

Лабораторна робота №5

Дослідження розсіювальних властивостей S-DES

Мета:

1. Дослідити властивості розсіювання та перемішування бітів у спрощеній криптосистемі S-DES.
2. Дослідити залежність вихідних параметрів від структури блоків заміни алгоритму.

Обладнання:

- персональний комп'ютер з встановленою операційною системою;
- будь-яка мова програмування.

Завдання:

1. Вивчити розсіювальні та перемішувальні властивості S-DES від структури ключа та вхідного тексту.
2. Вивчити залежність вихідних бітів від структури матриць заміни криптоалгоритму.

Література:

1. В.Столлингс. Криптография и защита сетей. М.: «Вільямс», 2001. – 672 с.
2. Б.Шнайер. Прикладная криптография. Протоколы, алгоритмы, исходные коды на языке С. 1996.

Теоретичні відомості

Спрощений DES – це алгоритм шифрування, який має, скоріше, навчальне, ніж практичне значення. За своїми властивостями він подібний до DES, але має значно менше параметрів [1].

Цей алгоритм приймає на вході 8-бітний блок відкритого тексту та 10-бітний ключ, а на виході генерує 8-бітний блок шифрованого тексту. При розшифруванні на вхід алгоритму подається 8-бітний блок шифротексту і 10-бітний ключ, а на виході генерує 8-бітний блок відкритого тексту.

Алгоритм шифрування передбачає послідовне виконання п'яти операцій: початкової перестановки IP ; циклової функції, що складається з перестановок та підстановок; перестановки SW , коли дві половинки блоку по 4 біти переставляється місцями; ще одного застосування циклової функції; і, нарешті, перестановки IP^{-1} , оберненої до початкової. Послідовне використання кількох перестановок та підстановок, як це показано у [1-2], значно ускладнюють криптоаналіз.

Циклова функція приймає на вході не лише блок тексту, а й 8-бітний цикловий підключ, який утворюється з 10-бітного ключа.

Блок-схему алгоритму подано на рис. 1. З цього рисунку видно, що, оскільки це симетричний криптоалгоритм, він використовує для шифрування та розшифрування один і той самий ключ. Тому ключ має бути як на передавальній, так і на приймальній стороні. З цього ключа на певних етапах шифрування та розшифрування генерується два 8-бітних циклових підключа.

Процедура генерування циклових підключів

1. Спочатку переставляються біти ключа таким чином. Якщо 10-бітний ключ уявити у вигляді k_1, k_2, \dots, k_{10} , то перестановка P_{10} задається формулою:

$$P(k_1, k_2, k_3, k_4, k_5, k_6, k_7, k_8, k_9, k_{10}) = (k_3, k_5, k_2, k_7, k_4, k_{10}, k_1, k_9, k_8, k_6).$$

Можна також зобразити перестановку P_{10} у вигляді таблиці:

P10									
3	5	2	7	4	10	1	9	8	6

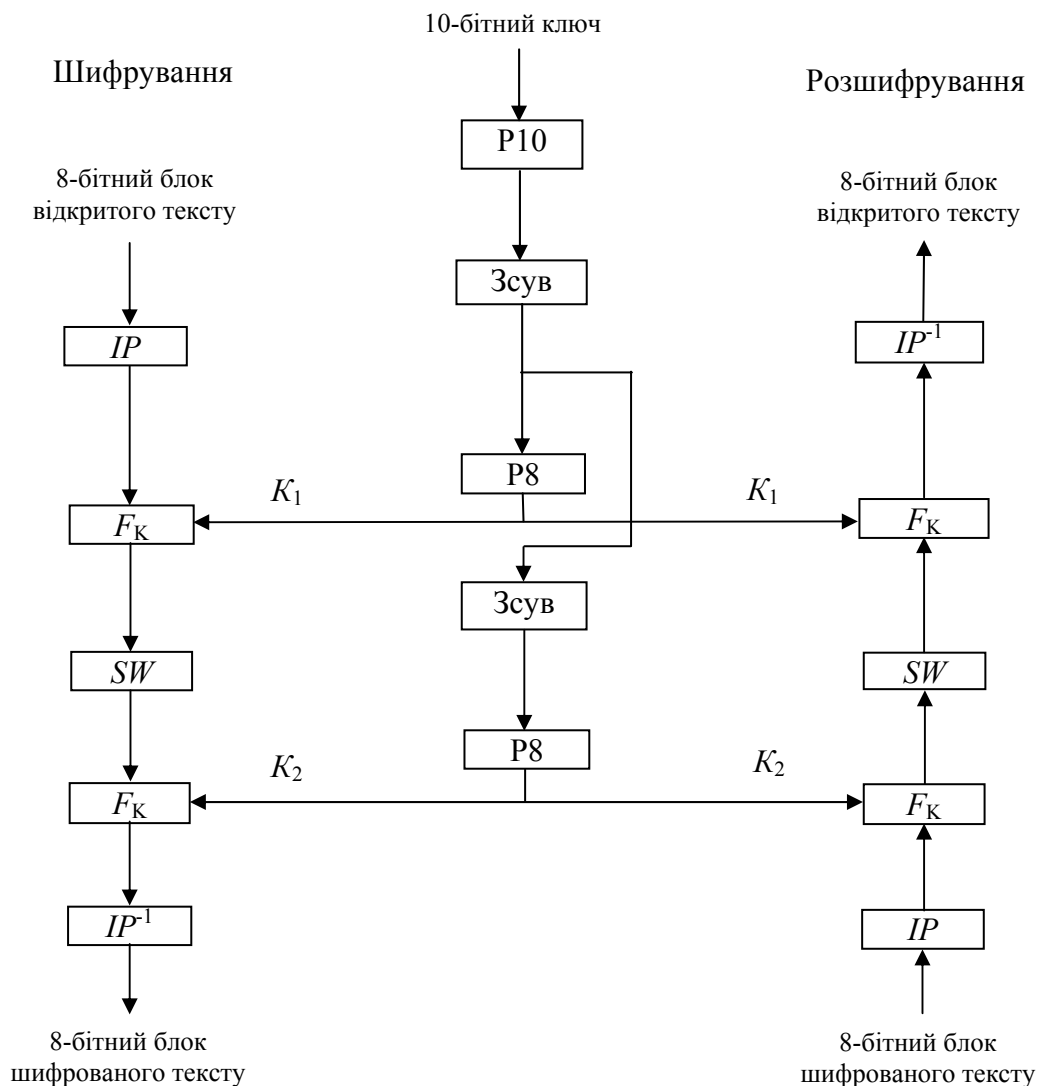


Рис. 1. Схема спрощеного алгоритме DES

Ця таблиця символізує позицію біту вхідних даних у вихідній послідовності: першм стає 3-й біт, другим – 5-й, третім – 2-й і т.д. Наприклад, ключ (1010000010) у відповідності з цією перестановкою перетворюється у послідовність (1000001100).

2. Ключ розділяється на дві половинки по п'ять бітів кожна. Окремо перша половина і окремо друга піддаються циклічному зсуву ліворуч на одну позицію. У нашому прикладі в результаті буде отримана послідовність (00001 11000).

3. Отримана послідовність піддається перестановці P8, в результаті якої з 10-бітного ключа обираються 8-бітова послідовність за таким правилом:

P8							
6	3	7	4	8	5	10	9

В результаті цієї операції ми отримаємо перший цикловий підключ (K₁). У нашому прикладі він буде мати вигляд (10100100).

4. Для генерування другого циклового підключа K₂ необхідно повернутися на крок назад, до двох 5-бітових рядків до застосування P8 та виконати для кожного з цих рядків

циклічний зсув ліворуч на дві позиції. У нашому прикладі значення підключів (00001 11000) перетворюються у (00100 00011).

5. Нарешті, застосувавши до цієї послідовності перестановку P8, отримуємо другий цикловий підключ K_2 . Для нашого прикладу результатом буде (01000011).

Шифрування S-DES

1. Початкова і кінцева перестановки (IP та IP^{-1}). На вхід алгоритму подається 8-бітний блок відкритого тексту, до якого застосовується початкова перестановка IP :

IP							
2	6	3	1	4	8	5	7

На завершальній стадії алгоритму виконується обернена перестановка IP^{-1} :

IP^{-1}							
4	1	3	5	7	2	8	6

Можна пересвідчитися, що ці дві таблиці дійсно обернені одна до другої тобто $IP^{-1}(IP(M))=M$.

2. Циклова функція F_K . Розіб'ємо вхідний блок тексту після IP -перестановки на два 4-бітні підблоки. Лівий 4-бітний блок позначимо через L , а правий - через R . Тоді циклову функцію можна записати у вигляді такої формули:

$$F_K(L, R) = (L \oplus F(R, K_i), R). \quad (1)$$

Тут K_i означає цикловий підключ, K_1 або K_2 ; \oplus - побітове XOR.

Тепер опишемо саму циклову функцію. На вході вона отримує 4-бітне значення (n_1, n_2, n_3, n_4), тобто праву половину вхідного блоку. Перша операція – операція розширення та перестановки. Її можна також зобразити табличкою:

Розширення з перестановкою							
4	1	2	3	2	3	4	1

Зручніше цю операцію зобразити у вигляді такої матриці:

$$\begin{matrix} n_4 & n_1 & n_2 & n_3 \\ n_2 & n_3 & n_4 & n_1 \end{matrix}$$

До цього значення додається 8-бітний підключ за допомогою операції XOR. Це можна зобразити наступним чином:

$$\begin{matrix} n_4+k_1 & n_1+k_2 & n_2+k_3 & n_3+k_4 \\ n_2+k_5 & n_3+k_6 & n_4+k_7 & n_1+k_8 \end{matrix}$$

Переіменуємо отримані елементи наступним чином:

$$\begin{matrix} p_{00} & p_{01} & p_{02} & p_{03} \\ p_{10} & p_{11} & p_{12} & p_{13} \end{matrix}$$

Перші чотири біти (тобто перший рядок цієї матриці) далі подаються на вхід модуля заміни (S-матриці), S_0 , на виході якого отримується 2-бітна послідовність. Другий рядок матриці подається на вхід другого модуля заміни, S_1 , на виході якого також отримується 2-бітна послідовність.

Модулі S_0 та S_1 задаються наступним чином:

$$S_0 = \begin{pmatrix} 1 & 0 & 3 & 2 \\ 3 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 3 & 1 & 3 & 1 \end{pmatrix}; \quad S_1 = \begin{pmatrix} 1 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 3 & 0 & 1 & 0 \\ 2 & 1 & 0 & 3 \end{pmatrix}$$

Рядки та стовпчики нумеруються, починаючи з нуля.

Ці модулі заміни працюють наступним чином. Перший і четвертий біти вхідної послідовності вважаються за двійкове представлення номера рядка, а другий та третій – номера стовпчика. Елемент, що знаходиться на перетині цих рядка і стовпчика задає двобітне вихідне значення. Наприклад, якщо $(p_{00}, p_{03}) = (00)$ та $(p_{01}, p_{02}) = (10)$, то вихідні два біти задаються значенням, яке знаходиться на перетині 0-го рядка та 3-го стовпчика, тобто буде числом 3, а у двійковому представленні – 11.

Аналогічну операцію виконують і з другим рядком $p_{10} \ p_{11} \ p_{12} \ p_{13}$.

Після застосування матриць заміни результат піддають перестановці P4 за таким законом:

P4			
2	4	3	1

Результат перестановки P4 і буде результатом функції F_K . Отримана послідовність бітів додається за модулем 2 з лівою половиною L вхідного блоку і буде новою лівою половиною. Права половина передається на вихід циклу без змін.

3. *Перестановка підблоків.* Як бачимо, за один цикл цикловою функцією обробляється лише ліва половина відкритого тексту, права половина залишається без змін. Для того, щоби зашифрувати й праву половину, використовується другий цикл, однак на його вхід треба подати переставлені підблоки: L і R поміняти місцями. Для цього й служить функція SW – перемикач блоків.

Після переставлення підблоків один цикл алгоритму закінчено.

До переставлених підблоків знову застосовується циклова функція, як це описано вище. При другому виклику циклової функції розширення з перестановкою, модулі S_0 та S_1 та P4 залишаються тими ж, тільки використовується підключ K_2 .

По закінченні другого циклу виконується IP^{-1} -перестановка і роботу алгоритму закінчено, тобто на виході маємо зашифрований текст.

Розшифрування зашифрованого тексту

Як видно з рисунку 1, розшифрування зашифрованого тексту виконується аналогічно шифруванню за винятком того, що ключі подаються у зворотному порядку.

Практична частина

1. Використовуючи будь-яку мову програмування створіть програму, яка б шифрувала та розшифровувала текстові повідомлення за допомогою криптосистеми S-DES.

2. Використовуючи створену програму, вивчіть її властивості розсіювання в залежності від вхідних бітів ключа. Для цього оберіть ключ «0000000000» та зашифруйте на ньому вибраний текст. Після цього змініть молодший біт ключа на 1 та виконайте шифрування того ж тексту. Скільки бітів зашифрованого тексту змінилося? Виконайте таку ж операцію з кожним бітом ключа. Поясніть отримані результати. Зміна якого біту ключа призводить до найбільших та найменших змін у зашифрованому тексті? Змініть ключ на «1111111111» та повторіть усі запропоновані операції. Оберіть випадковий ключ та виконайте усі дослідження для нього. Поясніть отримані результати.

3. Зашифруйте на випадковому ключі тексти «00000000». Поясніть отримані результати. Змінюйте по одному біту вхідного тексту та послідовно зашифровуйте його на

тому ж ключі? Скільки бітів зашифрованого тексту змінилося? Зміна яких бітів відкритого тексту найбільше і найменше впливає на зашифрований? Ті ж операції виконайте для тексту «11111111» та довільного відкритого тексту. Поясніть отримані результати.

4. Виконайте шифрування довільного тексту на довільному ключі. Змініть структуру матриць заміни, починаючи з першого рядка, виконуючи щоразу шифрування. Як змінюється результат? Визначте, яка зміна структури матриць призвела до найкращих та найгірших результатів.

Порівняйте розсіювальні властивості спрощеного алгоритму з такими ж властивостями реального DES. Дані про DES можна знайти у книзі [2] та Інтернеті.

5. Здійсніть атаку методом повного перебору ключів («грубою силою») та визначте скільки часу знадобилося Вашому комп'ютеру для повного перебору ключів. Скільки усього ключів довелося перебрати? Скільки ключів довелося перебрати до досягнення успіху атаки?

Зробіть висновок з лабораторної роботи. Підготуйте звіт з лабораторної роботи. Звіт повинен містити: а) результати досліджень; б) протокол Ваших дій; в) висновки з досліджень впливу зміни ключа, вхідного тексту та матриць заміни на результати шифрування; г) відповіді на контрольні запитання.

Контрольні запитання:

1. Оцініть криптостійкість цього алгоритму по відношенню до атаки «грубою силою».
2. Що спільного у цього алгоритму з реальним DES?
3. Які відмінності у цього алгоритму від DES?
4. Оцініть швидкодію Вашої реалізації цього алгоритму (у мегабітах або мегабайтах за секунду), виконавши тестове шифрування.
5. Які відмінності у швидкодії порівняно з DES? Дані про DES можна знайти у книзі Б.Шнаєра [2].
6. Якою Ви вважаєте оптимальну структуру матриць заміни цього алгоритму і чому?
7. Що таке «слабкі ключі»? До якого результату призводить шифрування на таких ключах? Наведіть приклади «слабких ключів».