Plag

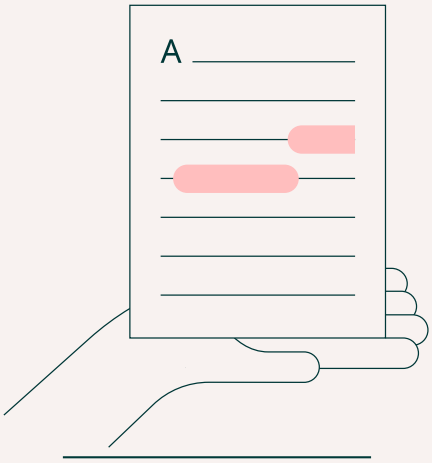
Звіт про оригінальність

● Оцінка схожості 29%

● Ризик плагіату НАЙВИЩИЙ

Ігор Кагало 2025-06-05 22:44

Посилання на звіт: ZSn4 / Посилання користувача: qfC8



Ось вона –
Ваша звіт про оригінальність!

Ми раді повідомити, що перевірка вашого документа завершена, і результати вже готові! Наші алгоритми старанно працювали, щоб знайти збіги в наших базах даних.

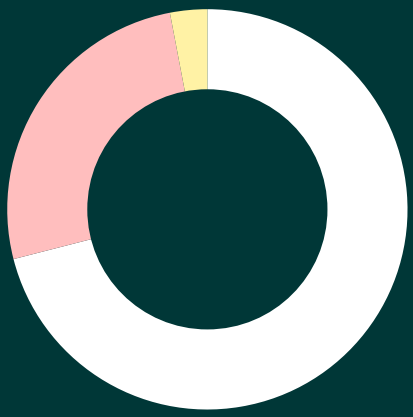
На наступних сторінках ви знайдете результати перевірки:

Бали
Збіги
Посилання

Ваш документ було перевірено за такими джерелами:

- ✓ База даних інтернет-джерел
- ✓ База даних наукових статей
- ○ Глибока перевірка (наш вдосконалений алгоритм)

Бали



Збіги тексту	26%
Перефразування	3%
Цитований текст	0%
Неправильне цитування	0%
Збігів не знайдено	71%

Ризик плагіату НАЙВИЩИЙ

Ризик плагіату вказує, як збіги тексту розподілені по документу. Вищий ризик виникає, коли збіги з'являються близько один до одного, наприклад, у тому самому абзаці або розділі.

Оцінка схожості 29%

Оцінка схожості показує, скільки слів або символів у вашому документі збігаються з текстами інших документів, включаючи перефразовані тексти або неправильні цитати.

Збіги

1 АНАЛІЗ СУЧАСНИХ КРИПТОГРАФІЧНИХ СИСТЕМ ЗАХИСТУ ДАНИХ

1.1 Основні означення і поняття

13 Повідомлення називається відкритим текстом. Зміна вигляду 13 повідомлення так, щоб заховати 13 його суть, називається кодуванням 13 або 13 шифруванням. Шифроване повідомлення називається шифротекстом. Процес перетворення шифротексту у відкритий текст називається розшифруванням.

13 Шифруванням повідомлень займається криптографія [1, 2].

Галузь, що охоплює криптографію і криптоаналіз, називається криптологією. а люди, які нею займаються, називаються криптологами.

Криптосистема це – сукупність алгоритмів, відкритих текстів, шифротекстів і ключів.

13 Відправник хоче послати своє повідомлення і бути впевненим, що це повідомлення не зможе ніхто перехопити і прочитати. Для цього текст повідомлення необхідно зашифрувати. Одержувач повинен розшифрувати одержану інформацію. Схему шифрування і розшифрування інформації показано на рис. 1.1.

Рисунок 1.1 Схема шифрування і дешифрування повідомлень

Позначимо відкритий текст через M. 17 Це може бути потік бітів, текстовий файл або бітове зображення. Для комп’ютера текст M - це двійкові дані. Відкритий текст створений для зберігання або передачі. Відкритий текст M - це повідомлення, яке повинне бути зашифроване .

Позначимо шифротекст через C. Це теж двійкові дані, іноді того ж розміру, що і M, іноді більші. Якщо шифрування супроводжується стисненням, то C є менше за M. Але шифрування може бути і розширенням вхідного тексту. Позначимо через E функцію кодування вхідного відкритого тексту, а через D - функцію декодування закодованого тексту Функція кодування E діє на M, створюючи C. 17 Математично це можна записати так [3]:

E(M) = C (1.1)

У зворотному процесі функція декодування D діє на C, відновлюючи M:
17 Математично це можна записати так:

17 D(C)= M (1.2)

Оскільки метою кодування і декодування повідомлення є відновлення початкового відкритого тексту, то повинна виконуватися рівність:

D(E(M))= M (1.3)

Окрім забезпечення конфіденційності, криптографія часто використовується для інших функцій :

1.2 Шифрування і розшифрування повідомлень

Криптографічний алгоритм або шифр є математичною функцією, 3 яка використовується для 3 шифрування і дешифрування повідомлення. 3 Ці функції пов'язані між собою: 3 одна для шифрування, а інша - для дешифрування.

3 Якщо безпека 21 алгоритму ґрунтується тільки 3 на збереженні самого алгоритму в таємниці, то це обмежений алгоритм. Обмежені алгоритми представляють тільки історичну 3 цінність. Велика 3 група користувачів не може використовувати обмежені 21 алгоритми, оскільки коли користувач покидає 3 групу, її члени повинні переходити на інший алгоритм. Алгоритм повинен бути замінений і в тому випадку, якщо 3 хто-небудь розкриє секрет алгоритму.

3 Обмежені алгоритми не допускають якісного контролю або стандартизації. У кожній групі користувачів є 3 свій унікальний алгоритм. Такі групи не можуть використовувати відкриті програмні продукти - зловмисник може купити такий же продукт і розшифрувати алгоритм. Обмежені алгоритми популярні тільки для повідомлень 3 з низьким рівнем безпеки.

3 Для підвищення безпеки шифрування інформації з метою її захисту 3 сучасна криптографія вирішує ці проблеми за допомогою ключа. Такий ключ може бути будь-яким значенням, що належить великій множині, яка називається простором можливих ключів [4].

1.3 Класифікація систем шифрування і розшифрування повідомлень

Процеси 3 шифрування і дешифрування залежать від ключа. Позначимо процес за шифрування відкритого повідомлення на ключі K через EK(M). Тоді функції (1.1) і (1.2) матимуть вигляд [2]:

C = EK(M). (1.4)

Процес розшифрування, тобто відкритого повідомлення із криптограми C за допомогою відомого ключа K, математично можна записати так:

M= DK(C). (1.5)

Функція DK повинна бути обернена до функції EK , тобто

,

В тому розумінні, що при правильному ключі K дозволяє отримати відкритий текст M, тобто виконується наступна рівність:

DK(EK(M))=M. (1.6)

Схему шифрування і розшифрування інформації з однаковими ключами показано на рис. 1.2.

Рисунок 1.2 Схема роботи криптосистеми з однаковими ключами

На передавальній стороні виконується шифрування відкритого повідомлення за допомогою функції шифрування EK(M) на ключі то одержуємо криптограму C, яку передають відкритим каналом зв'язку. На приймальній стороні, отримавши зашифроване повідомлення C , застосовується до нього обернене перетворення DK(C) і одержуємо відкрите повідомлення при використанні того самого ключа , як і на передавальній стороні. Розшифрування буде вірним, якщо повідомлення не було змінено під час передачі по каналу зв'язку.

Для успішної роботи учасники інформаційного секретного обміну повинні заздалегідь домовитися про алгоритм шифрування і розшифрування. Важливим фактором також є передача криптографічного ключа.

Оскільки ключ, що використовується для шифрування і дешифрування повідомлень однаковий, то такі системи називаються симетричними. Симетричні системи використовують переважно для перетворення тексту, які є комбінацією перестановок і замін.

Симетричні алгоритми характеризуються можливістю швидкого шифрування великих потоків інформації в каналах зв'язку. Вони забезпечують високу ступінь секретності. Симетричні алгоритми дають змогу використовувати одні і ті ж апаратні засоби для шифрування і дешифрування інформації [4].

Проблемою використання симетричних алгоритмів є зберігання і передача ключа розшифрування повідомлення, так як 5 він є секретною частиною криптосистеми як на передавальній так і на приймальній сторонах. Також потрібна велика кількість ключів для кожного одержувача секретного повідомлення.

Для усунення недоліків симетричних систем були винайдені несиметричні (асиметричні) способи шифрування повідомлень. В асиметричних алгоритмах при шифруванні і дешифруванні використовуються різні ключі. Ключ 5 для зашифрування інформації позначимо через K1. Це відкритий ключ, який не потрібно приховувати. Цей ключ суттєво відрізняється від відповідного ключа дешифрування K2. Тому одержимо наступні залежності:

EK1(M)=C (1.7)

DK2(C)=M (1.8)

Функція (1.3) матиме вигляд:

DK2(EK1(M))=M (1.9)

Безпека цих алгоритмів повністю ґрунтується на ключах. Це означає, що алгоритм може бути опублікований і проаналізований. Якщо конкретний ключ не відомий, то ніхто не зможе прочитати секретні повідомлення. Процес інформаційного обміну зображено схематично. Схему шифрування і розшифрування інформації з двома різними ключами показано на рис. 1.3 [2].

Рисунок 1.3 12 Схема роботи асиметричної системи 12 з двома різними ключами

12 Асиметричні 12 криптосистеми розв’язали основну проблему симетричних криптосистем, пов’язану з розповсюдженням ключів. Адже для за шифрування 12 інформації багатьма користувачами потрібно мати лише одну пару ключів: відкритий для шифрування і секретний для розшифрування. Перевагою асиметричної системи є те, що для створення багатокористувацької системи обміну зашифрованою інформацією не потрібно великої кількості ключів [3, 5].

Разом з тим асиметричні алгоритми набагато повільніші, ніж симетричні [3]. Повільність пояснюється складністю математичних 12 перетворень, на базі яких ґрунтуються асиметричні 12 криптосистеми , 12 та великої довжини ключів. Такі задачі виконуються досить повільно на комп’ютері.

Але асиметричний метод шифрування можна використати там, де симетричні алгоритми працювати не будуть, наприклад, для створення електронного цифрового підпису.

Симетричні алгоритми завдяки високій надійності та швидкодії найкраще підходять для захисту комп'ютерної інформації в комп'ютерах. Асиметричні алгоритми внаслідок малої швидкодії недоцільно застосовувати для шифрування великих потоків інформації. Асиметричну систему можна використати для обміну інформацією малих об'ємів, тобто можна нею шифрувати ключі для симетричних систем.

1.4 Змішані криптосистеми

Зараз широко використовуються комбіновані системи шифрування інформації. В комбінованих криптосистемах інформація шифрується за допомогою симетричних алгоритмів, а для передачі криптографічних симетричних ключів використовуються асиметричні алгоритми [2, 3].

1 Алгоритми з відкритими ключами не замінюють симетричні алгоритми і використовуються не для шифрування повідомлень, а для шифрування ключів:

1 Алгоритми з відкритими ключами працюють поволі. Але вимоги до об'єму 1 переданої інформації 1 також зростають, і завжди потрібно 1 буде 1 шифрувати дані швидше, ніж це зможе зробити криптографія з відкритими ключами.

1 Криптосистеми з відкритими ключами уразливі по відношенню до розкриття з вибраним відкритим текстом. Якщо C 1 = E(P), 1 де P - відкритий текст з n можливих відкритих текстів, то криптоаналітику потрібно тільки зашифрувати всі n можливих відкритих 1 текстів і 1 порівняти результати з C. Він не зможе розкрити ключ дешифрування, але він зможе визначити відкритий 1 текст.

1 Криптографія з відкритими ключами використовується для шифрування сеансових ключів, які використовуються симетричними алгоритмами для закриття потоку повідомлень.

1 Використання криптографії з відкритими ключами для розподілу ключів вирішує дуже важливу проблему розподіли 1 ключів. За допомогою приведеного 1 протоколу при необхідності зашифрувати повідомлення створюється сеансовий ключ, який знищується після закінчення сеансу зв'язку [2].

Схему роботи комбінованої системи зображено на рис. 1.4.

Рисунок 1.4 Функціональна 10 схема роботи комбінованої криптосистеми

10 Передавальна сторона вибирає відкритий ключ асиметричної системи приймальної сторони і зашифровує ним 10 ключ симетричного алгоритму. Зашифрований ключ 10 симетричного алгоритму 10 відправляється відкритим каналом зв'язку на приймальну сторону, де розшифровується секретним 10 ключем.

Відповідно, ключ шифрування 10 відомий обом учасникам обміну 10 інформацією.

10 Комбіновані криптосистеми внаслідок своєї універсальності в даний час дуже популярні.

1.5 Основи криптоаналізу

Сенс криптографії - в збереженні відкритого тексту (або ключа, або і того, і іншого) в таємниці. Передбачається, що зловмисники повністю контролюють лінії зв'язку між відправником і одержувачем .

Криптоаналіз - це наука одержання відкритого тексту без ключа. Успішно проведений криптоаналіз може розкрити відкритий текст або ключ, 3 виявити слабкі місця в криптосистемах. Метою криптоаналізу є розкриття таємниці повідомлення, яке передається.

Основну суть криптоаналізу, вперше сформулював Датчман А. Керкхофсон [4].

Є чотири основні типи криптоаналітичного розкриття. 3 Для кожного з них, звичайно, припускається, що криптоаналітик знає використовуваний алгоритм шифрування:

а) Розкриття 3 з використанням тільки шифротексту. У криптоаналітика є шифротексти 3 кількох 3 повідомлень, зашифрованих одним і тим алгоритмом 3 шифрування . 3 Завдання криптоаналітика полягає в розкритті відкритого тексту як 3 можна 3 більшого числа повідомлень, для того щоб дешифрувати інші повідомленння, зашифровані 2 тими ж ключами.

2 б) Розкриття з використанням відкритого тексту. У криптоаналітика є доступ не тільки до шифротекстів 2 декількох повідомлень, але і 2 до відкритого тексту цих повідомлень. Його завдання полягає в одержанні ключа (або ключів), використаного для шифрування інших повідомлень, зашифрованих тим ключем (ключами). Нехай відомо

P1, C1=Ek(P1),

P2, C2=Ek(P2)

. . .

Pi, Ci =Ek(Pi)

Треба одержати алгоритм для знаходження

Pi+1 з Ci+1=Ek(Pi+1).

в) 2 Розкриття з використанням вибраного відкритого тексту. У криптоаналітика не тільки є доступ до шифротекстів і відкритих текстів декількох повідомлень, але і 2 можливість вибирати відкритий текст для шифрування. Тут 3 криптоаналітик може вибирати шифровані 14 блоки відкритого тексту, які 14 дають 14 більше інформації про ключ шифрування 2 повідомлень. Його завдання полягає в отриманні ключа (або ключів), який 2 використовувався 2 для шифрування повідомлень, або алгоритму, що дозволяє дешифрувати нові повідомлення, зашифровані тим самим ключем (або ключами).

Нехай відомі співвідношення (1.10). Необхідно знайти

P1, P2, . . . Pi,

3 або алгоритм, як отримувати Pi+1 3 з Ci+1=Ek(Pi+1).

2 г) 2 Адаптивне розкриття з використанням відкритого тексту. Це окремий випадок розкриття з використанням вибраного 14 відкритого тексту. Криптоаналітик 2 не тільки може вибирати шифрований текст, але також може будувати свій подальший вибір на базі отриманих результатів шифрування. При розкритті 14 з використанням вибраного 2 відкритого тексту криптоаналітик міг вибрати для шифрування тільки один великий блок відкритого тексту, при адаптивному розкритті 14 з використанням вибраного 2 відкритого тексту він може вибрати менший блок відкритого тексту, потім вибрати наступний блок, використовуючи результати першого вибору і т. 2 д.

2 д). 2 Розкриття з використанням вибраного 14 шифротексту. Криптоаналітик 2 може вибрати різні шифротексти 2 для дешифрування. Він 2 має доступ до відкритих текстів, що дешифруються. Наприклад, , 3 у криптоаналітика є доступ до "чорного ящика", який виконує автоматичне дешифрування. 2 Його завдання полягає в отриманні ключа.Такий тип розкриття звичайно застосовується 3 до алгоритмів з відк 3 ритим ключем. Розкриття 3 з використання вибраного шифротексту є досить ефективним. 2 Розкриття з використанням вибраного 14 шифротекста часто 3 називають розкриттям з використанням вибраного тексту.

е) Розкритт з використанням вибраного 2 ключа Такий тип розкриття означає, що криптоаналитик може вибирати ключ, якщо у нього є деяка інформація про зв'язок між різними ключами.

2 1.6 Криптографічні протоколи

Сенс криптографії - у вирішенні проблем секретності, перевірки достовірності і цілісності переданої інформації.

1 Протокол - це порядок дій, двох або більше сторін, який 1 призначений для

вирішення завдання [3]. Порядок дій означає, що протокол виконується в певній послідовності, з початку до кінця двох або більше сторін. Це означає, що для реалізації протоколу потрібно принаймні дві людини, одна людина не зможе реалізувати протокол. Оскільки протокол призначений для вирішення певного завдання, то він повинен приводити до кінцевого результату. Характеристики протоколів:

Кожен учасник протоколу повинен знати протокол і послідовність його дій;

Кожен учасник протоколу повинен погодитися виконувати протокол .

Протокол повинен бути несуперечливим, кожна дія повинна бути однозначною, щоб не було нерозуміння між учасниками протоколу;

Протокол повинен бути повним, кожній можливій ситуації повинно відповідати одне значення;

Криптографічний протокол - це протокол, що використовує криптографію. Він протокол включає криптографічний алгоритм. Учасники протоколу можуть захотіти поділитися секретом один з одним, спільно генерувати випадкову послідовність, підтвердити один одному свою справжність або підписати контракт в один і той же момент часу. Загальне правило протоколу полягає в неможливості робити або дізнатися більше, ніж це визначено протоколом.

У повсякденному житті існують неформальні протоколи: замовлення товарів по телефону, голосування на виборах. Всі знають, як ними користуватися і вони працюють добре .Сьогодні все більше і більше людей спілкуються не особисто, а з використанням комп’ютерних мереж. Чесність і безпека протоколів людського спілкування ґрунтується на особистій присутності.

Для демонстрації роботи протоколу беруть участь двоє: відправник і одержувач. Вони беруть участь у всіх двосторонніх протоколах. Як правило, відправник посилає всі протоколи, а одержувач одержує.

2 ОПИС АЛГОРИТМІВ ШИФРУВАННЯ ТА ДЕШИФРУВАННЯ ТЕКСТІВ В РЕЖИМІ ЕЛЕКТРОННОЇ ШИФРУВАЛЬНОЇ КНИГИ

2.1 Основні поняття та характерні особливості побудови блокових шифрів

\

Суть блокових шифрів ґрунтується на поділі послідовності символів відкритого тексту на блоки фіксованої довжини. Ці блоки шифруються окремо. При цьому

однаковим блокам відкритого тексту відповідають однакові шифртексти. На сьогодні блокові шифри є найбільш поширеними. 15 До блочних шифрів належать вітчизняний та американський стандарти шифрування.

6 Блоковий шифр – це система підстановки в алфавіті блоків. Залежно від режиму блочного шифру підстановка може бути як одноалфавітною, так і багатоалфавітною,.

В блокових шифрах 9 одиницею кодування є блок з декількох байтів. Результат кодування залежить від усіх вихідних байтів цього блоку. Схема 19 застосовується при пакетній передачі інформації та кодування файлів. Блокові шифри відносяться до симетричних алгоритмів шифрування. Блоковіі шифри характеризуються наступними властивостями:

Висока 8 криптостійкість;

8 Простота 6 процедур шифрування і розшифрування текстів;

Висока надійність.

Криптостійкість шифру залежить від часу, необхідного 8 для розкриття шифру при використанні найкращого методу криптоаналізу. Надійність залежить від складності алгоритму розкриття шифру. Перетворення шифру повинно 8 використовувати наступні принципи (по К.Шеннону):

6 Розсіювання (diffusion) – тобто, зміна будь-якого знаку відкритого тексту або ключа впливає на велике 6 число знаків шифротексту, 6 що приховує статистичні властивості відкритого тексту;

6 Перемішування (confusion) – використання перетворень, що утрудняють отримання статистичних залежностей між шифротектстом і відкритим текстом.

6 Практично 8 всі сучасні Блочні шифри є композиційними – тобто, складаються з композиції простих перетворень. Саме по собі перетворення може і не забезпечувати потрібних властивостей, але їх ланцюжок дозволяє отримати необхідний результат.

Найбільшу популярність мають такі шифри: “шифр Фейстеля (Feistel)” до якого входять петлі Фейстеля та мережі Файстеля. При такому шифруванні вхідний блок для кожного перетворення розбивається на дві половини. Спочатку перетворюється тільки ліва половина, а права залишається незмінною. Потім обидві половини міняються місцями. Це перетворення здійснюється декілька разів і результатом шифру є кінцеві частини тексту.

В ролі функції шифрування використовується певна комбінація перестановок, підстановок, зсувів, додавань ключа і інших перетворень. При використанні

підстановок інформація проходить через спеціальні блоки, які називають S-блоками, в яких значення групи бітів замінюється на інші значення. За таким принципом побудовано багато алгоритмів для захисту інформації.

Ідея побудови криптографічної стійкої системи шифрування 20 шляхом послідовного застосування відносно простих криптографічних перетворень до різних частин відкритого тексту належить Шеннону.

На базі блокових шифрів ґрунтуються практично всі криптосистеми. Створення послідовності байт, 18 зашифрованих блоковими алгоритмами, дозволяє 18 шифрувати 18 пакети інформації необмеженої довжини. Характерна особливість блокових алгоритмів полягає в перетворенні 18 блоку вхідної інформації фіксованої довжини в блоки такої ж довжини.

Для утворення блочного шифру послідовність символів повідомлення ділиться на блоки , кожний з яких має однакову довжину . Кожний блок перетворюється в блок шифротексту довжини . Правило 4 перетворення залежить від ключа шифрування . Структурну схему блокового шифрування повідомлення шляхом поділу його на блоки показано на рис. 2.1.

Рисунок 2.1 – Структурна схема блокового шифрування

На багатократному використанні до блоків відкритого тексту конкретних базових перетворень ґрунтуються блокові шифри. Базові перетворення повинні легко реалізуватися програмно і бути аналітично складними перетвореннями.

Основними принципами 4 перетворення повідомлень, що забезпечують високу стійкість блокового шифрування є:

Перестановка символів (нелінійне перетворення підстановкою);

Перемішування символів;

Багаторазове повторення попередніх пунктів.

Перестановка ускладнює встановлення взаємозв’язку між відкритим текстом і шифротекстом. Перетворення окремих частин блоку поширює вплив одного знаку відкритого тексту на велике число знаків шифротексту.

Алгоритм шифрування складається з декількох ітерацій. Кожна ітерація використовує перестановки та перемішування символів. При реалізації кожного циклу шифрування використовуються однотипні операції. Розшифровується шифротекст з використанням цих операцій в оберненому порядку.

При поділі повідомлення на короткі блоки, 4 в криптограмі збережеться статистика повідомлення. Опонент 4 може використати її 4 для ефективного криптоаналізу. Для блоків великої довжини досить складно задати загальне нелінійне перетворення. Тому 4 необхідно вибирати 4 підблоки помірної довжини . До кожного підблоку застосовуються нелінійні перетворення і потім виконуються перестановки символів підблоків у межах усього блоку . Схему побудови блокових шифрів показано на рис. 2.2.

Рисунок 2.2 – Схема побудови блокових шифрів

Перетворення блоків залежать від 4 секретного ключа, що використовуються для шифрування. Ключ при кожній новій ітерації замінюється 4 на новий, отриманий перетворенням початкового ключа.

2.2 Класифікація блокових шифрів

Всі криптоалгоритми по використанню кількості ключів 9 поділяються на симетричні та асиметричні. За принципом їх конструювання симетричні алгоритми є потокові та блочні.

По вигляду криптографічного перетворення симетричні криптосистеми діляться на шифри заміни, перестановки та композиційні шифри.

На рисунку 2.3 зображена структурна схема загальної класифікації криптоалгоритмів.

Рисунок 2.3 – Структурна схема класифікації криптоалгоритмів

При використанні шифру заміни кожний 22 елемент початкового тексту взаємно-однозначно 7 замінюється одним, або декількома знаками деякого алфавіту. Шифр простої заміни замінює кожний знак вхідного алфавіту на деякий знак з того ж алфавіту, Результат 22 заміни 9 не залежить від розташування 22 знаку 6 у відкритому тексті. Ключами 7 для шифрів заміни є таблиці.

7 Шифри перестановки відрізняються від шифрів заміни тим, що при зашифруванні буква відкритого тексту переходить не у фіксований знак алфавіту, а в іншу букву того ж відкритого тексту, внаслідок чого букви розташовуються на нових місцях, тобто переставляються. Ключем 16 для даного шифру також служить таблиця заміни, тільки не букв алфавіту, а їх 16 індексів в 16 тексті, який підлягає шифруванню.

Блокові шифри діляться на:

зчеплення блоків;

зворотній зв’язок за шифром;

зворотній зв’язок за виходом;

електронної шифрувальної книги.

2.3 Блокове шифрування в режимі електронної шифрувальної книги

Головною перевагою алгоритму шифрування в режимі електронної шиірувальної книга є його проста реалізація. Недоліком алгоритму є його мала стійкість до розкриття. Це пов’язано з тим, що за відносно невеликої довжини блоків та великої довжини тексту, який необхідно передати, певні блоки можуть повторюватись. Це надає криптоаналітикові певну інформацію щодо змісту повідомлення.

При формуванні криптограми з повідомлення алгоритм шифрування можна використовувати у різних модифікаціях. Режим блокового шифрування, який при незмінному ключі шифрування , однакові блоки повідомлення перетворює в однакові блоки криптограм, називається шифруванням режимі електронної шифрувальної книги.

Для роботи з електронною кодовою книгою повідомлення розбивається на блоки однакової довжини. Кожний блок, незалежно від інших, шифрується в блок шифротексту. Шифрування в режимі електронної кодової книги допускає також процес розпаралелювання. Розпаралелювання дозволяє шифрувати одночасно декілька блоків. Це збільшує швидкість шифрування відкритого тексту. При цьому шифрування здійснюється при поступленні цілого блоку. жодне опрацювання тексту неможливе до надходження.

На рис. 2.4 показано схему роботи блокового шифру в режимі електронної шифрувальної книги.

Рисунок 2.4 – Схема блокового шифрування в режимі електронної шифрувальної книги

Режим шифрування за допомогою кодової книги допускає заміну зашифрованого повідомлення на якесь інше, тобто нав’язування опонентом без знання ключа помилкового повідомлення. Факт заміни може бути не виявлений після перетворення шифротексту в відкритий текст.

Перевагою алгоритму електронної кодової книги є те, що одиничні помилки в криптограмі в процесі розшифрування будуть поширюватися тільки в межах того блоку криптограми, де ця помилка виникла. На інші блоки дешифрованого повідомлення дана помилка не буде поширюватися.

Недоліком алгоритму є той факт, що при однаковому ключі шифрування однакові блоки повідомлення завжди шифруються в ті самі блоки криптограми. Цей факт

може використати опонент для одержання секретної 4 інформації про повідомлення навіть при відсутності ключа. Так, знайшовши вигляд криптограми для деякого повідомлення один разом, можна, зустрівши цю же криптограму в інший раз, безпомилково стверджувати без знання ключа, що передавалося те ж саме повідомлення.

4 2.4 Опис функцій для блокового шифрування

Блочні шифри перетворюють відкритий текст, представлений блоками двійкових послідовностей, у шифровані блоки двійкових послідовностей. Блокові шифри використовуються для шифрування потоків бітів. Алгоритм шифрування блоковим шифром полягає в наступному. Кожне повідомлення ділиться на блоки довжиною бітів.

Функції блокового шифрування можуть містити логічні операції та операції зсуву бітів:

логічне множення;

логічне додавання;

множення по модулю ;

арифметичний

арифметичний зсув бітів вправо;

циклічний зсув бітів вліво;

циклічний зсув бітів вправо.

Всі дії, які проводяться над блоковими бітових послідовностей криптоалгоритмами, ґрунтується на тому, що перетворюваний блок може бути представлений в виді цілого невід'ємного числа з діапазону, який відповідає його розрядності.

Характерною особливістю блочних алгоритмів є багатократне використання ключа. Це робить неможливим обернене розшифрування 9 по відношенню до ключа при відомих початкового і зашифрованого текстів. Перед початком шифрування послідовність двійкових символів початкового 4 повідомлення розбивається на блоки довжиною бітів (– парне).

Ключова послідовність змінюється від ітерації до ітерації. На кожній ітерації ключ міняється з вихідного ключа за допомогою перестановок або зсувів певних елементів ключа. Для цього з початкового ключа за допомогою заданого перетворення, що входить в опис алгоритму шифрування, формується послідовність ключів , , де – число

ітерацій.

Така структура циклу має ряд переваг:

Процеси 6 шифрування і дешифрування повідомлень збігаються, тільки послідовність ключа використовують у зворотному порядку;

Для розшифрування можна використати ті самі програмні модулі, що і для шифрування.

Стійкість криптосистеми суттєво залежить від ключової послідовності. Якщо ключова послідовність невелика, то стійкість системи є також невеликою.

Кожний біт шифрованого блоку даних є функцією від усіх його бітів і бітів ключа. Зміна одного біта даних впливає на зміну кожного біта криптограми.

2.5 Опис алгоритму для шифрування відкритих текстів

Процес шифрування текстів складається з наступних кроків:

Перетворення відкритого тексту в послідовність двійкових бітів;

Поділ послідовності бітів на блоки;

Перетворення відкритого тексту у шифротекст по одному блоку з використанням бітів ключової послідовності;

Передача шифрованої послідовності на приймальну сторону.

На рис. 2.5 показано структурну схему алгоритму шифрування відкритого тексту за допомогою блокового шифру в режимі шифрувальної кодової книги.

Рисунок 2.5 – Схема алгоритму шифрування в режимі електронної шифрувальної книги

Алгоритм шифрування відкритих текстів блоковим шифром в режимі шифрувальної кодової книги виконує наступні функції:

Читання відкритого тексту, призначеного для шифрування, з файлу;

Кодування літер початкового тексту в послідовність бітів;

Читання ключової бітової послідовності для шифрування з файлу;

Шифрування тексту в бітову послідовність за допомогою блокового шифру з ключем, який задається відправником повідомлення і тримається в секреті;

Перетворення бітової послідовності в шифротекст;

Запис шифротексту в файл та вивід його на екран.

Алгоритм працює з 64-бітовим блоком відкритого тексту. При використанні блокових шифрів відкритий текст перетворюється у шифротекст блоками по 64 біти з використанням 64-х бітів ключа. Ітераційний алгоритм перетворює 64-бітові 6 блоки відкритого тексту, які надходять від споживача в 64-бітові блоки шифротексту. Це симетричним алгоритмом. Для 6 шифрування і дешифрування тексту використовується однакова послідовність ключів. Ключова послідовність може змінюватися на кожній ітерації.

2.6 Опис алгоритму для дешифрування шифротекстів

Алгоритм дешифрування шифротекстів, зашифрованих в режимі електронної шифрувальної книги є симетричним, тому для шифрування тексту і дешифрування шифротексту використовуються однакові алгоритми і однакова бітова послідовність ключів. На кожній ітерації ключові послідовності можуть змінюватися. Алгоритм дешифрування шифротекстів в режимі шифрувальної кодової книги виконує наступні функції:

Читання шифротексту з файлу;

Кодування літер шифротексту в послідовність бітів;

Читання бітової послідовності ключа дешифрування з файлу;

Розшифрування шифротексту 8 з використанням ключа дешифрування, який передається відправником повідомлення і тримається в секреті;

Запис розшифрованих блоків послідовності бітів у файл;

Перетворення послідовності бітів шифротексту в літери початкового алфавіту;

Запис розшифрованого тексту в файл і вивід його на екран.

На рис. 2.6 показано схему розшифрування шифротексту на приймальній стороні з використанням ключової послідовності і одержання розшифрованого повідомлення на основі блокового шифру в режимі кодової шифрувальної книги.

Рисунок 2.6 – Схема розшифрування шифротекстів в режимі шифрувальної кодової книги

Помилки, які виникли в поточному блоці шифрованому тексті на приймальній стороні,

не тиражуються при розшифруванні наступних блоків, так як кожний блок дешифрується 6 незалежно один від одного.

6 Якщо в процесі передачі шифрованої інформації втратився біт шифротексту, то цілий блок, в якому втратився біт, не може бути правильно розшифрований. В цьому випадку сторони повинні провести повторну передачу і прийом повідомлення.

2.7 Загальна схема процесу 6 шифрування і дешифрування відкритих текстів

В дипломному проєкті розроблено алгоритм і програмне забезпечення для шифрування і розшифрування текстів за допомогою блокового шифру в режимі шифрувальної кодової книги. Особливістю цього шифру є використання однакових ключів для шифрування і розшифрування повідомлень. При розшифруванні ключ використовується у зворотному порядку.

Алгоритм шифрування виконує наступні функції:

1 Запис відкритого тексту в файл File_input.txt.

2 Читання відкритого тексту з файлу File_input.txt і перетворення його в послідовність бітів за допомогою програмного модуля d_input_bit.c.

3 Запис послідовності бітів в файл File_bit.txt.

4 Шифрування бітової послідовності за допомогою програмного модуля d_shyf_Book.c 8 з використанням ключа шифрування (файл File_KEY_ch.txt).

5 Запис шифротексту в файл File_Book.txt.

6 Вивід на екран інформації з файлу File_Book.txt у вигляді зашифрованої бітової послідовності.

Алгоритм розшифрування виконує наступні функції:

1 Читання шифротексту з файлу File_Book.txt.

2 Розшифрування шифротексту за допомогою програмного модуля d_rozshyf_Book.c 8 з використанням ключа розшифрування (файл File_KEY_de.txt).

3 Запис розшифрованої бітової послідовності в файл File_debit.txt.

4 Перетворення бітової послідовності File_debit.txt у відкритий текст за допомогою програмного модуля d_output_bit.c

5 Запис відкритого тексту в файл file_output.txt.

6 Вивід на екран інформації з файлу у вигляді відкритого тексту.

3 РОЗРОБКА ПРОГРАМНИХ ЗАСОБІВ ДЛЯ 9 ШИФРУВАННЯ ТА ДЕШИФРУВАННЯ ВІДКРИТИХ ТЕКСТІВ

3.1 Опис програмного коду для шифрування відкритих текстів в режимі електронної шифрувальної книги

Програмний код написано на мові С. Мову вибрано тому, що вона порівняно з іншими мовами програмування характеризується мінімальним часом розробки. А мінімізація часу розробки програм є важливою проблемою, з якою стикаються фахівці при шифровані і розшифрування повідомлннь.

Шифрування відкритих текстів за допомогою блокових шифрів в режимі 6 електронної кодової книги ґрунтується на перетворенні символів вихідного алфавіту в числові коди для їх подальшого перетворення в бітові послідовності. Це означає, що літерам алфавіту відкритого тексту присвоюються двійкові коди. В табл. 3.1 наведено двійкові коди символів, з використанням яких написано повідомлення, що підлягає шифруванню.

Таблиця 3.1 – Кодування символів вихідного алфавіту двійковими кодами

Символ

Числовий код

Двійковий код

Символ

Числовий код

Двійковий код

A

0

00000000

C

20

00010100

Б

1

00000001

Т

21

00010101

В

2

00000010

У

22

00010110

Г

3

00000011

Ф

23

00010111

Д

4

00000100

Х

24

00011000

Е

5

00000101

Ц

25

00011001

Є

6

00000110

Ч

26

00011010

Ж

7

00000111

Ш

27

00011011

З

8

00001000

Щ

28

00011100

И

9

00001001

Ь

29

00011101

І

10

00001010

Ю

30

00011110

Ї

11

00001011

Я

31

00011111

Й

12

00001100

.

32

00100000

К

13

00001101

;

33

00100001

Л

14

00001110

,

34

00100010

М

15

00001111

:

35

00100011

Н

16

00010000

-

36

00100100

О

17

00010001

пропуск

37

00100101

П

18

00010010

"

38

00100110

Р

19

00010011

Алгоритм працює з 64-бітовим блоком відкритого тексту. При використанні блокових шифрів відкритий текст перетворюється у шифротекст блоками по 64 біти з використанням послідовністю ключів на кожній ітерації. Ключова послідовність може змінюватися на кожній ітерації. Ітераційний алгоритм перетворює 64-бітові блоки відкритого тексту, які надходять від відправника повідомлення в 64-бітові блоки шифротексту.

На основі розробленого алгоритм написано програмне забезпечення для шифрування відкритих повідомлень на основі блокового шифру в режимі кодової книги. Програма демонструє основні прийоми роботи з шифруванням повідомлень. При шифруванні блоки відкритого тексту перетворюються в блоки шифротексту. Програма написана на мові програмування С.

Програма спроектована по принципу розділення її на окремі функціональні компоненти. В процесі роботи програми компоненти виконують конкретні функції і

взаємодіють між собою. Таким підхід дає змогу розробляти багатофункціональне програмне забезпечення.

Відкритий текст для шифрування розміщений в текстовому файлі File_input.txt в символьному форматі. Програмний модуль d_input_bit.c перетворює відкритий текст в бітову послідовність. Одержана бітова послідовність записується в файл. File_bit.txt. Повний текст програми на мові С для перетворення відкритих текстів у бітову послідовність наведено в додатку А.

Програма для шифрування відкритих текстів в режимі електронної кодової книги складається з наступних компонентів:

1 Підключення бібліотечних файлів, які містять прототипи стандартних функцій файлового вводу-виводу, функцій обробки символьної інформації та функцій роботи системи.

2 Задання довжини блоку і кількості ітерацій процесу шифрування.

3 Опис вказівників на змінні структурного типу FILE, які асоціюють конкретні фізичні файли на диску з потоками вводу-виводу.

4 Задання імен файлів для зберігання вхідного і зашифрованого текстів.

5 Відкриття файлів File_KEY_ch.txt і File_KEY_de.txt для читання послідовності ключів шифрування і запам’ятовування їх для дешифрування відповідно.

6 Відкриття файлів File_bit.txt і File_SHYF.txt для читання початкової бітової послідовності та для запису шифрованої бітової послідовності.

7 Читання ключа шифрування з файлу File_KEY_ch.txt запис його в масив.

8 Читання з файлу тексту повідомлення для шифрування і перевірка. Перетворення послідовності бітів з символьного формату в числовий.

9 Обчислення криптограми для кожного блоку на основі заданої ключової послідовності бітів і функціональних залежностей криптограми від початкового тексту.

10 Запис ключової послідовності бітів в файл File_KEY_de.txt для її використання при розшифруванні шифротексту.

Результатом роботи програмного модуля d_shyf_Book.c є створення файлу File_Book.txt для зберігання бітової послідовності шифротексту. Повний текст програми на мові С для шифрування відкритих повідомлень наведено в додатку Б.

3.2 Результат перетворення відкритого тексту в шифротекст

Відкритий текст задається в файлі File_input.txt. За допомогою програмного модуля d_input_bit.c вхідний текст перетворюється в числові десяткові коди, а потім – в послідовності бітів. Послідовність бітів записується в файл File_bit.txt.

Вхідними даними для програми є текст для шифрування. Текст для шифрування показано на рис. 3.1.

Рисунок 3.1 – Текст для шифрування

Результатом роботи програми є перетворення символів тексту для шифрування в послідовність бітів і запис закодованого повідомлення в файл File_bit.txt у вигляді послідовності бітів. На рис. 3.2 наведено результати перетворення відкритого тексту в бітову послідовність.

Рисунок 3.2 – Результати перетворення відкритого тексту в бітову послідовність

Вхідними даними для програми шифрування d_shyf_Book.c є послідовність бітів, яка розміщена в файлі File_bit.txt, що є кодом відкритого тексту і ключова послідовність, яка знаходиться в файлі File_KEY_ch.txt у вигляді бітової послідовності. Результатом роботи програми є зашифрований текст, який записується у файл File_Book.txt. На рис. 3.3 наведено ключ для шифрування, що міститься у файлі File_KEY_ch.txt.

Рисунок 3.3 – Файл File_KEY_ch.txt бітової послідовності ключа

На рис. 3.4 наведено фрагмент результатів шифротексту.

Рисунок 3.4 – Зашифрований текст у файл File_Book.txt

3.3 Опис програмного коду для розшифрування шифротекстів

Алгоритм шифрування в режимі шифрувальної електронної книги є симетричним. Тому для дешифрування шифротексту використовується однаковий алгоритм і однакова послідовність ключів, застосована в зворотному напрямі. Кількість ітерацій дешифрування повинна співпадати з кількістю ітерацій шифрування. Ключові послідовності змінюються на кожній ітерації в оберненому напрямі.

На основі розробленого алгоритму створено програмне забезпечення для дешифрування повідомлень, зашифрованих блоковим шифром в режимі електронної книги. Розроблене програмне забезпечення ілюструє процес розшифрування зашифрованих повідомлень. Програма дешифрує повідомлення з секретним ключем, який знаходиться в одержувача повідомлення. Результатом роботи програми є перетворення зашифрованого тексту, переданого по каналу зв’язку, в початковий.

Зашифрований текст для дешифрування у вигляді бітової послідовності нулів і одиниць зберігається в текстовому файлі File_Book.txt в символьному форматі.

Програмий модуль d_rozshyf_Book.c складається з наступних компонентів:

1 Підключення бібліотечних файлів, які містять прототипи стандартних функцій файлового вводу-виводу, функцій обробки символьної інформації та функцій роботи системи.

2 Задання довжини блоку і кількості ітерацій процесу дешифрування.

3 Опис вказівників на змінні структурного типу FILE, які асоціюють фізичні файли з потоками вводу-виводу fp3, fp4 і fp_key_de.

Потік fp3 зв'язаний з файлом File_shuf.txt, що містить зашифрований текст, потік fp4 зв'язаний з файлом File_debit.txt для зберігання послідовності бітів розшифрованого тексту. Потік fp_key_de асоціюється з файлом File_KEY_de.txt, що містить ключ дешифрування.

4 Задання імен файлів для зберігання зашифрованого і розшифрованого текстів.

5 Відкриття файлів File_Book.txt і file_KEY.txt для читання, який містить послідовність зашифрованих бітів, та для читання ключової послідовності.

6 Відкриття файлу File_debit.txt для запису розшифрованої бітової послідовності.

7 Читання ключа дешифрування та тексту для дешифрування з файлів File_KEY_de.txt і File_SHYF.txt, відповідно.

8 Розшифрування криптограми на основі заданого ключа і відповідних співвідношень та одержання розшифрованого тексту . Функція дешифрування в наведеному прикладі має вигляд .

9 Запис блоків розшифрованого тексту в файл.

15 Закриття файлів після закінчення роботи програми.

Результатом роботи програми дешифрування є створення файлу File_debit.txt, в якому знаходиться розшифроване повідомлення у вигляді бітової послідовності. Текст програми d_rozshyf_book.c для розшифрування повідомлень в режимі електронної кодової книги наведено в додатку В. Бітова послідовність розшифрованого текст уперетворюється у відкритий текст, який записується у файл File_output.txt. Текст програми d_output_bit.c для перетворення розшифрованої бітової послідовності в

початковий текстнаведено в додатку Г.

4.4 Результат перетворення шифротексту в відкритий текст

Вхідними даними для програми дешифрування є послідовність бітів криптограми, яка знаходиться у файлі File_Book.txt і ключова послідовність, яка знаходиться в файлі File_KEY_de.txt. Результатом роботи програми є розшифрований текст, який записується у файл File_debit.txt у вигляді бітової послідовності. На рис. 3.5 наведено ключ для дешифрування, що міститься у файлі File_KEY_de.txt.

Рисунок 3.5 – Файл File_KEY_de.txt бітової послідовності ключа розшифрування

На рис. 3.6 наведено фрагмент результатів розшифрованого тексту у вигляді бітової послідовності.

Рисунок 3.6 – Фрагмент результатів розшифрованого тексту

Результатом роботи програмного модуля для дешифрування текстів є розшифрований текст, який записується у файл File_output.txt. На рис. 3.7 наведено початковий текст.

Рисунок 3.7 – Результати перетворення шифротексту у відкритий текст

Аналіз одержаних результатів показує, що після проведеної операції шифрування відкритого тексту, наведеного на рис. 3.1, та операції розшифрування зашифрованого тексту, одержано початковий текст, представлений на рис. 3.7.

Посилання

Це джерела виділених збігів у вашому документі. Кожен збіг позначено темно-зеленим числом, яке відповідає вказаному тут джерелу. Джерела впорядковані за схожістю — чим вищий бал, тим сильніше збіг.

#	Джерело	%
1	moodle.znu.edu.ua	5.1%
2	moodle2.snu.edu.ua	3.7%
3	um.co.ua	3.5%
4	files.znu.edu.ua	3.0%
5	moodle.znu.edu.ua	1.8%
6	studfile.net	1.6%
7	studfile.net	1.3%
8	dspace.wunu.edu.ua	1.1%
9	isg-konf.com	0.6%
10	nubip.edu.ua	0.6%
11	skachatvs.com	0.6%
12	moodle.znu.edu.ua	0.6%
13	conf.ldubgd.edu.ua	0.6%
14	pdf.lib.vntu.edu.ua	0.4%
15	studfile.net	0.3%
16	learn.ztu.edu.ua	0.3%
17	soctech.narod.ru	0.3%
18	elartu.tntu.edu.ua	0.3%
19	ela.kpi.ua	0.2%
20	elar.khnu.km.ua	0.2%
21	dspace.pdau.edu.ua	0.1%
22	dspace.uzhnu.edu.ua	0.1%
23	ipme.kiev.ua	0.0%

Plag

Дякуємо, що перевірили
свій документ за допомогою
Plag!