МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
КОНОТОПСЬКИЙ ІНСТИТУТ
Центр заочної та дистанційної форми навчання

Кафедра електронних
приладів і автоматики

Кваліфікаційна робота бакалавра

# РОЗРОБКА АЛГОРИТМУ ПЕРЕДАЧІ ДАНИХ В ІНФОРМАЦІЙНИХ СИСТЕМАХ

Студент групи ЕП – 61 Я.М. Стеценко

Науковий керівник Ю. В. Столярчук

Нормоконтроль,
ст. викладач, к.т.н. О.Д. Динник

Конотоп 2020

# РЕФЕРАТ

Об’єктом дослідження кваліфікаційної роботи є розробка алгоритму передачі даних в інформаційних системах.

Мета роботи полягає у порівнянні типових варіантів реалізації алгоритму передачі даних в інформаційних системах та в огляді сфери застосування таких алгоритмів.

При виконанні роботи було проаналізовано види, модифікації, переваги та недоліки алгоритмів передачі даних в інформаційних системах, а також розроблений алгоритм передачі даних.

Алгоритм AES з DES призначений для шифрування та розшифровки блоків даних, що складаються з 64 біт під керуванням 64-бітного ключа. У криптографії шифр розширеного стандарту шифрування (AES) має 128-розрядний розмір блоку з розмірами ключів 128, 192 та 256 біт відповідно.

Робота викладена на _32_ сторінках, у тому числі включає _10_ рисунків, _5_ таблиць, список цитованої літератури із _23_ джерел.

КЛЮЧОВІ СЛОВА: АЛГОРИТМ, ПЕРЕДАЧЯ ДАНИХ, ІНФОРМАЦІЙНА СИСТЕМА, АЛГОРИТМ VIGENERE CIPHER, АЛГОРИТМ AES-DES.

**ЗМІСТ**

# ВСТУП

Сьогодні необхідність оновленої інформації стала неминучою для досягнення ефективного рішення у всіх сферах життя. Будь то промисловість, комерція, оборона, банківська справа, освіта, економіка чи політика, інформація потрібна всюди. [1]

Інформація в прямому ефірі, оскільки її потрібно постійно оновлювати і поновлювати.

Експонентний приріст інформації обумовлює необхідність збору, зберігання та отримання інформації в різних полях, коли це необхідно.

Наприклад (a) В умовах створення нової галузі інформація щодо вибору технології, навичок, грошей та матеріалів стає важливою вимогою для її зростання та безперебійного функціонування.

(b) На конкурентному ринку, перш ніж приймати рішення про ціну товару, виробник потребує інформації про цінову поліцію конкурентів, особливо про конкурентоспроможну продукцію, техніку продажу тощо.

Швидка еволюція цифрового обміну даними змусила це безпека інформації має велике значення при зберіганні та передачі даних. Оскільки велика кількість даних передається по мережі, перед їх надсиланням попередньо захистити всі типи даних. [2]

Проблема AES, найбільш широко використовуваного шифрування, полягає в тому, що він використовує безліч різновидових рівнянь, які мають лінійний характер. Таким чином, його можна порушити за допомогою алгебраїчного криптоаналізу.

Це створює серйозну загрозу, оскільки AES вважався непорушним, і тому він застосовувався у багатьох системах шифрування. У цьому

документі представлено розробку та реалізацію гібридного алгоритму AES-DES на основі 128 біт на базі гібриду як підвищення безпеки. [3]

# РОЗДІЛ 1
# ВИЗНАЧЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

## 1.1 Передача даних

Передача даних - це засіб передачі цифрових або аналогових даних через носій зв'язку на один або кілька пристроїв. Він дозволяє передавати та зв’язувати пристрої в різних середовищах:

- точка-точка;
- точка-багатоточка;
- багатоточка-багатоточка.

Передача даних може бути як аналоговою, так і цифровою, але в основному призначена для надсилання та прийому цифрових даних. Таким чином, передачу даних також називають цифровою передачею або цифровим зв’язком.

Він працює, коли пристрій має на меті передати об’єкт даних або файл на один або кілька пристроїв-одержувачів. Цифрові дані надходять від джерела пристрою у вигляді цифрових бітових потоків. Ці потоки даних розміщуються над комунікаційним середовищем для передачі до пристрою призначення. Зовнішній сигнал може бути або базовим, або смуговим. [4]

Крім зовнішнього зв'язку, передача даних може здійснюватися всередині між різними частинами одного пристрою. Відправка даних процесору з оперативної пам'яті (RAM) або жорсткого диска є формою передачі даних.

Види передачі даних:

- паралельна передача - кілька біт передаються разом одночасно в межах однієї тактової частоти імпульсу. Він передає швидко, оскільки використовує декілька вхідних та вихідних ліній для надсилання даних;

- паралельна передача використовує 25-контактний порт з 17 сигнальними лініями та 8 наземними лініями.

17 сигнальних ліній поділяються так:

- 4 рядки - ініціюйте рукостискання;
- 5 рядків - повідомляти та повідомляти про помилки;
- 8 рядків - передача даних

Послідовна передача - дані надсилаються побітно з одного комп'ютера на інший у двох напрямках. Кожен біт має тактову частоту пульсу. Вісім біт передаються за один раз з початковим і стоп-бітом, відомим як біт парності, який дорівнює 0 і 1 відповідно. Кабелі даних використовуються при передачі даних на більшу відстань. Кабель даних має D-подібний 9-контактний кабель, який з'єднує дані послідовно. [5]

Порівняння послідовної та паралельної передачі даних наведено в таблицы 1.1.

*Таблиця 1.1*

| **Основа для порівняння** | **Серійна передача** | **Паралельна передача** |
|---|---|---|
| Визначення | Дані протікають у двох напрямках, побіжно | Передача даних відбувається в декількох напрямках, 8 біт (1 байт) одночасно |
| Вартість | Економний | Дорогий |
| Кількість бітів, переданих за тактовий імпульс | 1 біт | 8 біт або 1 байт |
| Швидкість | Повільно | Швидкий |
| Програми | Використовується для міжміського зв'язку | Використовується для зв'язку на коротких відстанях |
| Приклад | Комп'ютер на комп'ютер | Комп'ютер на принтер |

Існує два типи послідовної передачі - синхронна та асинхронна. Обидва ці способи передачі використовують бітну синхронізацію. Бітова синхронізація необхідна для ідентифікації початку та кінця передачі даних.

Бітова синхронізація підтримує комп'ютер, що приймає, розпізнавати, коли дані починаються і закінчуються під час передачі. Тому бітова синхронізація пропонує контроль часу.

Асинхронна передача - в асинхронній передачі дані рухаються за підходом, що перебуває в половині, 1 байт або 1 символ одночасно. Він посилає дані постійним струмом байтів. Розмір переданого символу - 8 біт, з бітом парності, доданим на початку та в кінці, що робить його загалом 10 біт. Для інтеграції йому не потрібен годинник - скоріше, він використовує біти парності, щоб повідомити одержувачу, як перекласти дані. Це просто, швидко і економічно, і не вимагає двостороннього спілкування. [6]

Синхронна передача - при синхронній передачі дані переміщуються в повному парному підході у вигляді фрагментів або кадрів. Синхронізація між джерелом та ціллю потрібна, щоб джерело знав, з чого починається новий байт, оскільки між даними немає пробілів. Цей метод пропонує зв’язок у режимі реального часу між пов'язаними пристроями.

Порівняння синхронної та асинхронної передачі даних наведено в таблиці 1.2

*Таблиця 1.2*

| **Точка порівняння** | **Синхронна передача** | **Асинхронна передача** |
|---|---|---|
| Визначення | Передає дані у вигляді фрагментів або кадрів | Передає одночасно 1 байт або символ |
| Швидкість передачі | Швидкий | Повільно |
| Вартість | Дорогий | Економічно ефективним |
| Проміжок часу | Постійний | Випадкові |
| Чи є проміжки між даними? | Так | Ні |
| Приклади | Чати, телефонні розмови, відеоконференції | Електронна пошта, форуми, листи |

## 1.2 Режими передачі даних

Режим передачі даних визначає напрямок потоку інформації між двома пристроями зв'язку. Його також називають передачею даних або режимом спрямованості. Він визначає напрямок потоку інформації з одного місця в інше в комп'ютерній мережі.

У моделі шару шарів відкритого системного взаємозв'язку (OSI) фізичний шар присвячений передачі даних у мережі. Він, головним чином, визначає напрямок даних, в якому дані повинні подорожувати, щоб дістатися до приймальної системи або вузла.

Режими передачі даних на основі напрямку обміну, синхронізації між передавачем та приймачем та кількості бітів, що надсилаються одночасно в комп'ютерну мережу.

Режими передачі даних можна охарактеризувати у трьох наступних типах на основі напрямку обміну інформацією:

- симплекс;
- напівдуплекс;
- повний дуплекс.

Режими передачі даних можна охарактеризувати у наступних двох типах на основі синхронізації між передавачем та приймачем:

- синхронний;
- асинхронний.

Режими передачі даних можна охарактеризувати у наступних двох типах на основі кількості бітів, що надсилаються одночасно в мережу:

- серійний;
- паралельний.

Симплекс - це режим передачі даних, в якому дані можуть протікати лише в одному напрямку, тобто зв’язок є односпрямованим. У цьому режимі відправник може лише надсилати дані, але не може їх отримувати. Так само

приймач може приймати дані лише, але не може їх надсилати. На рис.1.1 зображено схематично симплексний режим передачі даних.

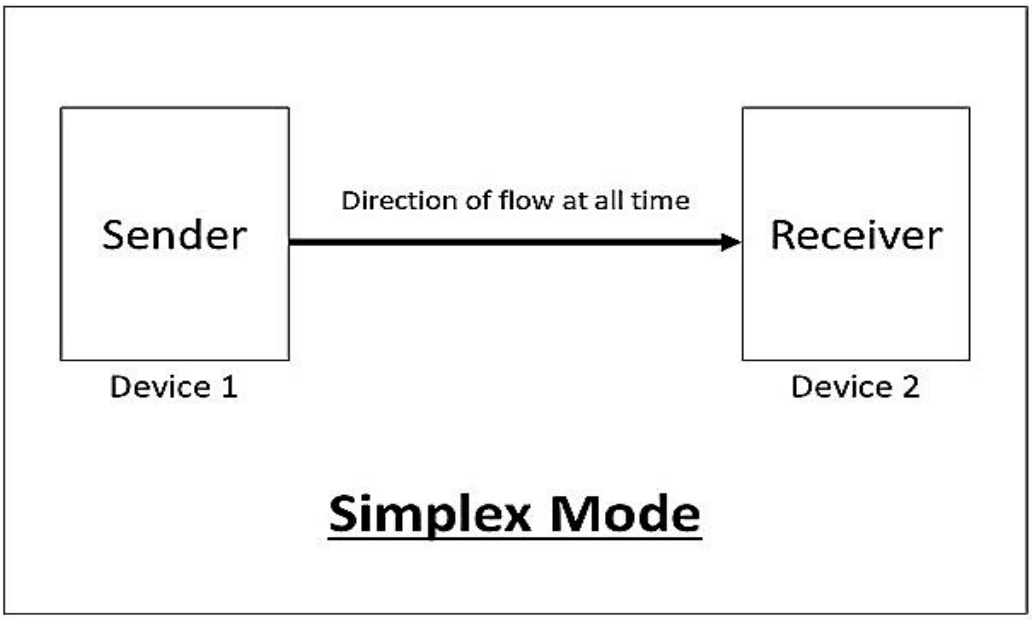



Рис.1.1. Симплексний режим передачі даних. [7]

Цей режим передачі не настільки популярний, тому що в цьому режимі не можливо здійснювати двосторонню комунікацію між відправником та одержувачем. В основному він використовується у сфері бізнесу, як у продажу, що не потребує відповіді. Це схоже на вулицю з одностороннім рухом.

Наприклад, передача радіо та телебачення, клавіатура, миша тощо.

Нижче наведено переваги використання режиму передачі Simplex:

Він використовує повну потужність каналу зв'язку під час передачі даних.

У ньому найменше або взагалі немає проблем з трафіком даних, оскільки дані протікають лише в одному напрямку.

Нижче наведені недоліки використання режиму передачі Simplex:

Він має односпрямований характер, не має взаємозв'язку між пристроями.

Не існує механізму передачі інформації назад відправнику (Немає механізму підтвердження).

Напівдуплекс - це режим передачі даних, в якому дані можуть протікати в обох напрямках, але в одному напрямку за один раз.

Його також називають напівдуплексним. Іншими словами, кожна станція може одночасно передавати та приймати дані, але не одночасно. Коли один пристрій надсилає інший, він може отримувати лише та навпаки. На рис.1.2 зображено схематично напівдуплексний режим передачі даних.

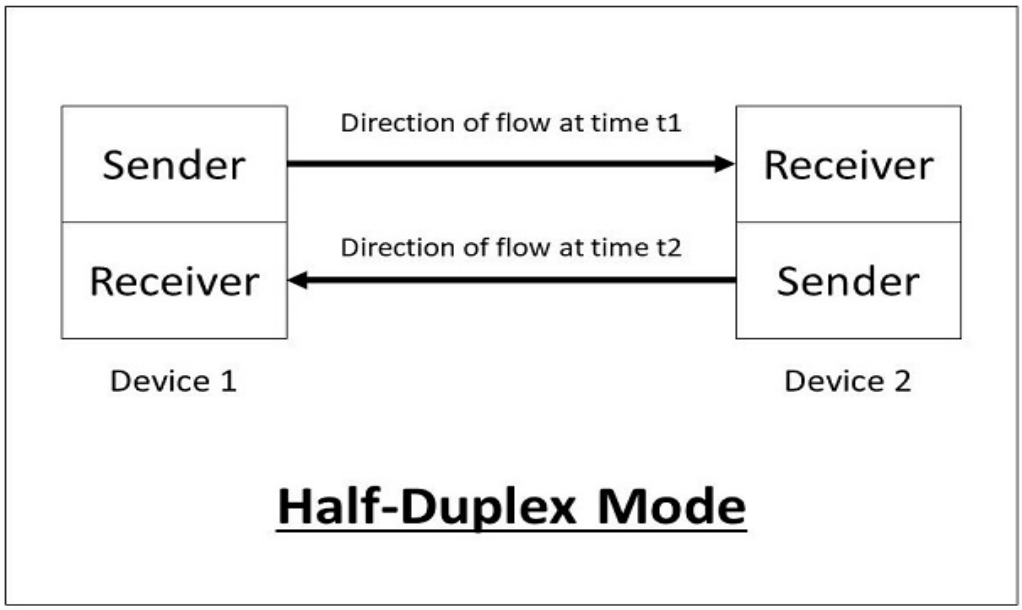



Рис.1.2. Напівдуплексний режим передачі даних [8]

У цьому режимі передачі може бути використана вся ємність каналу для кожного напрямку. Лінії передачі можуть нести дані в обох напрямках, але дані можуть надсилатися лише в одному напрямку за один раз.

Цей тип режиму передачі даних може використовуватися у випадках, коли немає необхідності спілкування в обох напрямках одночасно. Він може використовуватися для виявлення помилок, коли відправник не надсилає або отримувач не отримує дані належним чином.

У таких випадках дані потрібно знову передати одержувачем. Наприклад, Walkie-Talkie, Інтернет-браузери тощо. Нижче наведено переваги використання напівдуплексного режиму передачі:

- полегшує оптимальне використання каналу зв'язку;
- забезпечує двосторонній зв’язок.

Нижче наведені недоліки використання напівдуплексного режиму передачі:

- двосторонній зв’язок не може встановлюватися одночасно одночасно;
- затримка передачі може відбуватися, оскільки одночасно можливе спілкування.

Повнодуплексний - це режим передачі даних, в якому дані можуть перетікати в обох напрямках одночасно. Він носить двонаправлений характер. Це двосторонній зв'язок, в якому обидві станції можуть передавати та приймати дані одночасно. На рис.1.3 зображено схематично повнодуплексний режим передачі даних.

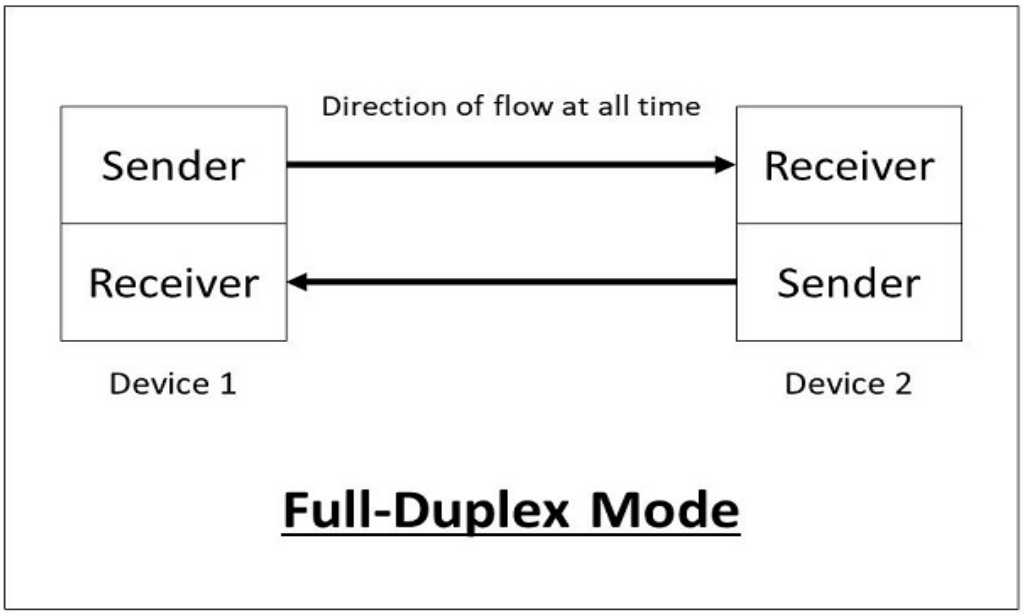



Рис.1.3. Повнодуплексний режим передачі даних [9]

Режим Full-Duplex має подвійну пропускну здатність порівняно з напівдуплексним.

Потужність каналу розділена між двома напрямками зв'язку. Цей режим використовується тоді, коли комунікація в обох напрямках потрібна одночасно. Наприклад, телефонна мережа, в якій обидва особи можуть говорити та слухати один одного.

Нижче наведено переваги використання режиму повнодуплексної передачі:

• двосторонній зв’язок може здійснюватися одночасно в обох напрямках;

• найшвидший спосіб спілкування між пристроями.

Нижче наведені недоліки використання напівдуплексного режиму передачі:

- потужність каналу зв'язку ділиться на дві частини;

- не існує спеціального шляху для передачі даних;

- має неправильне використання пропускної здатності каналів, оскільки існують два окремі шляхи для двох комунікаційних пристроїв.

Відповідно до синхронізації між передавачем і приймачем:

Режим синхронної передачі - це режим зв'язку, в якому біти надсилаються один за одним без будь-яких бітів запуску / зупинки або проміжків між ними. Насправді, і відправник, і одержувач рухаються тим самим системним годинником. Таким чином досягається синхронізація.

У синхронному режимі передачі даних байти передаються як блоки у безперервному потоці бітів. Оскільки в блоці повідомлень немає бітів запуску та зупинки.

Відповідач за групування бітів належить до відповідальності. Одержувач рахує біти по мірі їх надходження та групує їх у вісім біт одиниці. Одержувач безперервно отримує інформацію з тією ж швидкістю, що і передавач її надіслав.

Він також слухає повідомлення, навіть якщо жодних бітів не передається.

У синхронному режимі біти надсилаються послідовно, без поділу між кожним символом, тому стає необхідним вставити деякі елементи синхронізації з повідомленням, це називається " Синхронізація на рівні символів ".

Наприклад, якщо є два байти даних, скажімо, (10001101, 11001011), вони передаватимуться в синхронному режимі так як зображено на рис.1.4.

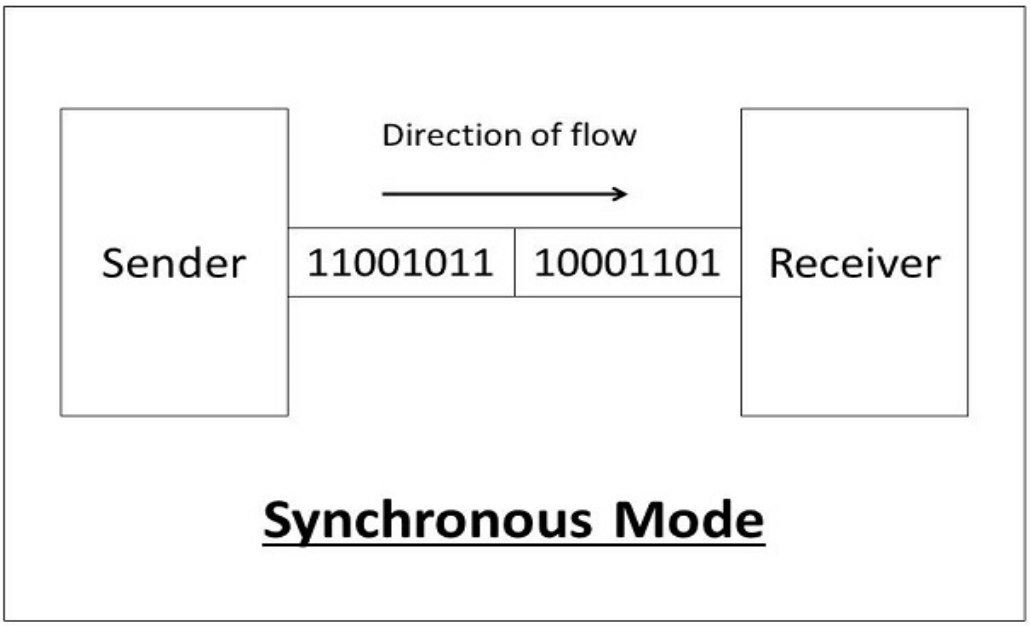



Рис.1.4. Синхронний режим передачі даних [10]

Наприклад, зв’язок у процесорі, оперативній пам’яті тощо. З переваг використання режиму синхронної передачі є швидкість передачі, оскільки немає розриву між бітами даних. З недоліків використання режиму синхронної передачі це дуже дорого.

Режим асинхронної передачі - це режим зв'язку, при якому в повідомленні під час передачі вводиться запуск і біт зупинки. Біти запуску та

зупинки забезпечують правильну передачу даних від відправника до приймача.

Як правило, початковий біт - "0", а кінцевий біт - "1".

Тут асинхронний означає "асинхронний на рівні байтів", але біти все ще синхронізовані. Тривалість часу між кожним символом однакова і синхронізована.

В асинхронному режимі зв'язку біти даних можуть надсилатися в будь-який момент часу. Повідомлення надсилаються з нерегулярними інтервалами і одночасно може надсилатися лише один байт даних. Цей тип передачі найкраще підходить для передачі даних на короткі відстані.

Наприклад, якщо є два байти даних, скажімо, (10001101, 11001011), вони передаватимуться в асинхронному режимі наступним чином як наведено на рис.1.5.

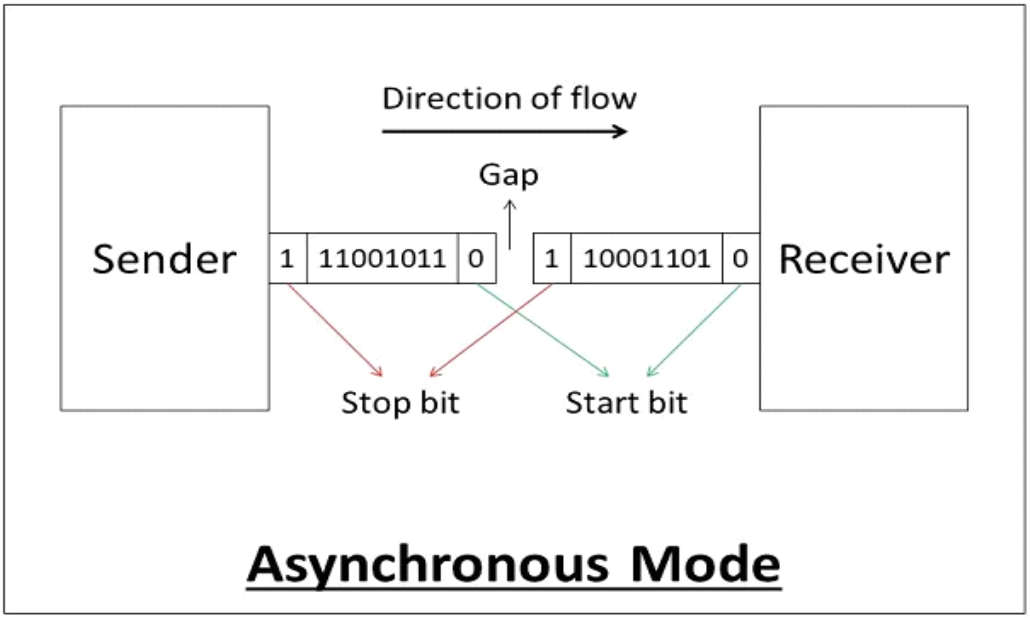



Рис.1.5. Асинхронний режим передачі даних [11]

Наприклад, введення даних з клавіатури на комп'ютер. Нижче наведено переваги використання режиму асинхронної передачі:

- дешевий і ефективний режим передачі;

- точність передачі даних висока через наявність бітів запуску та зупинки.

Нижче наведені недоліки використання режиму асинхронної передачі а саме передача даних може бути повільнішою через прогалини між різними блоками даних.

Режим передачі послідовних даних - це режим, в якому біти даних надсилаються послідовно один за одним по одному каналу передачі. На рис.1.6. наведено схематично режим передачі послідовних даних.

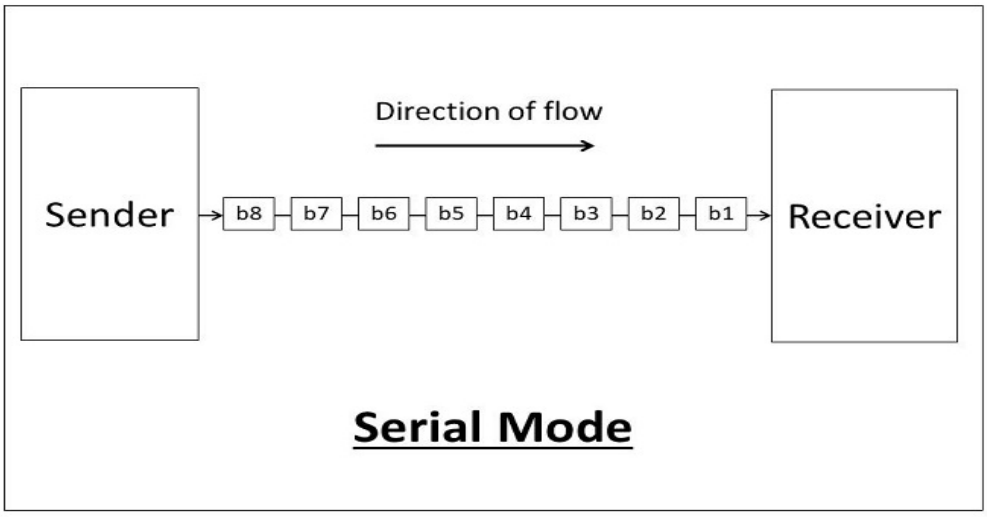



Рис.1.6. Режим передачі послідовних даних. [12]

Для зв'язку потрібна єдина лінія передачі. Біти даних приймаються синхронно один з одним. Отже, існує виклик синхронізації передавача та приймача. При послідовній передачі даних система займає кілька тактових циклів для передачі потоку даних. У цьому режимі зберігається цілісність даних, оскільки вона передає біти даних у визначеному порядку один за одним.

Цей тип передачі найкраще підходить для передачі даних на далекі відстані, або кількість відправлених даних порівняно невелика. Наприклад, передача даних між двома комп'ютерами за допомогою послідовних портів.

Нижче наведено переваги використання режиму послідовної передачі:

• можна використовувати для передачі даних на великі відстані, оскільки це надійно;

• кількість проводів і складність менше;

• рентабельність.

З недоліків використання послідовного режиму передачі це швидкість передачі даних повільна завдяки одному каналу передачі.Режим паралельної передачі даних - це режим, в якому біти даних надсилаються паралельно одночасно. Іншими словами, одночасно відбувається передача n-бітів.

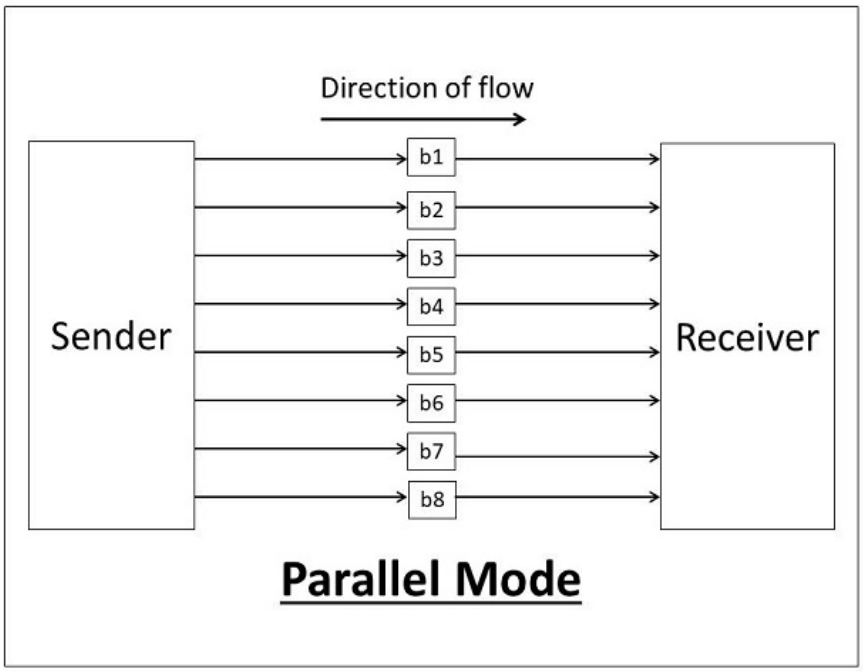



Рис.1.7. Режим паралельної передачі даних [13]

У таких режимах передачі використовуються кілька ліній передачі. Отже, кілька байтів даних можуть передаватися за один системний такт. Цей спосіб передачі використовується, коли велику кількість даних потрібно

надсилати за менший проміжок часу. В основному використовується для зв'язку на коротких відстанях. Для n-бітів нам потрібні n-лінії передачі. Отже, складність мережі зростає, але швидкість передачі висока. Якщо дві або більше ліній передачі занадто близькі один до одного, можливо, є ймовірність втручання в дані, погіршуючи якість сигналу. Наприклад, передача даних між комп'ютером та принтером.

# РОЗДІЛ 2
# КРИПТОГРАФІЧНИЙ АЛГОРИТМ VIGENERE CIPHER ПРИ ЗАХИЩЕННІ ПЕРЕДАЧІ ДАНИХ

## 2.1 Дані як основа алгоритмізації

Інформація - найцінніший актив, який слід захистити від крадіжки. Дані можуть бути у формі важливої інформації, яка не повинна широко розповсюджуватися, оскільки має небезпечний або життєво важливий зміст.

Доставка цього типу інформації повинна здійснюватися ретельно і не знати інших людей.

Якщо інформація викрадена і потрапляє до рук людей, які не несуть відповідальності, то ці дані можуть бути неправомірно використані або використані як джерело незаконного пошуку грошей [14].

Для забезпечення цієї інформації потрібні хороші прийоми, щоб перетворити цю інформацію в рядкові слова, які не можуть зрозуміти інші. У світі комп'ютерів інструменти для цього називаються криптографією.

Криптографія - це мистецтво перетворювати оригінальне повідомлення в непрочитане повідомлення, щоб повідомлення не було зрозумілим, коли воно було прийняте безвідповідальною людиною.

Криптографія в цілому непроста.

Але існує маса простих криптографічних методів. Криптографічні методи досить безпечні для використання і можуть бути захистом, щоб

уникнути атак. Метод, що використовується для захисту даних у цьому дослідженні, є Vigenere Cipher.

Цей метод є одним із методів заміщення, в якому символ простого тексту буде замінений символами таблиці ASCII шляхом зміщення положення символу на ключ.

У процесі шифрування цей алгоритм використовує спосіб шифрування простого тексту в шифротекст, щоб вихідне повідомлення було закодовано.

Алгоритми шифрування - це функції, які використовуються для виконання шифрування та функції дешифрування.

Дані - це дуже важлива інформація, яку необхідно зберігати в таємниці. Дані можуть бути у вигляді посередньої інформації або інформації, що є дуже важливим, коли інші люди можуть не знати вмісту даних.

Дані є частинами цифрової інформації. Зазвичай воно формується у певний спосіб і можуть бути різними способами, такими як цифри або текст.

Це інформація у двійковому цифровому форматі. Дані - це різновид технологічної інформації. Він ідентифікує інформацію зі свого джерела і розбивається на окрему невелику інформацію

## 2.2 Криптографія

Криптографія - це здатність методів шифрування, коли «вихідний текст» (простий текст) шифрується за допомогою ключа шифрування у «випадковий текст, який важко читати» (шифротекст) тим, хто не має ключа дешифрування.

Дешифрування за допомогою ключа дешифрування може відновити вихідні дані.

Ймовірність отримання оригінального рукопису тим, хто не має ключа розшифровки протягом короткого часу, дуже мала. Техніка шифрування, що використовується в класичній криптографії - це симетричне шифрування, де ключ дешифрування такий самий, як ключ шифрування.

Для криптографії publickey потрібні методи асиметричного шифрування, де ключ дешифрування не той самий, як ключ шифрування. Шифрування, дешифрування та генерація ключів для методів асиметричного шифрування потребують більш інтенсивної mputation, ніж симетричного шифрування, оскільки для асиметричного шифрування використовується величезна кількість.

Алгоритм Vigenere Cipher - це метод кодування тексту алфавіту за допомогою ряду паролів Цезаря на основі літер ключових слів. Пароль Vigenère - це проста форма поліальфабетичного коду заміни. Перевага цього пароля порівняно з Цезаром та іншими моноалфабетичними кодами полягає в тому, що вони не так вразливі до методу розшифровки, який називається аналіз вимог.

Віньєре став рушієм для громадянської війни в Америці, а конфедеративна армія використовувала код віньєре в американській громадянській війні. Беббідж і Касіскі успішно порушили код віньєнера в середині 19 століття [15].

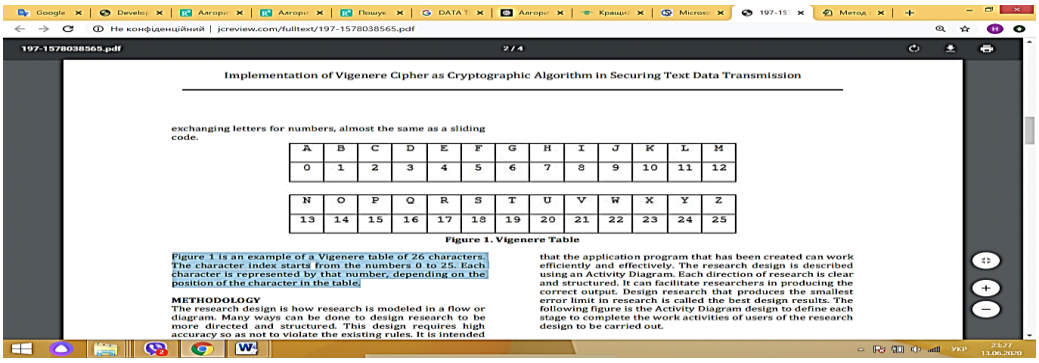
Цей тип алгоритму шифрування дуже добре відомий, оскільки його легко зрозуміти та реалізувати. Техніка отримання шифротексту може бути виконана з використанням підстановки чисел або прямолінійного квадрата.

Техніка підстановки віньєре з використанням чисел виконаний обмін літерами на цифри, майже такий самий, як розсувний код.

В таблиці2.1наведено Vigenere з 26 символів. Індекс символів починається з чисел 0 до 25. Кожен символ представлений цим числом, залежно від положення характера в таблиці.

Таблиця 2.1. Алгоритм Vigenere з 26 символів

*Таблиця 2.1*



Implementation of Vigenere Cipher as Cryptographic Algorithm in Securing Text Data Transmission

exchanging letters for numbers, almost the same as a sliding code.

| A | B | C | D | E | F | G | H | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |

| N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |

**Figure 1. Vigenere Table**

Figure 1 is an example of a Vigenere table of 26 characters. The character index starts from the numbers 0 to 25. Each character is represented by that number, depending on the position of the character in the table.

**METHODOLOGY**

The research design is how research is modeled in a flow or diagram. Many ways can be done to design research to be more directed and structured. This design requires high accuracy so as not to violate the existing rules. It is intended that the application program that has been created can work efficiently and effectively. The research design is described using an Activity Diagram. Each direction of research is clear and structured. It can facilitate researchers in producing the correct output. Design research that produces the smallest error limit in research is called the best design results. The following figure is the Activity Diagram design to define each stage to complete the work activities of users of the research design to be carried out.

Дизайн дослідження полягає в тому, як моделюється дослідження в потоці або діаграмі. Можна зробити багато способів, щоб дизайн був більш орієнтованим та структурованим.

Ця конструкція вимагає високої точності, щоб не порушити існуючі правила. Мається на увазі, що створена прикладна програма може працювати ефективно та ефективно.

Дизайн дослідження описується за допомогою Діаграми діяльності. Кожен напрямок досліджень чіткий і структурований. Це може полегшити дослідників у виробництві правильних виходів.

Дизайнерські дослідження, що створюють найменшу межу помилок у дослідженні, називаються найкращими результатами проектування. На рис.2.1. зображено алгоритм роботи для визначення кожного етапу для завершення робочої діяльності користувачів дослідницької конструкції, яка повинна здійснюватися.

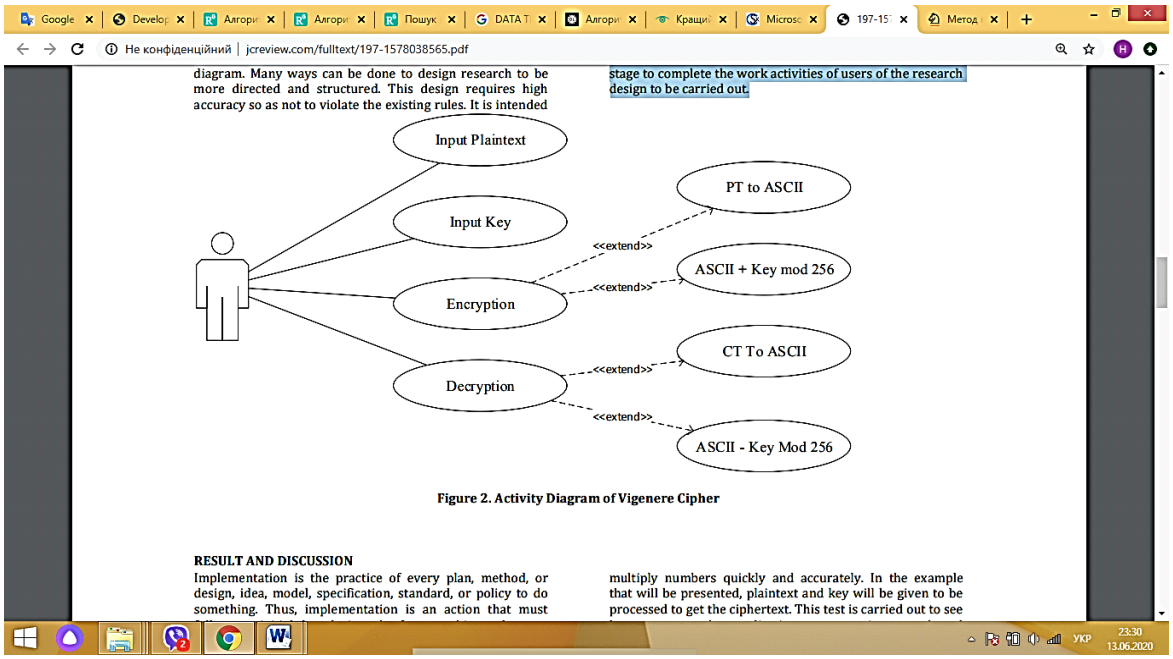



Рис.2.1. Алгоритм роботи Vigenere Cipher [16]

Реалізація - це практика кожного плану, методу чи дизайну, ідеї, моделі, специфікації, стандарту чи політики щось робити. Таким чином, mplementation - це дія, яка повинна слідувати будь-якій початковій думці, щоб щось сталося.

У контексті інформаційних технологій впровадження програмного забезпечення або апаратних засобів включає всі процеси після продажу, які беруть участь у тому, що добре працює в його середовищі, включаючи аналіз вимог, установку необхідної конфігурації, налаштування, виконання, тестування, інтеграція системи, навчання користувачів, доставка та виготовлення.

Розрахункові іспити призначені для оцінки здатності прикладної програми додавати, віднімати, ділити та множувати числа швидко та точно. У прикладі, який буде представлено, для отримання шифротексту буде оброблено відкритий текст та ключ для оброблення.

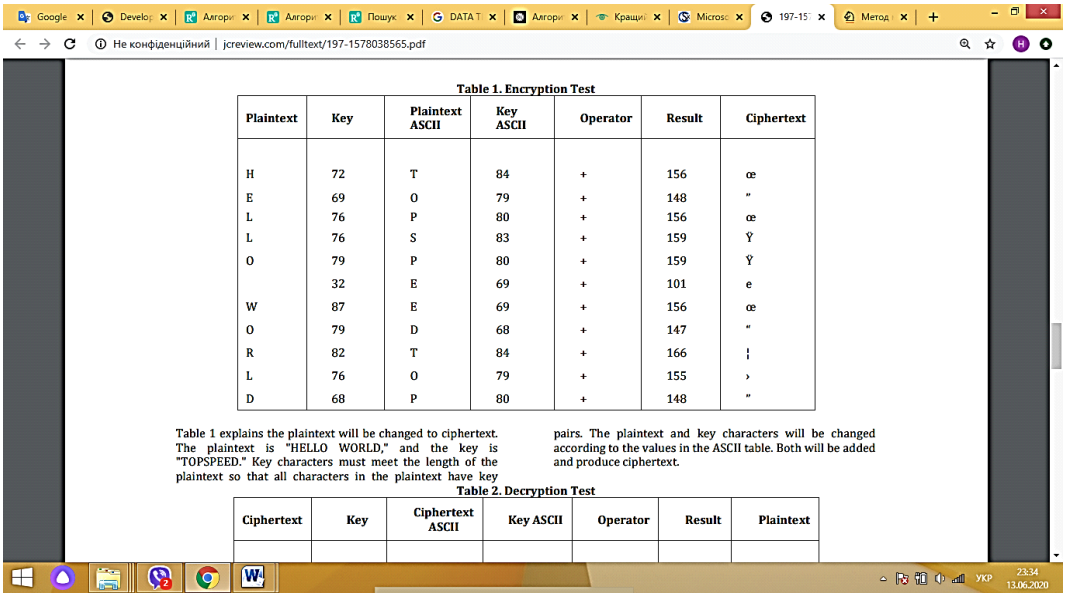
Цей тест проводиться, щоб побачити, наскільки точно створена прикладна програма і чи це обчислення, виконані вручну. Процес складається з двох процесів, таких як процес шифрування та процес дешифрування. Наступний розрахунок - це повне пояснення та розрахунок

процесу шифрування та дешифрування в алгоритмі Vigenere Cipher, надавши два простого тексту та ключі.

У таблиці 2.2 пояснюється, що відкритий текст буде змінено на шифротекст. Простий текст - "правильний світ", а ключ - "топсий". Ключові символи повинні відповідати довжині простого тексту, щоб усі символи в простому тексті мали пари ключів. Простий текст та ключові символи будуть змінені відповідно до значень таблиці ASCII. Обидва будуть додані і створювати шифротекст.

Таблиця 2.2. Зміна відкритого тексту на шифротекст

*Таблиця 2.2*



Не конфіденційний | jcreview.com/fulltext/197-1578038565.pdf

**Table 1. Encryption Test**

| Plaintext | Key | Plaintext ASCII | Key ASCII | Operator | Result | Ciphertext |
|---|---|---|---|---|---|---|
| H | 72 | T | 84 | + | 156 | œ |
| E | 69 | O | 79 | + | 148 | " |
| L | 76 | P | 80 | + | 156 | œ |
| L | 76 | S | 83 | + | 159 | Ÿ |
| O | 79 | P | 80 | + | 159 | Ÿ |
| | 32 | E | 69 | + | 101 | e |
| W | 87 | E | 69 | + | 156 | œ |
| O | 79 | D | 68 | + | 147 | " |
| R | 82 | T | 84 | + | 166 | ¦ |
| L | 76 | O | 79 | + | 155 | › |
| D | 68 | P | 80 | + | 148 | " |

Table 1 explains the plaintext will be changed to ciphertext. The plaintext is "HELLO WORLD," and the key is "TOPSPEED." Key characters must meet the length of the plaintext so that all characters in the plaintext have key pairs. The plaintext and key characters will be changed according to the values in the ASCII table. Both will be added and produce ciphertext.

**Table 2. Decryption Test**

| Ciphertext | Key | Ciphertext ASCII | Key ASCII | Operator | Result | Plaintext |
|---|---|---|---|---|---|---|

Шифрований текст, згенерований у попередній таблиці, буде повернутий таким чином, що він створює простий текст. Таблиця 2.3 є результатом процесу дешифрування з шифротексту, отриманого в таблиці 2.2. Результати не змінилися, щоб розрахунок Vigenere Cipher не зазнавав помилок і збоїв.

Таблиця 2.3. Розшифрований текст за допомогою алгоритму

*Таблиця 2.3*

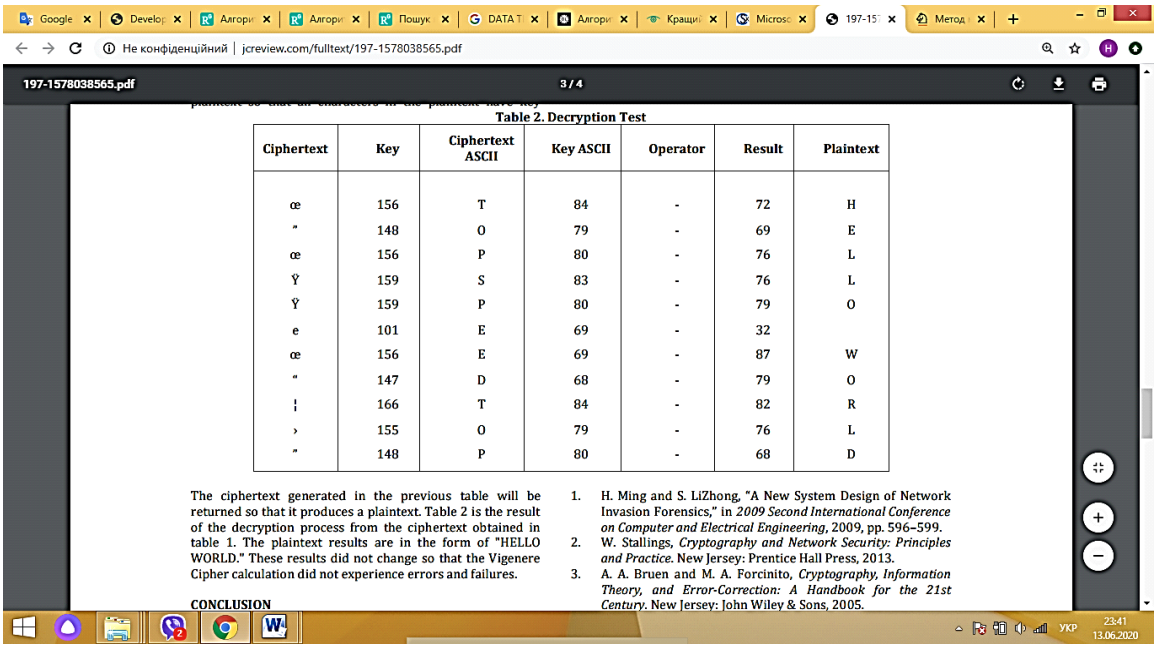



**Table 2. Decryption Test**

| Ciphertext | Key | Ciphertext ASCII | Key ASCII | Operator | Result | Plaintext |
|---|---|---|---|---|---|---|
| œ | 156 | T | 84 | - | 72 | H |
| " | 148 | O | 79 | - | 69 | E |
| œ | 156 | P | 80 | - | 76 | L |
| Ÿ | 159 | S | 83 | - | 76 | L |
| Ÿ | 159 | P | 80 | - | 79 | O |
| e | 101 | E | 69 | - | 32 | |
| œ | 156 | E | 69 | - | 87 | W |
| " | 147 | D | 68 | - | 79 | O |
| ¦ | 166 | T | 84 | - | 82 | R |
| › | 155 | O | 79 | - | 76 | L |
| " | 148 | P | 80 | - | 68 | D |

The ciphertext generated in the previous table will be returned so that it produces a plaintext. Table 2 is the result of the decryption process from the ciphertext obtained in table 1. The plaintext results are in the form of "HELLO WORLD." These results did not change so that the Vigenere Cipher calculation did not experience errors and failures.

**CONCLUSION**


1. H. Ming and S. LiZhong, "A New System Design of Network Invasion Forensics," in *2009 Second International Conference on Computer and Electrical Engineering*, 2009, pp. 596–599.
2. W. Stallings, *Cryptography and Network Security: Principles and Practice*. New Jersey: Prentice Hall Press, 2013.
3. A. A. Bruen and M. A. Forcinito, *Cryptography, Information Theory, and Error-Correction: A Handbook for the 21st Century*. New Jersey: John Wiley & Sons, 2005.


# РОЗДІЛ 3

# ГІБРИДНІ АЛГОРИТМИ БЕЗПЕКИ ДЛЯ ПЕРЕДАЧІ ДАНИХ З ВИКОРИСТАННЯМ AES-DES

## 3.1 Алгоритм DES як стандарт шифрування даних.

Алгоритм призначений для шифрування та розшифровки блоків даних, що складаються з 64 біт під керуванням 64-бітного ключа. У криптографії шифр розширеного стандарту шифрування (AES) має 128-розрядний розмір блоку з розмірами ключів 128, 192 та 256 біт відповідно. Інтеграція AES з DES полягає у підвищенні безпеки для режиму введення тексту, зображення, аудіо та відео.

У роботі викладені можливі слабкі місця в поточному алгоритмі шифрування AES, особливо щодо криптоаналізу на основі алгебрики. Для розуміння необхідності мінімізації алгебраїчних атак на AES там пропонується ідея інтеграції AES до DES. Звідси розробка гібридного алгоритму AES-DES.

Криптоаналіз AES AES заснований на принципі проектування, відомому як мережа перестановки заміни. Це швидко як в програмному, так і в апаратному забезпеченні. На відміну від свого попередника DES, AES не використовує мережу Feistel.

AES має фіксований розмір блоку 128 біт і розмір ключа 128, 192 або 256 біт, тоді як Rijndael можна задавати розмірами блоків і ключів у будь-якому кратному 32 біті, при цьому мінімум 128 біт. Розмір блоку має максимум 256 біт.

AES працює над матрицею байт основного стовпця розміром 4 × 4, що називається станом (версії Rijndael з більшим розміром блоку мають додаткові стовпці у штаті). Більшість обчислень AES проводиться у спеціальному кінцевому полі.

Шифр AES задається у вигляді кількості повторень раундів перетворення, які перетворюють вхідний простий текст у кінцевий вихід тексту шифру. Кожен раунд складається з декількох етапів обробки, включаючи той, який залежить від ключа шифрування. Набір зворотних раундів застосовується для перетворення тексту шифру назад у початковий простий текст, використовуючи той самий ключ шифрування.

Розмір ключа, використаний для шифру AES, визначає кількість повторень раундів перетворення, які перетворюють вхідний звичайний текст у кінцевий вихід, який називають текстом шифру. Кількість циклів повторення така:

- 10 циклів повторення для 128-бітних клавіш;
- 12 циклів повторення для 192-бітних клавіш;
- 14 циклів повторення 256-бітних клавіш.

DES - алгоритм блочного шифрування, який приймає рядок бітів простого тексту з фіксованою довжиною і перетворює його через ряд складних операцій в інший рядок біт-тексту шифру тієї ж довжини. У випадку з DES розмір блоку - 64 біт.

DES також використовує ключ для налаштування перетворення, щоб розшифрування може бути виконано справжнім користувачем, який використовується для шифрування ключа. Ключ нібито складається з 64 біт, проте лише 56 з них фактично використовуються алгоритмом. Вісім біт використовуються виключно для перевірки парності, а потім відкидаються.

Отже, ефективна довжина ключа становить 56 біт, і вона ніколи не котирується як така. Кожен 8-й біт обраного ключа відміняється, тобто позиції 8, 16, 24, 32, 40, 48, 56, 64 видаляються з 64-бітового ключа, залишаючи після себе лише 56-бітний ключ.

F-функція, функціонує на половині блоку (32 біта) одночасно і складається з чотирьох етапів: 1. Розширення - 32-розрядний напівблок розширюється на 48 біт за допомогою перестановки розширення шляхом дублювання половини біт.

Вихід складається з восьми 6-бітових фрагментів (8 * 6 = 48 біт), кожен з яких містить копію 4 відповідних вхідних бітів, плюс копію негайно сусіднього біта з кожного з вхідних фрагментів в будь-яку сторону [17].

Змішування клавіш - результат поєднується з підпунктом за допомогою операції XOR. Шістнадцять 48-бітових підключень по одному

для кожного раунду виводяться з головної клавіші за допомогою розкладу ключів.

Заміна - після змішування в підклаві блок розділяється на вісім 6-бітних шматочків перед обробкою S-скриньками або коробками заміни. Кожен із восьми S-коробок замінює свої шість вхідних бітів чотирма вихідними бітами відповідно до нелінійного перетворення, передбаченого у вигляді таблиці пошуку. Ящики S забезпечують ядро безпеки DES без них, шифр був би лінійним та тривіально зламаним [18].

Перестановка - 32 виходи з S-коробок переставляються відповідно до фіксованої перестановки, P-коробки. Це розроблено так, що після розширення вихідні біти кожного S-box в наступному раунді поширюються на 6 різних S-боків.

DES - 16 рядів Основний процес шифрування 64-бітового блоку даних та 56-бітного ключа за допомогою DES складається з:

- початкової перестановки (IP);
- 16 раундів складного обчислення;
- залежного від клавіш;
- f Кінцевої перестановки, будучи оберненою IP

На рис.3.1 зображено алгоритм роботи DES.

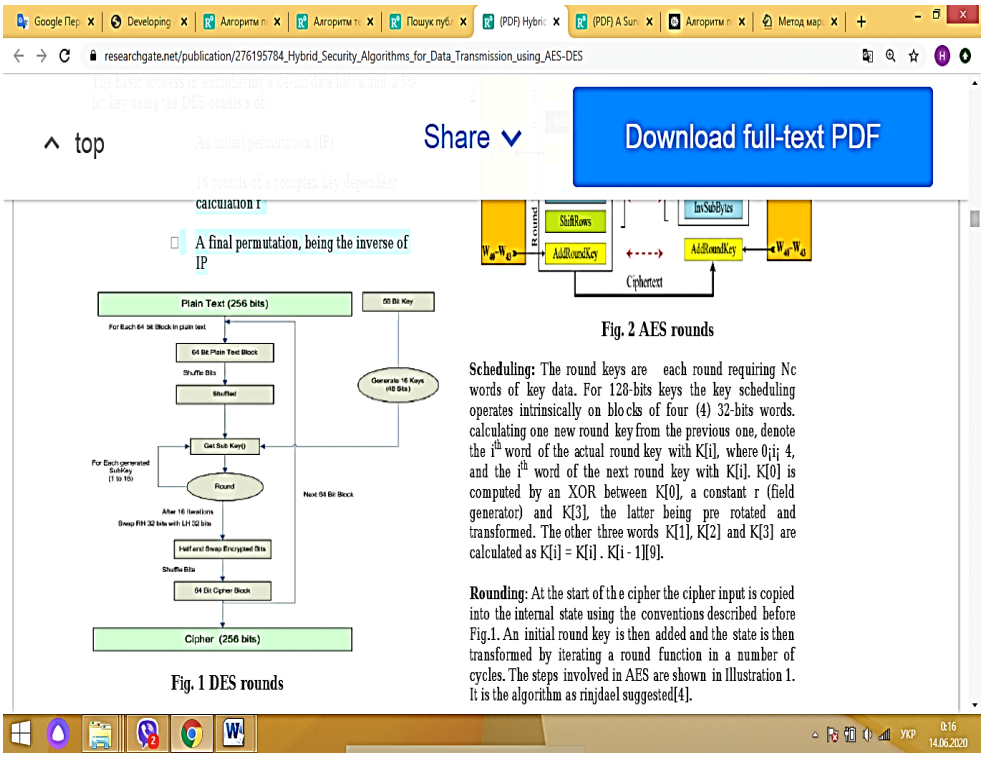



Рис.3.1. Алгоритм DES [19]

AES Байт у Rijndael - це основний блок даних для всіх операцій шифрування. Такі байти інтерпретуються як елементи кінцевого поля за допомогою поліноміального подання, де байт b з бітами b0, b1, ... b7 являє собою кінцеві елементи поля.

Кінцеві польові операції, такі як додавання та множення, необхідні для планування та округлення ключів. Додавання двох елементів кінцевого поля досягається додаванням коефіцієнтів для відповідних потужностей у їхніх многочленних поданнях, це додавання виконується в GF (2), тобто модулі 2, так що 1+ 1 = 0.

Додавання є не що інше, як виконання XOR між двома виразами.

Множення кінцевого поля складніше, ніж складання, і досягається шляхом множення генератора полінових полів, а решта береться за результат. Оскільки існує 256 можливих многочленів, для конкретного генератора поля може бути створена таблиця пошуку. Отже таблиця пошуку містить 256 * 256 записів. На рис.3.2 зображено AES тури

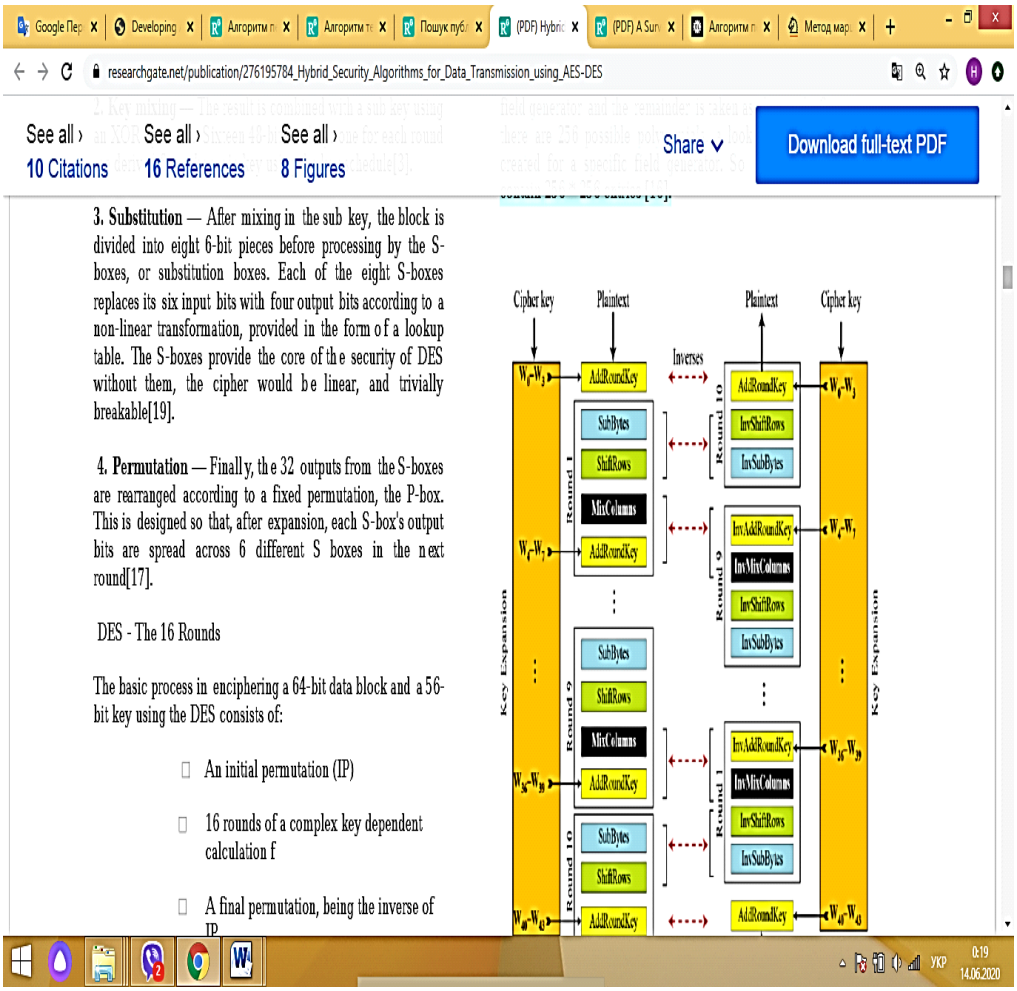



Рис.3.2. AES тури [20]

Круглі клавіші - це кожен раунд, який вимагає Nc слів ключових даних. Для 128-бітових клавіш планування клавіш функціонує по черзі на блоки з чотирьох (4) 32-бітних слів. обчислюючи одну нову круглу клавішу від попередньої, позначте i-то слово фактичної круглої клавіші за допомогою K [i], де 0¡i ¡4, а i-е слово наступної круглої клавіші з K [i]. K [0] обчислюється XOR між K [0], постійною r (генератор поля) і K [3], остання попередньо обертається і трансформується. Інші три слова K [1], K [2] і K [3] обчислюються як K [i] = K [i]. K [i - 1] [9].

На початку шифру вклад шифру копіюється у внутрішній стан, використовуючи умовні позначення, описані на рис. 3.1.

Потім додається початкова кругла клавіша, а потім стан трансформується шляхом повторення круглої функції за кілька циклів. Етапи, що беруть участь в AES, показані на ілюстрації 1. Це алгоритм, як запропонував rinjdael.

Кругла функція: один круг називається циклом, і кожен цикл має чотири кроки. Кожен етап перетворення описаний нижче. Додати круглий ключ: Це перший крок трансформації.

Функція круглого ключа Xor, оголошена на Ілюстрації 1, повинна виконувати бітовий Xor матриці стану та матриці круглих ключів. Суббайти: Трансформація: Тут знайдена зворотна матриця стану і виконується тонке перетворення .

Зсувні рядки: Перетворення рядків зсуву працює окремо на кожному з останніх трьох рядків матриці стану шляхом циклічного зміщення байтів у рядку. Другий ряд зсувається час виконання вліво, третій ряд зміщується два рази, а четвертий ряд зміщується тричі.

Стовпчики змішування: Перетворення стовпців міксів обчислює нову матрицю стану S шляхом множення лівого матриця S на поліноміальну матрицю [21].

Розширений стандарт шифрування (AES) - це блок-шифр, який, як стандарт, ратифікований Національним інститутом стандартів і технологій США (NIST), був обраний за допомогою процесу, помітно більш відкритого та прозорого, ніж його попередник, старіння шифрування даних.

Стандартний (DES). Цей процес виграв похвалу від відкритої криптографічної спільноти та допоміг підвищити впевненість у безпеці алгоритму виграшу з боку тих, хто підозріло ставився до попередника DES [22].

### 3.2. Гібридний AES -DES

У запропонованому алгоритмі (Hybrid AES-DES) мета була досягнута шляхом комбінування двох алгоритмів, званих DES та AES.

Для шифрування даних необхідно:

- вхід розглядається як текст, зображення (.jpeg), аудіо (8-бітний файл .wav низького рівня) або відео (.avi) перетворюється на 128-бітний звичайний текст;

- 128-бітний текст поділяється на два набори 64-бітових простих текстових даних;

- 64-бітний звичайний текст подається як вхід до алгоритму DES, який шифрується для надання зашифрованого 64-бітного тексту;

- два набори зашифрованих 64-бітових текстів потім об'єднуються як єдині 128-бітні зашифровані дані, які надалі застосовуються до алгоритму AES для подальшого шифрування [23].

Для дешифрування даних:

- 128-бітові зашифровані дані застосовуються до алгоритму AES, які забезпечують розшифрований набір 128 бітних даних;
- один набір 128-бітних даних далі розділяється на два 64-бітові набори даних;
- набори даних потім додатково застосовуються до алгоритму DES для отримання двох розшифрованих наборів 64 біт;
- два набори 64-розрядних розшифрованих даних об'єднуються в єдині 128-бітні дані.

# ВИСНОВКИ

Отже, вивчивши різні режими передачі, можна зробити висновок, що під час вибору режиму передачі даних потрібно враховувати деякі моменти:

- швидкість передачі;
- відстань, яку проходить інформація.
- вартість та простота встановлення;
- стійкість екологічних умов.

З результатів дослідження можна зробити три висновки.

Vigenere Cipher працює шляхом зміщення персонажів.

У Vigenere Cipher є ключ, який можна визначити запис до потрібної кількості клавіш.

Vigenere Cipher повинен використовувати модуль, щоб зашифрований символ не перевищував обмеження символів у таблиці ASCII.

Алгоритм AES з DES призначений для шифрування та розшифровки блоків даних, що складаються з 64 біт під керуванням 64-бітного ключа. У криптографії шифр розширеного стандарту шифрування (AES) має 128-розрядний розмір блоку з розмірами ключів 128, 192 та 256 біт відповідно. Інтеграція AES з DES полягає у підвищенні безпеки для режиму введення тексту, зображення, аудіо та відео.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ


1. Shang W. Development and Trend Analysis of Computer Network Security in China[J].Electronic Technology and Software Engineering,2016,(1):196-197.

2 Kang L Z. Current Situation and Development Trend of Network Security Technology[J].Network Security Technology and Application,2015,(4):176-179.

3 Xin Y. Research and Implementation of Bus IC Card Asymmetric Key Management System[D]. Beijing Jiaotong University,2016.

4 Xin L.Research on Technical Architecture of Big Data Security and Privacy Protection[J].Research on Information Security,2016,2(3):244-250.

5 Pan Y.Research on Data Encryption Scheme Algorithms Supporting Homomorphic Arithmetic Operations[J].Journal of Communications,2015,36(1):167-178.

6. H. Ming and S. LiZhong, "A New System Design of Network Invasion Forensics," in 2009 Second International Conference on Computer and Electrical Engineering, 2009, pp. 596–599.

7. W. Stallings, Cryptography and Network Security: Principles and Practice. New Jersey: Prentice Hall Press, 2013.

8. A. A. Bruen and M. A. Forcinito, Cryptography, Information Theory, and Error-Correction: A Handbook for the 21 Century. New Jersey: John Wiley & Sons, 2005.

9. F. H. Khan, R. Shams, F. Qazi, and D.-E.-S. Agha, "Hill Cipher Key Generation Algorithm by using Orthogonal Matrix," Int. J. Innov. Sci. Mod. Eng., vol. 3, no. 3, pp. 5–7, 2015.

10. M. den Hengst and M. Warnier, “Cyber Crime in Privately Held Information Systems: Personal Data at Stake,” in 2013 \European Intelligence and Security Informatics Conference, 2013, pp. 117–120.

11. Iswanto, “Avoiding local minima for path planning quadrotor based on modified potential field,” Int. Rev. Aerosp. Eng., vol. 11, no. 4, pp. 146–154, Aug. 2018.

12. Iswanto, O. Wahyunggoro, and A. I. Cahyadi, “3D object modeling using data fusion from laser sensor on quadrotor,” in AIP Conference Proceedings, 2016

13. W. Stallings, “Cryptography and Network Security Principles and Practices,” 4th Editio., 2010

14. Y. Kumar, R. Munjal, and H. Sharma, “Comparison of Symmetric and Asymmetric Cryptography with Existing Vulnerabilities and Countermeasures,” Comput. Sci. Manag. Stud., vol. 11, no. 3, pp. 60–63, 2011.

15. A. Hidayat, “Algoritma Kriptografi Vigenere Cipher,” 2012. .

16. Dony Ariyus, Pengantar Ilmu Kriptografi. Yogyakarta: Andi Offset, 2008.

17. Carlos Cid, Sean Murphy and Matthew Robshaw “Computational and Algebraic aspects of the Advanced Encryption Standard”, In Proceedings of the Seventh International Workshop on Computer Algebra in Scientific Computing . 2004

18 Aida Janadi “ AES immunity Enhancement against algebraic attacks by using dynamic S-Boxes”, Information and Communication Technologies from Theory to Applications, ICTTA 2008.

19 Jing Wang & Guo-ping Jiang, “Improved DES algorith based on irrational number”, IEEE Int. Conference Neural Networks & Signal Processing. 2008.

20 M.B. Vishnu & S.K. Tiong, “Security Enhancement of Digital Motion Image Transmission Using Hybrid AES-DES Algorithm”, IEEE Int. 2008.

21 Tingyuan Nie, “A Study of DES and Blowfish Encryption Algorithm”. 2009.

22 Yuan Kun, Zhang HanLi Zhaohui, “An Improved AES algorithm based on chaos”, Multimedia Information Networking and Security, INES '09. International Conference. 2009

23 Tingyuan Nie, Chuanwang Song, Xulong Zhi, “ Performance Evaluation of DES and Blowfish Algorithms”, Biomedical Engineering and Computer Science International Conference. 2010