразом:

$$\phi(ch)_{x,y} = \sum_{\theta=8}^{1} b_\theta(ch)_{x,y} \times 2^{(\theta-1)}, \qquad (4.2)$$

де $\phi(ch)_{x,y}$ - компонента, яка відноситься до каналу ch, та знаходиться на позиції (x, y) у межах спектрального сегменту $q_{i,j}$, та утворюється у результаті виконанна операції дискретного косинусного перетворення блоку $q'_{i,j}$ 8x8 вихідного контейнеру; тут індекс ch може приймати значення Y (яскравісний канал), cr або cb, (хроматичний синій та хроматичний червоний канали відповідно);

$b_\theta(ch)_{x,y}$ - біт θ-го розряду у межах каналу ch.

Після того, як компоненти $\phi(ch)_{x,y}$ початкового сегмену $q_{i,j}$, поданого у спектральному просторі, переведеног до двійкового вигляду, далі виконується зчитування сукупності біт нульового (НЗБ) розряду, тобто $\theta = 0$, з наступною побудовою бітової площини $\Phi(ch)^{(\theta)}_{k,\ell}$ відповідно до зазначеного далі принципу:

$$\Phi(ch)^{(\theta)}_{k,\ell} = \bigcup_{x=1}^{H}\bigcup_{y=1}^{W} b_\theta(ch)_{x,y} \,|\, \theta = 0. \qquad (4.3)$$

На той випадок, що перевірці підлягає контейнер на базі bmp, принцип побудови бітової площини буде аналогічним окрім того, що розглядатимуться не компоненти $\phi(ch)_{x,y}$, а пікселі $\psi(ch)_{x,y}$ для яких параметр ch буде приймати значення R, G або B відповідно.

Далі відносно утвореної множини $\Phi(ch)^{(\theta)}_{k,\ell}$, а точніше - $\Phi(ch)^{(0)}_{k,\ell}$, реалізується процедура обчислення середньої довжини серії двійкових елементів. Важливим тут є те, що зазначена процедура виконується послідовно та окремо для рядків і стовпців досліджуваного сегменту $q_{i,j}$ [20, 22].

При цьому, сутність процедури розрахунку середньої довжини $\bar{\ell}^{(r)}_{i,j}$ серії біт за рядками описується виразом:

$$\bar{\ell}_{i,j}^{(r)} = \frac{\sum_{\varphi=1}^{8} \alpha_{\varphi}}{8}, \tag{4.4}$$

де α_{φ} - кількість бінарних переходів, локалізованих у межах одного рядку.

У свою чергу, повністю аналогічно реалізується процедура обчислення середньої довжини $\bar{\ell}_{i,j}^{(c)}$ серії біт за стовпцями, як показано далі:

$$\bar{\ell}_{i,j}^{(c)} = \frac{\sum_{\psi=1}^{8} \alpha_{\psi}}{8}, \tag{4.5}$$

де α_{ψ} - кількість бінарних переходів, локалізованих у межах одного стовпця.

Далі, за результатами знаходження для усіх блоків $q_{i,j}$ спектрального опису контейнеру значень величин $\bar{\ell}_{i,j}^{(c)}$ і $\bar{\ell}_{i,j}^{(r)}$ для умов $\theta = 1$, виконується порівняння пар показників, одержаних для кожного з блоків у його межах між собою.

У цьому випадку за умови, що справедливим є залежність, подана наступним виразом:

$$\bar{\ell}_{i,j}^{(c)} \approx \bar{\ell}_{i,j}^{(r)} \to 7, \tag{4.6}$$

у першому наближенні вважається, що попередньо блок $q_{i,j}$ розглядається таким, до складу якого вбудовано приховувані дані, які, при цому, внесені у контейнер з використанням побітового способу інкапсуляції.

При цьому, умова (4.5) у загальному випадку сприймається як необхідна, проте недостатня інформативна ознака, що однозначно вказує на присутність заповненого контейнеру.

Виходячи з цього, далі для того, спростувати, або – навпаки, підтвердии гіпотезу про те, що зображення, яке досліджується, є

контейнером, необхідно проаналізувати усю сукупність досліджуваних блоків $q_{i,j}$. У даному разі, утворюються дві сукупності блоків $q_{i,j}$, серед яких наступні:

- сукупність $q_{i,j}^{(u)}$ блоків, що відповідають умовам (4.5), відносно яких попередньо робиться висновок про те, що вони можуть бути контейнерами;

- сукупність $q_{i,j}^{(s)}$ блоків, які формально, керуючись умовами (4.5), не є елементами стегосистеми, тобто, не містять модифікацій.

Після того, як множини $q_{i,j}^{(u)}$ та $q_{i,j}^{(s)}$ побудовано, для них далі виконується дослідження їх взаємної локалізації. Така операція передбачає виконання обходу множини $q_{i,j}^{(u)}$, яку було одержано попередньо. У рамках її реалізації початковій координаті задається значення (1,1), що відповідає початку координат зображення-контейнеру.

Для цього блоки $q_{i,j}^{(u)}$ сканується окремо за рядками та за стовпцями, у процесі чого виконується фіксація координат (i,j), які їм відповідають.

При цьому у випадку, що за результатами виконання такого сканування було знайлено блоки $q_{i,j}$, для яких виконується умова (4.6) та які розміщуються безпосередньо поруч у рядку/стовпцю, далі робиться висновок про те, що зображення, яке досліджується, є заповненим стеганоконтейнером [22].

Водночас, слід взяти до уваги те, що тоді, коли умова (4.6) не виконується у повній мірі, а отже - актуальною є наступна залежінсть:

$$\bar{\ell}_{i,j}^{(c)} \vee \bar{\ell}_{i,j}^{(r)} \to 7, \tag{4.7}$$

це, у свою чергу, розглядається як нечітка ознака наявності контейнеру НЗБ. Відтак, за умови справедливості залежності (4.7) приймається рішення щодо доцільності залучення додаткових алгоритмів стеганографічного дослідження.

Так, рис. 4.6 містить приклад процесу аналізу зображення-конейнерну, яке, у свою чергу, являє собою контейнер та одночасно з цим має у своєму складі ділянки, яким, з одного боку, відповідають необхідні ознаки

реалізованого інкапсулювання даних, а з іншого боку, такі ознаки можуть бути у наслідками впливу помилок.

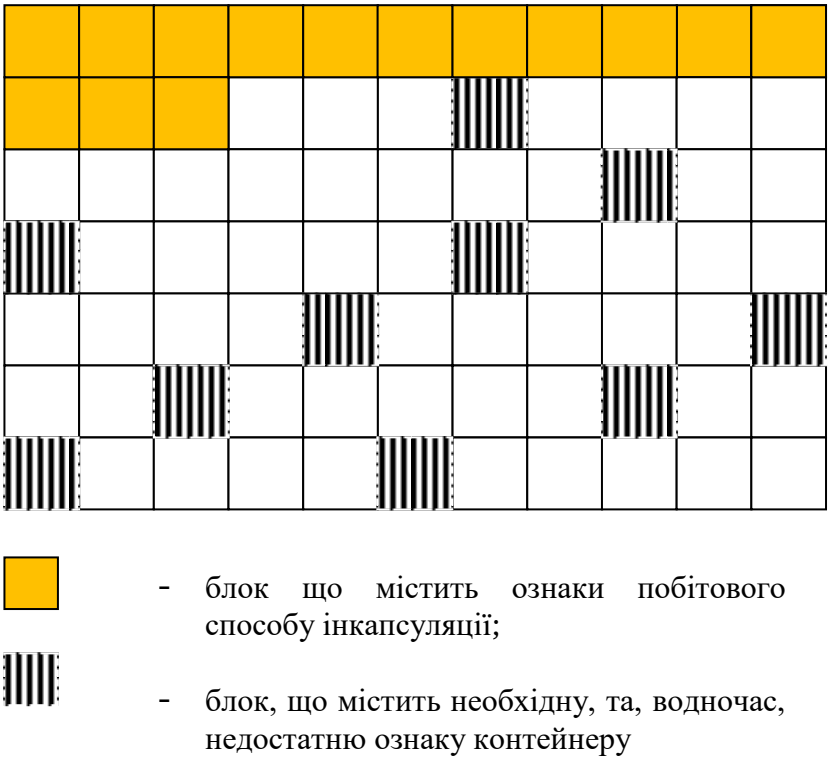


Рисунок 4.6 – Приклад даних, одержаних шляхом виконання аналізу графічного контейнеру з точки зору присутност НЗБ-модифікацій, реалізованих з використанням побітового способу інкапсуляції

4.3.3 Обмеження алгоритму виявлення ознак інкапсуляцій на рівні НЗБ

Розглянутий підхід дозволяє з великою вірогідністю виявити факт внесення змін у НЗБ-область у випадках використання побітового способу інкапсуляції. Це справедливо як для режиму суцільного заповнення, так і для випадків, коли простір НЗБ заповнюється фрагментарно. Проте у ряді випадків підхід демонструє низьку ефективність, а саме:

- коли вбудовування приховуваних даних у простір НЗБ реалізується на засадах, які не орієнтуються на алгоритми побітового способу інкапсуляції;

- інкапуліція приховуваних даних виконується з використанням зон контейнеру, до яких входять т.з. області шумоподібного заповнення «природнього», як демонструють ділянки 1-2, зображені рис. 4.7 а);

- інкапсуліція на НЗБ-засадах реалізується на рівні попередньо виявленої контурної інформації.

а)

б)

Рисунок 4.7 – Вихідне зображення з ділянками шумоподібного заповнення а) НЗБ-простір даного зображення б)

Складність виявлення факту інкапсуляції у наведеному прикладі полягає у тому, що внесені у контейнер зміни на невеликих ділянках, що входять до виділених зон 1 та 2 рис.4.7 а), тим паче без використання побітового способу інкапсуляції, будуть сприйматися як шуми квантування.

Отже, питання їх виявлення при цьому перетворюється на нетривіальну задачу.

4.4 Алгоритм виявлення ознак контейнеру, отриманого на базі методу нерівномірних інтервалів

4.4.1 Сутність методу

Метод орієнтується на модифікацію біт $b_\theta(ch)_{x,y}$ НЗБ на рівні блоків $q_{i,j}$ спектрального формату опису. Проте, на відміну від методів, що орієнтуються на побітовому способу інкапсуляції у його різних варіаціях, у рамках методу масиви біт на рівні НЗБ не вбудовуються. Замість цього використовується інкапсуляція даних δ_ζ приховуваного повідомлення у окремі біти $b_\theta(ch)_{x,y}$ контейнеру, що знаходяться один від одного на деякій відстані d.

Так, на першому технологічному кроці роботи методу виконується транспонування матриці H простору НЗБ, за результатами чого утворюється вектор $\vec{H}$ біт НЗБ, тобто:

$$\vec{H} = H^T \tag{4.8}$$

Операція, зазначена виразом (4.8) необхідна для того, щоб спростити алгоритм вбудовування, оперуючи далі одномірним вектором. При цьому, величина d_ζ зміщення позиції розраховується у діапазоні $d_\zeta = \overline{2;\Xi}$, де Ξ - кількість біт у приховуваному повідомленні, розраховується як алгебраїчна сума значень біт $b_\theta(ch)_{x,y}$ на позиції (x,y), з якої починається процес інкапсуляції, тобто:

$$d_\zeta = \sum_{\theta=1}^{7} b_\theta(ch)_{x,y}, \tag{4.9}$$

тут розрахунок суми виконується починаючи з 2 бітової площини, тобто, з $\theta = 1$, відповідно, біт з простору НЗБ не береться до уваги, що необхідно для одностайності зчитування прихованих даних.

Таким чином, кожна наступна позиція $(\zeta+1)$, у яку буде вбудовано символ (на рівні компоненти $\phi(ch)_{x,y}$), визначається як:

$$(\zeta+1)=\zeta+d_{\zeta}. \tag{4.10}$$

У такому випадку, ураховуючи довільний зміст компонент $\phi(ch)_{x,y}$, слід зазначити, що у загальному випадку справедливим є наступне твердження:

$$[d_{\zeta};d_{\zeta+1}]\neq[d_{\zeta+n};d_{\zeta+n+1}] \tag{4.11}$$

Відтак, схемаично процес розподілу позицій біт для інкапсулювання може бути подано рис. 4.8.

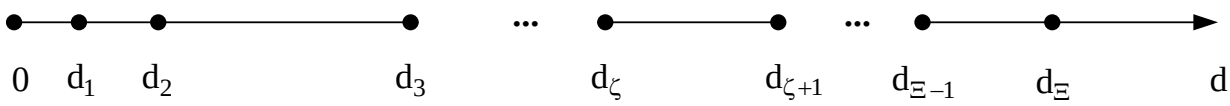


Рисунок 4.8 – Графічна інтерпретація процесу вибору позиції для інкапсуляції біта приховуваного повідомлення у рамках методу нерівномірних інтервалів.

Як видно з аналізу алгоритму, характер розподілу біт прихованого повідомлення у межах контейнеру, по-перше, наслідуватиме ознаки шуму квантування.

По-друге, виходячи з того, що, ураховуючи особливості архітектуру методу, заповненість навіть області НЗБ за його участю апріорі не буде більшою, ніж 60%.

Таким чином, використання універсальних статистичних методів у даному разі не буде доцільним.

Водночас, так як відстань d зміщення наступної точки інкапсуляції статистично у більшості випадків $d>0$, відповідно, методи виявлення ознак інкапсуляцій, поданий виразами (4.4 – 4.7) у цьому випадку також не буде ефективним.

Таким чином, для виявлення ознак інкапсулювання за даним методом, необхідно задіювати специфічний пошуковий шаблон.

4.4.2 Пошуковий шаблон для виявлення ознак контейнеру, утвореного за методом нерівномірних інтервалів

Складність реалізації пошукового шаблону для виявлення ознак інкапсулювань зумовлена, окрім чинників, зазначених у п. 4.4.1, невизначеністю, пов'язаною з тим, що апріорі невідомо, з якої саме позиції буде розпочато вбудовування даних.

Теоретично стартова позиція (x,y) інкапсуляції може приймати які завгодно значення, тобто, тут, у принципі, $x = \overline{1; M}$ та $y = \overline{1; N}$.

Разом з тим, найчастіше для початку вбудовування даних застосовується координата або x=0, та/або y=0.

Отже, при формальній побудові процесу виявлення ознак контейнеру, необхідно виконати аналіз змісту біт НЗБ за двома сценаріями, а саме:

- коли транспонування (вираз (4.8)) виконано відносно рядків;
- коли транспонування реалізовано за стовпцями.

Тоді, теоретично, за кожним зазначеним сценарієм може бути утворено M або N бінарних послідовностей $b(M)$ та $(b(N))$, для яких справедливо наступне:

$$\begin{aligned} b(M) &\neq b(N) \\ \ell(b(m)) &= \overline{M;1}; \\ \ell(b(n)) &= \overline{N;1}. \end{aligned} \qquad (4.12)$$

Тобто, так як невідомою є стартова точка (x,y) інкапсулювання, аналізу може підлягати уся множина n-x та m-x двійкових послідовностей, довжини яких змінюватимуться у діапазонах, як показано системою виразів (4.12).

Розглянутий підхід до виявлення ознак інкапсуляції за даним алгоритмом є дієвим при незначних величинах M та N.

Разом з тим, з ростом значень M та N можливість рішення завдання реконструкції масиву ймовірних інкапсуьованих біт з їх подальшим дослідженням обмежується доступним обсягом обчислювальної потужності.

4.5 Пошуковий шаблон для виялення ознак модифікації НЗБ на базі виявлення контурного змісту контейнеру

За великим рахунком, розпізнавання самого факту розміщення (певним чином) приховуваних даних у контур зараз на сьогодні є особливо складним завданням, як наслідок того, що:

- заповнюваність стеганоконтейнеру на базі контурної складової НЗБ зазвичай є не суттєвою, що рідко перевищує поріг 1%, тим самим розлячи недоцільним застосування методів статистичного дослідження;

- на результативність процедури аналізу НЗБ, у ході якого розраховується середні довжини бінарних переходів у рамках кожного блоку $q_{i,j}$, суттєвим чином впливають особливості змісту зображення, при цьому, що очевидно, результативність не гарантується.

Виходячи з цього, для випадку необхідності перевірки контурних областей, доцільно використати методи локалізації контурів у межах зображень [20, 22], який передбачається застосовувати спільно з алгоритмом дослідження НЗБ [18, 20].

Так, на першому етапі роботи у межах аналізованого зображення, що розглядається як потенційний контейнер, виявляються контурні складові, що на наступному технологічному етапі підлягають поглибленому аналізу.

У даному випадку для локалізації контурів можуть використовуватися такі підходи, як:

- виявлення контурів на базі методу ковзаючої маски;
- виявлення ознак присутності контуру б блоці $q_{i,j}$ на базі дослідження його статистичних характеристик.

При цьому, з точки зору мінімізації обчислювального навантаження (на випадок аналізу змісту надходячих даних у реальному часі це є гостро актуальним), доцільним для застосування є другий підхід. У рамках даного підходу пропонується двотактова схема пошуку ознак присутності контуру[18].

Так, протягом першого такту виконується обчислення нормованого усередненого динамічного діапазону $d_{nrm}(q_{i,j})$ блоку на рівні рядків та стовпців, відповідно до наступного принципу: $\psi_{x,y}^{(max)}$

$$\begin{cases} d_{nrm}^{(x)}(q_{i,j}) = \prod_{x=1}^{8} \frac{(\psi_{x,y}^{(max)} - \psi_{x,y}^{(min)})}{8}; \\ d_{nrm}^{(y)}(q_{i,j}) = \prod_{y=1}^{8} \frac{(\psi_{x,y}^{(max)} - \psi_{x,y}^{(min)}),}{8} \end{cases} \quad (4.13)$$

де $d_{nrm}^{(x)}(q_{i,j})$ - нормований усереднений динамічний діапазон блоку на рівні рядків;

$d_{nrm}^{(y)}(q_{i,j})$ - нормований усереднений динамічний діапазон блоку на рівні стовпців;

$\psi_{x,y}^{(max)}$ та $\psi_{x,y}^{(min)}$ - локалізовані найбільша та найменша величина пікселя у рядку/стовпцю.

Після цього, у ході поточного такту, у межах ймовірного контейнеру, що перевіяється, знаходиться піксель, з максимальним значенням $d_{nrm}^{(x)max}(q_{i,j})$ та/або $d_{nrm}^{(y)max}(q_{i,j})$. Далі встановлюється величина $d_{int}^{(max)}(q_{i,j})$ у відповідності з виразом:

$$d_{int}^{(max)}(q_{i,j}) \mid d_{nrm}^{(x)max}(q_{i,j}) \vee d_{nrm}^{(y)max}(q_{i,j}). \quad (4.14)$$

Відповідно у випадку, коли для довільного блоку $q_{i,j}$ контейнеру виконується наступна умова:

$$d_{int}(q_{i,j}) \in [d_{int}^{(max)}(q_{i,j}); \vartheta d_{int}^{(max)}(q_{i,j})], \quad (4.15)$$

такий блок попередньо буде віднесено до множини C' ймовірно-контурних. У свою чергу, множник ϑ з виразу (4.15) не є фіксованим та знаходиться у діапазону $\vartheta = \overline{0,6; 0,9}$. При цьому, конкретна величина з діапазону використовується виходячи з наявних особливостей контейнеру, що аналізується.

Під час наступного, другого такту, виконується перевірка гіпотези про попередню належість блоку $q_{i,j}$ до множни C', що виконувалося за виразами (4.14) та (4.15).

У ході даног такту, для кожного з блоків $q_{i,j}$, попередньо віднесених до множини C', виконується розрахунок величини $\Omega(\psi_{x,y})$ мультиплікативної потужності компонент $\psi_{x,y}$ блоку $q_{i,j}$. Для цього використовуються компоненти, починаючи з $\psi_{1,1}$ (на позиції (1,1)) до $\psi_{4,1}$, величини яких зчитуються у ході сканування за принципом зиг-загу, який є стандартним для jpeg. При цьому, буде задіяно 10 компонент НЧ-зони блоку. Саму процедуру розрахунку величини $\Omega(\psi_{x,y})$ демонструє виразом:

$$\Omega(\psi_{i,j}) = \prod_{\delta=1}^{10} \psi_{x,y}^{(\delta)}, \tag{4.16}$$

де $\psi_{x,y}^{(\delta)}$ - компонента, що належить до зони низьких частот блоку $q_{i,j}$.

Далі так само, як і у випадку формули (4.15), орієнтуючись на значення величини $\Omega(\psi_{i,j})^{(max)}$, встановлене попередньо, у межах попередньо сформованої множини C' локалізуються блоки $q_{i,j}$, для яких виконується така умова:

$$q_{i,j} \in [\Omega(q_{i,j})^{(max)};\ \mu\Omega(q_{i,j})^{(max)}], \tag{4.17}$$

де μ є також множником, значення кого також нижче 1, при цьому його конкретне значення також взначається експериментальним шляхом.

У решті решт, процедура визначення належності блоку $q_{i,j}$ сукупності множини C блоків контурного типу реалізується відповідно до виразу:

$$q_{i,j} \in C \mid (q_{i,j} \in C') \ \ \& \ \ (q_{i,j} \in [\Omega(q_{i,j})^{(max)};\ \mu\Omega(q_{i,j})^{(max)}]) \tag{4.18}$$

Після цього, як множини C блоків у межах поточного контейнеру виявлено, далі відносно кожного з них виконуються дії, спрямовані або на більш глибокий аналіз контурних зон, або на блокування ймовірного зловмисного змісту, вбудованого у НЗБ-простір контейнеру.

4.6 Обробка виявлених контурних областей на випадок протидії НЗБ-модифікаціям

Залежно від того, якому типу трафіку за рівнем пріоритетності належить поточний контейнер, що було включено до множини C, відносно нього може бути задіяно одну з наступних стратегій обробки:

1. Поглиблений аналіз контурної складової з застосуванням переліку пошукових шаблонів у локальній області. Сенс такого рішення зумовлюється різною статистикою розподілу бінарних елементів у цілому контейнері та у межах контурних зон.

Дана стратегія є актуальною для трафіку, не критичного щодо затримки, наприклад, для типів scavenger class чи best effort.

2. Вилучення пакету, що містить фрагменти контейнеру, з потоку, з наступним дослідженням у «пісочниці». Стратегія може бути застосована відносно трафіку додатків та сервісів, що допускають деякий відсоток втрат пакетів.

3. Часткове руйнування контейнеру. Дана стратегія актуальна у випадках, коли досліджуваний файл, з одного боку, має часткові ознаки ознаки контейнеру, чого не достатньо для класифікації його як такого, що може містити приховані дані. При цьому, з іншого боку, даний файл належить трафіку сервісу/додатку, для яких втрата пакетів та затримка їх надходження є критичною з позиції збереження функціональності.

Відтак, обробка ймовірних контейнерів у таких випадках передбачає часткове їх руйнування, без зниження їх функціональності. Як раніше було зазначено, зокрема, для графічних пакетів це є припустимим, що зумовлено:

- високою робастністю графічних файлів;
- значним рівнем надмірності;
- специфікою реалізації більшості стеганографічних алгоритмів, що передбачають інкапсуляцію даних в області зображень, не критичні з точки зору їх візуального сприйняття.

Наприклад, для блокування даних, інкапсульованих у контейнер на базі методу нарівномірних інтервалів, НЗБ-модифікації у режимі як побітового заповнення, так і у прив’язці до контурів може бути використано такі механізми, як [18, 21]:

1. Накладення бінарної маски на матрицю НЗБ.

Для цього попередньо генеруються двомірні масиви випадкових бінарних величин $b_0(Y)_{x,y}^{(rnd)}$, $b_0(cb)_{\nu,\upsilon}^{(rnd)}$ та $b_0(cr)_{\nu,\upsilon}^{(rnd)}$ для кожного з каналів – яскравісного та хроматичногоо опису, елементи кожного з яких отримуються за результатом виконання операції:

$$\begin{cases} b_0(Y)_{x,y} := \mathrm{rand}[1;0]; \\ b_0(cb)_{\nu,\upsilon}^{(rnd)} := \mathrm{rand}[1;0]; \\ b_0(Y)_{x,y}^{(rnd)} := \mathrm{rand}[1;0]; \end{cases} \quad x = \overline{1; M}, x = \overline{1; N}; \quad \nu = \overline{1; M/\sigma}, \upsilon = \overline{1; M/\varsigma}, \tag{4.19}$$

де M та N - розмірність маски для яскравісної складової, вона дорівнює розміру блоку контейнеру;

M/σ та M/ς - розмірності масок для хроматичних складових НЗБ;

σ та ς - дільники, величина яких залежить від поточного режиму колірної субдискретизації.

Далі виконується по елементне складення за модулем 2 величин біт НЗБ та згенерованих випадкових величин на відповідних координатах, тобто:

$$\begin{cases} b_0(Y)_{x,y} := b_0(Y)_{x,y} \ \mathrm{XOR}\ b_0(Y)_{x,y}^{(rnd)}; \\ b_0(cb)_{\nu,\upsilon} := b_0(cb)_{\nu,\upsilon} \ \mathrm{XOR}\ b_0(cb)_{\nu,\upsilon}^{(rnd)}; \\ b_0(cr)_{\nu,\upsilon} := b_0(cr)_{\nu,\upsilon} \ \mathrm{XOR}\ b_0(cr)_{\nu,\upsilon}^{(rnd)} \end{cases} \tag{4.20}$$

Це, у свою чергу, веде до повного руйнування НЗБ складових у кожному з каналів, не вносячи при цьому суттєвого викривлення у контейнер та зберігаючи його семантичні цілісність.

2. Обнулення змісту НЗБ.

Даний механізм передбачає заповнення матриць НЗБ кожного з каналів нульовими бітами:

$$\begin{cases} b_0(Y)_{x,y} := 0; \\ b_0(cb)_{\nu,\upsilon} := 0; . \\ b_0(cr)_{\nu,\upsilon} := 0 \end{cases} \tag{4.21}$$

У результаті цього також досягається ефект, аналогічний випадку застосування бінарної маски випадкового змісту.

ВИСНОВКИ

Згідно з вимогами чинного технічного завдання, під час виконання кваліфікаційної роботи було здійснено:

1. Аналіз ймовірних каналів надходження потенційних загроз для веб-вузла, серед яких, зокрема:

- вузли внутрішньої мережі, що мають доступ до самого ресурсу (адміністратор ресурсу, контент-менеджмент, адміністратори розділів тощо);

- зовнішні джерела.

2. Дослідження сутність та механізмів реалізації атак, побудованих за APT-ідеологією.

3. Виявлення ролі методів стеганографічного маскування даних, як елементу реалізації розвинутих стійких загроз.

4. Аналіз структури трафіку, що надходить як до веб-вузла, так і до клієнтського вузла що бере участь у адмініструванні веб-узлів.

У ході цього визначено потенційний перелік типів файлів та їх форматів, що першочергово можуть бути використані зловмисниками у якості стаганоконтейнерів.

5. Побудову стратегії обробки трафіку реального часу, у ході якої виконується аналіз найбільш критичних з позиції безпеки його складових. Така стратегія базується на механізмі рівноймовірної вибірки окремих пакетів з надходячого потоку.

За рахунок цього створюються умови для:

- забезпечення стегоаналізу потенційно підозрілих пакетів у реальному часі;

- локалізації та подальшого блокування зловмисного контенту з високою ступінню ймовірності.

У рамках даної стратегії передбачається:

- послідовне застосування уніфікованих та спеціалізованих алгоритмів стегоаналізу у єдиному технологічному циклі;

- вироблення рішення за результатами комплексного аналітичного дослідження, що може включати у себе заходи з блокування трафіку та його джерела, встановлення дозволу на прийом трафіку або застосування заходів поглибленого аналізу пакетів у захищеному середовищі.

6. Дослідження аналітичних алгоритмів, спрямованих на виявлення модифікацій у межах графічних контейнерів.

У ході цього було досліджено принципи побудови уніфікованих пошукових шаблонів, а також шаблонів, орієнтованих на виявлення ознак контейнерів, утворених з використанням НЗБ-підходу, а саме:

- за методом побітового вбудовування;
- з використанням контурних областей;
- на базі нерівномірних відрізків.

Також виконано дослідження механізмів обмеженого руйнування ймовірного контейнеру зі збереженням його семантичної цілісності для блокування зловмисного змісту на випадок, коли за формальними ознаками чіткого рішення про наявність модифікації не може бути прийнято.

Таким чином, кожен з пунктів технічного завдання виконано у повній мірі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Microsoft security report [Електронний ресурс] – Режим доступу: https://microsoft.com/securityinsights.
2. Кибератаки – определение, виды, профилактика [Електронний ресурс] – Режим доступу: https://techarks.ru/category/security/ Кибератаки – определение, виды, профилактика.html.
3. Противодействие целевым атакам (APT) и угрозам 0 дня (Zero Day) – Octava.ua [Електронний ресурс] – Режим доступу: https://octava.ua/services/products-and-solutions/protivodejstvie-zero-day-atakam.
4. Противодействие APT-атакам: автоматизация с MessilaBot или традиционные методы? [Електронний ресурс] – Режим доступу: http://nbj.ru/publs/banki-i-biznes/2016/12/19/protivodeistvie-art-atakam-avtomatizatsija-s-messilabot-ili-traditsionnye-metody/?full.
5. WTF is APT? Продвинутые атаки, хитрости и методы защиты [Електронний ресурс] – Режим доступу: https://xakep.ru/2018/07/20/wtf-is-apt/.
6. Can we test ATP defenses even if we can’t agree on how to define APTs? [Електронний ресурс] – Режим доступу: https://news.sophos.com/en-us/2015/10/23/can-we-test-apt-defenses-even-if-we-cant-agree-on-how-to-define-apts.
7. APT infographics [Електронний ресурс] – Режим доступу: https://sophos.files.wordpress.com/2014/04/apt-infographic_wr.pdf.
8. Bejtlich R. The Practice of Network Security Monitoring: Understanding Incident Detection and Response / Richard Bejtlich. – San Francisco: Search Press Inc, 2013. – 341 с.
9. Shostack A. Threat Modeling: Designing for Security / Adam Shostack., 2014. – 624 с.
10. Конахович Г. Ф., Пузыренко А. Ю. Компьютерная стеганография. Теория и практика. - К.: МК-Пресс, 2006. - 288 с
11. Шаньгин В.Ф. Информационная безопасность и защита информации. ДМК-Пресс., 2017, 702 с.

12. Шелухин О.И., Канаев С.Д. Стеганография. Алгоритмы и программная реализация. Горячая линия – Телеком, научно-техническое издательство 2017, 592 с.
13. Рябко, Б.Я. Основы современной криптографии и стеганографии [Электронный ресурс] : [монография] / А.Н. Фионов, Б.Я. Рябко. — М. : Горячая линия – Телеком, 2010 .— 233 с.
14. Грибунин В. Г., Оков И. Н., Туринцев И. В. Цифровая стеганография. М.: СОЛОН-Пресс, 2016, - 315 с.
15. Алексеев, А.П. Стеганографические и криптографические методы защиты информации : учеб. пособие по дисциплине "Информатика" / В.В. Орлов, А.П. Алексеев .— Самара : ИУНЛ ПГУТИ, 2010 .— 289 с.
16. Гизунов Д.С. Методика автоматизированного обнаружения скрытой информации в компьютерных файлах / Д.С. Гизунов, О.А. Демченко, Е.И. Никутин // Известия ТРТУ. – 2006. – Т. 71, № 16. – С. 49-53.
17. Provos N. Detecting steganographic content on the internet / N. Provos, P. Honeyman. // Technical Report CITI 01-1a, University of Michigan, 2001.
18. Fridrich Y. Steganography in Digital Media: Principles, Algorithms and Applicaticks. Cambridge Press, 2010. 462 p.
19. Кустов В.Н., Параскевопуло А.Ю. Простые тайны стегоанализа / В.Н. Кустов, А.Ю. Параскевопуло // Защита информации, INSIDE. – 2005. – № 4. – С. 72-78.
20. Быков С. Ф. Алгоритм сжатия JPEG с позиции компьютерной стеганографии Защита информации. Конфидент. - СПб.: 2000, № 3
21. Юренский П.В. МЕТОДЫ СТАТИСТИЧЕСКОГО И НЕЙРОСЕТЕВОГО СТЕГОАНАЛИЗА СКРЫТЫХ КАНАЛОВ // Инновации в науке: научный журнал. – № 1(89). – Новосибирск., Изд. АНС «СибАК», 2019. – С. 11-13.
22. Голуб В.А. Комплексный подход для выявления стеганографического скрытия в JPEG-файлах / В.А. Голуб, М.А. Дрюченко // Инфокоммуникационные технологии. – 2009. – Т. 7, № 1. – С. 44-50