НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмуванняіспеціалізованих комп‘ютернихсистем

«До захисту допущено»

Завідувач кафедри

_____ТарасенкоВ.П.
(підпис) (ініціали,прізвище)

“___”червня2019р .

Дипломний проект

на здобуття ступеня бакалавра

знапрямупідготовки **6.050102«Комп'ютернаінженерія»**

на тему: “Система захисту інформаційних потоків в комп‘ютерних мережах SDN””

Виконав: студент IV курсу, групи КВ-52

Здирко Владислав Володимимрович ____
(підпис)

Керівник доц. каф. СПіСКС,к.т.н.ОрловаММ. _____ _____
(підпис)

Консультантз нормоконтролю, доц.каф.СПіСКС,к.т.н.КлятченкоЯМ._____
_____ (підпис)

Рецензент_____
(посада,науковийступінь,вченезвання,науковийступінь,прізвищетаініціали) (підпис)

Засвідчую, що у цьому дипломномупроекті немає запозичень з праць інших авторів без відповіднихпосилань.
Студент_____
(підпис)

Київ – 2019 року

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп‘ютерних систем

Рівень вищої освіти – перший (бакалаврський)

Напрям підготовки 6.050102 «Комп’ютерна інженерія»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ТарасенкоВ.П.
(підпис) (ініціали, прізвище)
«___»червня2019 р.

ЗАВДАННЯ
на дипломний проект студента

Здирка Владислава Володимировича

1. Темапроекту:«Система захисту інформаційних потоків вкомп’ютернихмережахSDN’»,керівникпроектуОрловаМаріяМиколаївна,доц.каф.СПіСКС,к.т.н.,затверджені наказомпо університету від «22» квітня 2019р.
№1330-С

2. Терміноданнястудентомпроекту «08» червня 2019р.

3. Вихідні дані до проекту: див. технічнезавдання.

4. Змістпояснювальної записки:аналізіснуючихрішеньтаобґрунтуваннятемидипломногопроекту;особливості розробки ПЗ;розробкаПЗ.

5. Перелік графічного матеріалу (із зазначенням обов’язкових креслеників, плакатів, презентацій тощо):організація блоків в технології blockchain.Схемаструктурна; архітектура системи.Схемаструктурна; принцип взаємодії основнихмодулівдодатку.Схемаструктурна; принцип роботивеб-додатку.Схемаструктурна.

6. Консультантирозділів проекту[1]

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Клятченко Я.М., доц. каф. СПіСКС, к.т.н.		

7. Датавидачізавдання:10.10.2018р.

Календарний план

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1.	Видача завдання на дипломне проектування	10.11.2018	
2.	Розробка технічного завдання	25.11.2018	
3.	Аналіз існуючих рішень	15.12.2018	
4.	Вибір середовища розробки	05.01.2019	
5.	Розробка програмного продукту	28.02.2019	
6.	Відлагодження програмного продукту	20.03.2019	
7.	Підготовка пояснювальної записки	29.04.2019	
8.	Оформлення матеріалів проекту	18.05.2019	
9.	Попередній огляд матеріалів диплому на кафедрі	28.05.2019	

Студент _____ Здирко В.В.
(підпис)

Керівникпроекту _____ Орлова М.М.
(підпис)

[1] Консультантомне може бути зазначено керівника дипломногопроекту.

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	аркушівКількість	№ прим.	Примітки
1	А4	ІАЛЦ.045440.002 ТЗ	Система захисту	2		
			інформаційних потоків			
			в комп’ютерних мережах			
			SDN”			
			Технічне завдання			
2	А4	ІАЛЦ.045440.003 ТП	Система захисту	2		
			інформаційних потоків			
			в комп’ютерних мережах			
			SDN”			
			Відомість технічного			
			проекту			
3	А4	ІАЛЦ.045440.004 ПЗ	Система захисту	50		
			інформаційних потоків			
			в комп’ютерних мережах			
			SDN”			
			Пояснювальна записка			
4	А1	ІАЛЦ.045440.005 Д1	Система захисту	4		
			інформаційних потоків			
			в комп’ютерних мережах			
			SDN”			
			Організація блоків в			
			технології blockchain.			

З	Арк.	№ докум.	Підпис	Дата	
Розробив		Здирко В.В.			
Керівник		Орлова М.М.			
Н.		Клятченко			
Зав. каф.		Тарасенко В.П.			

ІАЛЦ.045440.001 ОА

«Розподілена база даних для управління коштамипідприємства. Веб-орієнтована частина»

Опис альбому

Літ.	Аркуш	Аркушів
	1	2

НТУУ «КПІ ім. ІгоряСікорського», ФПМ, КВ-41

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	аркушівКількість	№ прим.	Примітки
5	А1	ІАЛЦ.045440.006 Д1	Система захисту	1		
			інформаційних потоків			
			в комп’ютерних мережах			
			SDN”			
			Алгоритм-схема роботи			
			Модуля формування			
			повідомлення			
6	А1	ІАЛЦ.045440.007 Д1	Система захисту	1		
			інформаційних потоків			
			в комп’ютерних мережах			
			SDN”			
			Алгоритм-схема роботи			
			Модуля відправки			
			повідомлення			
7	А1	ІАЛЦ.045440.008 Д1	Система захисту	1		
			інформаційних потоків			
			в комп’ютерних мережах			
			SDN”			
			Алгоритм-схема роботи			
			Модуля прослуховування			
8		Диск CD-ROM	Текст пояснювальної	1		
			записки.			
			Графічні матеріали			

Зміст

					ІАЛЦ.045470.002 ТЗ		
Зм.	Арк.	№ докум.					
Розроб.		Здирко В.В.			*«Система захисту інформаційних потоків в комп'ютерних мережах SDN»* ***Пояснювальназаписка***	Аркуш	Аркушів
Перев.		Орлова М.М.				1	50
						НТУУ „КПІ ім. Ігоря Сікорського“ФПМ КВ-52	
Н.контр.		Клятченко Я.М.					
Затв.		Тарасенко В.П.					

1. НАЙМЕНУВАННЯТА ГУЛУЗЬРОЗРОБКИ

Найменуваннярозробки:Системазахистуінформаційнихпотоківв комп’ютерних мережах SDN

Галузь застосування: організація захищеного каналу зв’язку для передачі таємних повідомлень.

2. ПІДСТАВИДЛЯРОЗРОБКИ

Підставою для розробки є завдання на виконання роботи першого (бакалаврського) рівня вищої освіти, затверджене кафедрою системного програмування і спеціалізованих комп’ютерних систем Національного технічного університету України «Київський gолітехнічний sнститут ім. Ігоря Сікорського».

3. ЦІЛЬТАПРЕЗНАЧЕННЯ

Створення програмного засобу та дослідження способів передачі данних для системи захисту інформаційних потокі в комп’ютерних мережах.

4. ДЖЕРЕЛАРОЗРОБКИ

Джерелом інформації є технічна танауково-технічналітература,технічна документація,публікаціїу періодичних виданнях та електронні статті у мережіІнтернет.

5. ТЕХНІЧНІВИМОГИ

5.1 Вимоги до продукту, що розробляється:

1. забезпечення таємностіпередачіповідомлення;

2. забезпечення стабільностіроботи;

3. обхідможливих накладених накористувачаобмежень;

5.2 Вимоги до програмного забезпечення:

1. Операційна система на базі Linux

5.3 Вимоги до апаратного забезпечення:

1. компютерна базі процесора Intel Pentium ікраще;

2. оперативна память 512 Мбайт ібільше;

3. наявність доступу докомпютерноїмережі;

4. мережева карта підтримуючапротоколиIPv4 iIPv6;

6. ВИМОГИ ДО ПРОЕКТНОЇ ДОКУМЕНТАЦІЇ

У процесі виконання проекту повинна бути розроблена наступна документація:

1) пояснювальна записка;
2) керівництвокористувача;
3) креслення:

Модульформування повідомленнясхемаалгоритма роботиМодульвідправки повідомленьсхема алгоритмароботи

Модульобробки повідомленьсхемаалгоритма роботиМодульпрослуховуваннясхемаалгоритмароботи

7ЕТАПИ РОЗРОБКИ

№ з/п	Назва етапів роботи та питань, які мають бути розроблені відповідно до завдання	Термін виконання
1.	Видача завдання на дипломне проектування	10.11.2018
2.	Розробка технічного завдання	25.11.2018
3.	Аналіз існуючих рішень	15.12.2018
4.	Вибір середовища розробки	05.12.2018
5.	Розробка програмного продукту	28.02.2019
6.	Відлагодження програмного продукту	20.03.2019
7.	Підготовка пояснювальної записки	29.04.2019
8.	Оформлення матеріалів проекту	25.05.2019
9.	Попередній огляд матеріалів диплому на кафедрі	30.05.2019

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	аркушівКількість	№ прим.	Примітки
1	А4	ІАЛЦ.045440.004 ПЗ	Система захисту	50		
			інформаційних потоків			
			в комп'ютерних мережах			
			SDN"			
			Пояснювальна записка			
2	А1	ІАЛЦ.045440.005 Д1	Система захисту	1		
			інформаційних потоків			
			в комп'ютерних мережах			
			SDN"			
			Алгоритм взаємодії			
			модулів ПЗ			
3	А1	ІАЛЦ.045440.006 Д1	Система захисту	1		
			інформаційних потоків			
			в комп'ютерних мережах			
			SDN"			
			Модуль формування			
			повідомлення схема роботи			
4	А1	ІАЛЦ.045440.007 Д1	Система захисту	1		
			інформаційних потоків			
			в комп'ютерних мережах			
			SDN"			

Зм	Арк.	№ докум.	Підпис	Дата
Розробив		Карпусь В.В.		
Керівник		Орлова М.М.		
Н.		Клятченко		
Зав. каф.		Тарасенко В.П.		

ІАЛЦ.045440.003 ТП

«Розподілена база даних для управління коштамипідприємства. Веб-орієнтована частина»

Відомість технічного проекту

Літ.	Аркуш	Аркушів
	1	2

НТУУ «КПІ ім. ІгоряСікорського», ФПМ, КВ-41

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	аркушівКількість	№ прим.	Примітки
			Модуль відправки	1		
			повідомлення			
			схема роботи			
5	А1	ІАЛЦ.045440.008 Д1	Система захисту	1		
			інформаційних потоків			
			в комп’ютерних мережах			
			SDN”			
			Модуль прослуховування			
			Схема роботи			
6		Диск CD-ROM	Текст пояснювальної	1		
			записки			
			Графічні матеріали			

Зм	Арк.	№ докум.	Підпис	Дата	***ІАЛЦ.045440.003 ТП***	Арк. 2

Вступ

На сьогоднішній день проблема інформаційної безпеки якніколиактуальна.Компютернімережі повністю інтегрувались в сучасне суспільство, не існуєжодноговиробництва чи корпорації, які б не використовувалисвсесвітнюмережу.А отже в інформаційному середовищі за рахунок все більшогоросту,виникає велика кількість потенційних вразливих мість, - де інформаціяможебутиперехоплена. Щоб уникнути можливихвтратінформації застосовуються новіметодишифрування, та утаємничуванняфакту передачіінформації.

Для цих цілейпідходитьстеганографія-тайнопис,приякомуповідомлення,закодованетаким чином, що невиглядаєяк повідомлення, на відміну від криптографії. Таким чином непосвяченалюдина,яка не має відповідних прав, принципово неможерозшифрувати повідомлення, оскільки не знає про фактйогоіснування.Якщо криптографіяприховуєзміст повідомлення,тостеганографіяприховуєсам факт існуванняповідомлення.

На основі постулатів цієї науки було сформовано технологію цифрових водяних знаків, яка забезпечує підтвердження авторства за рахунок унікальних міток, які приховуються в текстових документах, зображеннях, звукових і відео файлах, які не можна визначити звичайними методами.

Отже в теперішніх умовах важливо дослідити сучасні методи реалізації таємної передачі даних і реалізувати систему захисту інформаційних потоків що буде відповідати критеріям захисту даних. Це необхідно реалізувати за сучасними стандартами, які б викликали найменшу підозру в факті таємної передачі даних, і з найбільшою простотою реалізацію, щоб в разі потреби швидко пристосовуватись до змінених умов таємної передачі повідомлення.

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ

Бітрейт	- кількість біт, які використовуються для передачі і обробки даних в одиницю часу. Використовується при вимірюванні єфективності швидкості передачі даних через канал.
Ретрансльовані пакети	- пакети, які по різним причинам повторно передаються в компютерній мережі.
Стегаконтейнер	- контейнер, який має в собі таємне повідомлення.
Стек потоколів	- організований набір протоколів, достатній для організації роботи вузлів мережі.
SDN	- мережапередачіданих, в якій рівенькеруваннямережою не звязаний із фізичним,апаратнимрівнем. Рівенькеруванняв такій мережіреалізується програмно.
API	- інтерфейс прикладного програмування. Набір процедур, функцій, структур, конструкцій, якими однакомпютернапрограмавзаємодієзіншою.
IDS	- програмне або апаратне забезпеченя, призначене для виявлення фактів несанкціонованого доступу до комп'ютерної системи чи мережі або несанкціонованого управління ними в основному через Інтернет.
OpenFlow	-протокол управління процесом обробки даних, що передаються по мережі передачі даних маршрутизаторами і комутаторами, який реалізує технологію програмно-конфігуровної мережі SDN.
TCP	-один з основних протоколів передачі даних мережі

	даних.У стеку протоколів TCP / IP виконує функції транспортного рівня моделі OSI.
TLS	- протокол захисту транспортного рівня: криптографічний протокол, що забезпечує захищену передачу даних між вузлами в мережі Інтернет

Постановка задачі

Створення програмногозасобу,який би дозволяв безпечнопередаватитаємну інформацію через мережуінтернет.Аналіз та дослідження способівпередачітаємної інформації вкомп'ютерних мережахтаSDN(software- defined networking) мережах.

Необхідні пункти при створення заданої системи:

1) проветиогляді аналіз існуючих систем таметодів;
2) розробити архітектуру створюваногоПЗ;
3) розробитимодулідля заданоїархітектури;
4) розробити програмне забезпеченнямодулів;
5) зійснити тестування ПЗ.

Розроблена система має задовільняти такі вимоги:

1) формування повідомлення тайогошифрування за обраноюметодикою;
2) формування пакетів заметодикоюмодифікаціїзаголовку пакету;
3) фідправка повідомленняотримувачу;
4) отримання повідомленнякористувачемтайогоподальша обробка.

1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ОБҐРУНТУВАННЯ ТЕМИ ДИПЛОМНОГО ПРОЕКТУ

1.1 Загальний опис проблеми

В сьогоднішньому світіконфіденційність інформації завжди стоїть під сумнівом, оскільки доброчесність сервісів, які надають послуги по передачі або збереженню даних є сумнівними. Хоча на рівні законодавства країн прийняті закони про конфіденційніть інформації, і їх обробка без згоди власника карається. Це не заважає опрацьовувати пересональні дані користувачів. Таким чином, при достатній кількость ресурсів і знань злочинець може отримати доступ до персональнтх даних. Тому приймаються міри захисту, які забезпечує криптографія.

В криптографії використовіються такіметодияк симетричне, асиметричне шифрування тахешфункції. Вони забезпечують шифрування даних, але не утаємничення. Як наслідок виникає проблема,колиоднаособахочепередати дані інформації через мережу Інтернет фактпередачіє беззаперечним і повідомленняможе бутиперехопленне. Це пояснюється тим, що не має повністю сектретного табезпечногоканалупередачівідодногокористувачадо іншого. Дані які передаються через мережуІнтернет,можутьбутипрочитанимим та скопійованими на транзитнихточкахпровайдерів. Сервера, маршрутизатори або інші,прочитуютьмережеві пакети задля запобіганняпередачіданих зпомилкоюі інших причин.Такас т р у к т р а в с е с в і т н ь о ї м е р е ж і д о з в о л я єекономитиресурси та їх правильно перерозприділяти, оскількипередачанеправильно сформованогопакетубез ранньої діагности породжує велеку кількість проблем.

1.2 Аналіз існуючих рішень

В сучасному світі, бізнес в області інформаційних технологій зростає за експонентноимзаконом.Із ростом потреб ринку зростає необхідність збільшення обчислювальних потужностей, систем збереження даних. Проте класичний спосіб формування потужностей для вирішення цих проблем непідходить,оскільки традиційні класичні мережі статичні і не відповідають швидкій динаміці зміни потреб. Таким прикладом є мережа Інтернет сервіси за надання доступу до відео матеріалів (стрімсервіси) під час настання певноїгодинидоби, кількість запитів до цього сервісу зростає в 2-4 рази. Це породжує необхідність мативеликий запас обчислювальних потужностей, який в більшій частині доби невикористовуєтьсянаповну.Відповіднокомпаніязіштовхується зоднієюздвохпроблем. Перша проблема: якщокомпаніямає великі обчислювальні потужності, які під час використання не працюють наповну.Це призводить до збільшення часу амортизації цих потужностей, що призводить дозначних економічнихвтрат.Друга проблема:компаніїв піковігодинине вистачає потужностей,тодісервіс починає нестабільно працювати, можливі збої, а при нестабільній роботі зменшується кількість клієнтів. Тобтозначнозменшується дохідкомпанії.Рішенням цих проблеми,можебути передачапотужностейоднієїкомпаніїіншійкомпаніїна необхідний період. Таким чином, першакомпанійвгодиниспаду,повинна передати потужності під управління і використання другій, за що першакомпанія отримає прибуток і обчислювальні потужності не будутьпростоювати. Влятогощоб успішноздійнюватиподібні операції, насьогодні широко використовуютьтехнологію SDN(software-definednetworking).

Головнаідея SND полягає в відділенняпередачітрафіка від управління пристроями, які здійснюютьпредачу.Зазвичай в традиційнихкомутаторах,маршрутизаторах процеси управління тапередачінеможливо розділити.Тобтоспеціальні мікросхеми, які функціонують в цихмодулях,реалізують

функції реакції на отриманий трафік щоб знати повідомленя з якого порту і в який пересилати дані, тобто управління всієї мережею, залежить від налаштування конкретно кожної частини (вузла) для обробки даних.

Для фомування технології SDN для формування маршруту було розроблено велека кількість протоколів, які взаємодіючи між собою здійснюють необхідні дії, але їх функціонування не узгоджуються з роботою сусіднього модуля. Таким чином, проводити швидку і надійну рекунфігурацію мережі неможливо, оскільки потрібно налаштовувати кожен пристрій окремо. При використанні технології SDN формується Conrol layer, який бере на себе всю логіку управління, при цьому маючи загальну картину взаємодії та можливість налаштування кожного пристрою. Управляюче програмне забезпечення (SDN Control Software) контролює роботу всієї системи.

Таким чином, для зміни мережі не потрібно змінювати роботу конкретного пристрою, потрібно виконати запит на зміну в Control layer і SDN Control Software виконає реконфігурацію мережі.

На практиці реалізована наступна схему взаємодії (Рисунок 1.1):

1. Application layer - рівень управління, який міститьвбудованийвеб-сервіс чи API іпротоколиуправління.
2. Control layer - рівень управліннятрафіком,вякому вбудованіалгоритми та функції, щовиконують задачууправління пристроями,забезпечуючиуправліннятрафіком.
3. Infrastructure layer — інфраструктурний рівень, який забезпечує фізичнупередачуданих; це рівень пристроїв та мережевихпакетів.

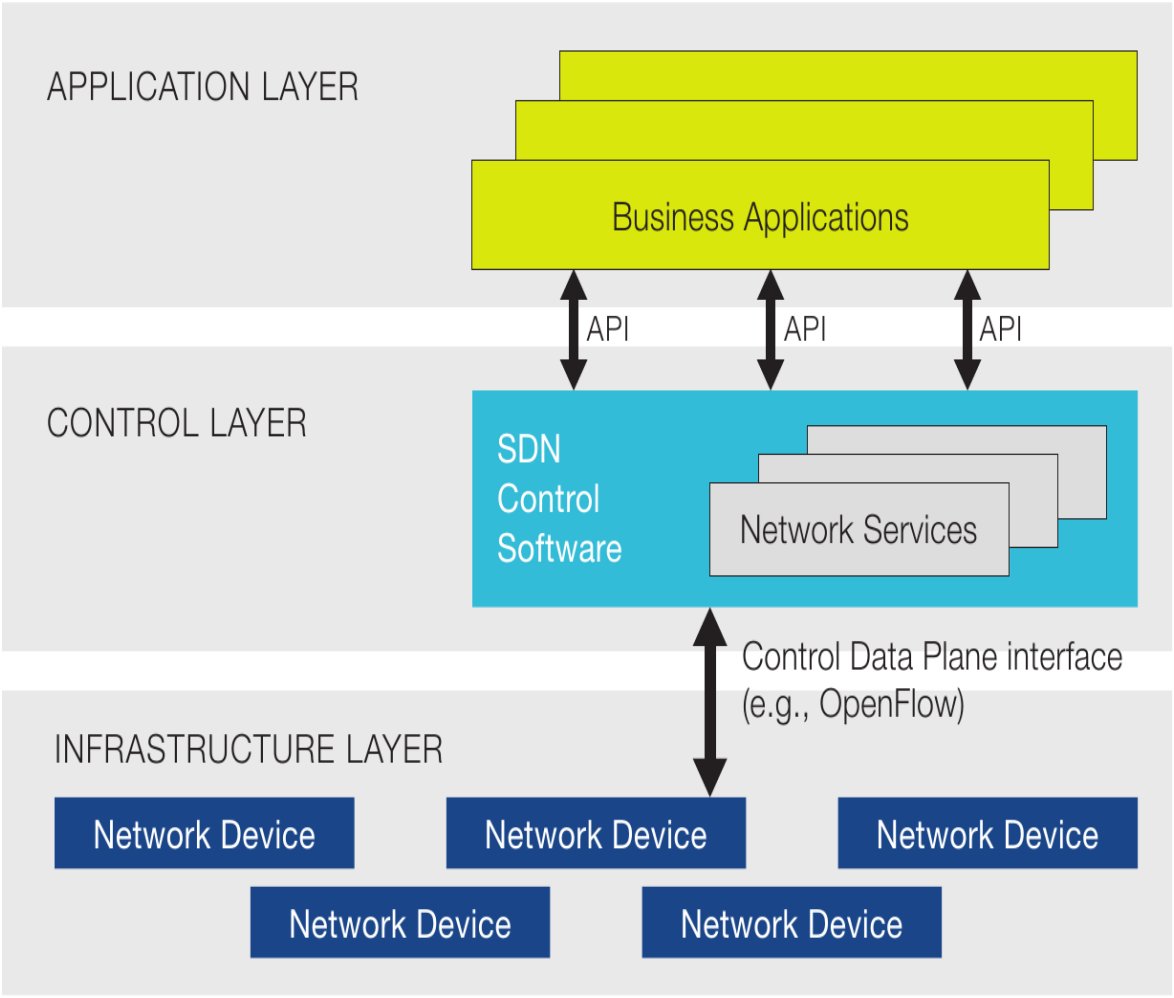

Рисунок 1.1 - SDN архітектура Реалізація

SDN на практиці дозволияє мати:

- централізоване управліннясистемою;
- віртуалізацію фізичних ресурсівмережі;
- швидкереагування мережі наналаштування;
- спрощене управління мережевимипристроями;
- зменшеннязатратна управліннямережами;
- простотевзаємодії;
- оптимізаціяпередачітрафіку.

Основним елементом технолохії SDN єпротоколOpenFlow,який забезпечує

взаємодію контролерів з мережевими пристроями.Використовуючи протоколOpenFlow,контролердодає,модифікує і видаляє записи в таблиці потоків. Крімтого,вінможезапитуватиукомутаторайогохарактеристики і зібранустатистику,конфігуруватикомутаторійогоокремі порти.

На сьогодні потужні компанії, такі як IBM, Cisco, Juniper Networks та інші використовують комутатори та маршрутизатори які сумісні з OpenFlow.

Архітектура SDN, суттєво відрізняється від класичної мережевої інфраструктури, але в ній залишаються потенційні вразливості з точки зору інформаційної безпеки. Розподіл доступу при роботі з контролером, авторизація та аутентифікація при роботі мережевого додатка (Application layer) з конторолером - ці а також інші аспекти, які вимушені прийматись до уваги при проектуванні SDN-мереж.

Зточкизору інформаційної безпеки, найбільш вразливимкомпонентому всій архітектурі SDN є мережевий контролер,атакана якийможепризаести до критичних наслідків у всій інфраструктурі[12]. Розподілення доступу мережевихдодатківпри їх роботі з SDN контролером — це актуальна проблема . Ситуація в якійбудь-якиймережевийдодаток можезмінити flow-таблицікерованіданим контролеромкомутатором,не відповідає сучасним вимогам інформаційної безпеки. Різні типидодатківпотребують різних типівдоступу,чим більшебудутьописані обмеженнякожногододатку(в відповідності до поставленої задачі), тим більш безпечноюбудедана конфігурація мережі.Можутьзастосовуватисьтакожрізнімоделідоступу для вирішення даної задачі, зокрема такі як рольові, мандатні, атакожкомбінаціїцихмоделейз урахуванням специфіки захисту інфраструктури[15].

Основні загрози, які виникають зі сторони мережевих пристроїв, що працюють на базі технології програмно-конфігуровних мереж SDN, є варіації таких атак, як „відмова в обслуговуванні“, „підміна контролера“ та інші. Пренесення „аналітичної“ частини мережі на контролери дозволяє перенести

загрози з мережевих пристроїв на програмне забезпеченнямодулів,які забезпечують функціонування мережі, а саме: контролер мережі та мережеві додатки,які звертаються до контролера.

Найбільш простим і ефективним способом порушення правильності функціонування мережі SDN є атаки типу „відмова в обслуговуванні“. Небезпекавиходитьіз роботи самої мережі SDN та роботиSDN-комутаторапри отриманні невідомого пакета (який непідходитьпідіснуючіправила віснуючіflow-таблиці). В такій ситуації можливі дваваріанти:

- пакетзалишається впам ікомутатораанаконтролерприходить тільки заголовкипакета[15];
- пакет повністю відправляється на контролер дляаналізу;

Обидва способи є відкритими для спроб формування атаки шляхом формування потоку пакетів для SDN-мережі. Можливі наступна реакції на описані вищі події:

1. Комутаторпочинаєформувативелику кількість повідомлень дляпередачіневідомих пакетів на контролер. Витрачаються процесорні ресурсикомутатора,збільшується витрата пам'яті. Особливо сильно витрачається пам'ять втомувипадку,якщокомутатор буферизуєсамі пакети і пересилаєконтролерутільки їхзаголовки.[15]

2. Потік пакетів відкомутаторана контролер завантажує канал зв'язку між контролером ікомутатором.Якщо середовище зв'язку є розділеним,тозниження оперативності доставки повідомленьможутьвідчути на собі всікомутатори.Підвищений вплив на канал зв'язкубуденадано в ситуації,коликомутаторпересилає пакети для аналізу повністю[15].

3. Контролер приймає та обробляє потік повідомлень,витрачаючипроцесорний час і пам'ять своїх ресурсів. Формування черг повідомлень змусить легітимні повідомленняочікуватисвоєї черги і знижує оперативність прийняття рішень вмережі[15].

4. Контролер генерує потік різних повідомлень у відповідь на запитиатакованогокомутатора.Витрачаються ресурси каналу зв'язку міжкомутатороміконтролерами[15].

5. Комутаторприймаєкомандивід контролера івиконуєїх,витрачаючиресурси процесора і пам'ять. Якщокомандимістять в собі створення нових правил таблиць потоків,товідбувається їх лавиноподібне збільшення, час перевіркикожногонового пакета по таблиці збільшується, зростають витрати на обслуговуваннятакоїтаблиці, атакожобсяги таблицьпотоків.[15]

Наслідком реалізації атаки є наступними:

- повне вичерпання ресурсівкомутатора.Врезультатіправдиві пакетиніколинебудутьоброблені даним мережевимвузлом,або їх обробкабудезізначнимизатримками;
- контролербудеперевантажений вхідними запитами врезультатінезможеоброблюватиуправляючі повідомлення, викликанілегітимнимтрафіком;
- каналзв'язкуміжконтролеромікомутаторомнезабезпечутєдоставкиповідомлення, оскільки вінбудеперевантаженим потокамиданих.

Переваги і недоліки архітектури SDN для безпеки мережевої інфраструктуриширокови вчені,[15] але оцінка вразливостей архітектури повиннаґрунтуватисяне тільки на аналізі теоретичної архітектури, але і на експериментах ірезультативпровадженняпротоколуOpenFlow в промислові мережі.Насьогодні визначенінаступні типи загроз для OpenFlow-мереж[15].

Режими роботи контролера (реактивної або проактивний) можутьбути легкоідентифіковані зловмисниками без застосування специфічнихпідходіві програмного забезпечення. Ідентифікація заснована на затримці першогопакетудля новогопотокутрафіку і доступна длякожногокористувача,підключеногодо мережі, аботакого,щовикористовуєсервіси даної

інфраструктури з зовнішніх мереж. В результаті деякі атаки можуть використовувати конкретний режим роботи контролера [12].

Наприклад, несанкціонована установка правил обробки накомутаторах,яка призводить до зниження ефективності або порушення роботи мережі, простіше реалізується в реактивному режимі в зв'язку з особливістюпідходуконтролера до управління таблицями потоків в цьому режимі. У той же час виконання цієї атаки на реактивний контролер є більш складною, оскільки потрібна,комплексна атака,і ймовірністьшвидкоговиявленняфактуатакизначновище.

Загрози безпеці, актуальні для більшості інформаційних систем, такі як сканування портів і визначення мережевихслужб,є критичними для архітектури SDN через уразливості каналу OpenFlow і наявностізначноготрафіку управління, який передається міжкомутаторамиі мережевими контролерами. Відзначимо вразливість архітектури SDN до DoS-атак -однимз найнебезпечніших для архітектури з централізованоїточноїуправління.Такоскількимодуліуправління не мають функцій управління, вони повинніматистабільне підключення до мережевого контроллера, щоб забезпечуватипередачуданих. У реактивному режимі контролера навітькороткийперіоднедоступності OpenFlow-контролераможевикликати багатопроблем для мережевої інфраструктури, а довгострокове блокування мережевого контролераможебутивикористано для повної зупинки обробку мережевого трафіку або реалізація більш складноїатаку.Наприклад,некоректниймережевий контролерможезайняти місцезаблокованого,що ставить під загрозу всюмережеву інфраструктуру.

Іншою потенційною проблемою є вразливість програмної інфраструктури OpenFlow-мережі. Можливість програмування мережі та наявність відкритих програмних інтерфейсів, що використовуються для інтеграції з мережевим контроллером, є ще одними точками для появи

вразливостей програмного забезпечення. У той час як контролери OpenFlow, розроблені професійними командами,і підтримувані великими спільнотами, теоретично можуть бути надійно захищеними, програмні модулі для контролера, які розробляються мережевими інженерами для потреб конкретної мережевої інфраструктури, можуть створити непередбачувані вразливості, які вплинуть на безпеку всієї мережі. Ця проблема безпеки є результатом низького рівня стандартизації рівнів управління і додатків в архітектурі SDN [13].

Щеодназагроза, характерна для традиційної мережевої інфраструктури, - атаки з підміною - має більш високий негативний потенціал для мереж SDN, ніж в традиційних мережах. У той час як вся мережа управляється центральних контролером, ризик підміни управляючого-трафіку в мережі OpenFlow досить високий. Підмінаможепризвести до несанкціонованого доступу до мережевих пристроїв, отримання контролеромнекоректноїстатистики або даних про стан мережі, щоможепозначитися на роботі всієї мережі. Однією з причин уразливості OpenFlow- мереж до атак з підміною є надмірна гнучкість стандартуOpenFlow.Стандарт дозволяє реалізувати взаємодію між мережевим контролером ікомутаторамина базіпротоколуТ С Р б е з ш и ф р у в а н н я , а п і д т р и м к а протоколуТ L S є н е о б о в ' я з к о в о ю д л я реалізації.

Всі спостережувані загрози безпеки пов'язані не зі специфічними версіямипротоколуOpenFlow,ці загрози актуальні для всіх випущених версійпротоколу(1.0-1.5) і,скорішза все, не можутьбутиусунені через фундаментальних особливостей архітектури SDN.

Щоб подолати ці перешкоди, Marce Winandy та Neil Davies запропонували шість основних принципів проектування, які вважаються обов'язковими для безпечної архітектури контролерів SDN[13].

Вони випливають з кращих практик системної безпеки іохоплюютьзагальнімоделібезпеки зі згаданих дослідницькихробіт,атакожважливі вимоги галузі. Для ілюстрації наведемо ці принципи як вершини багатогранника, нагадуючиформуалмазногоекрануванняконтролера(Рис.1.2).Якийназвали алмазнимпідходомдля безпекиSDN[13].

Complete Mediation (Повне посредництво) -Коженраз,колисуб'єкт намагається отримати доступ доресурсу,система повинна опосередкувати дію. Цей принцип вимагає систематичногоконтролюдоступу до ресурсів, щоб доступ до них перевірявсящоразу,щобпереконатися,що суб'єктмаєвідповідні привілеї. Посередник повиненбутилогічному н і к а л ь н и м „авторитетом“ для цієї перевірки, і він може скористатися перевагами функцій, що надаються розподіленими системами[13].

Compartmentalization (відокремлення) - Цей принцип, так званий Sandboxing, застосовує правило про те, що виникнена проблема безпеки повиннабутиобмежена вконкретномувідсіку,що містить його. Це добре відома концепція у всіхконтекстах,які вимагають безпеки. Для SDN він застосовується до ділових такеруючихш а р і в , д е додаткиповиннібутивідокремлені та ізольовані між собою та від самого контролера[13].

Code Size Minimization (Мінімізація розмірукоду)- Важливимаспекомбезпеки є обробка повідомлень стороною. Щоб цього досягти необхідно по- перше, мінімізація рядківкоду(LOC) це зменшує можливість помилокі „backdoor“ для атаки, що може бути використана зловмисником. По-друге, тонкий програмний шар може бути напівформально або офіційно перевірений, щоб дати математичні докази того, що він правильно відповідає вимогам, які він вимагає[13].

Capability-based Control(контроль можливостей) - дозволяє додавати авторизації (тобто можливості) суб'єктам, які вимагають послуги, замість того, щоб приєднувати їх до ресурсів, що її надають (підхід до Списку контролю доступу). Можливість повністю переноситься і представляє здатність виконувати привілейовані операції. У динамічному середовищі, де SDN-додатки та їх контекст безпеки можуть часто змінюватися, управління, що базується на можливостях, може бути легше керовано та перевірено[13].

Concurrency(паралелелізм) - У розподіленій системі важливо, щобкомпонентивиконувалися і працювали разомодночасно.Це надзвичайно важливо в середовищах, таких як SDN, де контролери можуть розподілятися в кластерах, а середовищедужединамічне. Операції, такі як вибори лідерів і реплікація вузлів, є прикладами, де необхідна безпечнапаралельність[13].

Compatibility(сумісність)-У різноманітті контролерів SDN, доступних в даний час на ринку, інтеграція механізмів безпеки не повинна істотно впливати на розробку та реалізацію існуючих контролерів. Рішення для забезпечення безпеки SDN має бути прозорим для контролера для того, щоб дозволити клієнтам застосовувати безпеку в розгорнутих інфраструктурах і зберігати «відкритість» SDN спочатку[13].

В роботібулазапропонована система, яка розроблена нао с н о в і „алмазного принципу“ в архітектурі Secure Controller (SCONA), яка об'єднує ці принципи. На (Рисунок 1.5)зображено огляд компонентів SCONA, кожен з яких розроблений для безпроблемного вирішення проблеми безпеки SDN від шару програми (бізнесу) до площини даних.

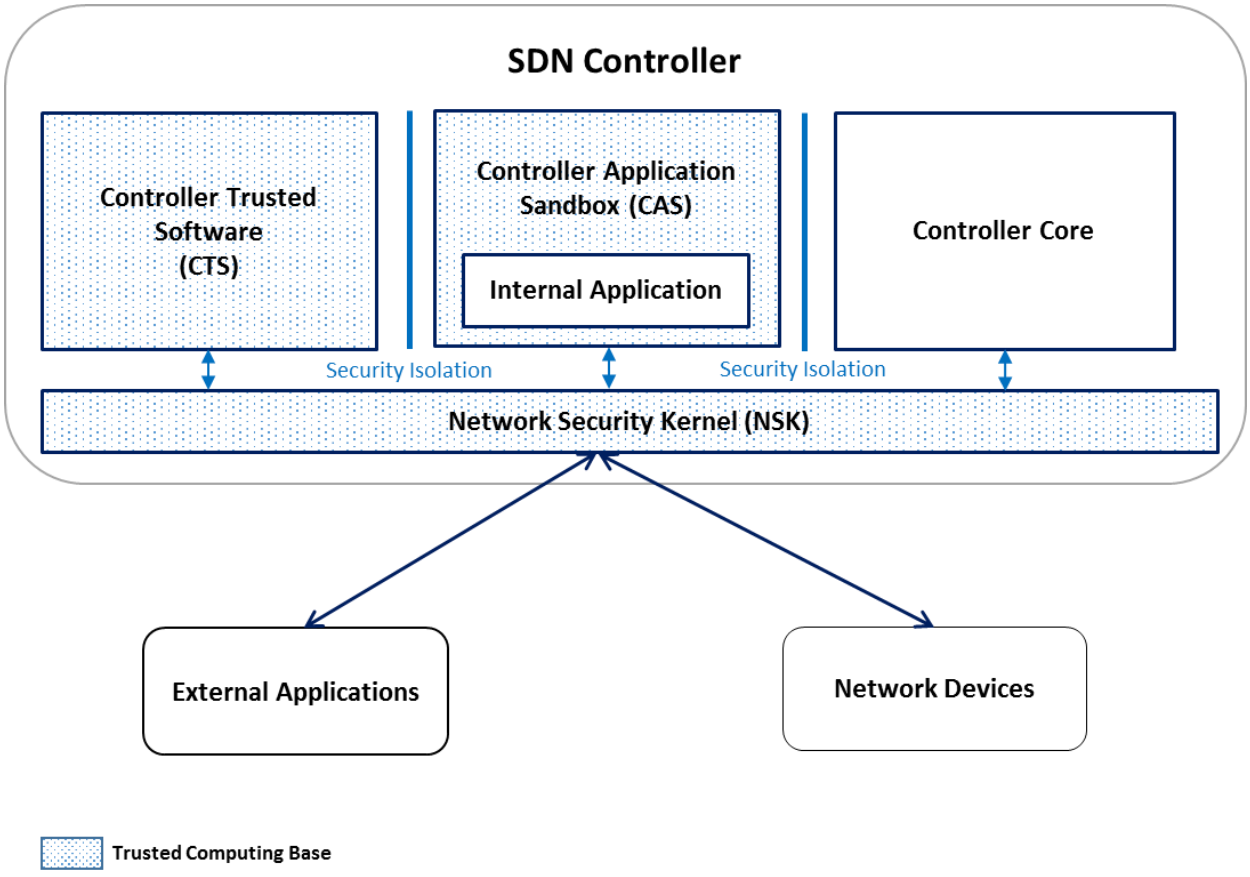

Рисунок 1.5 — архітектура SCONA

Основним компонентом SCONA є ядро мережевої безпеки (N[illegible]).SCONA Controller Applications Sandbox (CAS) і Controlled Trusted Software (CTS) реалізують принцип розділення, тобто внутрішні програми можуть бути використані для нівілювання безпеки загального контролера і забезпечення середовища для надійних функцій безпеки вищого рівня, наприклад, моніторингу поведінки програми. Інша функціональність контролера SDN, яка не пов'язана з безпекою, називається компонентом Controller Core (CC).

SCONA та алмазний підхід намагаютьсяподолатич а с т о частковуперспективу,запропоновану попередніми дослідженнями та існуючими галузевими реалізаціями.Ключовавідмінність і новизнапідходуполягає втому,щ о S C O N A л о г і ч н о охоплюєконтролер. Це ефективно забезпечує не

тільки безпеку додатків, але й обмежує атаки, що надходять з площини даних на додатки, та сам контролер[13].

Проведений в розділі аналіз аналітичний огляд свідчить проважливістьдослідженнятарозвиткузахищенихканалівзв кув класичних системахоскільки,процеспереходудотехнологіїSDNтісноповя йз проблемами захисту потоків інформації,атакожзахисту існуючих каналівзвязку.Дляцьогопотрібно аналізувати вразливості не тільки архітектури мережіSDN,алейкласичноїархітектурикомп тернихмереж,оскільки повністю замінити класичну архітектуру не представляється можливим. В класичній архітектурі існує декілька варіантів створення таємного,захищеногоканалузв ку.Одинізних- цеформуваннязакритогоканалув відкритій мережі за рахунок формування і розробкисвогостекупротоколів,якіївикористовуютьсяв захищеному режимі.Розглянемоінший спосіб, а саме шифрування даних. Основні недоліки шифрування полягають втому,що необхідні додаткові обчислювальні ресурси для шифрування тарозшифрування повідомлення.Такожкрім ресурсів необхіднийдодатковийчас длявиконанняцих операцій. Основна функція шифрування полягає в збереженні секретності, та цілосностіпередачіінформації. Але не дивлячись на всі процедури шифування при достатній кількості ресурсів і часузавждиможливий взлом шифру і розсекречення інформації. Оскільки фактпередачіінфомації неприховується.

2. ОСОБЛИВОСТІ РОЗРОБКИ СИСТЕМИ

2.1 Використані методологій та їх опис.

Для унеможливлення перехоплення навіть зашифрованих даних використовується стеганографія. Спільною рисою всіх стеганографічних методів є те, що приховане повідомлення, або додаткова інформація, вбудовуються в деякий нешкідли-вий код. В результаті створюється стеганоповідомлення, яке потім відкрито транспортується адресату каналом зв'язку або зберігається в такому вигляді.

Передачаінформації відкритими каналами звязку відкриваєбагатоможливостей дляприхованої передачіданих. Сектретні повідомлення можутьбутивбудованів зображення,звуковіфайли, текст тощо. Ці способи є стійкими до виявлення, проте існує низка програмних засобів, які на основі статистичнихметодіввиявляють закономірності у стегоконтейнерах. Всі попередні способи стеганографії є цифровою стеганографією.

В сучасній літературі зазвичайвикористовуєтьсянаступна класифікаціяметодівстеганографії які наведені на рисунку 2.1.

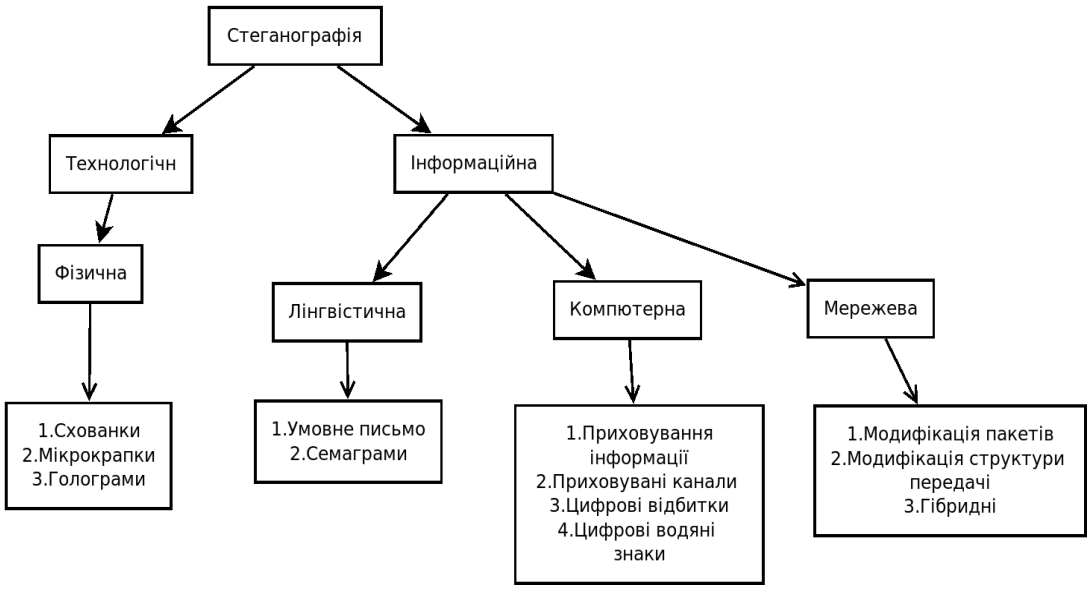

Рисунок 2.1 - Класифікація методів стеганографії

Щоб забезпечти захищену передачу даних існує багато стеганографічних методів.В подяльшому розділі розглянемо деякі з них і проаналізуємо їх можливості.

Найбільш відомими методиками модифікації контейнера є модифікація зображення. Серед них найпопулярнішою методикою є так звана методика заміни молодших або найменш значних біт інформації (LSB-Least Significant Bit) схема роботи представлена на рисунку 2.2 .

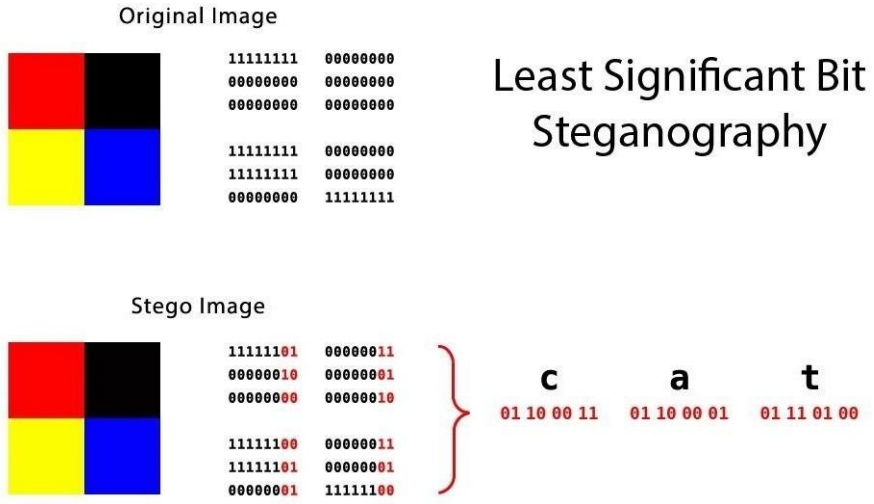

Рисунок 2.2 - Представлення методу LSB- Least Significant BitЦифрові зображення представляють собою матрицю піксилів. Молодший значимийбіт зображення несе в собі найменшу кількість інформації проколірта яскравістьтомуприйогозмінілюдськийзір неможеп о м і т и т и р і з н и ц ю , в н а с л і д о к ч о г о , можнавикористовуватийогодлявбудовуванняданих. А якщомодифікувати2 молодших біта, це збільшить розмірможливогоповідомлення вдвічі. Таким чином,маючизображення 1280 на 720 піксилів можна теоретичновикористатидо 1843200 бітприхованихданих. Проте існує ряд проблем зі стегеостійкістю цьогометода,асаме:

- Зображення маєбутиунікальним, оскільки порівнючимодифікованезображення з оргіналом можна миттєво зчитати всідані.

- Візуальні атаки, при якихбудуютьсяокремі зображення на основі окремих милодшихбіт.Приклад візуальною атаки (рисунок 2.3 і рисунок2.4).

Приклад приякомучітковидно збільшення єнтропії, що в звичайних немодифікованихзображеннях не спостерігається .

Рисунок 2.3 — Зображення, в якому присутня стеганограма

Рисунок 2.4 - Представлення візуальної атаки на стеганоконтейнер

Існує цілий ряд аналітичних програм, та методів які за допомогою статистичних атак на стеганоконтейнер дають однозначну відповідь на наявність в зображеннях повідомлення, такі як RS-метод, атака “хи-квадрат”

Тому стійкість секретності передачі такого повідомлення є дуже низькою. Також доволі значним недоліком є необхідності створення оргінального зображення.

Передачу прихованихданих у мережевій стеганографії здійснюють через таємні каналипередачіінформації.Такіканали можутьіснувативбудь-якомувідкритому каналі, вякомуіснує деяка надмірність.

Мережева стеганографія, що є підвидом цифрової стеганографії, останнім часом набула популярності завдяки методикам коли прихована інформація передається при використанні особливостей функціонування протоколів мережі інтернет. Типові методи мережевої стеганографії містять зміни властивостей мережевих протоколів. Крім того, може

використовуватисьвзаємозвязок міждвомаабо більше різнимипротоколамиз метою більш надійногоприховуваання передачісекретногоповідомлення.

Методи мережевої стеганографії можна розділити на три групи (рисунок 2.5):

1. MP-методи, при яких змінюються дані у поляхзаголовкупротоколуабо полекорисногонавантаження(Зміна сутностіпакета)
2. MS-методи у яких змінюється послідовністьпередачіпакетів, або навмисне винекненнявтрат.

3. HB-гібридний тип який містить в собі взаємодіюдвохпопередніхметодів.

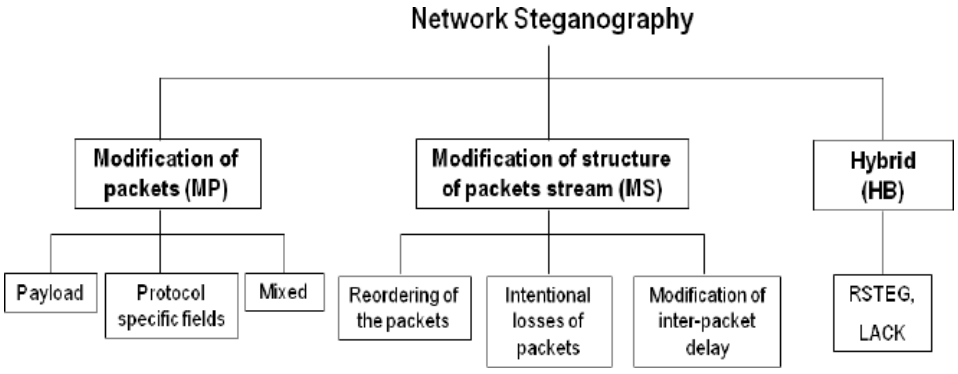

Рисунок 2.5 - Класифікація мережевої стеганографії

Методмодифікації мережевих пакетів Transcoding Steganography (TranSteg), якмй міняєкориснунагрузкуVoIP-пакета,такожкористуєтьсяпопулярністю за рахунок програм які забезпечують голосовий та відеозв'язок.

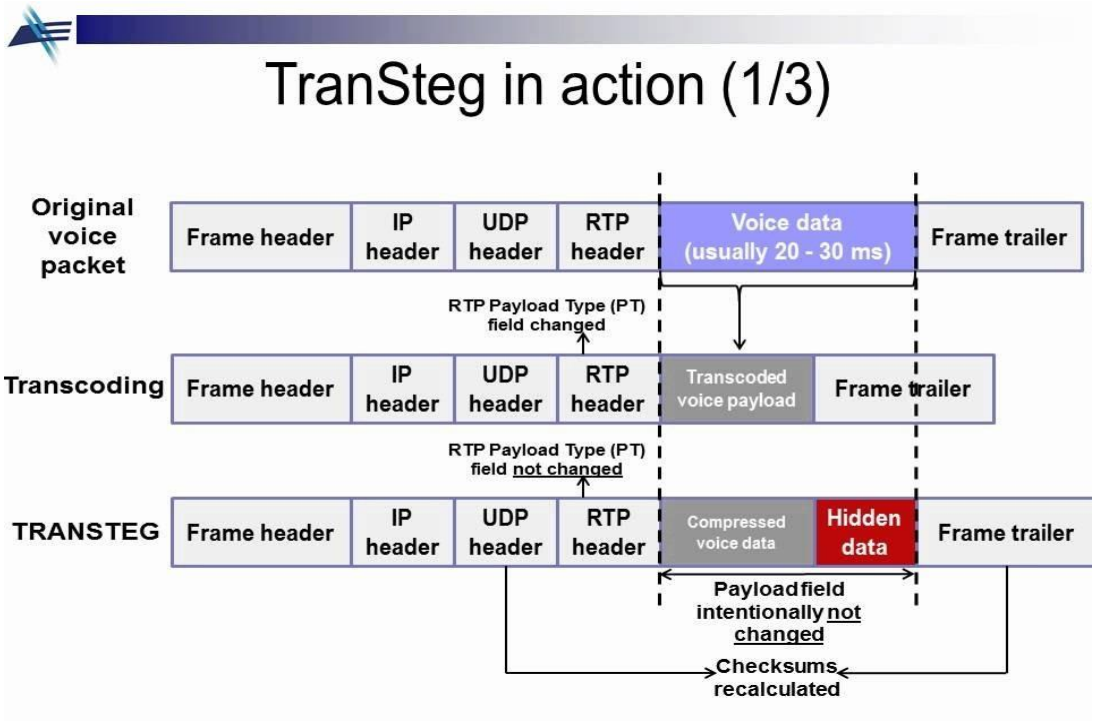

Рисунок 2.6 - Схема роботиметодуTranSteg

Ідеяметодазаключається в стисненні корисної нагрузки мережевого пакета за рахунокперекодуванняіможезастосовуватисьвсюдиде можливе стиснення відкритих даних, неважливо чи звтратоюцих даних чи без. Саме стиснення необхідне для зменшення розміру відкритих даних, щоб вивільнити місце для стеганограми. Як приклад, голосові данівисокоїякостіперекодовуютьсяв низькийбітрейт,по можливості з мінімальними втратами якості, а на вільне місце записуються даністеганограми.

Методдозволяє забезпечитидужевилику пропускну спроможність.Такпольськівчені,асамеW.Frączek,W.Mazurczyk, .Szczypiorski.всвоєму дослідженні [1, 2, 5] при використанні TranSteg отримализультатив 32Кб/ спредачітаємної інформації при збільшенні затримкипередачіна (1мс) на відмінупередачі пакетубез стеганограми [2,5]. Цейметод дужестійкий до виявлення оскільки потрібноконтролюватиі аналізувати каналпередачів реальному часі. Як наслідокзначнапропускна спроможність, стійкість до виявлення виявлення.Недолікомє дуже важкареалізація, оскількид л я

успішнї передачі стеганограми необхідно зясувати, які кодеки використовуються для формування голосового потоку, і підібрати кодеки з найменшою різницею втрати якості передачі. Також необхідно зазначити що при використанні цього методу втрата якості неминуча.

МетодLAC(LostAudioPacketsSteganography)працюєзпротоколомVoIP,звязокздіснюєтьсячерезIP-телефонію іскладаєтьсяздвохчастинслужбової та розмовної (рисунок 2.7). Вобохчастинах відбуваєтьсяпередачавдвохнапрямках. ДляпередачівикористовуєтьсясигнальнийпротоколSIP (Session Initiation Protocol) і RTP(Real-time Transport Protocol). Післявстановленнязвкупочинаєтьсяфазапередачіданих,деаудіопотокRTP передається вдвохнапрямкаходночасно.І саметомуцей алгоритм ефективний, оскільки пропускна спроможність вища, ніж у всіх інших алгоритмів, девикористовуєтьсяаудіопакети. Принцип функціонування методуLACполягаєвнаступному:передавачвибираєодинізголосовихпотоків ійогокорисне навантаження замінюється бітами секретного повідомлення, далі відбувається процес затримкипакету,як наслідокпакет,який прийшов в неназначений йому час, відкидається. Але оскількипакетспеціальномодифікований,токористувачякийвстановивсеансзвку,невідкидає його, а перехоплює, і вибираєвбудованув пакет секретну інформацію.

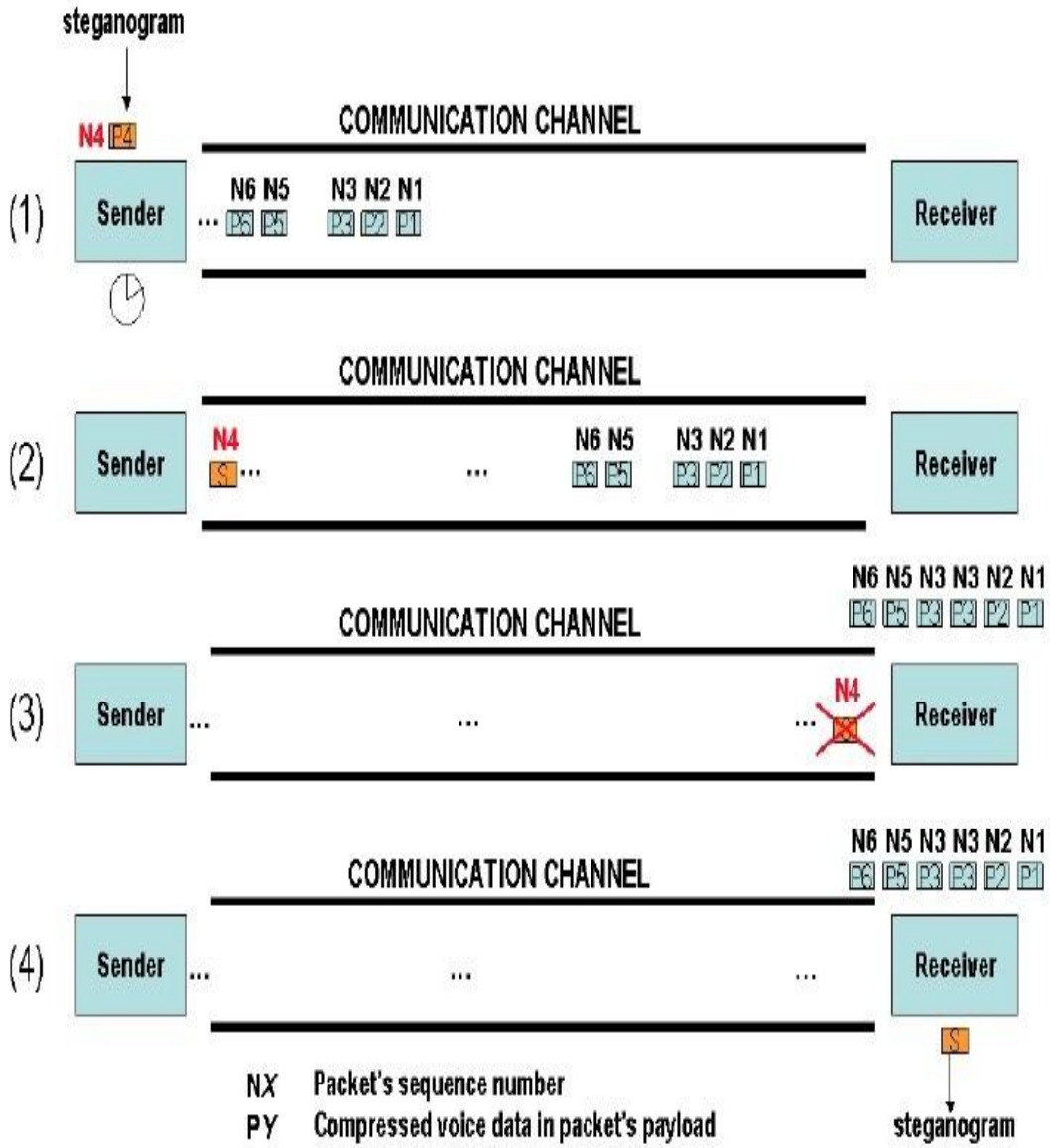

Рисунок 2.7 -СхемафункціонуванняметодуLAC

Стегоаналіз для пошуку таємної передачі даних при використанні метода LAC майже нічого не виявляє, оскільки втрата пакетів в IP-мережах є звичайною ситуацією, і тому штучні втрати які введені цим методом, важко виявити, якщо вони не перевищують допустиму кількість втрат в даному каналі зв язку.

ЯквисновокметодLAC характеризуєтьсясередньоюстійкістюдознаходження.Оскількипривикористанніметоду L AC приперевищені природногорівнявтратипакетів,виникаєявнепогіршенняякостізв ку,що

може викликати підозри у стороннього спостерігача. Також його недоліком є складна реалізація.

Метод RSTEG (Retransmission Steganography) оснований на механізмі повторної передачі пакетів (рисунок 2.8). Відправник передає пакет, а отримувач не відповідає пакетом з підтвердженням. Спарцьовує механізм повторної передачі пакетів, в який при повторній передачі вкладається стеганограма. Після цього приходить пакет від отримувача з підтвердженням коректної відправки.

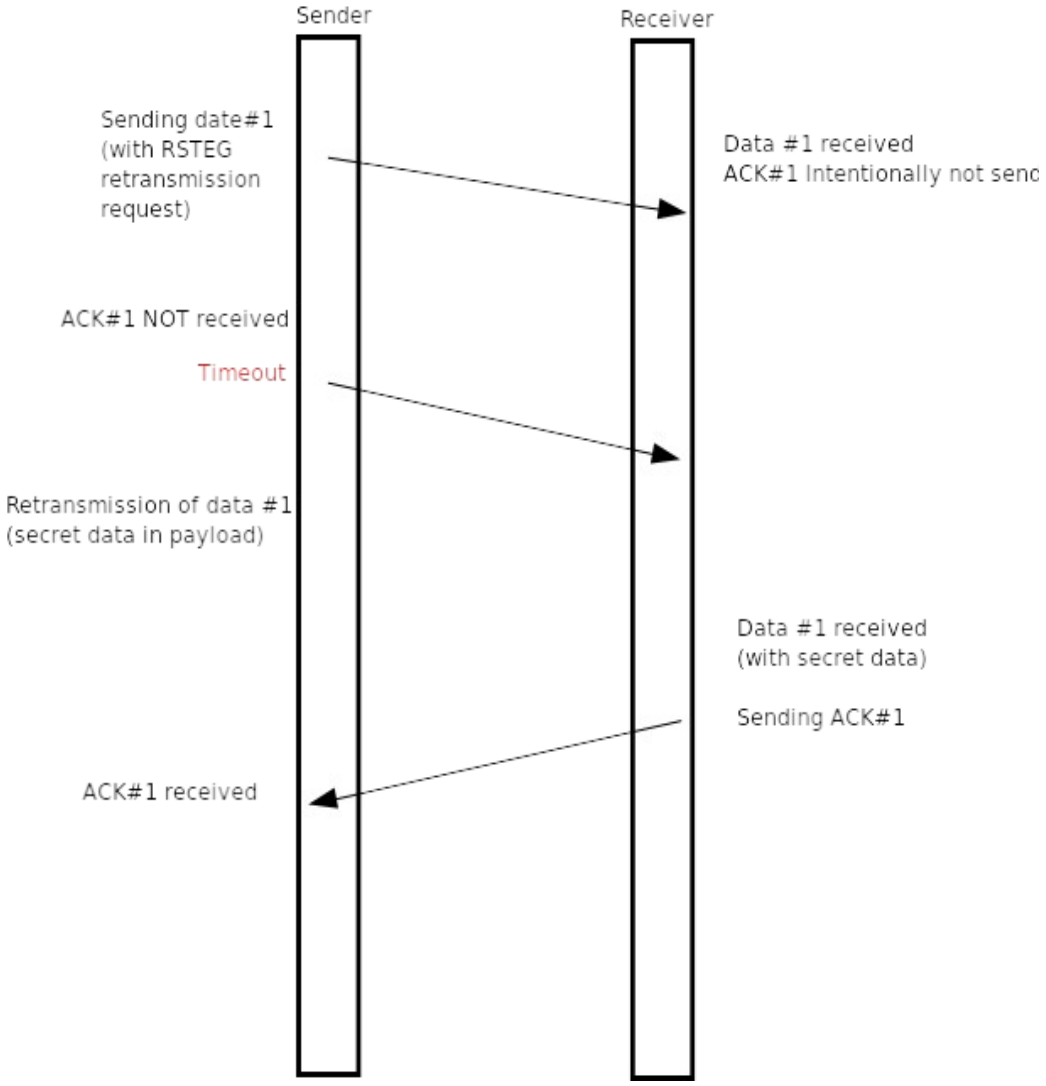

Рисунок 2.8 - Схема роботи методу RSTEG

Продуктивність RSTEG залежить віддвухфакторів: розміру пакета і частоти зякоюгенеруються пакети[12]. Оскільки незалежно відметода,я к и й

використовується,вірогідність викриття таємної інформаціїзавждиіснує,точим більше інформації внесено в поток даних, тим більша вірогідність, що вонабудевиявленаметодами стегеоаналізу.Якщо збільшується кількість пакетів, щовикористовуютьсядляпередачітаємних даних,тоі збільшується кількість ретрансльованих пакетів, щозначнозбільшує вірогідність виявленняфактутаємноїпередачіінформації. Дотогож втрата пакетів в мережі ретельно контролюється, а RSTEGметод викорустовуєлегальний трафік для створеннявтрат,таким чином збільшуються загальнівтратив мережі. І якщо рівеньвтратпісляпочаткуроботи по пересиланню таємних данихметодомRSTEG все щебудевідповідати рівнюочікуванихі прогнозованихвтратдля цієї мережі,тоце забезпечить секретність, оскільки небудепривертатиувагу.Щобпереконатись,що загальна кількістьвтраченихпакетів нормальна і доля RSTEG пакетів не занадто велика в порівнянні з іншимизв'язкамитієїжмережі,рівеньретрансльованихпакетівповиненжорсткоконтролюватись і динамічнозмінюватирівень RSTEG пакетів, щоб невидаватисекретний каналпередачіданих.

Методистеганографії з використанням ретрансляції пакетів RSTEG є гібридним,томупропускна спроможність приблизно дорівнюєметодамз модифікацію пакетів і вища в порівнянні зметодамизмінення порядкупередачіпакетів.МетодRSTEG добрепідходитьдля стекупротоколівTCP/IP [12], і приконтрольованомурівні ретрансляції даних не повиненвикликатипідозр в посторонього спостерігача.Недоліком,цього методує складна реалізація.

Методмодифікації заголовків пакетіввикористовуєнадлишковість в службових поляхпакету.Оскільки велика кількість пакетів якіпроходятьчерезмережу,постійно реконфігуровуються,тодля найкращихрезультатівпередачіданих існує велика кількістьзначеньі полів, які відповідають за певну характеристикупередачі цьогоповідомлення.Ц е

можнавикористовуватидля конфігурації навиглядзвичайних пакетів, де місцем для секретних данихбудутьслужбові інформаційні поляцьогопакету.Наприклад припередачіодногопакетуполе ID, невикористовуєтьсяце можнавикористатив своїх цілях і записати в ці 2 байти секретне повідомлення з 2байт,при цьому для спостерігача небудежодноїпідозри в наявності секретного повідомлення в полі ID, оскількиінколинавідь без необхідності деякі програми формують одинарні пакети з заповненим полем ID. Перевагами цього способа є протота реалізаці, та середня стійкості. Недоліками єдужемала кількість таємно переданих даних, атакожризикбутипоміченим при спробпередачівеликоїкількості даних, оскількизначнозростає кількість одноразових повідомлень, що неможене привернутиувагу.

Досліджуючиметодмодифікації заголовків пакетів на базі стекупротоколівTCP/IPбувзнайдений спосібпередачітаємних повідомлень, а саме модифікація даногометоду шляхомвикористання одноразових мережевих посиланнь.

Проаналізуємо даний підхід.Методосновний на створенні одноразових URL, які генеруються на сервері з високим ступенем довіри,наякомукористувачформує зашифроване секретне повідомлення. Після чого посилання,якебулостворенекористувачем,залишається відомим тількийому.Далі заметодоммодифікації заголовків пакетів, формується стеганограма, в яку записується одноразове посилання, іключдля розшифрування повідомлення. Після чого відбуваєтьсяпередачаотримувачу,отримувачпринадходженніпакетів вибирає за відомим йомуметодомдані зі стеганограми. На бізі отриманих даних формується одноразове посилання іключдля розшифрування, після чого,користувач переходитьза одноразовим посиланням. На сервері при регістраціїфактупереходуза посиланнямавтоматичнов и д а л я є т ь с я с т в о р е н а с т о р і н к а і д а н і з а л и ш а ю т ь с я л и ш е н а п р и с т р о ї , з якогокористувачперейшов за посиланням.Використовуючи

раніше отриманийключ(в і н можебути8-байтним) розшифровується повідомлення.Такимчином дляпередачіN-ної кількості байт інформаціївикористовутьсятака кількість байт стеганограми, яка дорівнює довжині посилання іключа,розмір яких варіюється від 8 - 128 байт дляключаі 8 - 40 байт для посилання.Такимчином, в найкращому випадку можна отримати N байт інформації за допомогою секретно переданих 16байт.Основнимнедолікомцьогометоду,це те, що потрібноматиповністю довіренний сервер, що є дорогим способом. Безпекацього методає вищою за середньою, оскільки кількість пакетів, які після модифікаціїбуливідправлені, є мінімально можливою. Таким чином, майже відсутня можливість реєстраціїпередачітаємного повідомлення.

Із дослілжених методологій було обрано метод, який задовільняє вимогам розробленої системи, тобто є простим в реалізації та характеризується досить стійкою до зміни стандартів мережі, і має средній рівень захисту, а саме метод модифікації заголовку пакету.

В основі майже всіхкомпютернихмереж лежить стекпротоколуTCP/IP,вінвключаєв себе певну кількістьпротоколівякі забезпечуютьпередачуданих по мережею, атакожвзаємодію мережевих пристроїв за допомогою службових та управляючих повідомлень. Одним із таких службовихпротоколівєпротоколICMP(Internet Control Message Protocol - протоколміжмережевийкеруючихповідомлень), призначенняякогополягає ввідправціповідомленьпривинекненніпомилок,перевіркизв ку,приформуванні маршрутівпередачітощо. На рисунку 2.9 зображений заголовпакетуIP.

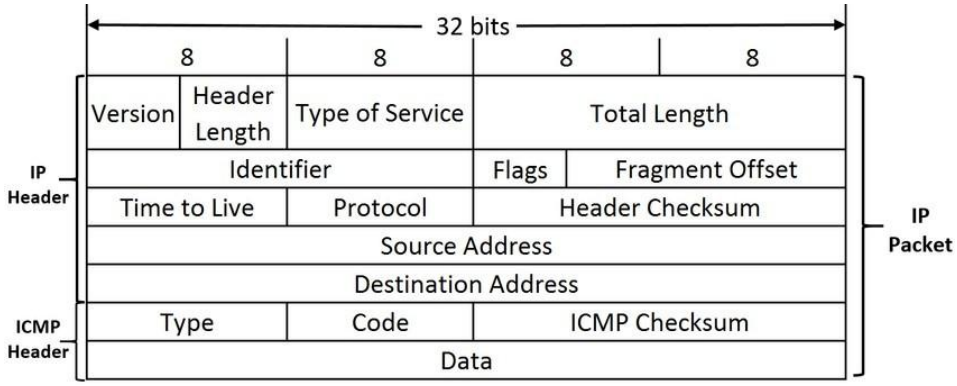

Рисунок 2.9- Структура IP пакета

В залежності від необхідної функції пристрій, який формуєICMP-пакет,вказуєвйого заголовкувідповіднізначенняв полі “тип”,якеівизначає форматпереданого пакета. При цьому в відповідності до стандартів для деякахтипів ICMP-повідомлень необхідно заповнити поле даних для отриманнянеобхідногорозміра пакета длятого,щоб сформуватикомандур i n g .

В результаті незважаючи на те що ICMP є службовим протоколом його присутність чи відсутність його даних в полі корисного навантаження пакета не може використовуватись як ознака передачі таємної передачі даних.

Процес інтеграції інформації в пакет, який потім інкапсулюється в інший пакет нижнього рівня і представляє собою основу функціонування цифрових мереж передачі даних.

Для сучасних компютерних мереж, що використовують стек TCP/IP, підтримка протоколу ICMP є обовязковою, що дає беззаперечні переваги при використанні цього протоколу для передачі секретних повідомлень. Однією з таких переваг є те, що періодична розсилка ICMP-повідомлень вузлами мережі, яка відбувається в наслідок помилки в IP-адресах пакетів або відсутність маршрутів до інших мереж, що призводить до великої кількості ICMP-пакетів, які містять службову інформацію, серед яких зручно сховати пакети з вбудованою в них стеганограмою.

Відключення даногопротоколуа б о йогоблокування за допомогою мережевих екранів призводить дотого,що порушується робота мережі, а отже не є хорошим способом захисту від пересилки стеганографії.Також не можна не відмітити складність внутрішньої структури пакета івеликоїкількості можливихкомбінацій полівзаголовкаікорисноїнавантаження. В стандартах ICMP описано 40 різних типів формування пакета, що можутьвиникатиібутикоректними[8].

В теперішніхумовахінформаційних війн проблема таємноїпередачіданих особливо актуальна. В останній час стали популярніметоди,при якихтаємнаінформаціяпередаєтьсячерезкомп ютернімережізвикористанням особливостей роботипротоколів передачіданих. Типовіметодимережевої стеганографіївключаютьв себе зміну властивостей пакетів, такі як зміну стану полівпакету,або зміну часунадходження.Показано що насьогоднініодиніз реальнихметодівне є досконалим і при збільшенні кількості інформації , що передається зростає ризик їїзнаходження методамистегоаналізу.

2.2 Використані технологій та їх опис.

Робота виконана на базі операційної системи Ubuntu. Для розрбленого програмного зпбезпечення було використані такі технології, як Python - високорівнева мова програмування, що орєнтована на продуктивність розробника, тобто його швидкість написання коду. Ще мов має простий синтаксис, а також є кросплатформенність за рахунок свого інтерпретатора. Інтерпретатор Python значно повільніший в порівнянні з іншими мовами, але має значну базу бібліотек, які пришвидшують швидкість розробки.

Для обраноїметодикибуловикористано бібліотеку Scapy - безкоштовупрограмну бібліотека для роботи з мережевими пакетами, основними перевагамиякоїє можливість створення власних протоколів,наглядність

отриманних даних, простота використання та підтримка Python. На рисунку 2.10 представлено створення пакету і його подільше представлення всередовищіScapy.

Рисунок 2.10 - Формування пакета за допомогою Scapy

Для аналізу пакетів (рисунок2.11)булавикористана програма Wireshark — програма, яка аналізує весь вхідний і вихідний трафік, має зручний інтерфейс з можливістю вибору фільтрів, атакожм о ж л и в і с т ь

побайтного представлення пакетів і їх полів, функцію збереження вибірки пакетів чи самого пакета.

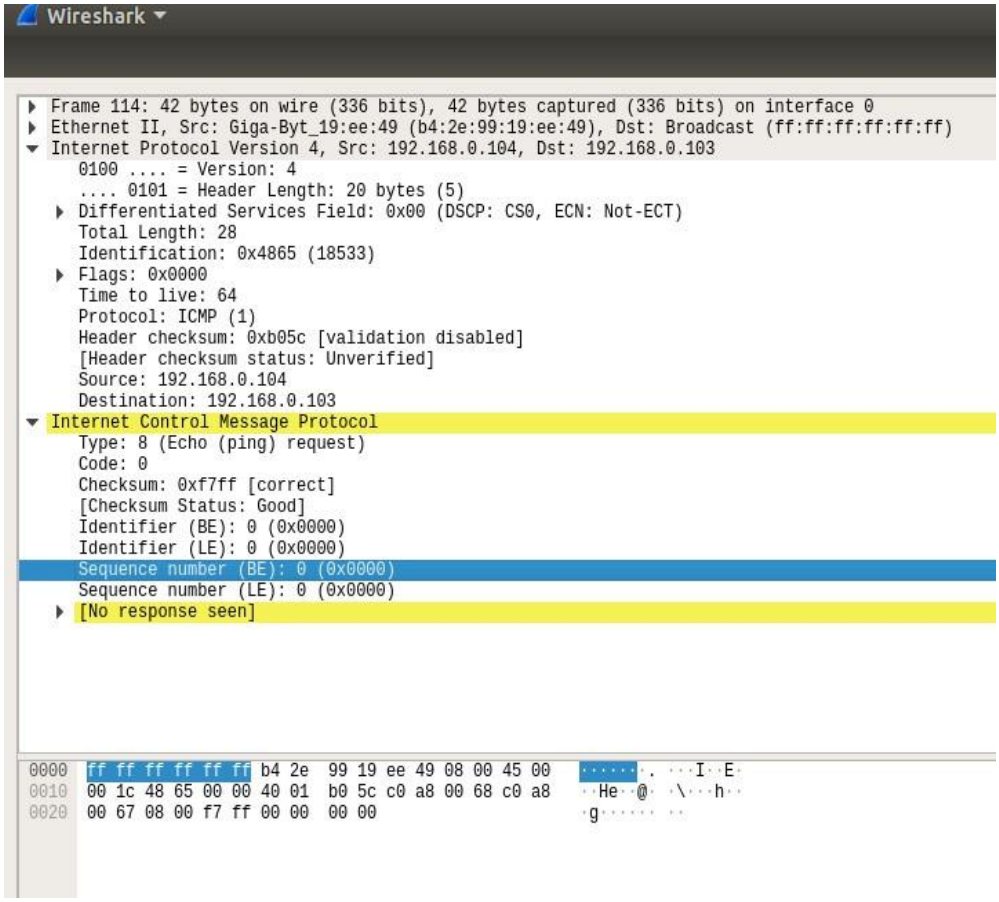

Рисунок 2.11 - Аналіз вихідного пакета

3.РОЗРОБКА

3.1 Реалізація метода модифікації заголовку пакету на основі стеку протоколів TCP/IP а також ICMP.

Практична реалізація метода модифікації пакетів зводиться до:

1) формування пакета з необхідними даними в поляхзаголовку;

2) відправку сформованого пакетаотримувачу;

3) отримання пакетакористувачемшляхомвиділенняйогосеред загального трафіку;

За допомогою програмного коплексу-бібліотеки Scapy було здійснено генерацію пакетів. Бібліотека Scapy виконує будь-яку взаємодію через інтерпретатор Python'a. Це здійснюється за допомогою викликів функцій із формуванням параметрів. За допомогою Scapy можна створювати пакети високого рівня (мережевого та прикладного), а Scapy автоматично доповнить формування нижніх рівнів, крім того можна збирати вручну, починаючи з канального рівня.

Для нашої задачі за допомого бібліотеки створемо пакет який буде передавати таємні дані і протестуємо його. Розробка за допомогою мови Python. На рисунку 3.1 показано створення пакету IP/TCP за допомогою Scapy.

```
>>>
>>> packet=IP(dst="192.168.10.10")/TCP(dport=22)/"TEST"
>>>
```

Рисунок 3.1 - Створення пакету

```
def two_byte_sender(firs_byte,second_byte):


    # Сотворюємо пакет для передачі стеганограми
    payload = ord(firs_byte) * 0x100 + ord(second_byte)
    pkt = IP(src="192.168.0.104", dst="192.168.0.103", id = payload) / ICMP(type = 8)
    # Отправляем пакет и ждём ответа
    sr1(pkt)
```

Рисунок 3.2 - Код формування пакету

Сформований пакет має адресу отримувача-192.168.0.103 та відправника-192.168.0.104. Відправлено таємне повідомлення “AB”, яке записане в поле Indentification.

За допомогою програми-аналізатора Wireshark спостерігаємо, що повідомлення було відправлено в мережу на рисунку 3.3.

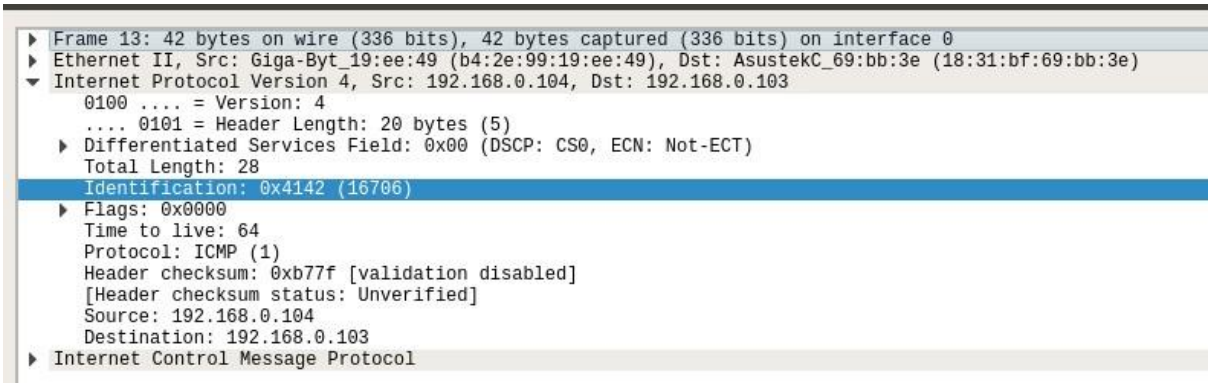

Рисунок 3.3 - Відправка пакету в мережу з заданими параметрами

Здійснемо побайтовий аналіз пакету за допомогою Wireshark

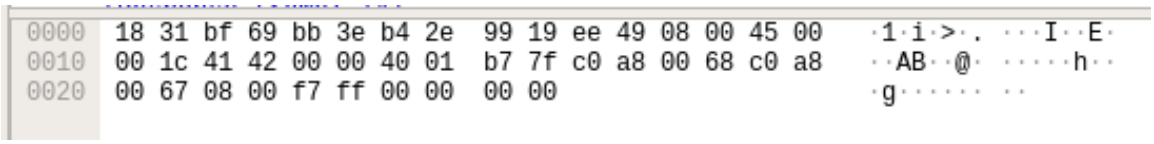

Рисунок 3.4 - Побайтове представлення пакета

Як бачимо, сформований пакет правильний, і в ньому спостерігається сховане відправником повідомлення “AB” показано на рисунку 3.5 .

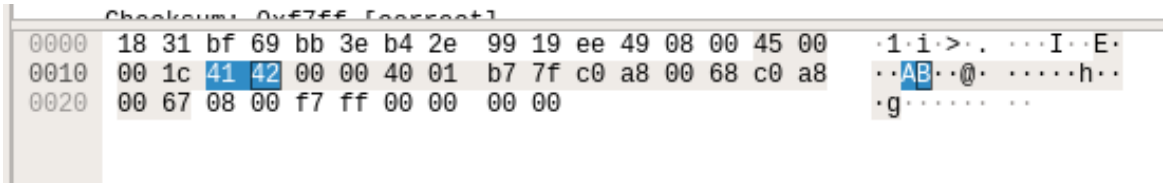

Рисунок 3.5 — Визначення таємного повідомлення

Припередачіодноразових повідомлень доцільновикористовуватиID-пакета, оскільки поле ID необхідне для формуваннявпорядкованоїпослідовності фрагментів даниходногоповідомлення. Тобтов випадку єдиного пакета це поле невикористовується,отже можнавикористати йогодляпередачітаємного повідомлення, а це 2 байти інформації з повідомлення.Такасхема передачівимушує створювати затримку між пакетами, щобвикористовуватиполе ID не запризначенням. Такимчином,корисненавантаження на 1пакетдорівнює 2 байтам. При цьому необхідно дотримуватись необхідних затримок, щоб невикликатипідозр.

При подальшому аналізі можна виявити, що якщо ICMP пакет буде також одним, то можна використовувати поле Sequence, яке вказує яким має бути наступний пакет. Для задач, де необхідна відправка одного довгого повідомлення, воно розбивається на декілька пакета. Оскільки в нашому прикладі це один пакет, то теоретично можна використати це поле для преедачі повідомлення.

Після того визначено поле, яке можна використовувати для введення стеганограми, відішлемо стеганограму. І як показано на рисунку пакет успішно сформований і відправлений. В побайтному представленні пакета можна побачити що таємне повідомлення “HELL”, де 2 байти віддано - на “HE”, а ще 2байти на “LL” зображений на рисунку 3.6 .

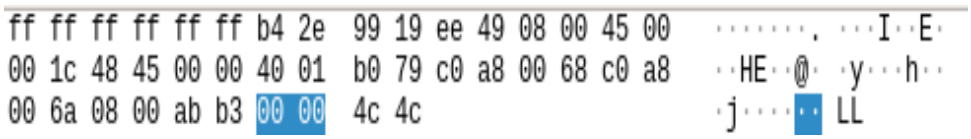

Рисунок 3.6 - Побайтове представлення повідомлення

Повідомлення було доставлено і успішно прийнято. Отже практична реалізація метода модифікації заголовків пакетів успішно виконана.

Розглянемо далі процес автоматизації цього методу. Розроблена програма має функції формування, шифрування, відправки та прийому повідомлення, а також їх розшифрування. Для цього складемо архітектурний опис програмного засобу.

Програма складається з 5 модулів.

На в додатку №1 представлена схема взаємодії модулів.

1. Інтерфейсниймодульн е о б х і д н и й д л я швидкоїі простої взаємодії зкористувачем.Дозволяє вибратиметодвідправки, адрес і самеповідомлення.
2. Модульформування повідомлення - після введеннякористувачемвсіх необхідних даних він зашифровує дані, і формує кількість необхідно створенихпакетів.
3. Модульвідправки повідомлення - відправляє повідомлення за заданою адресою.
4. Модульпрослуховування -модульв активному режимі слухає всі порти або тільки той порт який необхіднийкористувачу,і при отриманні повідомлення,якезадовільняє умови використання запропонованогометодувідправляєйогодля подільшоїобробки.
5. Модульобробки повідомлення - за зарання уговореним способомвизначаєстеганограму з отриманих повідомлень,використовуючинеобхідний, відомий спосіб і розшифровує його.

1.Інтерфейсниймодуль.

Інтерфейсний модуль забезпечує просту та швидку взаємодію з розробленою системою. На рисунку 3.7 представлено режим введення повідомлення.

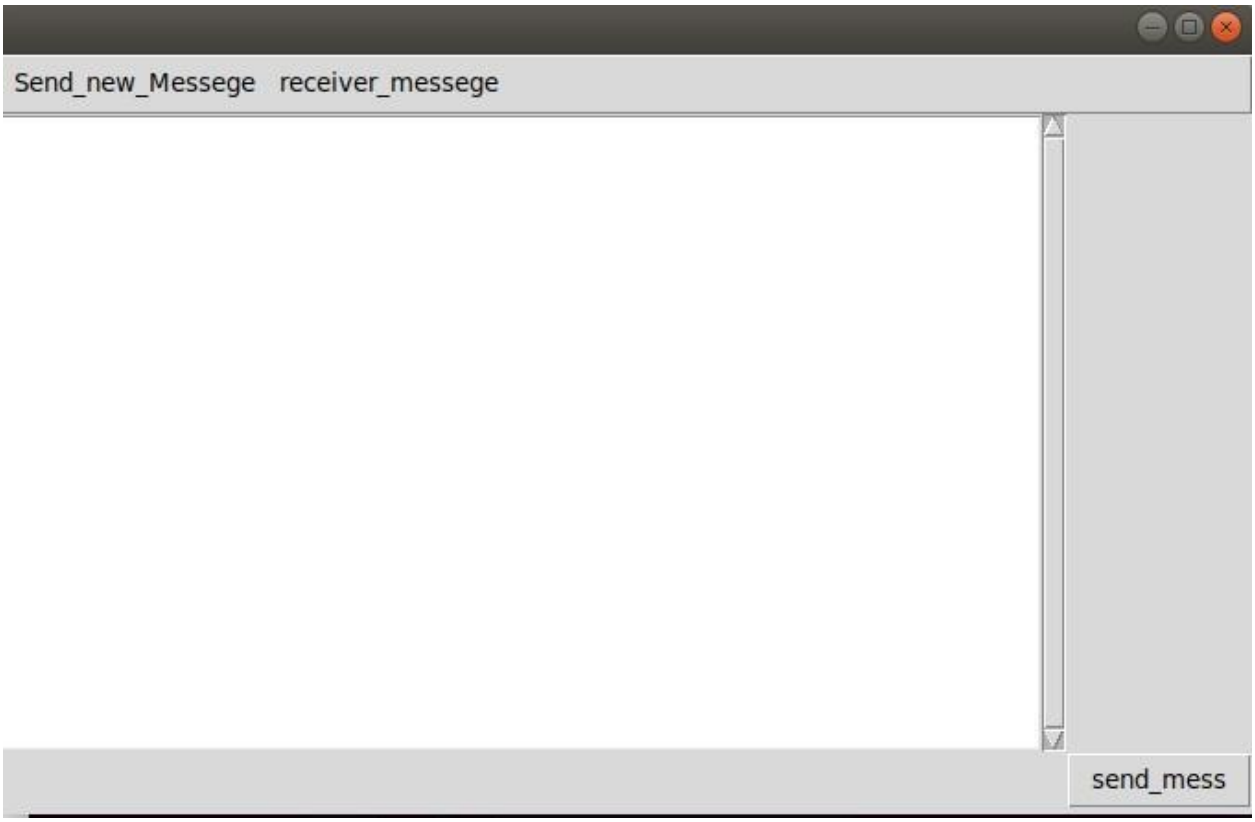

Рисунок 3.7 — Поле введення повідомлення

Після введення повідомлення і натиснення кнопки (send_mess) виникаєвікнопідтвердженнявідправки рисунок 3.8, вякому обов язковозаповненимимаютьбутиполя ey iDst_IP.Поле еуслужитьдлявказанняключа шифрування, щобшифруватинаписане повідомлення, полеможебутинезаповнене втакомуразі повідомлення небудезашифроване, а відправленне без шифрування. Поле Dst_IP необхідне для вказання адресакудинеобхідно доставити повідомлення. ПолеTimesend при заповнені створює заплановануподіїіпочинаєвідправку повідомлення в вказаний час.

Рисунок 3.8 — підтвердження відправка повідомленя

В вкладці receiver_messege рисунок 3.9,знаходитьсяменю налаштування прийому повідомлення. Поля scr_address і Port_receiver не є обовязковими до заповнення, як і вибірпротоколупакетів. У разі пустих полів після старту прийому повідомлень. Відбувається активне прослуховування мережі, в цьому режимі всі вхідні пакетибудутьвиводитись в вікні sniffer рисунок 3.10.

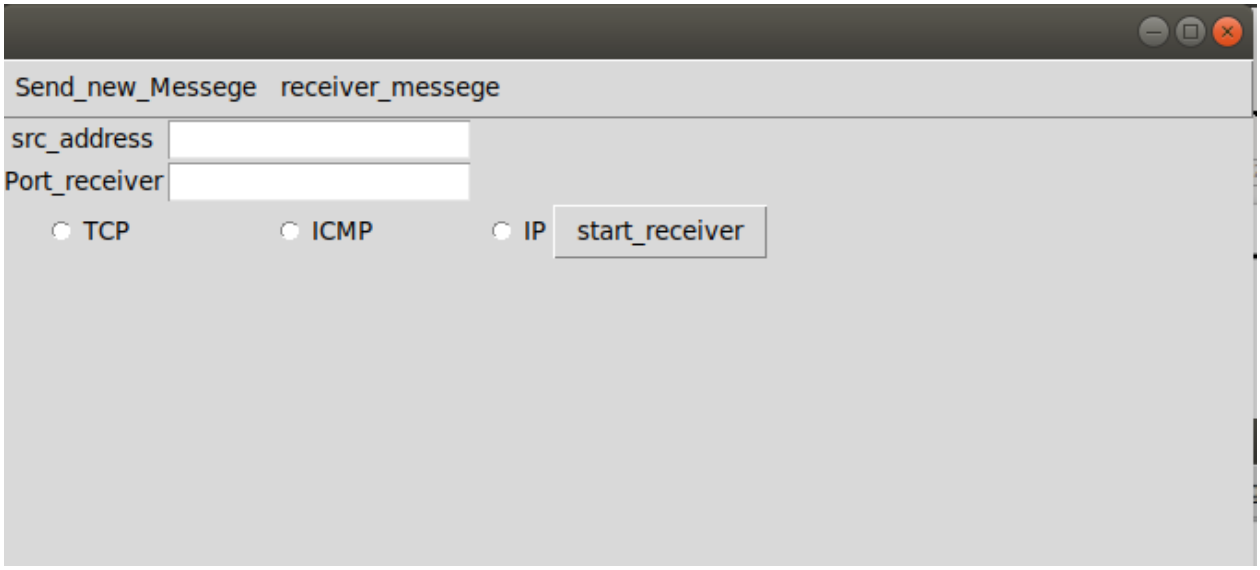

Рисунок 3.9 — Вкладка scr_address

Рисунок 3.10 — Вікно виводу інформації про вхідні пакети Sniffer

Якщоприйнятеповідомленнябулозашифроване,товнеобхіднеполе | | | | | | еувводимоключшифрування рисунок3.11.

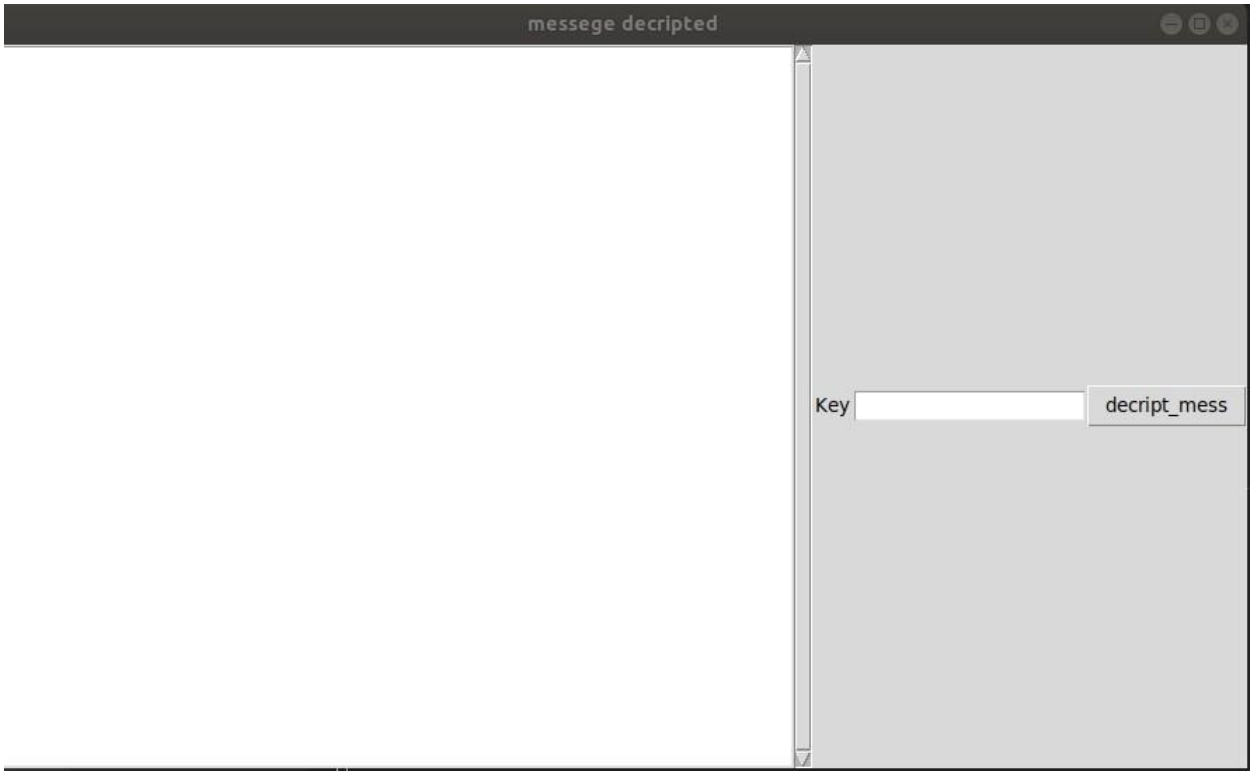

Рисунок 3.11 — Вікно з отриманим повідомленням

2.Модуль формування повідомлення .

Післятого,яккористувачзаповнить поля, щовказуютьна адресата та поля, які відповідають за вибірметода,якийбудевикористовуватись,виникає необхіднітсь в попередній обробці повідомлення. Насамперед повідомлення,щопередаєтьсятаємнимканаломзв язку,немаєвикликатипідозр,тобтойоговиглядмаєбутинайбільш віддаленим від структурованого, для чого застосовуються криптографічніметоди.

Скористаємся криптографічнимметодом**DES**(*Data EncryptionStandard*),схемароботи предсавлена на рисунку 3.12, алгоритмом для симетричного шифрування розроблений фірмою IBM і в 1977 роціприйнятий за офіційний стандарт владою США [5]. В основі алгоритму лежить використанняметодаФейстеля с 16 циклами іключом,який має довжину 56біт.Алгоритмвикористовує комбінаціюнелінійних і лінійних перетворень.

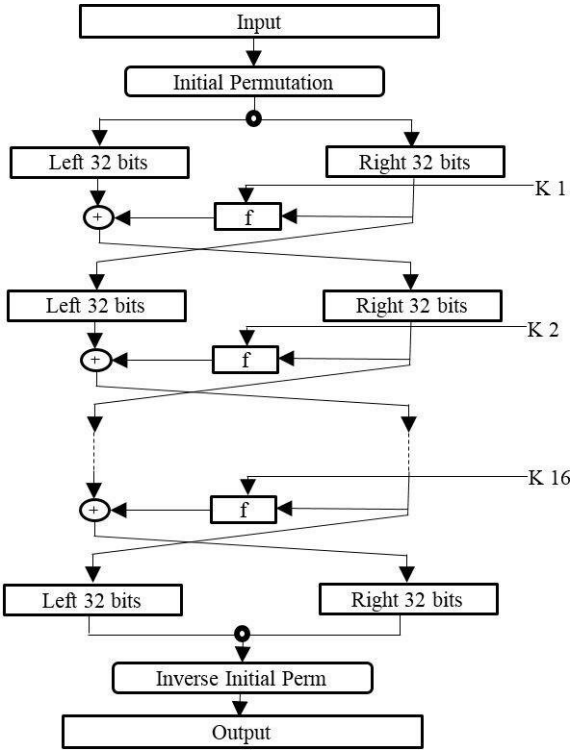

Рисунок 3.12 - Схема шифрування алгоритма DES

DESбувнаціональним стандартом США в 1977-1980 рр., Але в даний час DESвикористовується(зключемдовжини 56 біт) тільки для застарілих систем, найчастішевикористовують йогобільш кріптоустойчивость вид(3DES,DE).Длянашихжецілей шифруваннязадопомогоюDESпідходить.Оскільки він має реалізовані функції на базі бібліотеки Crypto яку підтримує Python. Крімтогошвидкість шифрування в нього висока, і основна мета - це повністювиключитиймовірністьзнаходженняструктурованних даних, що і забезпечує алгоритм. На рисунку 3.13 представлено приклад шифрування, який написаний на мові Python.

```
1  from Crypto.Cipher import DES
2
3  key = b'12345678'
4
5  def pad(text):
6      while len(text) % 8 != 0:
7          text += b' '
8      return text
9
10 des = DES.new(key, DES.MODE_ECB)
11 text = b'Hello world'
12 padded_text = pad(text)
13
14 encrypted_text = des.encrypt(padded_text)
15 print(encrypted_text)
16
```

Рисунок 3.13 - Приклад шифрування методом DES на мові Python

Після того, як дані були зашифровані, рахуємо скільки пакетів необхідно для передачі всього повідомлення. Оскільки за один пакет можна передати 4 байта, то необхідна кількість пакетів дорівнює кількості символів повідомлення поділеного на 4. Таким чином, у нас формується набір пакетів, які відправляємо наступному модулю.

3.Модульвідправки повідомлень(Sender)

Після розбиття повідомлення на пакети цей модуль починає формувати стеганограму, тобто він модифікує пакети за обраною методикою і за вказаною адресою здійснює передачу пакетів. При цьому для нормальної роботи модуль не потребує підтвердження зі сторони приймача. Після цього вся використана інформація, тобто саме повідомлення, налаштування відправки видаляються.

```
from scapy.layers.inet import *
class paket_sender:

    def icmp_prot_uses(firs_byte,second_byte,four_byte):
        payload = ord(firs_byte) * 0x100 + ord(second_byte)
        mess=ord(four_byte[0])*0x100+ord(four_byte[1])
        pkt = IP(src="192.168.0.104", dst="192.168.0.103", id = payload) / ICMP(seq=mess,type = 8)
        sr1(pkt)
        pkt.show()

    def two_byte_sender(firs_byte,second_byte):


        # Сотворюємо пакет для передачі стеганограми
        payload = ord(firs_byte) * 0x100 + ord(second_byte)
        pkt = IP(src="192.168.0.104", dst="192.168.0.103", id = payload) / ICMP(type = 8)
        # Отправляем пакет и ждём ответа
        sr1(pkt)

    def six_byte_sender(two_byte,four_byte_inform):

        steg_2=ord(two_byte[0])* 0x100+ord(two_byte[1])
        steg_4=ord(four_byte_inform[0])* 0x100+ord(four_byte_inform[1])+ord(four_byte_inform[2])+ord(four_byte_inform[3])
        print("steg_4",steg_4)
        pkt=IP(src="192.168.0.104",dst="192.168.0.101",id=steg_2)/TCP(seq=steg_4)

        sr1(pkt)
```

Рисунок 3.14 - Приклад коду по формуванню та відправленню пакету

4.Модульпрослуховування(Listner)

Відповідно доспочаткуо б у м о в л е н и м и н а л а ш т у в а н н я м и отримувачналаштовуєпрограму.Далі активізована програмакористуючисьмодулем(Listner рисунок 3.15), починає в активному режиміпрослуховувативесь вхідний трафік. Отримавши перше повідомленнявизначаєскільки пакетів

чекати. Далі відбувається процес поступового визначення стеганограми, причому визначення виконується послідовно з надходженням пакетів.

```python
from scapy.all import *
from scapy.layers.inet import ICMP
# Настраиваем прослушивание пакетов
#   filter -- только icmp
#   timeout -- слушаем только 10 секунд
#   count -- ждём не больше 100 пакетов
#   iface -- только на интерфейсе eth1


packets = sniff(filter = "icmp", timeout = 10, count = 100)
print(packets)
# Итерируемся по всем полученным пакетам
for pkt in packets:
    if pkt[ICMP].type != 8:
        continue
        # Просим красиво напечатать
    pkt.show()
```

Рисунок 3.15 - Приклад Listnera

ізналаштуваннями5.Модульобробкиповідомлення.

Як тільки повністю надійде повідомлення, починається процес розшифрування відповідно довизначеногоабо переданим повністю безпечним каналомключа.Після процесу розшифруваннякористувачмає змогу переглянути повідомлення. Після закінчення роботи програма повністю видаляє отриманні дані та введеніналаштування.

3.3 Тестування програмного забезпечення

Для тестування програмного засобу створено і відправлено повідомлення у реальній мережі. На рисунку 3.16 показано створене повідомлення за допомогою розробленого програмного засобу.

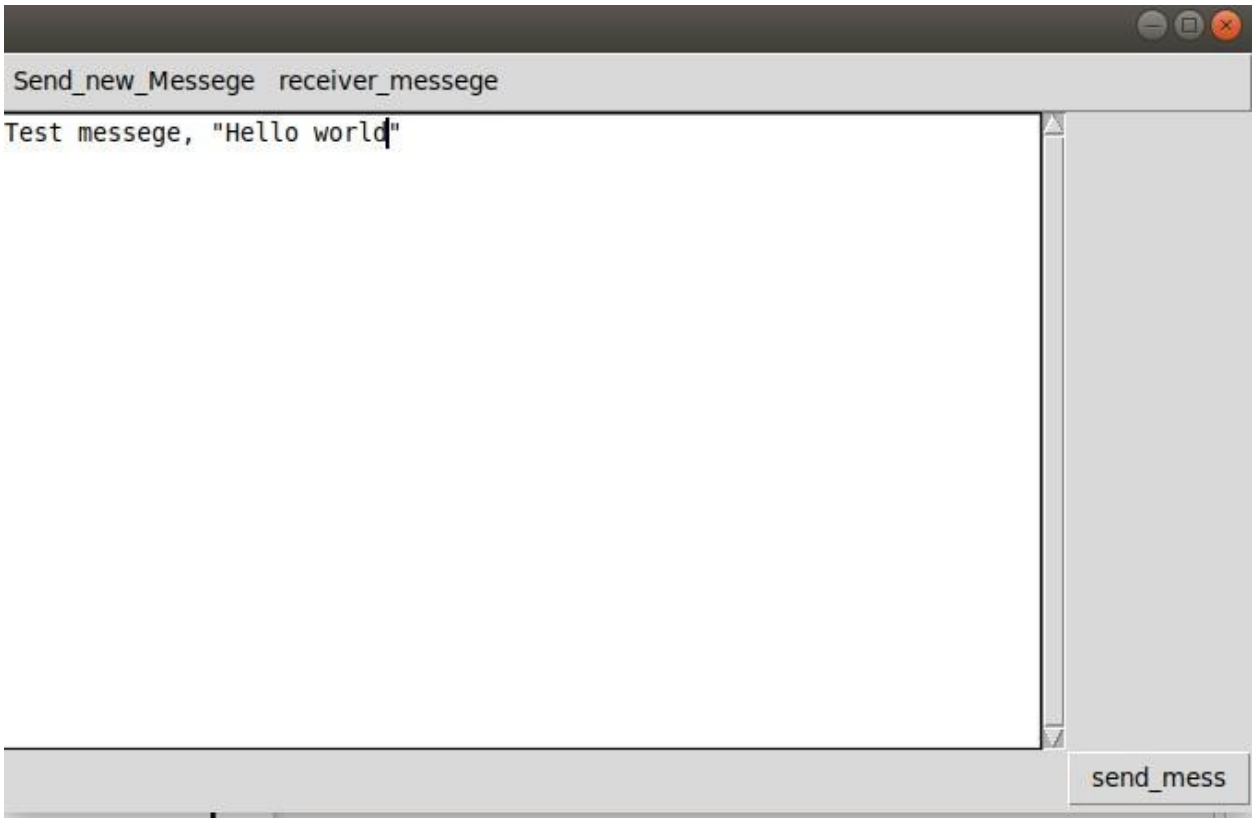

Рисунок 3.16 — Тестове повідомлення

В тей самий час на стороні приймача, увімкнутий режим прийому повідомлення. Аналізатор трафіка моніторить вхідні пакети (рисунок 3.17) і при виявленіпакетуякий задовільняє умови передає пакети вмодульобробки повідомлення.

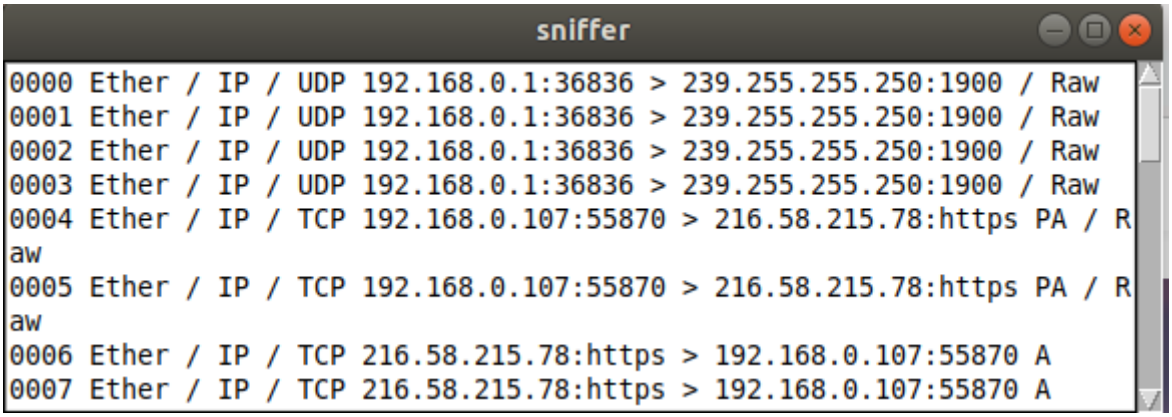

Рисунок 3.17 — Аналіз вхідного трафіку

При відправки повідомлення вказуємо ключ шифрування (рисунок 3.18).

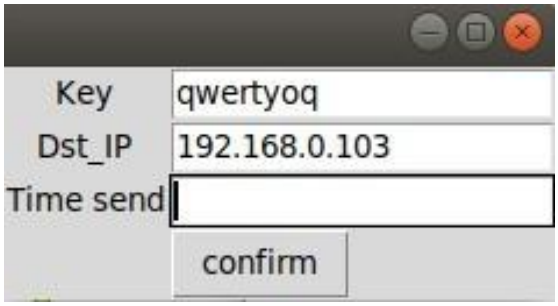

Рисунок 3.18 — Підтвердження відправки і вказання ключа шифрування

Після прийому повідомлення необхідно його розшифрувати (рисунок 3.19). Для цього необхідно ввести ключ, якщо ключ правильний то повідомлення можна прочитати. Якщо ключ є неправильним то повідомлення неможливо прочитати.

Рисунок 3.19 — Представлення прийнятого повідомлення ійогорозшифрування

Проведене тестування дає змогу оцінити роботу програми. За отриманими результатами при тестуванні в мережі провайдера PI-telekom, програмний засіб та методика працює. Таємне повідомлення було вдало передане і прийняте.

Висновоки

За заданим завданням було проведено дослідження сучасних технологій які забезпечують таємний мережевий зв’язок. Розроблено програму, яка дозволяє виконати передачу прихованих даних через мережу, приховуючи їх під виглядом службових повідомлень. При цьому однозначним плюсом створеної системи є те, що завдяки особливості роботи вибраного метода і стеку протоколів, не є можливим блокування надісланих повідомлень. Оскільки блокування стеку протоколів призвиде до порушення нормальної роботи мережі, а для виявлення стеганографічного повідомлення необхідно проводити глибокий аналіз всіх вхідних пакетів, що потребує великих обчислювальних затрат і може призводити до затримки передачі пакетів. Недоліки даної системи полягають в тому, що існує вірогідність виявлення факту передачі даних хоч і досить мала, і потребує великих зусиль та ресурсів. Також не було випробувано дану систему на мережах регіонально рівня. Що не дає змоги сказати про її правильну роботу при збільшенні віддаленості користувачів.

Поставлена задача виконана успішно хоча і має декілька недоліків. Створена система може використовуватись для обходу накладених на користувача обмежень, а також формування таємного каналу передачі даних.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. W.Mazurczyk,J.Lubacz,.Szczypiorski —Onsteganographyin lostaudio packets.2013

2. WojciechFrączek,rzysztofSzczypiorskiPerfect undetectability of network steganography Perfect undetectability of network steganography. Security Comm. Networks, 2016, 9: pp.2998–3010.

3. RFC 792 — Internet Control Message Protocol.URL: tools.ietf.org/html/rfc792.

4. BelkinaT.A.Molodoy uchenyy (Rus), 2018,No11.URL: moluch.ru/archive/ 197/48821/.

5. W.Frączek,W. Mazurczyk,.Szczypiorski.StreamControlTransmission ProtocolSteganography.WarsawUniversity ofTechnology,Institute of Telecommunications

6.Golubev E.A., Emel'yanovG.V.T-Comm- Telekommunikatsii itransport. 2009. NoS3. pp.185-186.

7. Programming Python, 4th Edition Powerful Object-OrientedProgramming Mark Lutz 2010 1632p.

8.. Тарасов Д.О. Класифікація та аналіз безкоштовних програмних засобів стеганографії / Д. О. Та-расов, А. С. Мельник, М. М. Голобородько // Інформаційні системи та мережі. ВісникНУ «Львівська політехніка». — 2010. — No 673. — С. 365–374.

9. БарсуковВ.С. Оценка уровня скрытностимультимедийныхстеганографических каналов хра-нения ипередачиинформации / В. С.Барсуков,А. П.Романцов// СпециальнаяТехника.—2000. — No1.

10. П.П.УрбановичЗащита инофрмацииметодамикриптографии, стеганогрфии и обфускации, 2010,216.

11. О.В. Вусельська,Р.В.Зюбіна, О.В.Фролов Систематизація та класифікація наявних стеганографічнихметодів приховуванняінформації.Стаття2016року
12. Retransmission Steganography AppliedWojciechMazurczyk, MiłoszSmolarczyk, rzysztofSzczypiorskiInstituteofTelecommunicationsWarsaw University ofTechnologyWarsaw,Poland
13. “The Diamond Approach for SDN Security“Angelo Liguori, Huawei, German Research Center; and Marcel Winandy, Huawei, GermanResearch CenterIEEE Softwarization,March2018
14. evinBenton, LJeanCamp,andChrisSmall.Openflowvulnerability assessment. In Proceedings of the second ACM SIGCOMM workshop onHot topics insoftwaredefinednetworking,p.151–152.2013.
15. Аспектыинформационной безопасности архитектуры SDN А. А. Захаров, Е. Ф. Попов, М. М.ФучкоВестник СибГУТИ. 2016. № 1
16.Greg Goth, "Software-Defined Networking Could Shake Up Morethan Packets," IEEE Internet Computing, July/August,2011.