УДК 681.3.067

РОЗРОБКА ДИНАМІЧНОГО АЛГОРИТМУ ШИФРУВАННЯ ДАНИХ НА ОСНОВІ DES

Асп. Яремчук Є. В., докт. техн. наук, проф. Азаров О. Д.

Вступ

Стандарт шифрування даних DES (Data Encryption Standart) був опублікований Національним бюро стандартів США (NBS) і набрав чинності у 1977 р. В основі DES лежить криптографічний алгоритм, який оперує блоками розміром 64 біта та ключем розміром 48 біт. Процес шифрування кожного блоку виконується послідовно в 16 циклах, ключ для кожного циклу формується довільно з 56-бітної ключевої послідовності [1—2].

Результати багаторічних досліджень [3—4] показали, що «зламати» шифр, отриманий за допомогою DES, можливо лише шляхом «силової атаки», тобто повним перебором усіх можливих ($2^{56} \approx 10^{18}$) ключових послідовностей. Однак і така довжина ключа (56 біт) не забезпечує належний рівень захищеності даних. Але, не зважаючи на це, криптографічний алгоритм DES має теоретичну цінність як приклад вдалого поєднання підстановки та перестановки, що забезпечило добре розсіювання та перемішування.

Авторами статті зроблена спроба формалізувати основні положення алгоритму DES так, щоб їх можна було використати для шифрування блоків довжиною $N = 2^k$, $(k \geq 3)$. Згідно стандарту шифрування даних усі функції-перетворення однозначно визначені до початку процесу шифрування, тобто є статичними, а в запропонованому алгоритмі ці функції-перетворення будуються динамічно під час виконання. Конкретний вигляд їх визначається довжиною блоку вхідних даних і різний для різних значень k

Загальний опис динамічного алгоритму шифрування даних

Згідно DES процес шифрування (рис. 1) включає такі вузлові моменти: початкова P_1 та кінцева P_2 перестановки, функція шифрування $f(R(i-1), K(i))$ та функція формування ключа $K(i)$, де i – номер циклу шифрування ($i = 1, \ldots, 2^{k-2}$). Для зручності усі перетворення, на зразок підстановки та перестановки, подаються у вигляді матриць розміром $n \times m$.

Нехай маємо блок вхідних даних, довжиною $N = 2^k$, $(k \geq 3)$, для стандарту шифрування даних $N = 2^6 = 64$. Визначимо інші параметри, необхідні для реалізації алгоритму:

кількість циклів шифрування 2^{k-2};

довжина ключової послідовності $n_0 = 2^{k-1}\left(\frac{3}{4}+1\right) = 2^{k-3} \cdot 7$;

довжина ключа $n_i = 2^{k-1}\left(\frac{3}{4}\right) = 2^{k-2} \cdot 3$, $i = 1, \ldots, 2^{k-2}$.

Над блоком вхідних даних виконується початкова перестановка P_1, яку в загальному вигляді можна зобразити у вигляді:

$$m = 2^3; \quad n = 2^{k-3};$$

$$a_{ij} = \begin{cases} N - (m - i \cdot 2), & \text{якщо } j = 1, \ i = 1, \ldots, \frac{n}{2}; \\ (N-1) - \left(m - (i - \frac{n}{2}) \times 2\right), & \text{якщо } j = 1, \ i = \frac{n}{2}, \ldots, n; \\ a_{ij-1} - m, & \text{якщо } i = 1, \ldots, n, \ j = 2, \ldots, m. \end{cases}$$

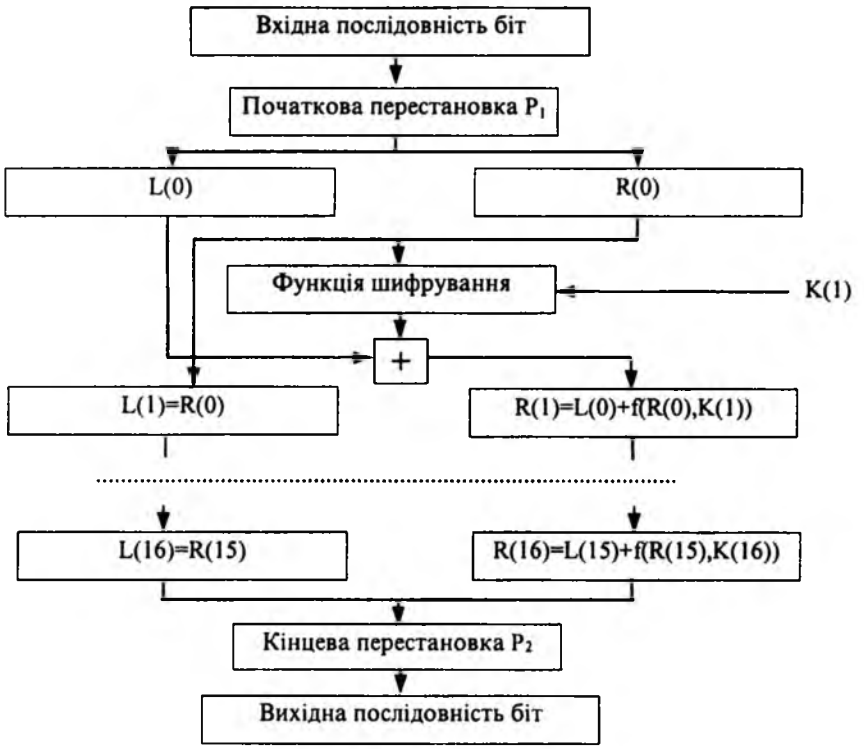

Рис.1. Процес шифрування

Приклад 1. Для алгоритму DES отримаємо:

$m = 2^3 = 8; \quad n = 2^{k-3} = 2^3 = 8;$

$$\begin{vmatrix} 58 & 50 & 42 & 34 & 26 & 18 & 10 & 2 \\ 60 & 52 & 44 & 36 & 28 & 20 & 12 & 4 \\ 62 & 54 & 46 & 38 & 30 & 22 & 14 & 6 \\ 64 & 56 & 48 & 40 & 32 & 24 & 16 & 8 \\ 57 & 49 & 41 & 33 & 25 & 17 & 9 & 1 \\ 59 & 51 & 43 & 35 & 27 & 19 & 11 & 3 \\ 61 & 53 & 45 & 37 & 29 & 21 & 13 & 5 \\ 63 & 55 & 47 & 39 & 31 & 23 & 15 & 7 \end{vmatrix}.$$

Після виконання усіх циклів шифрування здійснюється кінцева перестановка P_2, яка представляє собою матрицю, елементи якої знаходяться за формулами:

$m = 2^3; \quad n = 2^{k-3};$

$$a_{ij} = \begin{cases} N - \left(\frac{m}{2} - i\right), & \text{якщо} \quad i = 1, \quad j = 1, 3, \ldots, m-1; \\ mi, & \text{якщо} \quad i = 1, \quad j = 2, 4, \ldots, n; \\ a_{i-1j} - 1, & \text{якщо} \quad i = 2, \ldots, n, \quad j = 1, \ldots, m. \end{cases}$$

Перші $\frac{n}{2}$ рядки перестановки P_1, записані послідовно однин за одним, утворюють послідовність лівих або молодших біт ($L(0)$ – послідовність), а решта $\frac{n}{2}$ рядків – послідовність правих або старших біт ($R(0)$ – послідовність). Довжина обох послідовностей однакова і дорівнює 2^{k-1}.

Протягом 2^{k-2} циклів виконується процес шифрування, який виражається рекурентними співвідношеннями:

$$L(i) = R(i-1),\ i = 1, 2, \ldots, 2^{k-2};$$
$$R(i) = L(i-1) + f(R(i-1), K(i)),\ \ i = 1, 2, \ldots, 2^{k-2}.$$

Функція шифрування DES $f(R(i-1), K(i))$ (рис. 2) складається з функції розширення E, функції формування ключа $K(i)$ (не розглядається в цій статті), так званих S-функцій і перестановки P_3.

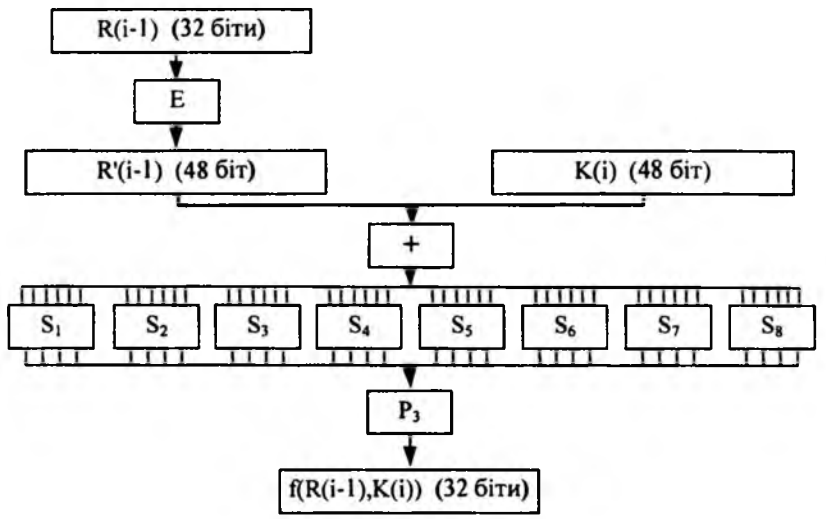

Рис. 2. Функція шифрування

Функція розширення E перетворює $R(i)$-послідовность розміром 2^{k-1} у послідовність $R'(i)$ розміром $2^{k-2} \cdot 3$ і має вигляд:

$m = 2^{k-3}$; $n = 2 \cdot 3$;

$$a_{ij} = \begin{cases} 2^{k-1}, & \text{якщо } i = 1,\ j = 1; \\ (a_{ij-1} + 1) \bmod a_{11}, & \text{якщо } i = 1, 2, \ldots, n,\ \ j = 2, 3, \ldots, m; \\ (a_{i-1m-1} + 1) \bmod a_{11}, & \text{якщо } i = 2, 3, \ldots, n,\ \ j = 1, 2, \ldots, m. \end{cases}$$

Згідно з DES S-функції подаються у вигляді таблиць, розміром 4×16 кожна, рядками якої є довільні перестановки послідовності за модулем 16. Кількість таких функції для алгоритму, що розглядається, знаходиться за формулою $n_S = 2^{k-3}$. Структура S-функції, розміри вхідних та вихідних блоків даних для них залишимо без змін, оскільки мотиви, які покладені у вибір саме такої функції шифрування розробниками фірми IBM, були засекречені [3]. Для формування перестановок пропонується використати алгоритм p-інверсій, який розроблений на основі алгоритму інверсій (описаного в праці Д. Кнута «Искусство программирования для ЭВМ» [5]).

Алгоритм А. Генерування перестановки методом p-інверсій

Початок алгоритму

Крок 1. Задання початкових даних

Довжина або порядок перестановки l;

Кількість інверсій n_i;

Параметр p, бажано приймати $0 < p \le \frac{l}{2}$;

Початкова перестановка $A = \{1, 2, \ldots, l\}$;

Лічильник перестановок $i = 1$.

Крок 2. Якщо $i > n_i$, то перейти на крок 5, інакше

якщо $(i+p) > l$, то $a'_i \leftrightarrow a_{(i+p)-l}$, інакше $a'_i \leftrightarrow a_{i+p}$.

Крок 3. $i = i + 1$.

Крок 4. Отриману перестановку $A' = \{a'_1, a'_2, \ldots, a'_l\}$ приймаємо за початкову;

Переходимо на Крок 2.

Крок 5. Отримана перестановка $A' = \{a'_1, a'_2, \ldots, a'_l\}$ є кінцевою.

Завершення алгоритму.

Приклад 2. Генерування перестановки за допомогою алгоритму А.

$l = 5,\ n_i = 4,\ p = 2;$
$A = \{1, 2, 3, 4, 5\};$
$i = 1 : A' = \{3, 2, 1, 4, 5\};$
$i = 2 : A' = \{3, 4, 1, 2, 5\};$
$i = 3 : A' = \{3, 4, 5, 2, 1\};$
$i = 4 : A' = \{2, 4, 5, 3, 1\}.$

Перестановка P_3 довжиною 2^{k-1} формується за допомогою алгоритму А і записується у вигляді матриці $n_S \times 4$.

Процедура формування ключа дуже важлива та складна операція, так як секретність даних, які зашифровані алгоритмом DES, визначається лише секретністю ключа. Зазначимо, що в розробці фірми IBM було передбачено незалежний вибір усіх $16 \times 48 = 768$ біт ключа, які використовуються у 16 циклах шифрування. Розмірність ключа була знижена NBS з власних міркувань [3]. Пропонується функція формування ключа, відмінна від описаної в стандарті шифрування даних. Основні принципові відмінності, що лягли в основу нової функції, такі:

– операції зсуву виконуються для молодших (або лівих) та старших (або правих) біт ключа незалежно, причому кількість позицій зсуву обирається теж незалежно;

– функція перестановки та вибору ключа $K(i)$ формується алгоритмом генерації перестановок (подібним до алгоритму А);

– початкова перестановка ключевої послідовності та функція вибору біт ключа P_k, будується аналітично.

Природно, алгоритм, побудований на основі цих узагальнень, не буде мати усі переваги DES, але такий підхід дозволяє згенерувати алгоритм шифрування, який би за своїми характеристиками був наближений до DES.

Висновки

1. Описаний підхід дозволяє виконувати шифрування блоків розміром 2^k з ключем $2^{k-2} \cdot 3$ біт.

2. Аналітичне зображення функцій-перетворень збільшує ефективність та універсальність програмної реалізації описаного алгоритму.

ЛІТЕРАТУРА

1. Дейтел Г. Введение в операционные системы. Т. 2. — М.: Мир. — 1987. — 398 с.

2. Диффи У., Хеллман М. Э. Защищенность и имитостойкость: Введение в криптологию // ТИИЭР. — 1979. — № 3. — С. 71—109.

3. Месси Дж. Л. Введение в современную криптологию // ТИИЭР. — 1976. — № 3. — С. 24—42.

4. Смид М. Э., Бранстед Д. К. Стандарт шифрования данных: будущее и настоящее // ТИИЭР – 1988. – № 5. — С. 43—54.

5. Кнут Д. Искусство программирования для ЭВМ. Т. 2. — М.: Мир. — 1976. — 724 с.

Кафедра обчислювальної техніки