**Харківський національний економічний університет**

**імені Семена Кузнеця**

**ЗВІТ**

**З ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 6**

**за дисципліною: “Теоретичні основи криптографії”**

**Виконав: студент факультету Інформаційних технологій**

**2 курсу, спец. Кібербезпека,**

**групи 6.04.125.010.21.2**

**Бойко Вадим Віталійович**

**Перевірив:**

**Чугай Андрій Михайлович**

**ХНЕУ ім. С. Кузнеця**

**2023**

1. Який розмір блоку даних, розмір ключа шифру та розмір ключів раунду в DES?

Головні риси шифру DES визначаються тим, що він ґрунтується на схемі Фейстеля з такими параметрами:

* довжина блоку – 64 біти,
* кількість раундів – 16,
* розмір ключа – 56 бітів,
* розмір кожного з підключів k1, k2, …, k16 – 48 бітів.

2. Яка кількість раундів у DES?

В алгоритмі DES – 16 раундів

3. Скільки змішувачів та пристроїв заміни використовується у першому способі зашифрування та зворотного розшифрування? Яка їх кількість використовується у другому способі?

У першому способі зашифрування та зворотного розшифрування в шифрі DES використовується один змішувач (Feistel Network) та 8 пристроїв заміни (S-boxes). У другому способі зашифрування та зворотного розшифрування в шифрі DES також використовується один змішувач (Feistel Network) та 8 пристроїв заміни (S-boxes), так само, як у першому способі.

4. Яка кількість перестановок використовується у алгоритмі шифру DES?

У алгоритмі шифру DES (Data Encryption Standard) використовується загалом 16 раундів шифрування (для зашифрування) або розшифрування (для розшифрування). Кожен раунд включає в себе ряд операцій, таких як заміна, перестановка та зсуви бітів, а також використання змішувача та 8 пристроїв заміни (S-boxes).

5. Скільки операцій xor використовується у DES?

У алгоритмі шифру DES використовується декілька операцій XOR, а саме:

* В операції "криптографічного зсуву" (Cryptographic Shift) використовується XOR для зсуву бітів вліво або вправо на різну кількість позицій, залежно від раунду шифрування.
* У процесі "змішування" використовується XOR для комбінування результатів попереднього кроку з 48-бітним підключем раунду.
* У кожному "пристрої заміни" (S-box) використовується XOR для комбінування 6-бітного вхідного блоку з 48-бітним підключем раунду перед подальшою обробкою в S-боксі.

Отже, загальна кількість операцій XOR в одному раунді DES складається з 1-2 операцій XOR для криптографічного зсуву, 1 операції XOR для змішування та 8 операцій XOR в пристроях заміни, що робить разом 10-11 операцій XOR на один раунд шифрування/розшифрування DES. Оскільки DES використовує 16 раундів, загальна кількість операцій XOR в DES складається з 160-176 операцій XOR на одне повне шифрування/розшифрування.

6. Чому в DES потрібна операція, яка здійснює перестановку з розширенням?

Операція перестановки з розширенням (англ. Expansion Permutation) в алгоритмі шифру DES (Data Encryption Standard) використовується для розширення розміру вхідного блоку даних перед подальшою обробкою. Ця операція додає додаткові біти до вхідних даних, збільшуючи їх розмір з 32 біти до 48 бітів.

Основна причина використання операції перестановки з розширенням в DES полягає в тому, що вона допомагає ввести додатковий рівень безпеки в шифр. Вхідні дані розширюються до більшого розміру, перед тим як вони комбінуються з раундовим підключем за допомогою операції XOR та подаються на вхід в пристрої заміни (S-бокси). Це робить атаки на шифр більш складними, так як кожен біт вхідних даних тепер має вплив на кілька бітів вихідних даних після пройдення S-боксів, збільшуючи криптографічну стійкість алгоритму.

Операція перестановки з розширенням також взаємодіє з операцією XOR з раундовим підключем, створюючи залежності між вхідними та вихідними бітами, що також забезпечує додатковий рівень безпеки та стійкості шифру. Таким чином, операція перестановки з розширенням є важливим кроком в алгоритмі DES, допомагаючи забезпечити високий рівень безпеки та стійкості шифру

7. Чому генератор ключів раунду DES вимагає видалення перевірочних бітів?

Генератор ключів раунду в DES (Data Encryption Standard) вимагає видалення перевірочних бітів з ключа з метою забезпечення безпеки та захисту від можливих атак на шифр. Тому генератор ключів раунду DES використовується для видалення перевірочних бітів з ключа перед подальшим використанням в алгоритмі шифрування. Це допомагає забезпечити безпеку ключа та підвищує стійкість шифру DES шляхом усунення можливих вразливостей, пов'язаних з перевірочними бітами.

8. Що таке дворазовий DES? Яка атака дворазового DES зробила його марною?

Дворазовий DES (Double DES) - це метод шифрування, який використовує дві послідовні операції шифрування DES для забезпечення більш високого рівня безпеки.

У дворазовому DES повідомлення спочатку зашифровується одним ключем DES, а потім результат шифрування зашифровується другим ключем DES. Це створює подвійний рівень шифрування, оскільки процес шифрування виконується двічі з різними ключами. Застосування двох ключів при шифруванні може здаватися більш безпечним, оскільки атакувачу потрібно знати два ключі, щоб розшифрувати повідомлення.

Однак, була відкрита атака на дворазовий DES, відома як "атака meet-in-the-middle" (зустріч у середині). Ця атака використовує властивість дворазового DES, коли виконується шифрування з двома різними ключами, проте кінцевий результат залежить від однакових замінних блоків, які використовуються при зашифруванні і розшифруванні.

Атака meet-in-the-middle вимагає збереження пам'яті і великої кількості обчислень, але вона дозволяє зловмисникам підібрати два ключі DES, використовувані в дворазовому DES, при відомому повідомленні і відповідному шифрованому тексті. Ця атака зробила дворазовий DES вразливим до підбору ключів, тому він не вважається безпечним. Однак, застосування потрійного DES (Triple DES) з трьома ключами DES може забезпечити вищий рівень безпеки в порівнянні з дворазовим DES.

9. Що таке триразовий DES? Що таке триразовий DES із двома ключами шифру? Що таке триразовий DES із трьома ключами шифру?

Триразовий DES (Triple DES) є розширенням стандарту шифрування DES, який використовується для забезпечення більшої безпеки шифрування за рахунок використання більшої кількості ключів.

Триразовий DES може використовувати два або три ключі шифру. Триразовий DES з двома ключами шифру (іноді також називається 2TDEA або 2-key Triple DES) використовує два незалежних ключі DES для здійснення трьох раундів шифрування. Процес шифрування в 2TDEA виглядає наступним чином: блок даних спочатку шифрується з використанням одного ключа DES, потім результат шифрування розшифровується з використанням другого ключа DES, і нарешті, результат знову шифрується з використанням першого ключа DES. Розшифрування відбувається в зворотньому порядку.

Триразовий DES з трьома ключами шифру (іноді також називається 3TDEA або 3-key Triple DES) використовує три незалежних ключі DES для здійснення трьох раундів шифрування. Процес шифрування в 3TDEA виглядає наступним чином: блок даних спочатку шифрується з використанням першого ключа DES, потім результат шифрування розшифровується з використанням другого ключа DES, потім знову шифрується з використанням третього ключа DES. Розшифрування відбувається в зворотньому порядку.

10. Значення послідовності вхідних даних DES дорівнює: 1234567890abcdef16. Визначити значення послідовності на виході блоку IP-перестановки.

Для визначення значення послідовності на виході блоку IP-перестановки з вхідної послідовності довжиною 64 біти, такої як "1234567890abcdef" у шістнадцятковому форматі, потрібно спочатку перевести її в бінарний формат, а потім застосувати таблицю перестановок.

Вхідна послідовність (у бінарному форматі): 0001001000110100010101100111100000010011010001010101101111011111

Тепер скористаємось таблицею перестановок: 58 50 42 34 26 18 10 2 60 52 44 36 28 20 12 4 62 55 47 39 31 23 15 7

Переведемо у шістнадцятковий формат буде: 0x85e813540f0ab405

11. Значення послідовності R0 у DES дорівнює: R0= f0aae8a516. Визначити значення послідовності на виході E-блоку перестановки та розширення.

Використовуючи двійкове значення R0:

R0 = f0aae8a516 = 11110000101010101110100010100101 та здійснивши заміну за допомогою табл., отримаємо: RE = 101010101011010011010001110001101110010010111111 = aab4d1c6e4bf.

12. Значення послідовності R1 у DES дорівнює: R1 = 116ba13316. Визначити значення послідовності на виході E-блоку перестановки та розширення.

Перетворимо R1 у двійкову систему числення:

R1 = 0001000101101011101010010011001100010011 Застосуємо таблицю розширення E, щоб отримати послідовність довжиною 48 біт:

RE = 01100001011011001101010001010001010101010100110000110010001 10011 Отже, значення послідовності на виході E-блоку перестановки та розширення дорівнює: 616cd451554c3233.

13. Значення послідовності на виході E-блоку перестановки та розширення в DES дорівнює = 8a2b57d029a616. Значення раундового ключа: k1 = 4568581abcce16. Визначити значення послідовності на виході операції xor функції фейстеля змішувача.

Спочатку переведемо наші значення в двійковий формат:

k1 = 4568581abcce = 10001010110100001011000000110101011110011001110

R1E = 8a2b57d029a6 = 100010100010101101010111110100000010100110100110

Тепер виконаємо операцію XOR (логічне "або-виключаюче") між двійковими значеннями k1 та R1E: 10001010110100001011000000110101011110011001110 100010100010101101010111110100000010100110100110

000000001111101111100111111101010101011000111000

Тепер, переводимо в 16-річну систему: fbe7f55638.

14. Значення послідовності на виході E-блоку перестановки та розширення в DES дорівнює = 9f3ca2bffea616. Значення раундового ключа: k12=c2c1e96a4bf316. Визначити значення послідовності на виході операції xor функції фейстеля змішувача.

RE12 = 9f3ca2bffea616

k12 = c2c1e96a4bf316

Перетворимо їх у двійковий формат:

RE12 = 100110110011110010100101101111111111111010100110 (двійковий)

k12 = 11000010110001111010100110100101001010111110011 (двійковий) Виконаємо операцію XOR: 100110110011110010100101101111111111111010100110

XOR

11000010110001111010100110100101001010111110011

010110011111101110001100000110101101110101000101

Тепер, переводимо в 16-річну систему => 5bfc630db5a516

15. Значення послідовності на виході операції xor змішувача DES дорівнює: = 5dfd4bd5b55516. Визначити значення на виході блоків заміни та стиснення: а) S2; б) S3; в) S4; г) S5.

А)

Спочатку перетворимо в двійкову систему: R12 = 5dfd4bd5b555 = 010111011111110101001011110101011011010101010101

Тепер, розділимо на 8 блоків по 6 бітів:

010111 011111 110101 001011 110101 011011 010101 010101

Якщо записати перший і шостий біти разом (010111), то отримаємо в двійковому вирахуванні 012, що виражається як число 110 при десятковому обчисленні. Остання частина бітів 10112 у двійковому обчисленні є 1110 у десятковому обчисленні. На перетині рядка 1 та стовпця 11 у табл. (S2-блок заміни та стиснення) знаходиться значення результату - 1010 у десятковому обчисленні або 10102 у двійковому обчисленні. Тоді вхід 0101112 дає вихід 10102.

Тепер, робимо ці перетворення з кожним блоком, а потім все скріплюємо: 011111 => рядок 1, стовпець 15 = 05 => 01012

110101 => рядок 3, стовпець 10 = 07 => 01112

001011 => рядок 1, стовпець 5 = 02 => 00102

110101 => рядок 3, стовпець 10 = 07 => 01112

011011 => рядок 1, стовпець 13 = 09 => 10012

010101 => рядок 1, стовпець 10 = 01 => 00012

010101 => рядок 1, стовпець 10 = 01 => 00012

Отже, S2 = 10100101011100100111100100010001

Б)  
Спочатку перетворимо в двійкову систему:

R12 = 5dfd4bd5b555 = 010111011111110101001011110101011011010101010101

Тепер, розділимо на 8 блоків по 6 бітів:

010111 011111 110101 001011 110101 011011 010101 010101

Якщо записати перший і шостий біти разом (010111), то отримаємо в двійковому вирахуванні 012, що виражається як число 110 при десятковому обчисленні. Остання частина бітів 10112 у двійковому обчисленні є 1110 у десятковому обчисленні. На перетині рядка 1 та стовпця 11 у табл. (S3-блок заміни та стиснення) знаходиться значення результату - 1410 у десятковому обчисленні або 11102 у двійковому обчисленні. Тоді вхід 0101112 дає вихід 11102.

Тепер, робимо ці перетворення з кожним блоком, а потім все скріплюємо:

011111 => рядок 1, стовпець 15 = 01 => 00012

110101 => рядок 3, стовпець 10 = 14 => 11102

001011 => рядок 1, стовпець 5 = 04 => 01002

110101 => рядок 3, стовпець 10 = 14 => 11102

011011 => рядок 1, стовпець 13 = 11 => 10112

010101 => рядок 1, стовпець 10 = 05 => 01012

010101 => рядок 1, стовпець 10 = 05 => 01012

Отже, S3 = 11100001111001001110101101010101

В)

Спочатку перетворимо в двійкову систему:

R12 = 5dfd4bd5b555 = 010111011111110101001011110101011011010101010101

Тепер, розділимо на 8 блоків по 6 бітів:

010111 011111 110101 001011 110101 011011 010101 010101

Якщо записати перший і шостий біти разом (010111), то отримаємо в двійковому вирахуванні 012, що виражається як число 110 при десятковому обчисленні. Остання частина бітів 10112 у двійковому обчисленні є 1110 у десятковому обчисленні. На перетині рядка 1 та стовпця 11 у табл. (S4-блок заміни та стиснення) знаходиться значення результату - 1210 у десятковому обчисленні або 11002 у двійковому обчисленні. Тоді вхід 0101112 дає вихід 11002.

Тепер, робимо ці перетворення з кожним блоком, а потім все скріплюємо:

011111 => рядок 1, стовпець 15 = 09 => 10012

110101 => рядок 3, стовпець 10 = 05 => 01012

001011 => рядок 1, стовпець 5 = 15 => 11112

110101 => рядок 3, стовпець 10 = 05 => 01012

011011 => рядок 1, стовпець 13 = 10 => 10102

010101 => рядок 1, стовпець 10 = 02 => 00102

010101 => рядок 1, стовпець 10 = 02 => 00102

Отже, S4 = 11001001010111110101101000100010

16. Значення послідовності на виході операції xor змішувача DES

дорівнює: : = e884876816. Визначити значення послідовності на виході S-блоків заміни та стиснення змішувача.

17. Значення послідовності на виході S-блоків заміни та стиснення змішувача в DES дорівнює: : = d5b8091516. Визначити значення послідовності на виході P-блоку прямої перестановки змішувача.

Використовуючи двійкове значення RS:

RS = d5b80915 = 1101 0101 1011 1000 0000 1001 0001 0101 та здійснивши заміну за допомогою таблиці, отримаємо:

RP = 0010 0000 0011 0001 0000 1010 1001 0010 = 20310a9216

18. Значення послідовності на виході P-блоку прямої перестановки змішувача DES дорівнює: : = 07d0a03c16. Значення послідовності:

L3 = 4f73c3b316. Визначити значення послідовності на виході функції змішувача xor.

Представляючи значення RP3 та L3 у двійковому вигляді та використовуючи правило порозрядного підсумовування даних по модулю 2 отримаємо:

00000111110100000010100000111100

XOR

01001111011100111100001110110011

01001000101000111110101110001111

Перетворене значення RL на шістнадцяткове матиме вигляд RL = 48a3eb8f