Міністерство освіти і науки України
Національний університет водного господарства
та природокористування

Кафедра обчислювальної техніки

04-04-192

МЕТОДИЧНІ ВКАЗІВКИ

для виконання лабораторних та самостійних робіт
з дисципліни
"Захист інформації в комп'ютерних системах"
студентами напряму підготовки
6.050102 "Комп'ютерна інженерія"
Частина ІІІ

Рекомендовано
методичною комісією
напряму підготовки
"Комп'ютерна інженерія"
Протокол № 7
від 03 березня 2017 р.

Рівне 2017

Методичні вказівки для виконання лабораторних та самостійних робіт з дисципліни "Захист інформації в комп'ютерних системах" студентами напряму підготовки 6.050102 "Комп'ютерна інженерія". Частина ІІІ. / П. В. Ольшанський, – Рівне: НУВГП, 2017, – 28 с.

Упорядник П. В. Ольшанський, старший викладач кафедри обчислювальної техніки.

Відповідальний за випуск
Б.Б. Круліковський, кандидат технічних наук, доцент, завідувач кафедри обчислювальної техніки.

Зміст

Лабораторна робота №6.

Тема: Режими використання блочних алгоритмів шифрування

Мета роботи. Вивчити особливості використання блочних шифрів в режимах електронної шифрувальної книги ECB , зчеплення блоків шифру CBC, зворотнього зв'язку по шифротексту CFB та вихідного зворотнього зв'язку OFB.

Теоретичні відомості

При використанні блочного шифру інформація поступає на вхід шифруючого пристрою (мікросхеми чи програми) порціями однакової довжини - блоками, кожен з яких шифрується окремо.

1. Режим електронної шифрувальної книги (electronic codebook, ECB) або інша назва - режим простої заміни.

Самий простий і поширений режим роботи - кожен блок тексту шифрується окремо і незалежно один від одного. Однакові частини тексту будуть також однаковими в зашифрованому вигляді, що значно ослаблює ефективність шифрування, бо в електронній пошті часто зустрічаються однакові чи дуже подібні документи, наприклад, грошові рахунки, запити, стандартні заголовки і закінчення документів, а також довгі послідовності однакових символів - нулів чи пропусків.

Позитивна властивість - якщо при передачі будуть спотворені окремі біти чи байти, спотворення не вплине на інші блоки і вони будуть успішно розшифровані. Цей режим реалізований в попередніх лабораторних роботах.

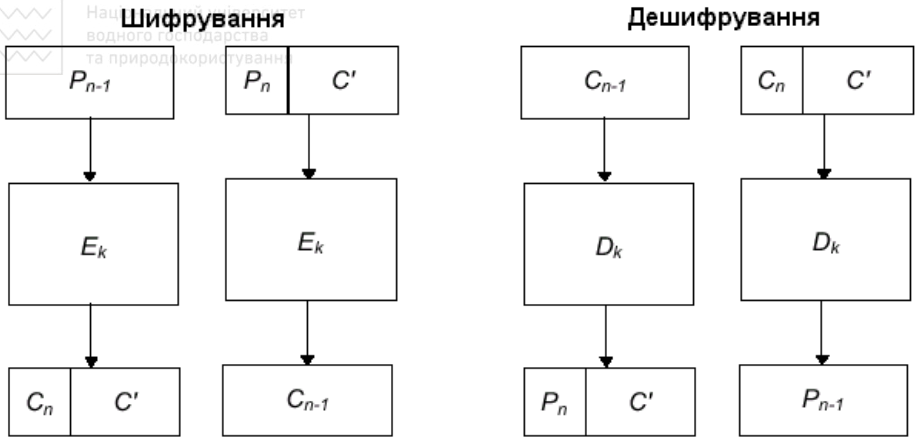

Рис.12 "Викрадення" останнього блоку в режимі ECB.

Проблема останнього неповного блока.

Більшість повідомлень точно не діляться на довжину блока шифрування, тому в кінці опиняється останній вкорочений блок. В режимах ECB і CBC блоки повинні бути 64-бітовими. Простим способом розв'язку цієї проблеми є **набивка**: останній блок доповнюється регулярним шаблоном - деякою комбінацією нулів і одиниць. Якщо потрібно вилучати набивку після дешифрування, в останній байт останнього блока записують кількість байтів набивки. Для коректної роботи методу потрібно доповнювати пустий блок навіть для випадку, якщо повідомлення містить ціле число блоків.

Інший варіант - **"викрадення" шифротексту** наведений на рис.17. $\boldsymbol{P_{n-1}}$ - останній повний блок відкритого тексту, $\boldsymbol{P_n}$ - останній, короткий блок відкритого тексту, $\boldsymbol{C_{n-1}}$ - останній повний блок шифротексту, $\boldsymbol{P_n}$ - останній, короткий блок шифротексту, $\boldsymbol{C'}$ - проміжний результат.

2. Режим зчеплення блоків шифру (cipher block chaining, CBC) Результати шифрування попередніх блоків впливають на шифрування чергового блока. Кожен блок шифротексту залежить не тільки від блоку тексту, який шифрується, але і від всіх попередніх блоків тексту.

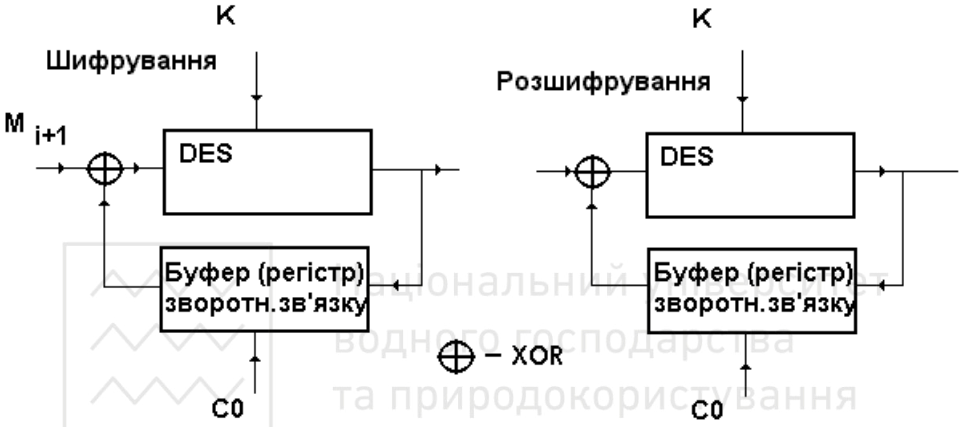

Рис.13 Режим зчеплення блоків шифротексту CBC.

Перед шифруванням чергового блоку тексту виконується побітова логічна операція XOR з ним та попереднім вже зашифрованим блоком, який зберігається у спеціальному буфері (регістрі) зворотнього зв'язку після закінчення попереднього кроку. Розшифровування відбувається за тією ж схемою.

Математичний запис шифрування $C_i = E_k (P_i \oplus C_{i-1})$

розшифровування $P_i = C_{i-1} \oplus D_k (C_i)$

Вектор ініціалізації. В методі CBC однакові блоки будуть виглядати по-різному, але однакові повідомлення, так само, як і в ECB, виглядатимуть однаково. Для запобігання цього на початку шифрування в регістр зворотнього зв'язку можна записувати

згенероване випадкове значення, яке можна передавати відкрито на початку повідомлення і завантажувати в буфер розшифровуючого пристрою. В цей нульовий блок C_0 рекомендується включати мітки дати та часу, які зчитуються із системного годинника, кодуються чи шифруються за певною схемою і ускладнюють підробку чи повторне посилання перехоплених документів (наприклад, грошових переказів) зловмисниками.

Проблема останнього блоку. Нехай останній блок містить L бітів. Після шифрування останнього повного блока знову шифрується шифротекст, з результату вибирається L бітів і виконується операція XOR з коротким блоком. Хоча останній блок не може бути розшифрований, але він може бути непомітно спотворений зловмисниками. Якщо останні біти повідомлення містять важливу інформацію, це небезпечно. Кращим методом є "викрадення" шифротексту (рис.14)

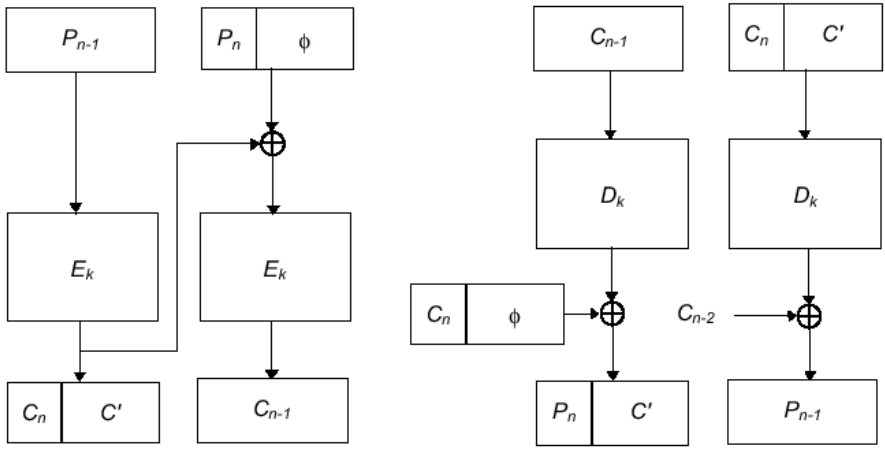

Рис.14 "Викрадення" останнього блоку шифротексту в режимі CBC.

Поширення помилки. Режим CBC характерний прямим зворотнім зв'язком з шифротекстом при шифруванні та інверсним оберненим зв'язком при дешифруванні. Єдина бітова помилка у відкритому тексті вплине на даний блок та на всі інші. Це не страшно, бо дешифрування інвертує цей ефект і відновлений після дешифрування текст міститиме ту саму єдину помилку. Частіше зустрічаються помилки в шифротексті. Блок, який містить єдину помилку, буде повністю спотворений при дешифруванні. Ця помилка вплине також при дешифруванні на наступний блок. Тому одна бітова помилка викликає спотворення двох блоків. Поширення помилки шифротексту є головним недоліком режиму CBC.

Потокові шифри. В багатьох випадках, наприклад, при передачі термінових команд чи повідомлень, інформацію потрібно передавати негайно, не чекаючи накопичення повного блоку, необхідного для початку шифрування. Для негайного шифрування послідовностей бітів чи байтів найчастіше використовуються алгоритми гамування - накладання послідовностей псевдовипадкових чисел (див. лабор. роботу 4). Можна також використовувати для шифрування блочні алгоритми (мікросхеми) в спеціальних криптографічних режимах для k бітів ($1<k<63$) зі зворотнім зв'язком. Це, зокрема, дозволяє шифрувати дані посимвольно ($k=8$).

3. Режим зворотнього зв'язку по шифротексту (cipher-feedback, CFB)

Використовується вхідний зсувний регістр з розміром, рівним розміру блока (64 біти). Спочатку в регістр, як і в режимі CBC заноситься довільний вектор C_0. Вектор шифрується і для крайніх

лівих ***k*** бітів результату виконується XOR з ***k*** бітами відкритого тексту для одержання ***k*** бітів шифротексту. Ці ***k*** бітів передаються, у вхідному регістрі відбувається зсув на ***k*** бітів вліво, причому зсув нециклічний, а на звільнене місце записуються ***k*** зашифрованих бітів (рис. 15).

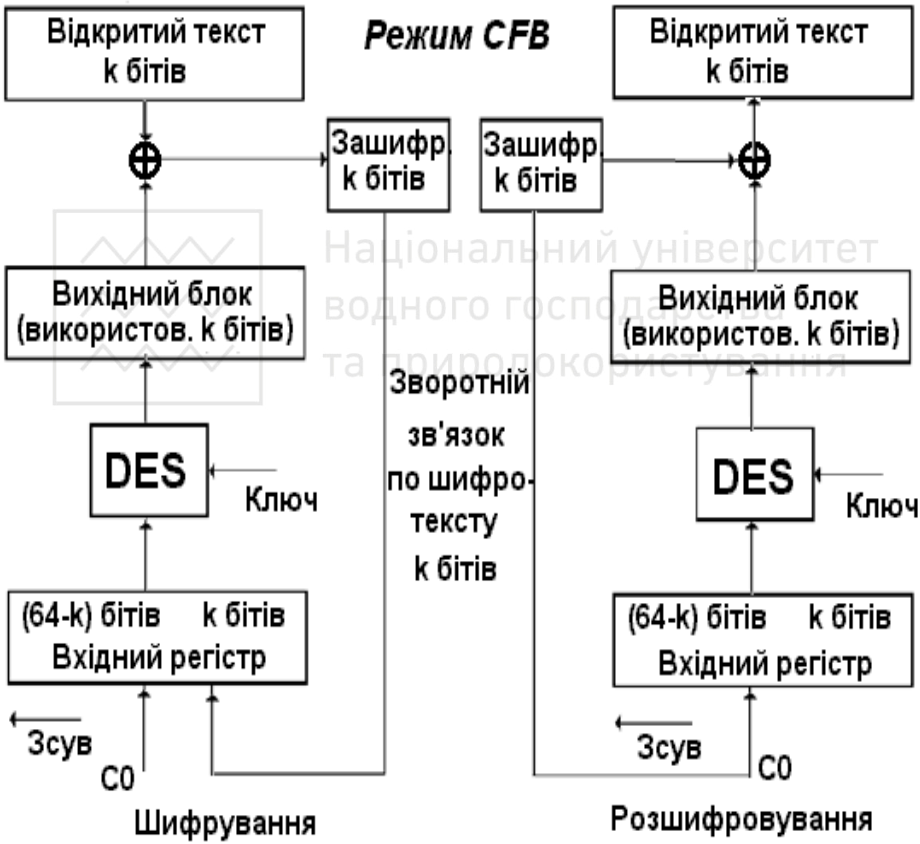

Рис.15 Структурна схема функціонування DES в режимі CFB (cipher-feedback)

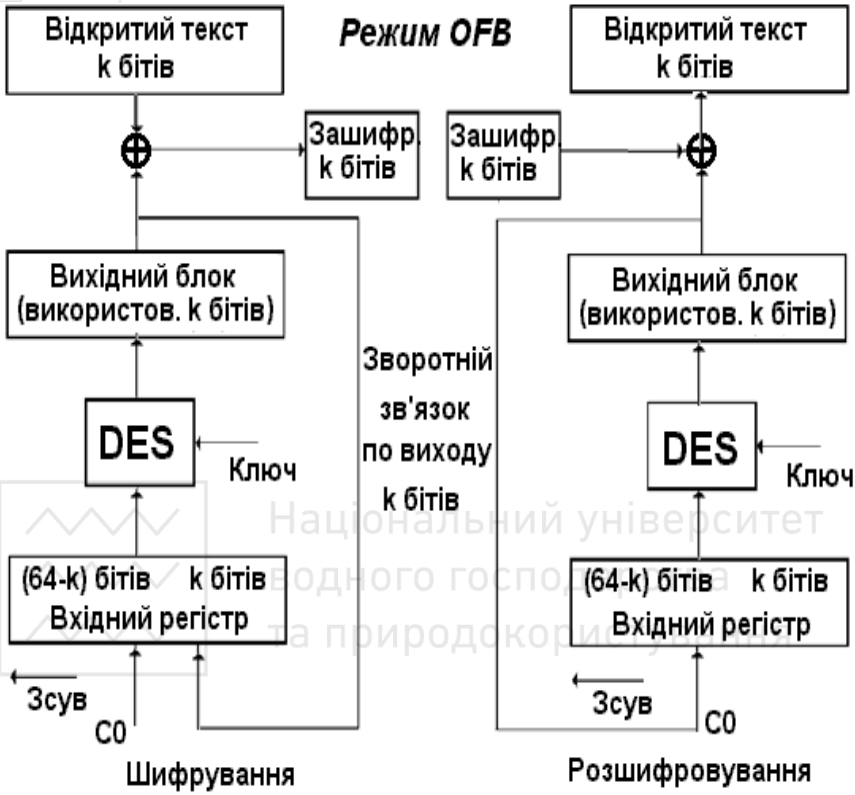

Рис. 16 Структурна схема функціонування DES в режимі OFB (output-feedback)

Поширення помилки. В режимі CFB помилка в шифротексті буде спотворювати розшифрований текст до тих пір, поки вона не покине вхідний регістр, наприклад, для побайтного шифрування (***k***=8), зіпсований байт покине вхідний регістр через 8 циклів, і зіпсованими буде 1+8=9 байтів.

4. Режим зворотнього зв'язку по виходу OFB (output-feedback)

Суттєвим достоїнством **OFB** є відсутність явища розмноження спотворень, які виникають під час передачі шифротексту, в процесі розшифровування останнього. Цікавою властивістю **OFB** є те, що він відображає будь-яку множину k-бітових блоків саму на себе: результатом шифрування будь-якого блока з цієї множини є блок з цієї ж множини. Це особливо важливо для реалізації байт-орієнтованих протоколів типу транспортних, в яких допускаються тільки такі восьмибітові комбінації, які відповідають певній підмножині символів.

В наведених схемах ***C0*** - 64-бітовий випадковим чином згенерований вектор, а зсув вхідного блоку виконується на ***k*** позицій вліво (причому зсув нециклічний).

Обидва режими використання DES для потокового шифрування позбавлені від головного недоліку ECB: ідентичним блокам відкритого тексту в загальному випадку відповідають різні блоки шифротексту.

Хід роботи

1. Доповніть алгоритм блочного шифрування в режимі ECB з попередньої роботи (див. Додаток 4) для коректного шифрування та розшифровування останнього блока із набивкою (заповненням його бітовим шаблоном).

2. Реалізуйте варіант алгоритму блочного шифрування в режимі ECB з попередньої роботи для коректного шифрування та розшифровування останнього блока із "викраденням" шифротексту.

3. Реалізуйте режим CBC для алгоритму блочного шифрування DES.

4. Доповніть алгоритм блочного шифрування в режимі CBC з попереднього пункту для коректного шифрування та розшифровування останнього вкороченого блока із набивкою.

5. Реалізуйте варіант алгоритму блочного шифрування в режимі CBC для коректного шифрування та розшифровуванням останнього блока із "викраденням" шифротексту.

6. Реалізуйте блочний алгоритм шифрування в режимі CFB для потокового побайтного шифрування (***k***=8)

7. Реалізуйте блочний алгоритм шифрування в режимі OFB для потокового побайтного шифрування (***k***=8)

Всі варіанти програм збережіть з різними назвами для звіту

Контрольні запитання.

- *Які недоліки властиві для режиму електронної записної книжки (прямої заміни)?*
- *Як вирішується проблема останнього неповного блоку в різних режимах? Запропонуйте свій алгоритм шифрування останньої неповної порції тексту.*
- *Які переваги та недоліки роботи блочного алгоритму в режимі CBC?*

- *Які особливості програмної та апаратної реалізації режиму CBC?*
- *Поясніть явище розмноження помилок в режимах зі зворотнім зв'язком. Чому в режимі OFB розмноження помилок не відбувається?*
- *Які алгоритми використовуються для поточного шифрування?*
- *Які режими використання блочного шифру можна запропонувати (крім 4 розглянутих в роботі)?*
- *Як відбувається виправлення помилок при розшифровуванні в різних режимах? Які режими є самосинхронними?*

Лабораторна робота №7.

Тема: Блочні алгоритми шифрування на основі DES

Мета роботи. Створити програму для шифрування файлів довільного типу вдосконаленими варіантами алгоритму DES, інтегрувати в створений раніше програмний пакет, оцінити швидкість, надійність та зручність апаратної та програмної реалізації.

Теоретичні відомості

Алгоритм DES розроблений майже 40 років тому, був рекомендований урядовим агенством США, і, незважаючи на недовіру і підозри, які він викликав у незалежних експертів, для свого часу був абсолютно надійним і найраціональнішим з відомих алгоритмів. Він набув поширення в комерційних та банківських системах, в системах зв'язку - мікросхеми DES вбудовані в багатьох мобільних телефонах.

Для аналізу і перевірки стійкості DES алгоритму були розроблені спеціальні теоретичні методи криптоаналізу - диференційний та лінійний, з допомогою яких можна розкрити окремі етапи чи неповні реалізації, однак до цього часу не існує практичних методів зламування повного алгоритму DES, крім повного перебору ключів.

Довжина блоку шифрування - 64 біти, довжина ключа - 56 бітів. Швидкодія сучасних комп'ютерів дозволяє з допомогою паралельних мережевих обчислень перебрати всі можливі 2^{56} ключів за декілька днів, а з допомогою спеціально побудованих суперкомп'ютерів навіть за декілька годин. Тому в наш час DES можна вважати недостатньо надійним. Більшість нових стандартів шифрування використовують повністю чи частково ідеї DES, відрізняючись більшою довжиною ключа.

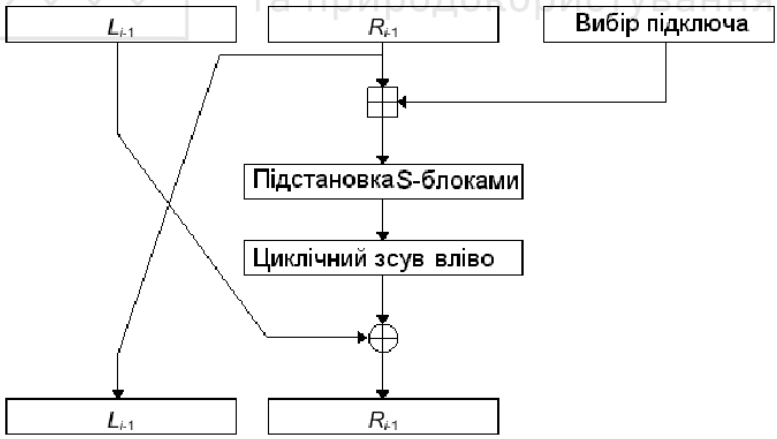

Рис. 17 Перетворення блока на одному етапі ГОСТ

Радянський стандарт, прийнятий в 1989 році, відомий під назвою ГОСТ 28147-89, аналогічний DES, але дещо простіший, що спрощує програмну реалізацію. Використовує 256-бітовий ключ

(замість 56) та 32 етапи (замість 16), що гарантує його надійність на багато десятиліть. Крім того, окремі деталі алгоритму (S-блоки) не розсекречуються, що є додатковою ключовою інформацією і збільшує криптостійкість. Для конкретних реалізацій можна генерувати власні S-блоки з допомогою генератора випадкових чисел. Використовується Державним банком Росії та в інших установах.

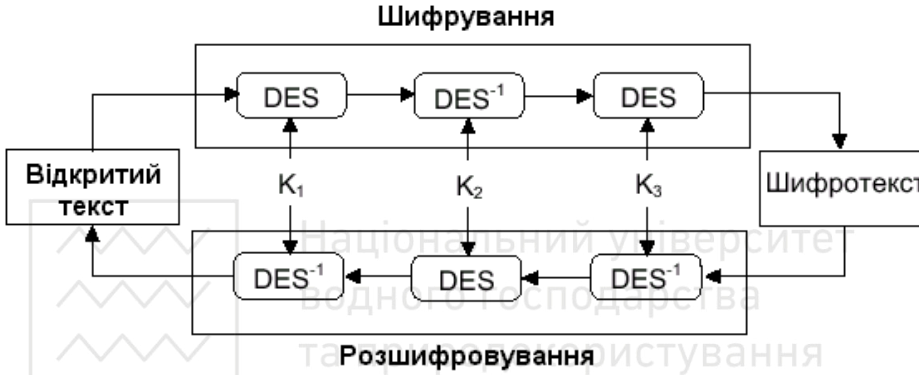

Рис.18 Потрійний DES

Потрійний DES.

Було запропоновано велику кількість варіантів подвійних та потрійних DES з подвоєною та потроєною довжиною ключа. Найбільш надійним алгоритмом на основі DES вважається потрійний DES, який використовує три блоки шифрування DES з різними ключами. Сумарна довжина ключа 3x56=168 бітів. Недолік - час шифрування втричі більший. Достоїнство - можна використати поширені на ринку мікросхеми для звичайного DES. Для оборотності алгоритму на середньому етапі використовується зворотній хід звичайного алгоритму DES.

$$DES3_{k1,k2,k3}(x) = DES_{k3}(DES^{-1}_{k2}(DES_{k3}(x)))$$

```
procedure DES3_alg(
            Dir_or_back:Direction; //напрям шифрування
            start_block64,          //вхідний блок
            key1,key2,key3  : block64bits;// 3 ключі
     var end_block64 :block64bits);// результуючий блок
var
    first_block64,second_block64 :block64bits;
    Reverse : Direction ;
begin
DES_alg(Dir_or_back, start_block64, key1,
                     first_block64 );
  if Dir_or_back=Direct then Reverse:=Back
                        else Reverse:=Direct;
  DES_alg(Reverse,  first_block64, key2,
                     second_block64 );
   DES_alg(Dir_or_back,  second_block64, key3,
                     end_block64 );
end;
```

DESX (DES eXtended). Розширений DES, запропонований Роном Рівестом в 1984 році, крім 56-бітового ключа, використовує ще два 64-бітових

$$DESX_{k,k1,k2}(x) = k_2 \text{ XOR } DES_k(k_1 \text{ XOR } x)$$

Довжина ключа 56+64+64=184 біти, k_1 - перший зашумлюючий ключ, k_2 - завершальний зашумлюючий ключ. Цей алгоритм простий, ефективно реалізується апаратно, може

використовувати стандартні реалізації DES, збільшує стійкість до атак на перебір ключів.

Рис.19 Головна форма проекту

Хід роботи

1. Відкрийте проект з попередньої лабораторної роботи в Delphi, доповніть кнопками Button10 - Button13 згідно рис.19.

2. Реалізуйте в обробниках натискання алгоритми шифрування файлів потрійним DES та розширеним DESX. Модифікуйте програму для послідовного введення трьох різних 8-байтових ключів чи одного довгого 24-байтового.

3. Запропонуйте власний варіант розширення алгоритму DES з подовженим ключем. Опишіть детально алгоритм в словесній формі та у вигляді підпрограми мовою програмування Паскаль чи Сі.

Контрольні запитання.

- *Які недоліки властиві для режиму електронної записної книжки (прямої заміни)?*
- *Які етапи алгоритму DES не впливають на надійність шифрування? Чи можна їх пропустити?*
- *Які методи криптоаналізу можна застосувати (теоретично) для зламування текстів, зашифрованих алгоритмом DES?*
- *Чи змінює алгоритм DES частотні характеристики тексту?*
- *Запропонуйте свій алгоритм шифрування останньої неповної порції тексту.*
- *В комбінації з якими іншими відомими Вам методами можна використовувати розглянутий в роботі алгоритм шифрування?*
- *Які недоліки в реалізації можуть ослабити надійність розглянутого алгоритму шифрування?*
- *Який тип файлів використовується в алгоритмі? Чому запропонований алгоритм шифрує файли будь-якого типу?*

Завдання для самостійної роботи.

1. Продемонструвати лавинний ефект у DES: написати програму, що обчислює відстань Хемінга для змін у тексті й у ключі на кожному кроці шифрування і виводить результати у вигляді таблиці чи графіка.

2. Довести, що якщо при використанні DES замінити будь-який біт на його доповнення у відкритому тексті і ключі, то відбудеться аналогічна заміна в закритому тексті.

3. Довести, що перестановка П(m): 0, 2, ... , 2^n-1 (взаємно однозначне відображення n-бітових цілих чисел у себе) у більше, ніж 60% випадків, має нерухому точку.

4. Довести, що розшифровування в DES співпадає з шифруванням (ключі в зворотньому порядку). Показати, що в DES перші 24 біти будь-якого підключа вибираються з однієї 28-бітової підмножини вихідного ключа, а інші - з іншої, непересічної з ним, 28-бітної підмножини цього ключа.

5. Скільки можливих ключів дозволяє використовувати шифр Плейфейра? (Представити у виді степеня двійки.)

6. Яка методика криптоаналізу шифру Хіла з обраним відкритим текстом?

7. Чому в алгоритмі Лемера 2^31-1 вибирається в якості m, хоча для 2^31, наприклад, операція порівняння реалізується простіше?

8. Показати, що в алгоритмі Лемера забезпечення повного періоду не гарантує якості генерованої псевдовипадкової послідовності. (Порівняти, наприклад, x_{n+1}=6x_n mod 13 і x_{n+1}=7x_n mod 13.)

9. Який максимальний період можна одержати для генератора: x_{n+1}=a x_n mod 24 (чому дорівнює a, які обмеження на x_0?)

10. Довести, що якщо m - просте і алгоритм Лемера з параметром a породжує послідовності максимального періоду, то при використанні параметра a^k також будуть породжуватися послідовності максимального періоду, якщо k менше m, а m-1 не ділиться націло на k. (Перевірити для x_0=1, m=1, a=3, 3^2, 3^3, 3^4.)

11. Чому дорівнює НСД (n, n+1)?

12. Скласти програму для частотного аналізу відкритих та зашифрованих файлів.

13. Скласти програму для перестановки всіх символів у файлі у зворотному порядку.

14. Розкласти на прості множники число 2^30+1.

15. Розкласти на прості множники число 2^30-1.

16. Розшифрувати перехоплену радіограму від Штірліца

У!ОМОДИДАТЇХПОСЯТЬЧЕХОЖЕДУ

Скласти програму для шифрування рядків тексту за таким самим алгоритмом. .

17. Скласти програму для генерації всіх перестановок із n символів.

18. Скласти програму для реалізації переможця конкурсу на найкращий стандарт блочного шифру - Rijndael.

Перелік питань до заліку з предмету "Захист інформації в комп'ютерних системах"

1. Основні методи криптоаналізу.
2. Сітка Фейстеля.
3. Стеганографія. Використання графічних, звукових та текстових файлів для прихованого передавання інформації.
4. Транспортне кодування.
5. Порівняння складності задач та алгоритмів.
6. Система з відкритими ключами PGP (Pretty Good Privacy).
7. Комп'ютерні злочини та засоби їх запобігання.
8. Алгоритм XOR та його криптоаналіз.
9. Найпростіші алгоритми архівації. Алгоритми RLE, Лемпеля-Зіва.
10. Алгоритм шифрування з відкритим ключем RSA .
11. Алгоритм оптимального стискання інформації Хаффмана.
12. Потоковий алгоритм шифрування DES.
13. Ймовірносні тести простоти чисел.
14. Технологія цифрового підпису.
15. Генератори випадкових і псевдовипадкових чисел. Використання їх для шифрування.
16. Відкритий обмін ключами за алгоритмом Деффі-Хелмана..
17. Короткий історичний огляд розвитку криптографії .
18. Переможець конкурсу на найкращий стандарт блочного шифру -Rijndael.
19. Електронний цифровий підпис.
20. Класифікація комп'ютерних вірусів за шкідливістю, механізмом поширення та особливостями алгоритмів.
21. Поняття інформаційної безпеки. Основні терміни і визначення.
22. Що таке дескриптор вірусу? Яку інформацію про вірус він містить?
23. Поняття про технічні, інформаційні та організаційні методики захисту.
24. Назвіть основні форми проявів комп'ютерних вірусів?
25. Що таке сигнатура вірусу?
26. Причини порушення інформаційної безпеки в сучасних комп'ютерних системах.
27. Які типи вірусів можна шукати за сигнатурами?
28. Функції захисту. Систематизація вимог.

29. Опишіть структуру і процес завантаження Сом-файлу.
30. Систематизація методик захисту.
31. Опишіть послідовність дій, виконуваних вірусами при зараженні Сом-файлів.
32. Поняття про політику безпеки і моделі безпеки.
33. Опишіть алгоритм завантаження Exe-файла.
34. Методи захисту даних.
35. Опишіть можливий сценарій зараження Exe-файла.
36. Систематизація криптографічних методів.
37. Поняття про комп'ютерні віруси. Види вірусів.
38. Технічні канали витоку інформації.
39. Правові основи захисту інформації. Основні закони і норми.
40. Схема Шенона.
41. Симетричні криптосистеми.
42. Перестановки і підстановки.
43. Одноалфавітні і многоалфавітні криптосистеми.
44. Потокові і блокові шифри.
45. Модулярні шифри.
46. Шифри Віжинера.
47. Автоматичний вибір ключа і біжучого ключа.
48. Шифри Плейфейера, Хіла, одноразового блокнота.
49. Барабанні шифри.
50. Сітка Фейстеля. Дифузія і конфузія (Розсіювання і перемішування). Структура. Алгоритм дешифрування.
51. DES. Лавинний ефект. Надійність. Криптоаналіз.
52. Режими роботи DES. Зчеплення блоків. Шифрований зворотній зв'язок. Подвійний і потрійний DES. Інші симетрично-блокові шифри.
53. Канальне і наскрізне шифрування в мережах передачі даних. Розподіл ключів.
54. Генерування випадкових чисел. Алгоритм Лемера.
55. Прості числа. Основна теорема арифметики. Теорема Евкліда про існування нескінченної множини простих чисел. Теорема про проміжки між простими числами.

56. Мала теорема Ферма.
57. Теорема Ейлера.
58. RSA: основні елементи криптосистеми. Шифрування і розшифровування.
59. Цифровий підпис.
60. Дискретні логарифми.
61. Схема обміну ключами Діфі-Хелмана та її модифікації.
62. Схема Ель Гамаля.
63. Криптосистеми, що базуються на задачі про рюкзак.
64. Греко-китайська теорема про залишки та її застосування.
65. Складність обчислень. Гіпотеза P=NP та стійкість криптосистем.
66. Односторонні функції. Гіпотеза про існування односторонніх функцій та стійкість криптосистем.
67. Псевдовипадкові генератори.
68. Інтерактивні системи доведення з нульовим розголошенням.
69. Модель протоколу з нульовим розголошенням.
70. Протокол, що базується на ізоморфізмі графів. Аутентифікація.
71. Невідслідковуваність і затінюючий підпис.

Додаток 5

Англо-український словник криптографічних термінів

Англійські терміни	*переклад на українську*
adaptive chosen-ciphertext attack	адаптивна атака з вибором шифротексту
adaptive chosen-message attack	адаптивна атака з вибором повідомлень
adaptive chosen-plaintext attack	адаптивна атака з вибором відкритого тексту
assign	призначити
asymmetric cryptosystem, public-key cryptosystem	асиметрична криптосистема, криптосистема з відкритим ключем
back	зворотній
birthday paradox	парадокс днів народження
breaking (cracking) a cipher	розкриття шифру
breaking (cracking) cryptosystem	розкриття криптосистеми

breaking (cracking) cryptographic protocol	розкриття криптографічного протоколу
cardinal number	натуральне число
chosen-ciphertext attack	проста атака з вибором шифротексту
chosen-message attack	проста атака з вибором повідомлень
chosen-plaintext attack	атака з вибором відкритого тексту
chosen-text attack	атака з вибором тексту
cipher	шифр
ciphertext	шифротекст
ciphertext-only attack	атака з відомим шифротекстом
collision	коллізія
common secret key	спільний секретний ключ
completeness	повнота
compression function	функція стискання
cryptoanalysis	криптоаналіз
cryptoanalyst	криптоаналітик
cryptographic protocol	криптографічний протокол
cryptography	криптографія
cryptology	криптологія
direction	напрям
deciphering	дешифрування
decryption	дешифруваннння
directed chosen-message attack	направлена атака з вибором повідомлень
discrete logarithm	дискретний логарифм
discrete logarithm problem	задача дискретного логарифмування
encryption	шифрування, зашифровування
execute	виконувати
gauge	шкала, покажчик
generic chosen-message attack	проста атака з вибором повідомлень

hide	приховувати
identification protocol	протокол ідентифікації абонента
illegal user	незаконний користувач
integer factorization problem	задача факторизації цілих чисел
integrity	цілісність
interactive protocol	інтерактивний протокол
judge	арбітр
key	ключ
key agreement (exchange) protocol	протокол розподілу ключів
key generation algorithm	алгоритм генерації ключів
key distribution (sharing) protocol	протокол розподілу ключів
key space	простір ключів
known-message attack	атака з відомими повідомленнями
known-plaintext attack	атака з відомим відкритим текстом
last	останній
legal user	законний користувач
merge	злити, об'єднати
message space	простір повідомлень
modal	формальний, модальний
one-time pad	одноразовий блокнот
one-way function	одностороння функція
one-way trapdoor function	функція із секретом
partial cracking	часткове розкриття
permutation, transposition	перестановка
plain text	відкритий текст
prime number	просте число
privacy	конфіденційність
private key	секретний ключ
private-key cryptosystem	криптосистема із секретним ключем, симетрична криптосистема

prover	той, хто доводить
pseudorandom generator	генератор псевдовипадкових послідовностей
public key	відкритий ключ
refresh	оновити
reverse	обернений, протилежний
secrecy	конфіденційність
secret key	секретний ключ
secret-key cryptosystem	криптосистема із секретним ключем
secure protocol (scheme)	стійкий, надійний протокол (схема)
security	стійкость, надійність
select	вибирати
sender	відправник
session key protocol	протокол генерації сеансових ключів
signature generation algorithm	алгоритм генерації підписів
signature verification algorithm	алгоритм перевірки підписів
simulator	моделююча машина
smart card	інтелектуальна картка
soundness	коректність
substitution	підміна
symmetric cryptosystem	симетрична криптосистема
threat	загроза
total breaking	повне розкриття
trusted authority, trusted center	центр довіри
valid signature	допустимий підпис коректний підпис
validator	валідатор
verification algorithm	алгоритм перевірки підписів
verifier	перевіряючий
zero-knowledge	нульове розголошення

Матеріали в Інтернеті

(Більшість розміщені на серверах www.cryptography.ru та www.kiev-security.org.ua **,** пошукові системи www.google.com, www.yandex.ru)

1. Шеннон К., Теория связи в секретных системах, в кн.: Шеннон К. Э., Работы по теории информации и кибернетике, М.: ИЛ, 1963.
2. Жельников В. Криптография от папируса до компьютера.
3. Баричев С. Основы криптографии.
4. Сапегин, Вегнер. Защита программного обеспечения персональных ЭВМ.
5. Menezes A., van Oorschot P., Vanstone S. , "Handbook of Applied Cryptography" , CRC press, 1996.

Література

1. Алферов А. П, Зубов А. Ю., Кузьмин А. С., Черемушкин А. В. Основы криптографии. Учебное пособие. 2-е изд., доп. М.: Гелиос АРВ, 2002.
2. Шнайер Б. Прикладная криптография. М.: Триумф, 2002.
3. Столлингс В. Криптография и защита сетей: принципы и практика. М.: Изд. дом Вильямс, 2001.
4. Введение в криптографию. /Под общ. ред. В.В. Ященко. - М.: МЦНМО: "ЧеРо", 1999.
5. Молдовян Р.А., Молдовян Н.А., Советов Б.Я. Криптография. - СПб: Лань, 2001.
6. Чмора А., Современная прикладная криптография, Гелиос АРБ, 2001.
7. Сидельников В. М., Черепнев М. А., Ященко В. В., Системы открытого распределения ключей на основе некоммутативных полугрупп, Доклады РАН, 1993, т. 332, № 5.
8. Столлингс В. , "Криптография и защита сетей" , М: Вильямс, 2001.
9. В.Г.Проскурин, С.В.Крутов, И.В.Мацкевич: "Защита в операционных системах", М: Радио и связь: 2000.
10. Аграновский А.В., Хади Р.А., Ерусалимский Я. М., "Открытые системы и криптография" , Телекоммуникации, 2000.
11. А.В.Аграновский, А.В.Балакин, Р.А.Хади, "Классические шифры и методы их криптоанализа", М: Машиностроение, Информационные технологии, №10, 2001.

12. С.Расторгуев, "Программные методы защиты информации в компьютерах и сетях", М:Издательство Агентства "Яхтсмен", 1993.
13. А. Ростовцев, "Алгебраические основы криптографии", СПб: Мир и Семья, 2000.
14. Устинов Г.Н. , "Основы информационной безопасности" , М: Синтег, 2000.
15. Анин Б. , "Защита компьютерной информации" , СПб: БХВ, 2000.
16. Романец Ю.В., Тимофеев П.А. , "Защита информации в компьютерных системах и сетях" , М: Радио и связь, 2001.
17. Саломаа А.: "Криптография с открытым ключом", Москва: "Мир", 1995. - 318с.
18. Олифер В., Олифер Н.: "Компьютерные сети", Спб: Издательство "Питер", 1999. - 672с.
19. С. Мафтик, "Механизмы защиты в сетях ЭВМ", изд. Мир, 1993 г.
20. В. Ковалевский, "Криптографические методы", Компьютер Пресс 05.93 г.
21. В. Водолазкий, "Стандарт шифрования ДЕС", Монитор 03-04 1992 г.
22. С. Воробьев, "Защита информации в персональных ЗВМ", изд. Мир, 1993 г.
23. Мэсси Жд. Л. Введение в современную криптологию. ТИИЭР, 1988, Т. 76, N 5, С. 24--42.
24. Погорелов Б. А., Черемушкин А. В., Чечета С. И. К вопросу о терминологии, используемой в криптографии. Вестник Томского университета. Приложение. Материалы научных конференций, симпозиумов, школ, проводимых в ТГУ. 2003, N 6, 53--57.
25. Stinson D. R. Cryptography: Theory and practice. CRC Press, N.Y., 1995.
26. Скембрей Дж, Мак-Клар С., Курц Дж. Секреты хакеров. Безопасность сетей - готовые решения. М.: Изд. дом Вильямс, 2001.
27. Акритас. А. Основы компьютерной алгебры. М.: Мир, 1994
28. Кузьминов Т. В. Криптографические методы защиты информации. Новосибирск. Институт систем информатики СО РАН, 1998.
29. Гнеденко Б. В. Основы теории вероятностей. М., 1989.
30. Медведовский И.Д., Семьянов П.В., Платонов В.В., "Атака через Интернет ", СПб: 1999.

31. Милославская Н.Г., Толстой А.И. , "Интрасети: доступ в Интернет , защита" , М.: ЮНИТИ-ДАНА, 2000.
32. Gutmann P.: "Network Security", University of Auckland, 1996.
33. Koblitz N., Algebraic Aspects of Cryptography, Springer, 1997.
34. Beker H., Piper F., Cipher System, Northwood Books, 1982.
35. Cryptology and computational number theory, Proc. of Symp. in Appl. Math., v. 42, 1990.
36. Luby M., Pseudorandmness and cryptographic applications, N.Y., Princeton Univ. Press, 1996.
37. Защита информации. ТИИЭР, Т.78, 5, 1988.
38. Ноден П., Китте К. Алгебраическая алгоритмика (с упражнениями и решениями). - М.: Мир, 1994

Методичні вказівки та матеріали

1. Методичні вказівки для виконання лабораторних та самостійних робіт з дисципліни "Основи захисту та кодування інформації" студентами спеціальності 7.080200, 8.080200 "Прикладна математика". Частина І. / П. В. Ольшанський, - Рівне: УДУВГП, 2004, -52 с.

2. Методичні вказівки для виконання лабораторних та самостійних робіт з дисципліни "Основи захисту та кодування інформації" студентами спеціальності 7.080200, 8.080200 "Прикладна математика". Частина ІІ./ П. В. Ольшанський, - Рівне: УДУВГП, 2004, -60 с.

3. Методичні вказівки для виконання лабораторної роботи «Боротьба з комп'ютерними вірусами» /Ольшанський П.В.- Рівне: УДАВГ, 1998.

4. Методичні вказівки для виконання лабораторної роботи «Архіватори» /Ольшанський П.В.- Рівне: УДАВГ, 1998.

5. Англо-український словник комп'ютерних термінів/П.В. Ольшанський, - Рівне: УДУВГП, 2002.

6. Методичні вказівки "Робота в операційних системах сімейства Microsoft Windows" /П.В.Ольшанський, -Рівне: УДУВГП, 2002.

7. Методичні вказівки "Розв'язування задач в системі програмування Borland (Turbo) Pascal 7.0 /П.В.Ольшанський, -Рівне: РДТУ, 1999.