# ЛАБОРАТОРНА РОБОТА № 9

# РЕЖИМИ ШИФРУВАННЯ БЛОКОВИХ ШИФРІВ

**Мета роботи:** ознайомитися з програмною реалізацією алгоритму симметричного блокового шифрування RIJNDAEL. Розробити програми його використання у різних режимах шифрування.

**Використовуване програмне забезпечення:** середа розробки GNU Octave, Python.

## Теоретичні відомості

* + 1. **Опис алгоритму шифрування RIJNDAEL**

На цей час в Україні, як і в Росії діє міждержавний стандарт симетричного шифрування ГОСТ 28147-89. В 1997 році національний інститут стандартів і технологій США (NIST) об’явив про початок програми по прийняттю нового стандарту криптографічного захисту для закриття важливої інформації урядового рівня. Фіналістом конкурсу став RIJNDAEL, AES (Advanced Encryption Standart) Федеральний стандарт шифрування США затверджений міністерством торгівлі США як стандарт 4 грудня 2001 року. Місце розробки 1997 рік, Бельгія. Автори Йоан Дамен (Joan Daemen) Винсент Раймен (Vincent Rijnmen).

Алгоритм симетричного блокового шифрування RIJNDAEL становить особливий інтерес як новий американський стандарт криптографічного захисту – стандарт ХХІ в. для закриття важливої інформації урядового рівня, що прийшов на заміну існуючому з 1974р. алгоритму DES, найпоширенішому криптоалгоритму у світі. В алгоритмі RIJNDAEL не виявлено слабостей у захисті.

Його відрізняють:

* + висока ефективність на будь-яких платформах;
	+ високий рівень захищеності;
	+ шифр добре підходить для реалізації в smart-картах через низькі вимоги до пам'яті;
	+ швидка процедура формування ключа;
	+ гарна підтримка паралелізму на рівні інструкцій;
	+ підтримка різних довжин ключа з кроком у 32 біта. Параметри шифру RIJNDAEL наведені в табл. 2.1.

Таблиця 2.1 – Параметри шифру RIJNDAEL

|  |  |
| --- | --- |
| pозмір блока, біт | 128, 192, 256 |
| pозмір ключа, біт | 128, 192, 256 |
| число раундів | 10, 12, 14 |
| pозмір ключового елемента, біт | 128, 192, 256 (дорівнює розміру блока) |
| число ключовых елементів | 11, 13, 15 (на 1 більше числараундів) |

Кількість раундів шифрування залежить від розмірів ключа та блока.

Таблиця 2.2 – Кількість раундів шифрування

|  |  |  |  |
| --- | --- | --- | --- |
| Розмір ключа | 128 | 192 | 256 |
| Розмір блока |
| 128 | 10 | 12 | 14 |
| 192 | 12 | 12 | 14 |
| 256 | 14 | 14 | 14 |

Один раунд шифрування на псевдокоді виглядає наступним

чином:

Round (State,RoundKey)

{

ByteSub(State); ShiftRow(State); MixColumn(State);

AddRoundKey(State,RoundKey);

}

Останній раунд шифрування на псевдокоді виглядає так: Round (State,RoundKey)

{

ByteSub(State); ShiftRow(State);

AddRoundKey(State,RoundKey);

}

У цьому криптоалгоритмі деякі операції виконуються над байтами, що розглядаються як елементи поля GF(28) . Елементами GF(28) є двійкові багаточлени степені N<8, що можуть бути задані рядком своїх коефіцієнтів. При такому представленні елементів поля додавання в полі GF(28) – це операція поразрядного XOR, а множення

– це звичайна операція множення багаточленів з узяттям результату

по модулю деякого незвідного двійкового багаточлена (*x*) , з

використанням операції XOR при приведенні подібних членів. У

*x* 1

RIJNDAEL обраний незвідний поліном показника 51.

1. *x*8 *x* 4 *x*3

Блок даних представляється у вигляді масиву значень.

Таблиця 2.3 – Представлення блоків даних

|  |  |  |  |
| --- | --- | --- | --- |
| *a*00 | *a*01 | *a*02 | *a*03 |
| *a*10 | *a*11 | *a*12 | *a*13 |
| *a*20 | *a* 21 | *a*22 | *a* 23 |
| *a*30 | *a*31 | *a*32 | *a*33 |

У процедурі ByteSub *aij* змінюється на *aij*1 в полі Галуа *GF* (28 ) .

*aij* \**aij*1

1(mod(*x*8 *x* 4

*x*3 *x*

1))

Афінне перетворення діє за правилом:

*Y C* \* *x*

0

*C*1 (mod(*x*8

1))

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *y*0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | *x*0 | 1 |
| *y*1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | *x*1 | 1 |
| *y*2 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | *x*2 | 0 |

*y*3 1



|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *y*4 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | *x*4 | 0 |
| *y*5 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | *x*5 | 1 |
| *y*6 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | *x*6 | 1 |
| *y*7 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | *x*7 | 0 |

1 1 1

0 0 0

1 \* *x*3

     

У процедурі ShiftRow останні три рядки стану циклічно зсовуються на різне число байтів (*C*1, *C*2, *C*3) у залежності від довжини блоку *Nb*.

Таблиця 2.4 – Зсув рядків

|  |  |  |  |
| --- | --- | --- | --- |
| *Nb* | *C*1 | *C*2 | *C*3 |
| 4 | 1 | 2 | 3 |
| 6 | 1 | 2 | 3 |
| 8 | 1 | 3 | 4 |

У процедурі MixColumn стовбці стану розглядаються як

багаточлени над

*GF* (28 )

і помножуються на багаточлен

*g*(*x*)

'03' *x*3

'01' *x*2

'01' *x*

'02' за модулем (*x* 4

1) :

.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *b*0 | 02 | 03 | 01 | 01 | *a*0 |
| *b*1 | 01 | 02 | 03 | 01 \* *a*1 |
| *b*2 | 01 | 01 | 02 | 03 | *a*2 |
| *b*3 | 03 | 01 | 01 | 02 | *a*3 |

Загальне число бітів раундових ключів дорівнює довжині блоку, помноженої на число раундів, плюс 1 (наприклад, для довжини блоку 128 біт і 10 циклів буде потрібно 1408 біт циклового ключа). Ключ шифрування розширюється в розширений ключ. Раундові ключі беруться з розширеного ключа в такий спосіб: перший раундовий ключ містить перші *Nb* слів, другий – наступні *Nb* слів і т.д.

* + 1. **Режими шифрування**

Під режимом шифрування розуміється такий алгоритм застосування блокового шифру, що при відправленні повідомлення дозволяє перетворювати відкритий текст у шифротекст і, після передачі цього шифротексту по відкритому каналу однозначно відновити первісний відкритий текст. Як видно з визначення, сам блоковий шифр тепер є тільки частиною іншого алгоритму – алгоритму режиму шифрування. Це обумовлено тим, що блоковий шифр працює тільки з окремим блоком даних, у той час як алгоритм

режиму шифрування має справу вже з цілим повідомленням, що може складатися з деякого числа *n* блоків. Більш того, повідомлення взагалі не зобов'язане складатися з блоків, у тім змісті, що це повідомлення не завжди можна розбити на ціле число *n* блоків. У цьому випадку в різних режимах шифрування приходитися доповнювати повідомлення різною кількістю біт.

Передбачено можливість функціонування блокових шифрів у наступних режимах шифрування:

* режим електронна кодова книга (ECB);
* режим зціплення блоків шифротексту (CBC);
* режим зворотного зв’язку по шифротексту (CFB);
* режим зворотного зв’язку по виходу (OFB);
* режим шифрування з лічильником (Counter).

На рисунках 2.1 – 2.5 приведені схеми шифрування в цих режимах. Використані наступні позначення:

Pі – і-й блок відкритого тексту;

Ek – функція зашифрування на ключі k; Cі – і-й блок зашифрованого тексту;

Dk – функція розшифрування на ключі k; Si – і-те значення лічільнику;

S0 – синхропосилка.



Рисунок 2.1 – Режим електронна кодова книга (ECB)



Рисунок 2.2 – Режим зціплення блоків шифротексту (CBC)



Рисунок 2.3 – Режим зворотного зв’язку по шифротексту (CFB)



Рисунок 2.4 – Режим зворотного зв’язку по виходу (OFB)



Рисунок 2.5 – Режим шифрування з лічильником (Counter)

Розглянуті режими шифрування представлені в документі *Recommendation for Block Cipher Modes of Operation. NIST Special Publication 800-38A. Technology Administration U.S.Department of Commerce. 2001 Edition,* що був виданий NIST США і має назву

«Рекомендації для режимів шифрування з блоковим шифром».

## Завдання на лабораторну роботу

* + 1. Розробити програму використання алгоритму RIJNDAEL в наступних режимах шифрування:
* режим електронна кодова книга (ECB);
* режим зціплення блоків шифротексту (CBC);
* режим зворотного зв’язку по шифротексту (CFB);
* режим зворотного зв’язку по виходу (OFB);
* режим шифрування з лічильником (Counter).

## Зміст звіту

* + 1. Титульний лист, тема і мета роботи.
		2. Тексти програм.
		3. Результати роботи програм.
		4. Висновки.

## Контрольні питання

* + 1. З яких операцій складається один раунд алгоритму RIJNDAEL?
		2. Поясніть роботу алгоритму в режимі електронна кодова книга (ECB).
		3. Поясніть роботу алгоритму в режимі зціплення блоків шифротексту (CBC).
		4. Поясніть роботу алгоритму в режимі зворотного зв’язку по шифротексту (CFB).
		5. Поясніть роботу алгоритму в режимі зворотного зв’язку по виходу (OFB).
		6. Поясніть роботу алгоритму в режимі шифрування з лічильником (Counter).
		7. Чому шифри RIJNDAEL називаються блоковими шифрами?
		8. Якими можуть бути довжина ключа и блока в RIJNDAEL?
		9. В чому полягає основний шаг криптосистем RIJNDAEL?
		10. Перечисліть основні режими шифрування блокових шифрів.
		11. Які основні параметри шифрів RIJNDAEL?