Приклад виконання

ПІБ: Gorbenko Vitaliy Ivanovich
Літери для шифрування: gorbenko
Представимо літери в двійковому вигляді за таблицею ASCII:

01100111 01101111 01110010 01100010 01100101 01101110 01101011 01101111

В результаті маємо 64 біта, що дорівнює розміру блоку в алгоритмі DES. За алгоритмом DES здійснюємо початкову перестановку IP бітів вхідного блоку за наступною таблицею:

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

Числа в даній таблиці означають: у позицію 1 записують 58-й біт вхідного блоку, у позицію 2 записують 50 -й біт, у позицію 3 записують 42-й біт і т.д.

В результаті отримуємо блок в 64 біти наступного вигляду:

11101111 00001100 11011010 00110111 00000000 11111111 01000011 01010111

Цей блок ділимо на ліву і праву частини по 32 біти.

Ліва частина L: 11101111 00001100 11011010 00110111.

Права частина R: 00000000 11111111 01000011 01010111.

Формуємо ключ раунду.

Для всіх варіантів ключ для шифрування однаковий: password. Ці вісім символів також представляємо у двійковому вигляді за таблицею ASCII:

01110	01100	01110	01110	01110	01101	01110	01100
000	001	011	011	111	111	010	100

Здійснюємо перестановку PC-1. За наступною таблицею:

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

Кількість елементів таблиці 56. Тобто в результаті отримуємо вектор у 56 біт. Відсутні біти 8-й, 16-й, 24-й, 32-й, 40-й, 48-й, 56-й, 64-й. Ці біти ніякої участі в процесі шифрування не приймають і використовуються для контролю парності. Правило використання даної таблиці: у позицію 1 записують 57-й біт вхідного блоку, у позицію 2 записують 49 -й біт, у позицію 3 записують 41-й біт і т.д.

Результат виконання перестановки PC-1:

00000000 11011111 11011111 01010101 11001011 00000000 00001101

Розбиваємо цю послідовність біт на дві частина по 28 біт.

Ліва частина C0:

00000000 11011111 11011111 0101

Права частина D0:

0101 11001011 00000000 00001101

Виконуємо циклічний зсув обох частин на 1 біт вліво:

Отримуємо C1:

00000001 10111111 10111110 1010

Отримуємо D1:

1011 10010110 00000000 00011010

З’єднуємо обидві частини:

00000001 10111111 10111110 10101011 10010110 00000000 00011010

До цієї послідовності застосовуємо перестановку PC-2 за наступною таблицею:

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

В таблиці 48 комірок. Результатом виконання цієї перестановки буде 48 бітовий вектор. Правила виконання перестановки: у позицію 1 записують 14-й біт вхідного блоку, у позицію 2 записують 17 -й біт, у позицію 3 записують 11-й біт і т.д.

В результаті отримуємо ключ раунду:

11100000 10101110 01101110 10011101 11011110 01011100

Функція раунду.
Входом даної функції є права частина R (32 біти) вхідного блоку:
00000000 11111111 01000011 01010111
Застосовуємо до R функцію розширення E за наступною таблицею:

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

В таблиці 48 комірок. Результатом виконання цієї перестановки буде 48 бітовий вектор. Правила виконання перестановки: у позицію 1 записують 32-й біт вхідного блоку, у позицію 2 записують 1 -й біт, у позицію 3 записують 2-й біт і т.д.

В результаті маємо 48 бітовий вектор:

10000000 00010111 11111110 10100000 01101010 10101110

Даний вектор і ключ раунду порозрядно складаються за модулем 2:

01100000 10111001 10010000 00111101 10110100 11110010

Ця 48 бітна послідовність розбивається на 8 блоків по 6 біт кожний:

011000 001011 100110 010000 001111 011011 010011 110010

Кожен з блоків подається на вхід відповідного S-блоку. Робота S-блоків описується наступними таблицями:

S1-блок

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S2-блок

15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5

0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S3-блок

10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S4-блок

7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S5-блок

2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

S6-блок

12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S7-блок

4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1

13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S8-блок

13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

Ці S-блоки функціонують за наступним правилом. Входом є 6 бітний вектор. Перший та останній біти визначають номер рядка (нумерація з 0). Середні чотири біти – номер стовпчика (нумерація з 0). Десяткове число у відповідній комірці записується у вигляді 4 бітного двійкового числа, що і є виходом S-блоку.

Подані на вхід 6 бітні послідовності після обробки на відповідному S-блоці перетворюються на наступні 4 бітні вектори:

0101 0010 1001 0001 0001 1011 0011 1101

Ці 32 біти подаються на вхід перестановки P:

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

В таблиці 32 комірки. Результатом виконання цієї перестановки буде 32 бітовий вектор. Правила виконання перестановки: у позицію 1 записують 16-й біт вхідного блоку, у позицію 2 записують 7 -й біт, у позицію 3 записують 20-й біт і т.д.

В результаті отримуємо значення функції раунду у вигляді 32 бітового вектора:

11111110 00100000 10101101 00100010

Значення функції раунду порозрядно складається по модулю 2 з лівою частиною вхідного блоку L:

00010001 00101100 01110111 00010101

Отриманий 32 бітний вектор є правою частиною 64 бітного вектора. Лівою частиною буде вектор R (тобто праві 32 біти 64 бітної послідовності на початку раунда).

Отже, остаточний результат раунду:

00000000 11111111 01000011 01010111 00010001 00101100 01110111 00010101

В контрольній роботі для спрощення виконується лише один раунд. Тому отриману послідовність вважаємо результатом 16 раундів.

Заключні дії алгоритму DES.

Результат 16 раундів розбиваємо на дві 32 бітні послідовності і міняємо їх місцями:

00010001 00101100 01110111 00010101 00000000 11111111 01000011 01010111

Застосовуємо до отриманих 64 біт наступну перестановку:

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

Правила виконання перестановки: у позицію 1 записують 40-й біт вхідного блоку, у позицію 2 записують 8 -й біт, у позицію 3 записують 48-й біт і т.д. Дана перестановка є оберненою до початкової перестановки IP.

Нарешті отримуємо остаточний результат:

00010001 00101100 01110111 00010101 00000000 11111111 01000011 01010111

Це і є 64 біти шифротексту.