Инновационный Евразийский университет

Кафедра
«Энергетика, металлургия и информационные технологии»

СЛАЙД-ЛЕКЦИЯ

по дисциплине

«Основы информационной безопасности»

Тема: Американский стандарт шифрования данных DES

Образовательные программы:
6В06101 «Информатика»
6В06102 «Информационные системы»
6В06103 «Вычислительная техника и программное обеспечение»

Разработчик:
старший преподаватель, м.и. И.И. Ляшенко

План лекции:

1. Краткая характеристика шифра DES

2. Алгоритм шифрования DES

1. Краткая характеристика шифра DES

Стандарт шифрования данных DES (Data Encryption Standard) опубликован Национальным бюро стандартов США в 1977г.

В 1980г. этот алгоритм был принят Национальным институтом стандартов и технологий США (НИСТ) в качестве *стандарта шифрования данных для защиты от несанкционированного доступа* к важной, но не секретной информации в государственных и коммерческих организациях США.

К *достоинствам* DES можно отнести простоту ключевой системы, высокую скорость аппаратной и программной реализации, достаточно высокую криптографическую стойкость алгоритма шифрования при заданной длине ключа.

Алгоритм DES, используя комбинацию ряда подстановок и перестановок, осуществляет шифрование 64-битовых блоков данных с помощью 56-битового ключа k.

2. Алгоритм шифрования DES

Процесс шифрования состоит в начальной перестановке битов входящего блока, 16-ти циклах шифрования k и конечной перестановке битов.

Все таблицы являются *стандартными* и должны использоваться при реализации алгоритма DES в неизменном виде.

Все перестановки и коды в таблицах подобраны разработчиками таким образом, чтобы максимально затруднить процесс вскрытия шифра путем подбора ключа.

Схема алгоритма DES

В приводимом описании алгоритма DES использованы следующие обозначения:

L_i и R_i - левая и правая половины 64-битового блока L_iR_i;

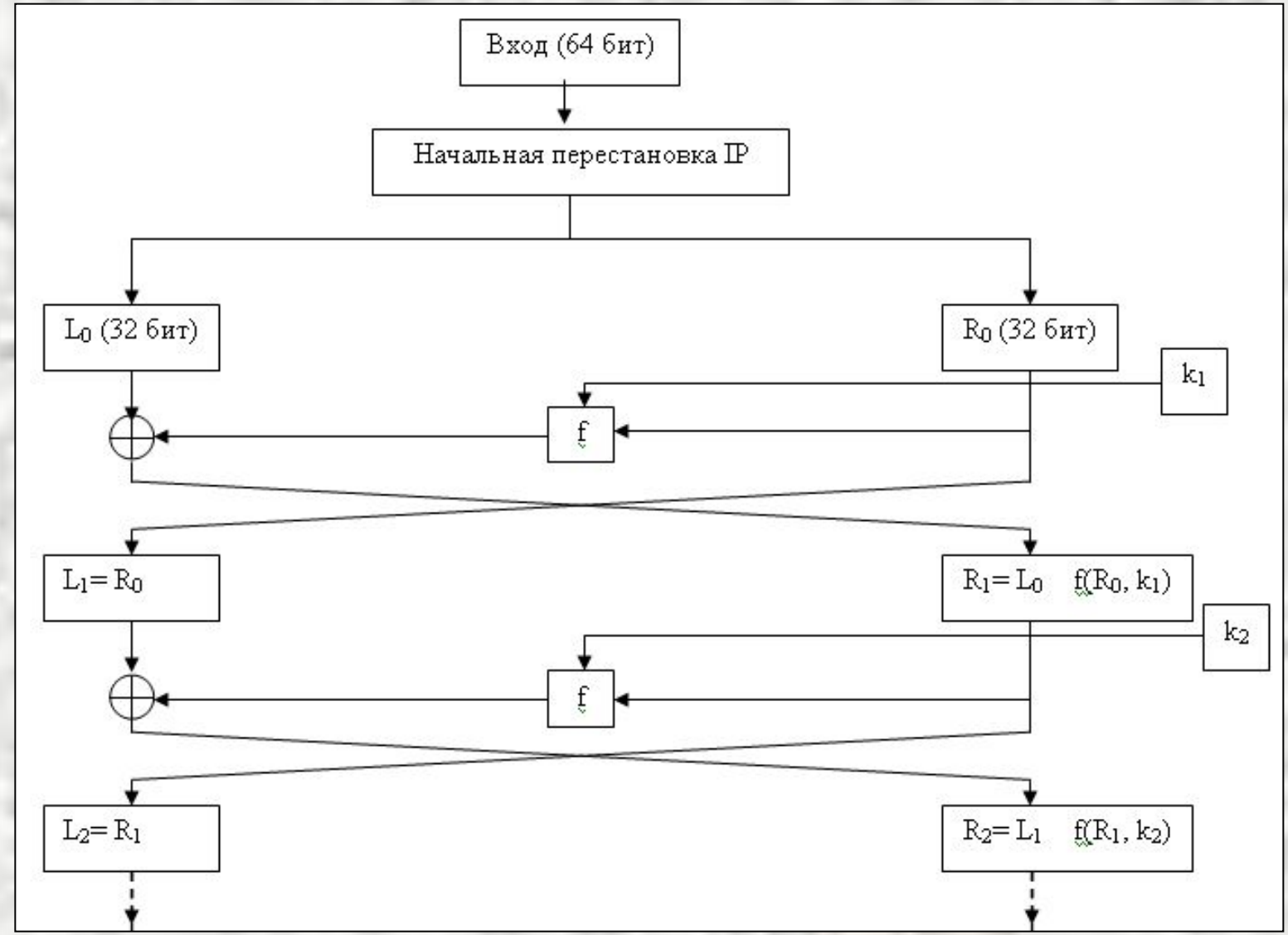
$\oplus$ - операция побитового сложения векторов-блоков по *mod 2*;

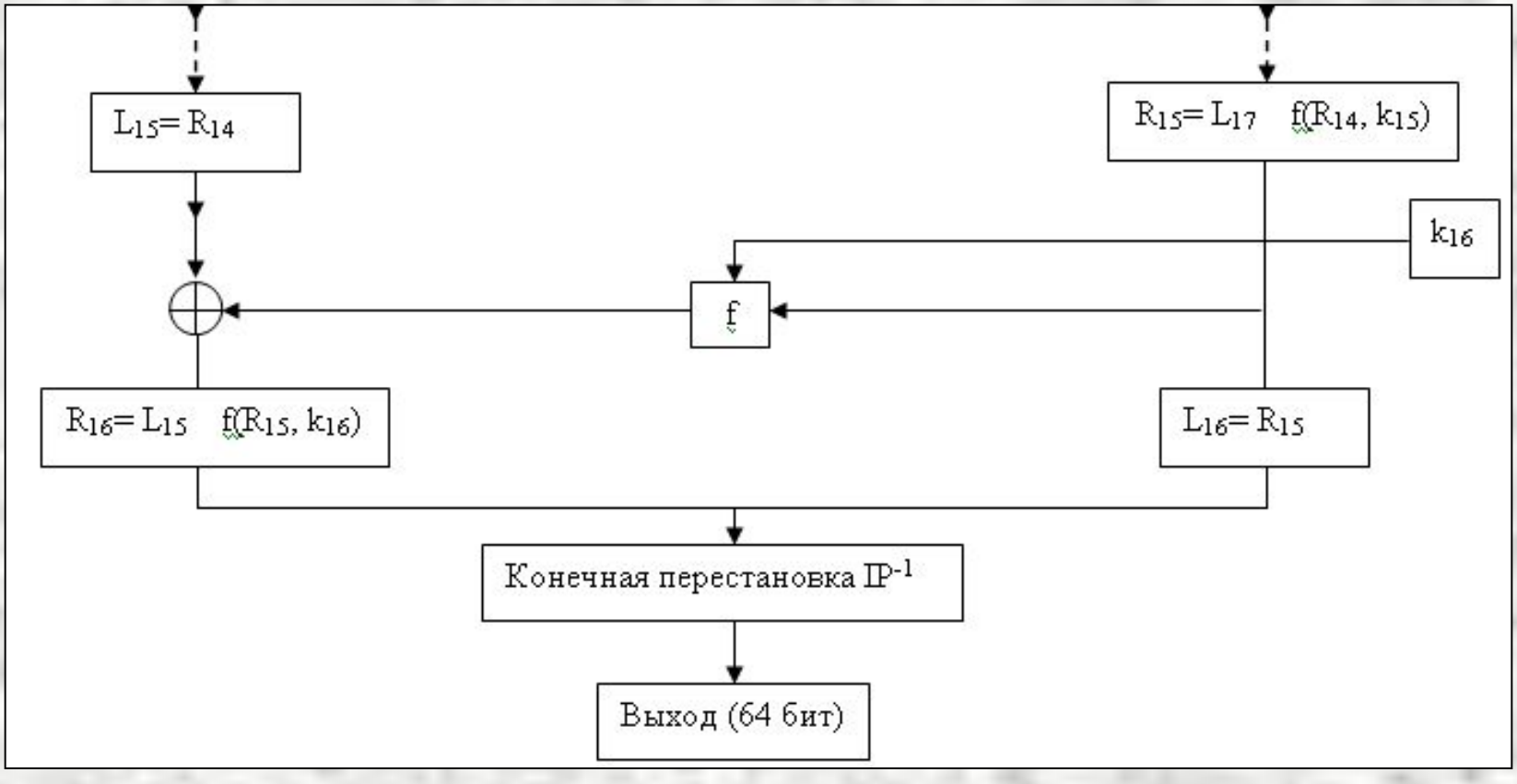
k_i – 48-битовые ключи;

F – функция шифрования;

IP – начальная перестановка степени 64.

Лекция 6. Американский стандарт шифрования данных DES




L15= R14
R15= L17 f(R14, k15)
k16
f
R16= L15 f(R15, k16)
L16= R15
Конечная перестановка IP-1
Выход (64 бит)

При зашифровании очередного блока T его биты подвергаются *начальной* перестановке IP согласно таблице:

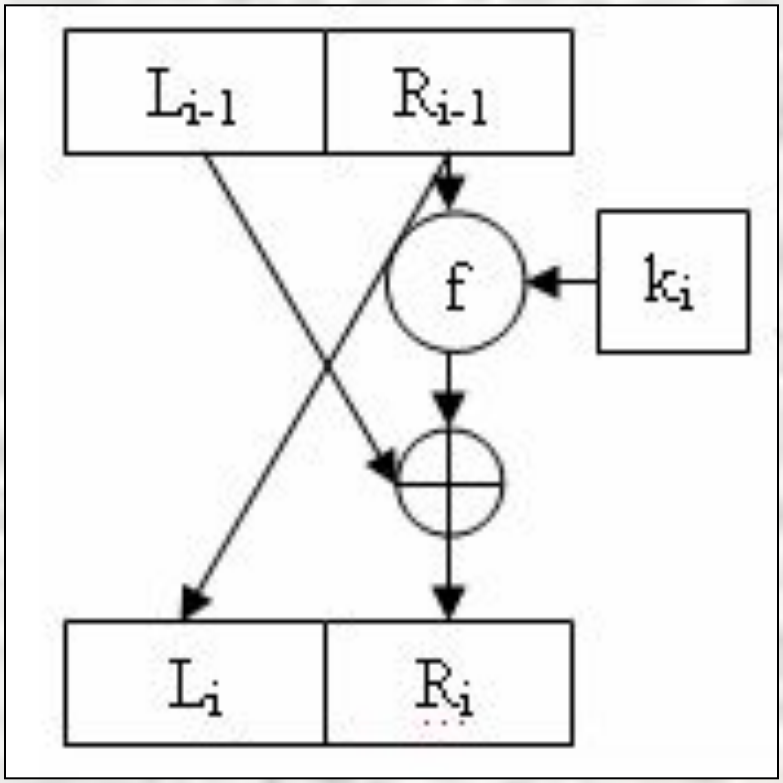
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

При этом бит 58 блока T становится битом 1, бит 50 – битом 2 и т.д. Полученный после перестановки блок IP(T) разделится на две половины:

- L0, состоящий из 32 старших бит;

- R0, состоящий из 32 младших бит.

Затем выполняется итеративный процесс шифрования, состоящий из 16 циклов *преобразования Фейстеля*.



Начальная перестановка IP.

Пусть

$T_{i-1} = L_{i-1}R_{i-1}$ – результат (i-1) итерации.

Тогда результат i-й итерации

$T_i = L_iR_i$ определяется формулами:

$$(*)\begin{cases} L_i = R_{i-1} \\ R_i = L_{i-1} \oplus f(R_{i-1}, k_i),\ i = \overline{1,16} \end{cases}$$

Функция f называется *функцией шифрования*.
Ее аргументами являются 32-битовый вектор R_{i-1} и 48-битовый ключ k_i, который является результатом преобразования 56-битового ключа шифра k.
Результатом последней итерации является блок $T_{16} = R_{16} L_{16}$.
По окончании шифрования осуществляется восстановление позиций битов применением к T_{16} обратной перестановки IP^{-1}.

При расшифровании данных все действия выполняются в *обратном порядке*, при этом вместо соотношений (*) следует использовать соотношения, пользуясь которыми можно «спуститься» от L_{16} и R_{16} к L_0 и R_0.

$$\begin{cases} R_i = L_{i-1} \\ L_{i-1} = R \oplus f(L_i, k_i), i = \overline{16,1} \end{cases}$$

Для вычисления среднего значения f используется:

- ✔ ***функция расширения E;***
- ✔ ***преобразование S, составленное из 8 преобразований S-блоков S_1, S_2, …, S_8;***
- ✔ ***перестановка P.***

Аргументами функции f являются вектор R_{i-1} (32 бита) и вектор k_i (48 бит).

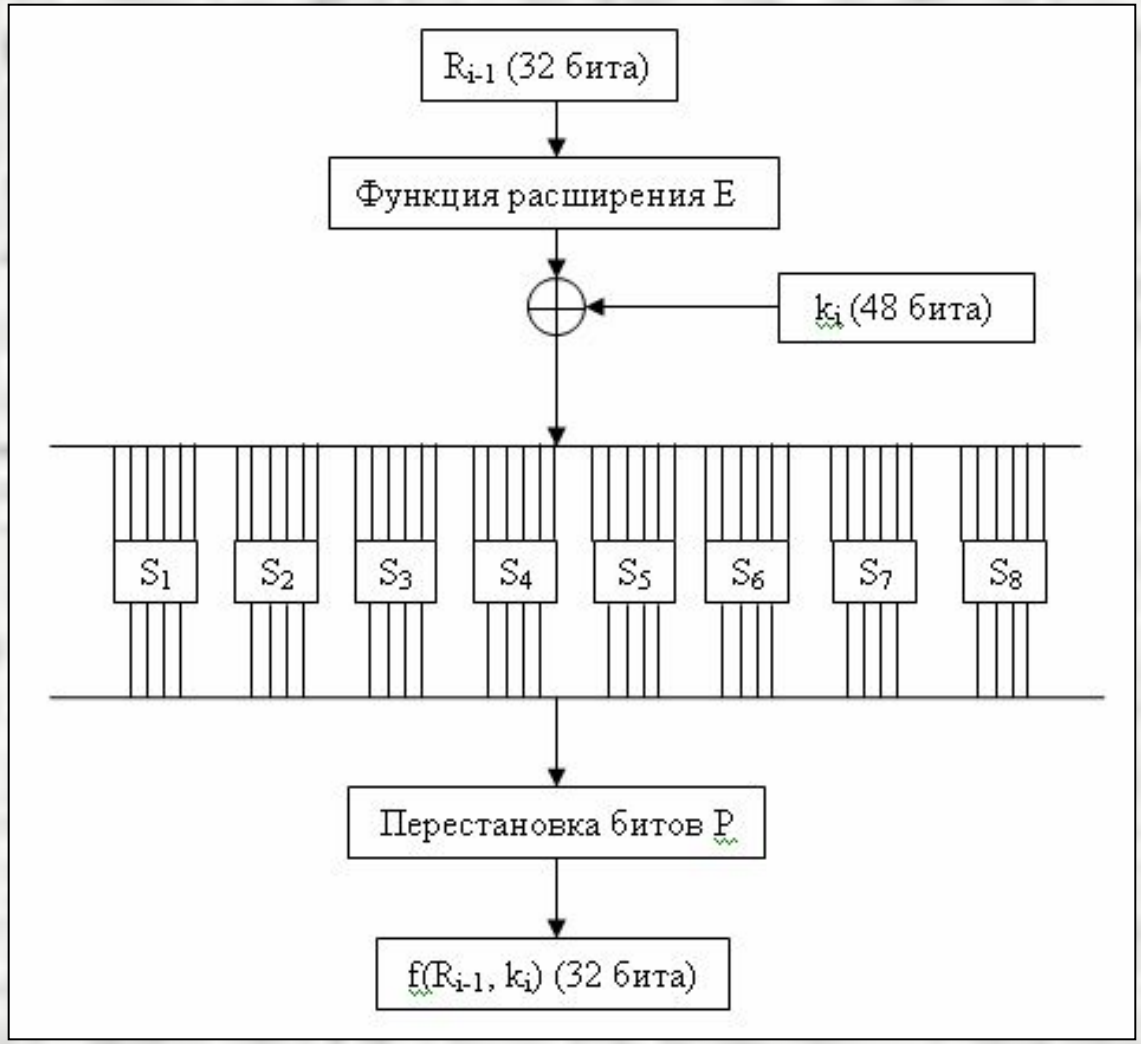


Схема вычисления значения функции $f(R_{i-1}, k_i)$

Функция E «расширяет» 32-битовый вектор R_{i-1} до 48-битового вектора $E(R_{i-1})$ путем дублирования некоторых битов вектора R_{i-1}, при этом порядок следования битов в $E(R_{i-1})$ указан в таблице:

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
29	29	30	31	32	1

Первые 3 бита $E(R_{i-1})$ – это соответствующие биты 32, 1, 2 вектора R_{i-1}, а последние 3 бита – это соответствующие биты 32, 32, 1 вектора R_{i-1}. Полученный результат складывается побитно по *mod 2* с текущим значением ключа R_i и затем представляется в виде 8 последовательных блоков B1, B2, …, B8.

$$E(R_{i-1}) + k_i = B_1 \ldots B_8$$

Далее каждый из блоков B_j трансформируется в 4-битовый блок B'_j с помощью подходящих таблиц S-блока S_j , список которого приведен в таблице:

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

Преобразование блока B_j в B'_j осуществляется следующим образом.

Пусть, например,

B_2=111010

Первый и последний разряды B_2 являются двоичной записью числа a, $0 \le a \le 3$.

Аналогично, средние 4 разряда представляют число b, $0 \le b \le 15$.

В примере $a=10_2=2_{10}$; $b=1101_2=13_{10}$.

Строки и столбцы таблицы S_2 пронумерованы числами *a* и *b*, таким образом, пара (*a*, *b*) однозначно определяет число, расположенное на пересечении строки с номером *a* и столбца с номером *b*.

В данном случае это число равно 3. Записывая его в двоичной форме, получаем B'_2=0011.

		НОМЕР СТОЛБЦА																
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
НОМЕР СТРОКИ	0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7	S_1
	1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8	
	2	4	1	4	8	13	6	2	11	15	12	9	7	3	10	5	0	
	3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13	
	0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10	S_2
	1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5	
	2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15	
	3	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9	
	0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8	S_3
	1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1	
	2	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7	
	3	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12	
	0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15	S_4
	1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9	
	2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4	
	3	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14	
	0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9	S_5
	1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6	
	2	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14	
	3	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3	
	0	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11	S_6
	1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8	
	2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	1	6	
	3	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13	
	0	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1	S_7
	1	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6	
	2	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2	
	3	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12	
	0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7	S_8
	1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2	
	2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8	
	3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11	

Значение $f(R_{i-1},k_i)$ теперь получается применением перестановки битов P, заданной таблицей к результативному 32-битовому блоку $B'_1, B'_2 \ldots B'_8$

На каждой итерации используется текущее значение ключа k_i (48 бит), получаемое из исходного ключа k случайным образом.

Сначала пользователь выбирает сам ключ k, содержащий 56 случайных значащих битов. 8 битов, находящиеся в позициях 8,16,…,64 добавляются в ключ таким образом, чтобы каждый байт содержал нечетное число единиц. Это используется для обнаружения ошибок при обмене и хранении ключей.

Значащие 56 бит ключа подвергаются *перестановке*, приведенной в таблице.

C_0	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
D_0	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

Эта перестановка определяется двумя блоками C_0 и D_0 по 28 бит в каждом. Так, первые три бита в C_0 есть соответствующие 57, 49, 41 биты ключа. Затем индуктивно определяются блоки C_i и D_i i=1,…, 16.

Если уже определены C_{i-1} и D_{i-1}, то C_i и D_i получаются из них одним или двумя левыми циклическими сдвигами согласно таблице:

I	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Число сдвигов	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Теперь определим ключи k_i, $1 \le i \le 16$.
Ключ k_i состоит из 48 битов, выбираемых из битов блока C_i D_i согласно таблице:

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

Первыми тремя битами в k_i являются биты 14, 17, 11 из блока C_i D_i. Отметим, что 8 из 56 бит (9, 18, 22, 25, 35, 38, 43, 54) из C_i D_i отсутствуют в k_i.

Замечания.

Нелинейность преобразований, осуществляемых DES, определяется только s-блоками. Их выбор не имеет достаточно обстоятельного обоснования. Высказывались мнения о том, что s-блоки имеют некоторую «лазейку», позволяющую осуществить контроль за шифрованной перепиской.

Официальная версия:

В 1976 г. АНБ заявило, что выбор s-блоков определен следующими требованиями:

1) каждая строка табличного задания каждого 8-блока должна являться перестановкой множества {0, 1, …, 15};

2) s-блоки не должны являться линейными или аффинными функциями своих входов;

3) изменение одного бита входа s-блока должно приводить к изменению по крайней мере двух битов выхода;

4) для каждого s-блока и любого входа x значение S(x) и S(x (0, 0, 1, 1, 0, 0)) должны различаться по крайней мере двумя битами.

Криптоанализ DES приводит ко многим нелинейным системам уравнений.

Известные аналитические методы вскрытия DES не дают существенного выигрыша по сравнению с полным перебором всего множества из 2^{56} ключей.

К недостаткам DES относится небольшое (по современным меркам) число ключей, что дает возможность их полного перебора на ЭВМ за реальное время.

Официально DES является стандартом шифрования данных до 31.12.1998г.

В 1997г. был объявлен конкурс на новый стандарт, который был назван AES (Advanced Encryption Standard). 2 октября 2000г. Национальный институт стандартов и технологий (НИСТ) США объявил победителя конкурса AES.

Контрольные вопросы:

1. **Указать достоинство стандарта шифрования DES.**
2. **Какой объем блоков данных и ключа в DES-алгоритме?**
3. **Описать схему алгоритма шифрования DES.**
4. **Описать итеративный процесс шифрования (сеть Фейстеля).**
5. **Указать недостаток алгоритма шифрования DES.**

Список используемых источников:

1. Бубнов А.А. Основы информационной безопасности. – М.: Академия, 2017. - 256 с.
2. Ерохин В.В. Безопасность информационных систем. - М. : Флинта, 2016. - 184 с.
3. Гашков С.Б. Криптографические методы защиты информации. - М.: Академия, 2010. - 300с.
4. Мельников В.П. Информационная безопасность. – М.: Академия, 2013. - 336 с.

Список используемых источников:

5. **Мельников В.П. Защита информации. – М.: Академия, 2014. - 304 с.**
6. **Бабаш А.В. Информационная безопасность. Лабораторный практикум. - М. : КНОРУС, 2013. - 136 с.**
7. **Платонов В.В. Программно-аппаратные средства защиты информации. – М.: Академия, 2014. - 336 с.**