УДК 004.056.055

В. А. ЛУЖЕЦЬКИЙ, А. В. ОСТАПЕНКО

Вінницький національний технічний університет, Вінниця

АНАЛІЗ АЛГОРИТМІВ СИМЕТРИЧНОГО БЛОКОВОГО ШИФРУВАННЯ

Анотація. Проведено аналіз алгоритмів симетричного блокового шифрування з точки зору способу реалізації перетворення вхідних даних у шифротекст. Сформульовано рекомендації для вибору перетворень, що забезпечать підвищення швидкості шифрування.

Ключові слова: симетричні блокові шифри (СБШ), криптографія, шифр, шифрування.

Аннотация. Проведен анализ алгоритмов симметричного блочного шифрования с точки зрения способа реализации преобразования входных данных в шифротекст. Сформулированы рекомендации для выбора преобразований что обеспечат повешение скорости шифрования.

Ключевые слова: симметричные блочные шифры (СБШ), криптография, шифр, шифрование.

Abstract. We analyzed the symmetric block cipher algorithms, depending on the method of converting input data into ciphertext. We proposed recommendations for selection of transformations that will provide increasing speed of encryption.

Key words: symmetric block cipher, cryptography, cipher, encryption.

Вступ

Кожний вид інформації має свої специфічні особливості, що суттєво впливають на вибір методів її шифрування. Велике значення відіграють об’єм та необхідна швидкість передачі даних. Потреба вирішення проблеми захисту електронної інформації обумовлює актуальність розробки шифрів, як одного із видів криптографічних перетворень, що використовують для захисту інформації в комп’ютерних системах та мережах.

Актуальність

На сьогоднішній день переважна більшість стійких криптосистем реалізована на основі симетричних блокових шифрів (СБШ). Алгоритми шифрування СБШ реалізують принцип, який полягає в тому, що багатократно виконується перетворення блоку даних з використанням секретного ключа шифрування. За підходами до реалізацій цього перетворення виділяють блокові шифри побудовані на основі мереж Фейстеля, чергування процедур перестановок і підстановок (SP-мереж), структури «квадрат» (Square) [1] та операцій за модулем [2]. Оскільки всі вони мають майже однакові характеристики швидкості шифрування, стійкості до відомих атак, тому важливо вибрати блоковий шифр, який найбільш повно враховує наявні у розробників систем захисту програмні та апаратні засоби.

Мета

Метою статті є аналіз сучасних симетричних блокових шифрів для вибору перетворень, що забезпечать підвищення швидкості шифрування.

Постановка задач

Як правило, алгоритми зашифрування та розшифрування СБШ є ітераційними і складаються з послідовності перетворень (рис. 1).

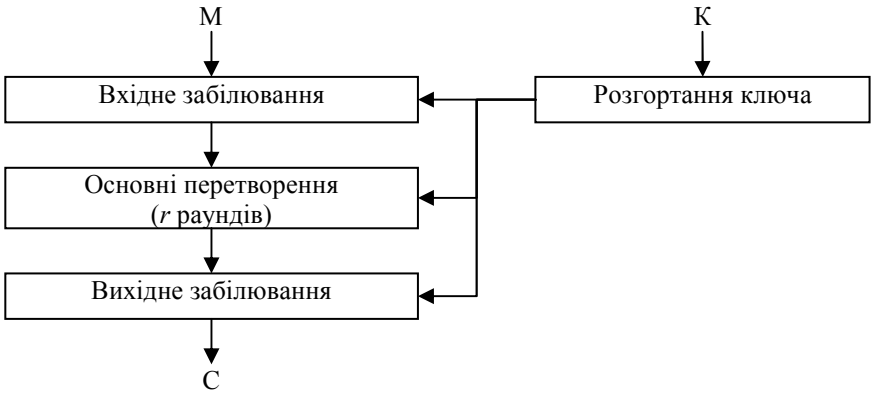

Рисунок 1 – Загальна схема перетворення даних у СБШ

В кожному раунді використовується окремий раундовий ключ, який отримується як результат процедури розгортання секретного ключа. Задачею даної процедури є формування необхідної кількості раундових ключів із секретного ключа, обмеженого розміру. Найпростіша її реалізація передбачає розбиття ключа на частини, які по черзі використовуються у раундах. Для зменшення розміру ключа дана процедура будується як складне багатоетапне перетворення, що модифікує секретний ключ [1]. Процедури

вхідного та вихідного забілювання змішують вхідні дані M і результат перетворення після r-го раунду з ключовою інформацією, шляхом використання операцій побітового додавання, XOR.

Основні перетворення СБШ можуть бути представлені певною кількістю послідовних використань слабкого блокового шифру, що називається раундом перетворення. Ці перетворення описуються однією і тією ж функцією, але в якості аргументів якої використовуються результат попереднього перетворення і відповідний раундовий ключ. Узагальнена схема раундового перетворення СБШ представлена на рис. 2, де k_i – раундовий ключ; m_i – блок вхідного повідомлення; F – функція раундового перетворення; c_i – блок криптограми після i-го раунду перетворення.

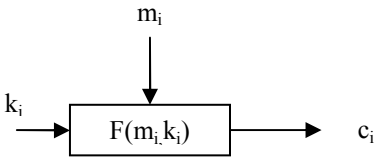

Рисунок 2 – Узагальнена схема раундового перетворення СБШ

В загальному випадку, функція раундового перетворення є послідовністю залежних від ключа нелінійних замін, «міксуючих» перестановок та циклічних зсувів [3].

Відповідно до способів комбінування секретних систем [4] для отримання нової секретної системи, основні перетворення даних у СБШ можуть бути представлені як добуток секретних систем:

$$S = \prod_{i=1}^{n} T_i, \tag{1}$$

де S – комбінована секретна система; T_i – i-а секретна система (раунд перетворення) з набору n секретних систем.

Тобто, використання сукупності слабких раундів утворюють стійкий шифр [4]. Оскільки набір і послідовність виконання операцій в раундах є незмінними, криптографічна стійкість розглянутих СБШ залежить від розміру секретного ключа, складності виконуваних операцій або кількості раундів перетворення, у разі використання простих операцій. Такий спосіб побудови перетворень СБШ значно полегшує як їх реалізацію (апаратну та програмну), так і їх криптографічний аналіз, що дозволяє будувати атаки на меншу кількість раундів, згодом розширюючи їх [1].

Незалежно від способу реалізації основних перетворень сучасні СБШ, мають задовольняти таким загальноприйнятим вимогам:

1. Знання алгоритму не повинно зменшувати крипостійкість шифру (правило Керкоффа) [4].
2. Стійкість алгоритму шифрування повинна залежати тільки від секретного ключа [4].
3. Використання довжини блоку не менше 128 біт (2×64 біт) [1]. Хоча розмір блоку 64 біт є на сьогодні безпечним, але зважаючи на швидкий ріст можливостей елементної бази, вже в недалекому майбутньому можливі вдалі реалізації певних видів атак.
4. Можливість використання різного розміру секретного ключа (128, 192, 256 біт) [1].

Недоліки алгоритмічної реалізації основних перетворень СБШ можуть призводити до зменшення їх криптографічної стійкості та зменшення швидкості процедур зашифрування, розшифрування. Тому, постає задача аналізу СБШ з точки зору способу реалізації перетворення вхідних даних у шифротекст для вибору перетворень, що забезпечать підвищення швидкості шифрування. У переважній більшості СБШ основні перетворення побудовані на основі мереж Фейстеля, чергування процедур перестановок і підстановок (SP-мереж), структури «квадрат» та операцій за модулем. Розглянемо детальніше ці підходи.

Блокові шифри на основі мереж Фейстеля

Відомі мережі Фейстеля з двома [1] та чотирма гілками [5]. В мережі Фейстеля на дві гілки оброблюваний блок розбивається на два підблоки L, R. Для кожного раунду обчислюється:

$$L_i = R_{i-1} \oplus F\left(L_{i-1}, k_{i-1}\right),\ R_i = L_{i-1}.$$

Результатом виконання r раундів є блоки L_r, R_r зашифрованої інформації. На рис. 3 наведено типову схему раунду перетворення на основі мережі Фейстеля з двома гілками.

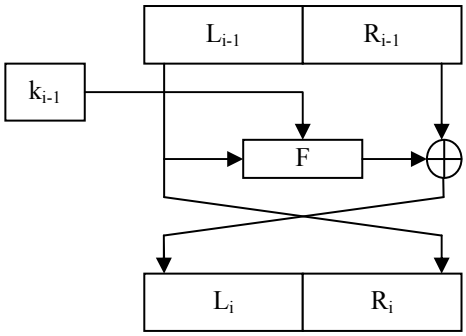

Рисунок 3 – Схема раунду перетворення на основі мережі Фейстеля з двома гілками

Основні кількісні характеристики сучасних СБШ на основі мережі Фейстеля з двома гілками [1] представленні в табл. 1.

Таблиця 1 – Характеристики СБШ на основі мережі Фейстеля з двома гілками

Алгоритм	Кількість раундів r	Довжина ключа K (біт)	Розмір блоку W (біт)
Blowfish	16	32-448	64
Camellia	18	128	128
	24	192/256	
CAST-128	12	40/64/80	64
	16	128	
CAST-256	48 (12×4)	128/160/192/224/256	128
DEAL	6	128/192	128
	8	256	
DES	16	56	64
DFC	8	128/192/256 (або 0-256)	128
ГОСТ-28147-89	32	256	64
E2	12	128/192/256	128
FEAL-N	4-32	64	64
Сімейство HPC	8	Без обмеження	Без обмеження
ICE	16	64	64
KASUMI	8	128	64
Khufu	8-64 (кратне 8)	64-512 (кратне 64)	64
LOKI97	16	128/192/256	128
Lucifer (4)	16	128	128
MAGENTA	6	128/192	128
	8	256	
MISTY1	4×n(8)	128	64
RC5	1-255	0-2040	32/64/128
TEA	64	128	64
XTEA	64	128	64

Аналіз табл. 1 дозволяє виділити такі особливості СБШ побудованих на основі мереж Фейстеля з двома гілками. У більшості шифрів задані конкретні фіксовані значення кількості раундів, довжини ключа та розміру блоку (DES, ICE, KASUMI, TEA та ін.), але в деяких (Blowfish та ін.) значення одного або декількох параметрів можуть бути без обмежень (Сімейство HPC) або варіюватись в межах кратності певному значенню (Khufu, MISTY1). Кількість раундів мінімально можлива від 4-х, 8-и і максимально до 32-х, 64-х. Шифр RC5 має діапазон можливих значень кількості раундів від 1 до 255 (рекомендований мінімум 4 раунди). Значення заданої довжини ключа від 40-64 до 512 біт, причому використання більшої довжини ключа також збільшує необхідну кількість раундів (Camellia, CAST-128, DEAL, MAGENTA). Переважна більшість СБШ передбачають використання блоків по 64 та 128 біт, лише Сімейство HPC не накладає обмежень на довжину блоку та шифр RC5 підтримує мінімальну довжину блоку 32 біт.

У кожному СБШ використовується деяка фіксована множина операцій, на базі якої і будується раундова функція, причому конкретний набір операцій суттєво впливає на швидкість процедур шифрування. Тому, проаналізуємо набір базових операції для СБШ, побудованих на основі мережі Фейстеля з двома гілками (табл. 2).

Таблиця 2 – Базові операції для СБШ на основі мережі Фейстеля з двома гілками

Алгоритм	Бієктивні математичні функції		Побітові циклічні зсуви (змінні/фіксовані)		Логічні		Таблична заміна	Перестановка (керована/ таблична)
	Дод. за mod2^N	Множ. за mod2^N	Вліво	Вправо	«І»	«АБО»		
Blowfish	(N=32)						4 табл.	
Camellia			(ф=1)		+	+	4 табл.	(т)
CAST-128	(N=32)	(N=32)	(з)				4 табл.	
CAST-256	(N=32)	(N=32)	(з)				4 табл.	
DEAL							8 табл.	(т)
DES							8 табл.	(т)
DFC	(N=64)	(N=64)					+	
ГОСТ-28147-89	(N=32)		(ф=11)				8 табл.	
E2		(N=32)	(ф=1)			+	+	(т)
FEAL-N	(N=8)		(ф=2)					
Сімейство HPC	(N=64)		(з/ф=2,8, 12,22,32)	(з/ф=4,5, 11,17,23)	+			
ICE							4 табл.	(к,т)
KASUMI			(ф=1)		+	+	2 табл.	
Khufu			(ф=8,24,16)				1-8 табл.	
LOKI97	(N=64)						2 табл.	(к,т)
Lucifer (4)							2 табл.	(т)
MISTY1					+	+	2 табл.	
RC5	(N=16-64)		(з)					
TEA	(N=32)		(ф=4)	(ф=5)				
XTEA	(N=32)		(ф=4)	(ф=5)				

Так серед бієктивних математичних функцій значно вживанішою є операція додавання за модулем 2^N причому значення N варіюється від 8 (FEAL-N) до 128 (RC5). Найбільш поширеними є побітові циклічні зсуви вліво на фіксовану кількість біт, лише шифри Сімейства HPC, TEA, XTEA використовують циклічний зсув вправо. Група шифрів для побудови раундової функції не використовує операції табличних замін та перестановок, при цьому шифруючи дані комбінацією бієктивних математичних функцій та операцій циклічного зсуву (FEAL-N, Сімейство HPC, RC5, XTEA). Найбільша кількість таблиць замін передбачена в шифрах DEAL, DES, ГОСТ-28147-89 та Khufu у разі виконання 64 раундів. Найменший набір базових операцій використаний у шифрах FEAL-N та RC5, а найбільший у шифрах Camellia та E2. Операції парної перестановки та XOR є незмінними базовими операціями мереж Фейстеля (рис. 3), тому вони за замовчуванням присутні в усіх представниках даного виду і не були внесені в табл. 2. Побітові циклічні зсуви можуть бути як з фіксованим значенням (Camellia, ГОСТ-28147-89 та ін.), так і з змінними значеннями, що залежать від ключа (RC5, CAST-256 та ін.). Аналогічно і з операцією перестановки яка може бути заданою таблицею (Camellia, Lucifer (4) та ін.), або ж бути залежною від певних параметрів (ICE, LOKI97). В шифрах CAST-128, CAST-256, TEA, XTEA та в Сімействі HPC використовується додавання і віднімання за mod 2N. Шифр E2 виконує операції (зсуву, заміни та перестановки) над байтами. Окрім циклічного зсуву в деяких шифрах (TEA, XTEA) виконується арифметичний зсув біт. У DES виконується розширююча перестановка, що розширює вхідне значення з 32 до 48 біт. Khufu для кожних 8 раундів передбачає генерування нової таблиці замін, а в LOKI97 таблиці замін обробляють вхідні дані різної розрядності (11 та 13 біт). В алгоритмі Lucifer(4) виконується керована секретним ключем операція табличної заміни, що визначає яка із двох таблиць замін буде використана у відповідному раунді. БШ MAGENTA не увійшов до табл. 2 оскільки його перетворення побудовані на комбінації лише операції піднесення до степеня твірного елемента поля GF(28) та XOR.

Більшість сучасних СБШ використовують мережу Фейстеля з двома гілками в якості основи [1]. Основні перетворення побудовані за допомогою даної мережі є оберненими, а процедури зашифрування та розшифрування відрізняються лише порядком використання раундових ключів, для шифрів з однаковою функцією шифрування, або порядком використання функцій шифрування при використанні гетеро-

генної структури (MARS, CAST-128, CAST-256). Це дозволяє легко реалізувати шифри як на програмному, так і на апаратному рівні, використовуючи один фрагмент апаратної схеми чи програмного коду, що забезпечує максимально компактну реалізацію і широкі можливості застосування [1].

Є три типи мереж Фейстеля на чотири гілки, що покладені в основу побудови СБШ (рис. 4), де $X_{1...}X_4$ – блоки вхідного повідомлення, $Y_{1...}Y_4$ – блоки шифротексту [1, 5].

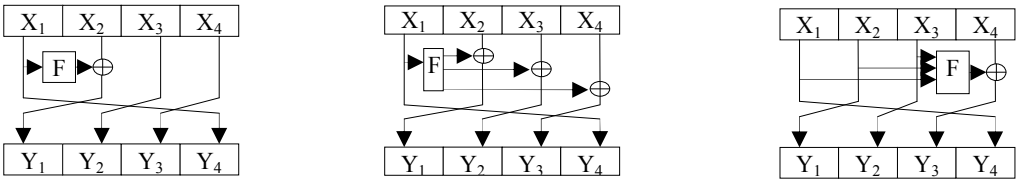

Рисунок 6 – Схема раунду перетворення на основі мереж Фейстеля з чотирма гілками:
а) тип 1; б) тип 2; в) тип 3

Кількісні характеристики СБШ на основі мереж Фейстеля з чотирма гілками [1] наведені в табл. 3.

Таблиця 3 – Характеристики СБШ на основі мереж Фейстеля з чотирма гілками

Алгоритм	Кількість раундів *r*	Довжина ключа *K* (біт)	Розмір блоку *W* (біт)	Тип структури
CLEFIA	18	128	128	1
	22	192		
	26	256		
IDEA	8+1	128	64	2
MacGuffin	32	128	64	3
MARS	32	128-1248	128	2
NUSH	36	128	64	3
	68	192	128	
	132	256	256	
RC2	16+2	8-1024	64	3
RC6	20	128/192/256	128	1
SPEED	>32 (кратне 4)	64-256 (кратне 16)	64/128/256	3
Skipjack	32	80	64	1
Twofish	16	128/192/256	128	1

Проаналізувавши табл. 3 можна відзначити такі особливості. Всі представники СБШ на основі мереж Фейстеля з чотирма гілками передбачають розбиття блоку вхідного повідомлення на 4 підблоки окрім шифру SPEED, що обробляє 8 підблоків. Переважна більшість використовує перший та третій тип структури, лише IDEA, MARS побудовані за принципом структури типу два. Кількість раундів зазвичай є фіксованим значенням, що збільшується відповідно до збільшення довжини ключа (CLEFIA, NUSH), лише в SPEED передбачено використання змінної кількості раундів у кількості не менше 32 (кратній 4). При чому шифр SPEED єдиний серед розглянутих у табл. 3 СБШ підтримує змінний розмір блоків. Максимальну довжину ключа дозволяють використовувати шифри MARS (1248 біт) та RC2 (1024 біт), мінімальну RC2 та SPEED 8 та 64 біт відповідно. Основний набір базових операцій для СБШ побудованих на основі мереж Фейстеля з чотирма гілками представлений в табл. 4. У шифрі Twofish також передбачене використання операції множення на матрицю даних, отриманих після операцій табличних замін. Шифр IDEA використовує множення за модулем (216+1). Операція побітового циклічного зсуву у шифрі NUSH виконується на змінну кількість біт, значення яких задано в таблиці і залежить від номеру раунда, також за схожим принципом почергово використовуються, в залежності від відповідної таблиці, логічні операції «І» та «АБО». Шифр RC2 в функції раундового перетворення використовує логічну операцію «НІ» (інверсія підблоку), та циклічний зсув вліво на змінну кількість біт, значення яких задано таблицею від 1 до 5. Конкретні значення N в операції додавання за модулем 2^N та фіксованого циклічного зсуву вправо у шифрах SPEED, NUSH залежать від розміру блоку, що обробляється. Аналіз табл. 4 дозволяє стверджувати, що найчастіше в СБШ побудованих на основі мереж Фейстеля з чотирма гілками використовується операція додавання за модулем 2^N, при цьому використання множення передбачено лише в RC6. Найбільша кількість базових операцій використовується в шифрах MARS, Twofish, а найменшу у IDEA.

Таблиця 4 – Базові операції для СБШ на основі мереж Фейстеля з чотирма гілками

Алгоритм	Бієктивні математичні функції		Побітові циклічні зсуви (змінні/фіксовані)		Логічні		Таблична заміна	Перестановка (керована/ таблична)
	Дод. за mod2^N	Множ. за mod2^N	Вліво	Вправо	«І»	«АБО»		
IDEA	(N=16)	+						
MacGuffin							8 табл.	(т)
MARS	(N=32)		(з/ф=13,5)	(ф=8)			3 табл.	
NUSH	(N=16-64)			(з)	+	+		
RC2	(N=16)		(з=1-5)		+			(т)
RC6	(N=32)	(N=32)	(з,ф=5)					
SPEED	(N=8/16/32)			(з,ф=3/7/15)	+			
Skipjack	(N=10)						+	
Twofish	(N=32)		(ф=1,8)	(ф=1)			4 табл.	

В роботі [5] були наведені ряд переваг СБШ побудованих на основі мереж Фейстеля з чотирма гілками. Так домінування незмінної частини (тип 3) дозволяє ускладнити характер залежності значення функції F від своїх аргументів, при цьому розмір зашифрованої частини є малим, тому для досягнення заданого рівня криптостійкості необхідно збільшити кількість ітерацій (NUSH), що впливає на швидкість шифрування. Коли ж відбувається домінування змінної частини (тип 1, 2) функція F залежить від меншої за розміром частини блоку, тому для неї спрощується процес встановлення закономірностей. Але після кожного раунда змінюється більша частина блоку, що ускладнює перетворення та компенсує цей недолік. При цьому стійкість СБШ побудованих на основі мереж Фейстеля з чотирма гілками до лінійного криптоаналізу більша ніж в СБШ побудованих на основі мережі Фейстеля з двома гілками [5].

СБШ на основі мереж Фейстеля мають такі недоліки:

1. Відносно невисока швидкість процедур зашифрування, розшифрування, оскільки один раунд обробляє лише половину (для мережі на дві гілки) або частину (для мереж на чотири гілки) блоку вхідного повідомлення, що вимагає збільшення кількості ітерацій [1,5].

2. Низькі оцінки стійкості до статистичних методів криптоаналізу [5].

3. Потреба в енергонезалежної пам'яті при використанні великої кількості таблиць замін (CAST-128, LOKI97, MacGuffin та ін.).

Блокові шифри на основі SP-мереж

У блокових шифрах побудованих на основі SP-мереж обробка даних зводиться, в основному, до замін (коли, наприклад, фрагмент вхідного блоку замінюється іншим фрагментом відповідно до таблиці замін *S*, яка може залежати від значення ключа) і перестановок P, залежних від ключа [1]. Узагальнена схема раунду основних перетворень наведена на рис. 5.

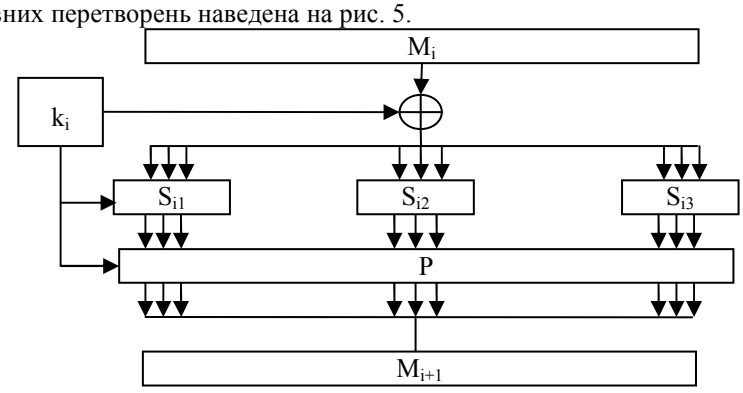

Рисунок 5 – Схема раунду перетворення на основі SP-мережі

Чим більше S-блок, тим важче знайти статистичні залежності для розкриття методами криптоаналізу. Більшість випадкових S-блоків нелінійні і характеризується високою стійкістю до лінійного криптоаналізу, причому із зменшенням кількості вхідних бітів стійкість знижується достатньо повільно [6]. P-блоки реалізують операції перестановки, за допомогою яких маскуються взаємозв'язки між відкритим текстом, шифротекстом і ключем. Особливість перестановки полягає в використанні таких перетворень, які виключають можливість відновлення взаємозв'язку статичних властивостей відкритого та зашифро-

ваного повідомлення [6]. Основні кількісні характеристики сучасних СБШ, побудованих на основі SP-мереж [1], представленні в табл. 5.

Таблиця 5 – Характеристики СБШ на основі SP-мереж

Алгоритм	Кількість раундів r	Довжина ключа K (біт)	Розмір блоку W (біт)
3-Way	11	96	96
ARIA	12/14/16	128,192,256	128
CS-Cipher	8	8-128	64
Diamond2	10-15	8-65536	128
Hierocrypt-L1	6	128	64
Khazad	8	128	64
SAFER+	8/12/16	128/192/256	128
Serpent	32	128/192/256	128
SC2000	7	128	128
	8	192/256	
Threefish	72	256/512	256/512/1024
	80	1024	

Аналіз табл. 5 показує, що більшість СБШ використовують в середньому 8-16 раундів, що значно менше порівняно з СБШ побудованими на основі мереж Фейстеля. Максимальна кількість раундів 80 виконується в шифрі Threefish для блоку 1024 біт і ключа 1024 біт відповідно, який оперує простими операціями. Найбільший діапазон значень ключа заданий в шифрі Diamond2 8-65536 біт. Значення розміру блоку в СБШ варіюється від 64 (CS-Cipher, Khazad) до 1024 біт (Threefish).

Проаналізуємо набір базових операції для СБШ побудованих на основі SP-мереж (табл. 6).

Таблиця 6 – Базові операції для СБШ на основі SP-мереж

Алгоритм	Дод. за $\text{mod}2^N$	Побітові циклічний зсув вліво (змінні/фіксовані)	Логічні операції	Таблична заміна	Перестановка	Множ. на матрицю
ARIA				4 табл.		+
CS-Cipher		(ф=1)	«І»/«НІ»	/+	+	
Hierocrypt-L1				+		+
Khazad				+		+
SAFER+	+				+	+
Serpent		(ф=1,3,5,7,13,22)		8 табл.	+	
SC2000			«І»	3 табл.	+	+
Threefish	(N=64)	(з)			+	

Операція XOR присутня у всіх розглянутих СБШ тому не винесена в табл. 6. Шифр ARIA використовує різну комбінацію таблиць замін для парних та непарних раундів. У CS-Cipher передбачена можливість виконання або табличної заміни або обчислення функції, значення якої можна визначити за таблицею чи обчислити, в залежності від наявних ресурсів. У шифрах SC2000, CS-Cipher виконується додаткове накладання констант на блок даних. Шифр Hierocrypt-L1 передбачає використання двох матриць 4×4 та 8×8. Усі операції SAFER+ виконуються за модулем 256. Таблиці замін Serpent згенеровані з таблиць DES. Також його лінійне перетворення може реалізовуватись у вигляді табличної заміни або ж як ряд обчислень (побітова циклічні зсуви вліво та XOR). Аналізуючи табл. 6 можна сказати про масове використання СБШ побудованими на основі SP-мереж операції множення на матрицю. Табличні заміни можуть бути реалізовані у вигляді таблиць, або ж як ряд операцій (CS-Cipher, Serpent, ARIA). Перестановки є фіксованими і задаються таблично (SAFER+, Serpent та ін.). Операція додавання за модулем використовується лише у Threefish, структура якого не передбачає S-блок.

СБШ, побудовані на основі SP-мереж, є стійкими до різних видів криптоаналізу, завдяки високому степеню нелінійності основних перетворень [6]. В загальному випадку процедури зашифрування та розшифрування суттєво відрізняються, що збільшує складність їх апаратної та програмної реалізації. Використання табличних замін суттєво впливає на швидкість процедур зашифрування та розшифрування та вимоги шифрів до енергонезалежної пам'яті [1]. Вищенаведені недоліки сповільнюють процес розробки та дослідження СБШ даного типу.

Блокові шифри на основі структури «квадрат» (Square)

Для структури «квадрат» характерним є представлення блока у вигляді двовимірного байтового масиву. Криптографічні перетворення можуть виконуватись над окремими байтами масиву, а також над

його стовпчиками та рядками [1]. Основні кількісні характеристики сучасних СБШ побудованих на основі структури «квадрат» [1, 7] наведені в табл. 7.

Таблиця 7 – Характеристики СБШ на основі структури «квадрат»

Алгоритм	Кількість раундів r	Довжина ключа K (біт)	Розмір блоку W (біт)
AES (Rijndael)	10	128	128
	12	192	192
	14	256	256
Anubis	8+K[32]	128-320 (кратне 32)	128
CRYPTON	12	8-256 (кратне 8)	128
Grand Cru	10	128	128
Noekon	16	128	128
Q	8/9	128/192/256	128
SQUARE	8	128	128

Аналіз табл. 7 показує, що для СБШ побудованих на основі структури «квадрат» характерним є представлення блоку 128 біт у вигляді масиву (4×4) байт, лише шифр Noekon оперує блоком представленим у вигляді 4-х рядків по 32 біт. Можлива кількість раундів від 8 (SQUARE) до 18 (Anubis), причому змінну кількість підтримує лише шифр Anubis та Q залежно від довжини ключа. Більшість представників передбачають використання заданого розміру ключа або змінний 8-320 біт (CRYPTON, Anubis). У зв'язку з своєрідним представленням блоку даних специфічними є і операції, що використовуються для СБШ даного типу, оскільки вони можуть виконуватись над рядками, стовпчиками, байтами та бітами блоку. Вхідне та вихідне забілювання зазвичай виконує комбінацію декількох операцій основного перетворення (SQUARE, Grand Cru та ін.).

Набір базових операції для СБШ побудованих на основі структури «квадрат» наведено в табл. 8.

Таблиця 8 – Базові операції для СБШ на основі структури «квадрат»

Алгоритм	Циклічні зсуви (байт/біт)	Таблична заміна (байт)	Перестановка (байт/біт)	Множення на матрицю
AES (Rijndael)	(байт)	+		
Anubis		+	(байт)	+
CRYPTON		5 табл.	(біт, байт)	
Grand Cru	(біт)	+	(байт)	+
Noekon	(біт)	+		
Q	(байт)	3 табл.		
SQUARE		+	(байт)	

У шифрі AES використовується операція множення стовпців. У CRYPTON для табличних замін використовуються 4 (різні для парних та непарних раундів) таблиці, а для перестановки біт використовуються маскуючи константи та логічна операція «І». Для вхідного забілювання у Grand Cru байти блоку послідовно додаються із значеннями таблиці замін (взята з AES). Також, значення циклічного зсуву вправо біт рядка залежить від ключа, а операція додавання виконується за модулем 256. У шифрі Q використовується однаковий ключ для всіх раундів шифрування та запозичені таблиці замін, з шифру AES, а інші із Serpent, при цьому передбачено використання циклічного зсув вгору у стовпчику на змінну кількість байт. Шифр SQUARE реалізує лінійне перетворення над рядком шляхом множення байт за модулем 28. У шифрі Noekon усі операції виконуються над бітами рядку. Аналіз базових операцій для СБШ побудованих на основі структури квадрат (табл. 8) дозволяє зробити висновки про специфіку використання цих операцій. Так циклічні зсуви можуть виконуватись не лише вліво та вправо оперуючи бітами та байтами але і вгору в стовпчику (Q). Таблична заміна байт передбачена використанням всіх розглянутих СБШ даного типу. Найбільшу кількість таблиць (5) використовує шифр CRYPTON, причому основною є одна таблиця, а чотири інші генеруються з неї за допомогою операції циклічного зсуву вліво. Операція перестановки замінює i-й рядок і i-й стовпчик, лише у шифрі CRYPTON додатково присутня перестановка біт. Найбільшу кількість базових операцій використано у шифрі Grand Cru, а найменшу у Anubis та Noekon.

Проведені дослідження у роботі [7] свідчать про високу стійкість цього класу блокових шифрів. Використання байт-орієнтованої структури дозволяє забезпечити ефективну реалізацію на 8-біних мікроконтролерах [1, 3]. Проте, використання табличних замін байт суттєво позначаються на швидкості шифрування блокових шифрів на основі структури Square.

Блокові шифри на основі арифметичних операцій за модулем

Основні перетворення даних СБШ передбачають використання операцій множення та додавання за модулем [2]. Характеристики таких СБШ, представленні в табл. 9.

Таблиця 9 – Характеристики СБШ на основі операцій за модулем

Алгоритм	Кількість раундів *r*	Довжина ключа *K* (біт)	Розмір блоку *W* (біт)
ABC	17	512	256
Caligo	6	128	128
		256	256
MMB	6	128	128
MultiSwap	12	374	64
xmx	8	256	256
		512	512
	12	768	768
	16	1024	1024

Аналізуючи табл. 9 можна відзначити, що максимальну кількість раундів (r=17) для СБШ, побудованих на основі операцій за модулем, використовує шифр ABC, а мінімальну (r=6) шифри MMB та Caligo. Значення довжини ключа шифрування варіюється від 128 біт (MMB) до 1024 біт (xmx), і залежить від розміру блоку (xmx, Caligo, MMB). Мінімальним розміром блоку оперує шифр MultiSwap (64 біт), а максимальним xmx (1024 біт). У шифрах ABC [8] та MMB [2] операція множення за модулем використана як функція побітового перемішування. У СБШ Caligo [9] виконується множення за модулем. Процес зашифрування одного блоку даних СБШ MultiSwap [10] складається з 12 раундів, з яких на 5 і 11 раундах виконується додавання підблоку даних за модулем 232 з ключем, а на решті раундів виконується множення підблока даних за модулем 232 з відповідним ключем. Основні перетворення блокового шифру xmx [11] базуються на операціях множення за модулем та XOR. Спочатку здійснюється побітове додавання *n*-бітного блока даних з секретним ключем *A*, отриманий результат множиться на теж саме значення *A* за модулем $2^n - 1$. В процесі розшифрування в якості множника використовують обернене мультиплікативне *A* за модулем *m*, що обчислюється з використанням розширеного алгоритму Евкліда. У роботі [12] запропоновано СБШ, який оперує з *n*-бітними блоками даних і *n*-бітними ключем. Процес зашифрування блоку даних виконують з використанням таких перетворень:

$$Y = f'\left(\left(X \oplus K_1\right) \cdot K_2 \bmod 2^n\right) \cdot K_3 \bmod 2^n,$$

де K_1, K_2, K_3 – підключі (K_1, K_3 – непарні); $f'(.)$ – функція дзеркальної перестановки u бітних підблоків *n*-бітного блоку даних. Процес розшифрування обернений по відношенню до зашифрування і вимагає пошук обернених мультиплікативних за модулем 2^n.

СБШ на основі операцій за модулем мають недоліки з точки зору необхідності використання платформ, що підтримують операції множення, додавання (32, 64 біт), для досягнення необхідної швидкості шифрування. Використання одного і того самого ключа для двох різних груп операцій на всіх раундах шифрування (xmx, MultiSwap) і відкритого значення модуля не забезпечує достатній рівень криптографічної стійкості шифру, проти атак, з використанням мультиплікативних диференціалів [13].

Висновки

Наведений аналіз відомих СБШ дозволяє сформулювати рекомендації для вибору перетворень, що забезпечать підвищення швидкості шифрування:

1. Використання великої кількості табличних замін збільшує вимоги шифру до енергонезалежної пам’яті (Camellia, CAST-128, E2, LOKI97 та ін.). Тому для підвищення швидкості шифрування необхідно передбачити можливість реалізації табличної заміни як таблиці або певною послідовністю алгебраїчних операцій, для вибору у конкретній системі.

2. Масове використання операцій орієнтованих на певний вид платформ (DFC, HPC, RC6, та ін.) суттєво зменшує швидкість шифрування СБШ на інших платформах, що обмежує можливості їх використання.

3. Використання складних багатоетапних процедур розгортання секретного ключа значно зменшують швидкість шифрування (Blowfish, Twofish, Hierocrypt-L1), але і використання елементарної процедури розгортання ключа (MAGENTA, TEA, Noekon та ін.) збільшує можливість успішної атаки на зв’язаних ключах (Related-key attack). Отже, для збільшення швидкості шифрування процедура розгортання секретного ключа повинна бути реалізована максимально просто з можливістю розпаралелення обчислень в багатопроцесорних системах.

4. В багатьох СБШ (FEAL-N, SPEED) збільшення рівня криптографічної стійкості досягається шляхом збільшення кількості раундів шифрування, що значно зменшує швидкість шифрування. Тому, стійкість шифру не повинна досягатись лише виключно збільшенням кількості ітерацій перетворення.

5. Використання складної функції раундового перетворення з великою кількістю базових операцій (Twofish, MARS, SC2000 та ін.) суттєво впливає на швидкість шифрування. До того ж ускладнює можливість детального аналізу стійкості шифру та унеможливлює доведення відсутності прихованих вразливо-

стей. Тому, структура основних перетворень СБШ повинна бути максимально зрозумілою та прозорою для її аналізу та дослідження.

6. Використання однорідних структур перетворень може призвести до того, що закономірність одного раунду може бути розповсюджена на весь ланцюжок основних перетворень, полегшуючи тим самим криптоаналіз. Тому, в останній час деякі автори криптоалгоритмів використовують неоднорідну (гетерогенну) структуру, в якості основи для СБШ. При цьому різні раунди шифрування можуть мати вразливості різних типів, але в сукупності забезпечувати високу стійкість перетворення.

Для підвищення основних характеристик блокових шифрів, рядом дослідників пропонуються альтернативні шляхи побудови перетворення даних одним з яких є використання принципів псевдонедетермінованих алгоритмів [14]. Можливість створення ними великої кількості модифікацій алгоритму шифрування теоретично робить неможливим попередні статистичні дослідження, які є базовими для найпотужніших сучасних методів криптографічного аналізу. Оскільки стійкість алгоритмів таких шифрів забезпечується, на відміну від існуючих, не складністю функції перетворення, а невизначеним порядком їх застосування (з точки зору зловмисника) та змінною структурою оброблюваної інформації, тому зникає потреба у використанні складних обчислень. Перетворення будуються на базі елементарних операцій, які найбільш просто та швидко реалізуються в сучасних мікропроцесорах, що обумовлює швидкість їх виконання.

Список літератури

1. Панасенко С. П. Алгоритмы шифрования. Специальный справочник / С. П. Панасенко. – СПб.: БХВ-Петербург, 2009. – 576 с.

2. Daemen J. Block ciphers based on modular arithmetic / J. Daemen, R. Govaerts // In Proceedings of the 3rd symposium on State and Progress of Research in Cryptography, W. Wolfowicz (ed.), Fondazione Ugo Bordoni, 1993. – pp. 80-89.

3. Винокуров А. Ю. Новые подходы в построении блочных шифров с секретным ключом / А. Ю. Винокуров. – Режим доступу до статті: http://www.enlight.ru/crypto/index.htm

4. Шеннон К. Работы по теории информации и кибернетике / К. Шеннон ; пер. с англ. – М.: Изд-во иностранной литературы, 1963. – 830 с.

5. Schneier B. Unbalanced Feistel Networks and Block-Cipher Design / B. Schneier, J. Kelsey // Counterpane Systems. – Режим доступу до статті: http://www.schneier.com/paper-unbalanced-feistel.pdf

6. Kam J. Structured design of substitution-permutation encryption networks / J. Kam, G. Davida // IEEE Transactions on Computers. – 1979. – Vol. 28, №10. – P. 747.

7. Knudsen L. The block cipher SQUARE / L. Knudsen, J. Daemen, V. Rijmen // Computer Science. – Springer-Verlag, 1997. – Vol.1267. – pp. 149-165. – Режим доступу до статті: http://citeseerx.ist.psu.edu

8. Schmidt D. ABC – A Block Cipher / D. Schmidt // Cryptology ePrint Archive. – 2002. – P. 50 – Режим доступу до статті: http://eprint.iacr.org/2002/062.

9. Machado A. Caligo, an Extensible Block Cipher and CHash, a Caligo Based Hash / A. Machado // Cryptographic Hash Algorithm Competition, 2006. – P. 11. – Режим доступу до статті: http://csrc.nist.gov

10. Screamer B. Microsoft's digital rights management scheme-technical details / B. Screamer. – 2001. – Режим доступу до статті: http://cryptome.org/ms-drm.htm.

11. M'Raihi D. XMX: A Firmware-Oriented Block Cipher Based on Modular Multiplications / D. M'Raihi and others // Lecture Notes in Computer Science. – 1997. – Vol. 1267. – pp. 166-171.

12. Сокирук В. В. Побудова статистично безпечного БСШ на основі арифметичних операцій за модулем / В. В. Сокирук, В. А. Лужецький // Інформаційні технології та комп'ютерна інженерія. – В.: ВНТУ, 2006. – №1. – С. 158-163

13. Borisov N. Multiplicative differentials / N. Borisov, M. Chew, R. Johnson // Fast Software Encryption: 9th Interna-tional Workshop on table of contents. – 2002. – Vol. 2365. – pp. 17-33.

14. Лужецький В. А. Блочний шифр на основі псевдонедетермінованих послідовностей криптопримітивів / В .А. Лужецький, А. В. Остапенко, // Наукові праці ВНТУ. – № 4 (2010). – Режим доступу до статті: http://www.nbuv.gov.ua.

Відомості про авторів

Лужецький Володимир Андрійович – д.т.н., професор, завідувач кафедри Захисту Інформації. Вінницький національний технічний університет, Хмельницьке шосе, 95, м. Вінниця, 21021.

Остапенко Аліна Василівна – аспірант кафедри захисту інформації. Вінницький національний технічний університет, Хмельницьке шосе, 95, м. Вінниця, 21021, e-mail: asja87@gmail.com