ЛЕКЦИЯ 5.

Стандарт шифрования данных DES (Data Encryption Standard)

В 1972 году **Национальное бюро стандартов** (***National Bureau of Standards***, **NBA**), ныне - **Институт стандартов и технологий (*National Institute of Standards and Technology*, NIST)**, выступило инициатором разработки единого − стандартного, криптографического алгоритма.

Было приведено несколько **критериев оценки** проекта:

- Алгоритм должен обеспечивать высокий уровень безопасности.

- Алгоритм должен быть полностью определен и легко понятен.

- Безопасность алгоритма должна основываться на ключе и не должна зависеть от сохранения в тайне самого алгоритма.

- Алгоритм должен быть доступен всем пользователям.

- Алгоритм должен позволять адаптацию к различным применениям.

- Алгоритм должен позволять экономичную реализацию в виде электронных приборов.

- Алгоритм должен быть эффективным в использовании.

- Алгоритм должен предоставлять возможности проверки.

- Алгоритм должен быть разрешен для экспорта.

Стандарт шифрования данных DES 23 ноября 1976 года был принят в качестве федерального стандарта и разрешен к использованию на всех несекретных правительственных коммуникациях.

Официальное описание стандарта было опубликовано **15 января 1977** года и вступило в действие шестью месяцами позже.

Американский национальный институт стандартов (***American National Standards Institute,*** **ANSI**) одобрил DES в качестве стандарта для частного сектора в **1981** году, назвав его

Алгоритмом шифрования данных (*Data Encryption Algorithm,* DEA).

Структура DES

1. **DES** представляет собой **блочный шифр**, он шифрует данные 64-битовыми блоками.

2. **DES** является симметричным алгоритмом: для шифрования и дешифрирования используются ***одинаковые алгоритм и ключ*** (за исключением небольших различий в использовании ключа).

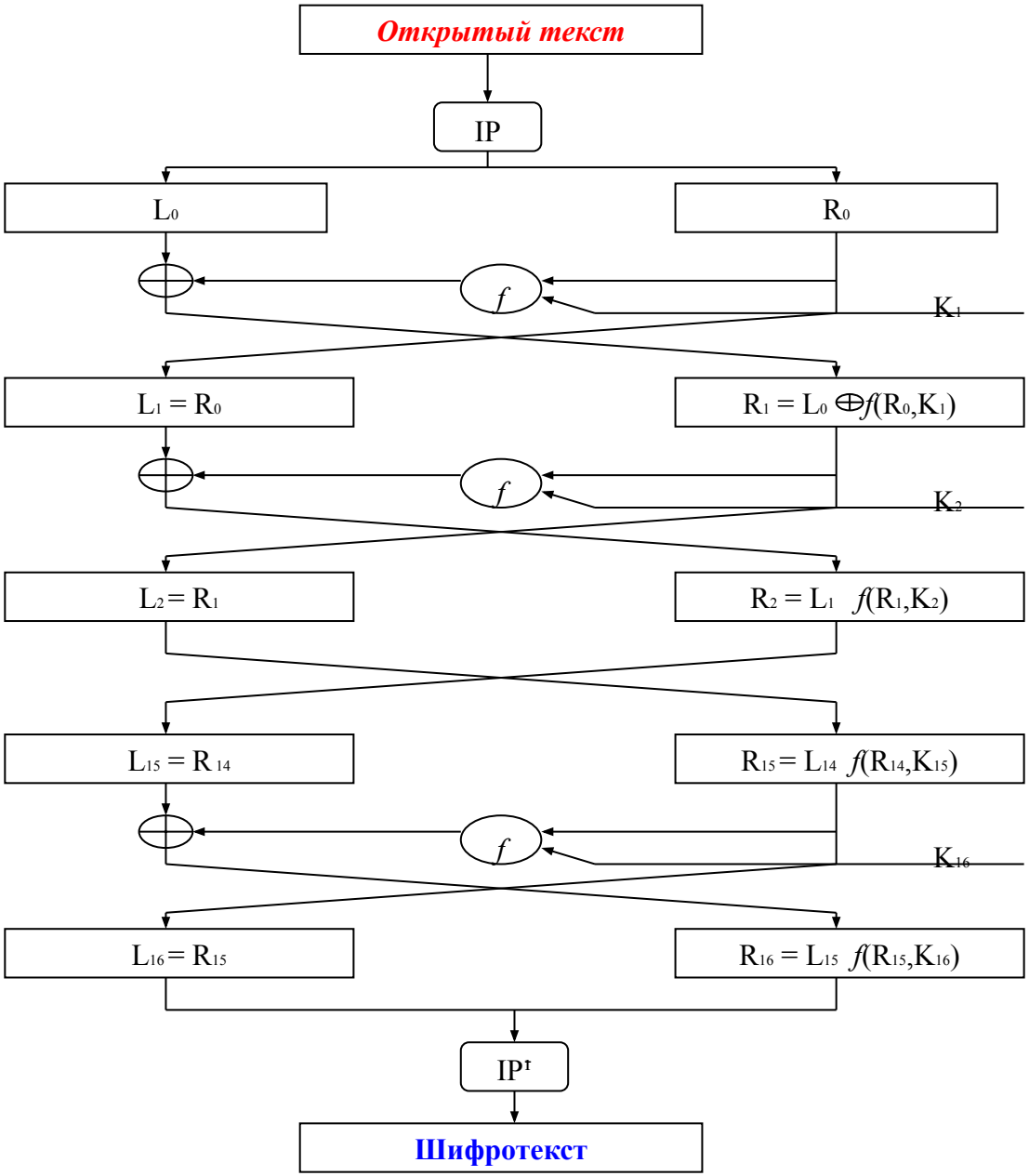
Длина ключа равна 56 битам.

Безопасность полностью определяется ключом.

3. Алгоритм представляет собой *комбинацию* двух основных методов шифрования: ***смещения и диффузии***.

Фундаментальным строительным блоком DES является применение к тексту единичной комбинации этих методов (подстановка, а за ней – перестановка), зависящей от ключа. Такой блок называется **этапом**.

4. **DES** состоит из **16** этапов, одинаковая комбинация методов применяется к открытому тексту ***16 раз.***



Алгоритм DES

Схема алгоритма

DES работает с 64-битовым блоком открытого текста:

1. после **первоначальной перестановки**
2. блок **разбивается на правую и левую половины** длиной по 32 бита;
3. затем выполняются **16 этапов одинаковых действий**, называемых **функцией *f***, в которых данные объединяются с ключом;
4. после шестнадцатого этапа **правая и левая половины объединяются;**
5. алгоритм завершается **заключительной перестановкой** (*обратной* по отношению к первоначальной).

На каждом этапе биты ключа *сдвигаются*, затем из **56** битов ключа выбирается **48** битов.

Функцией *f* выполняются *четыре* операции:

− правая половина данных увеличивается до **48** битов с помощью ***перестановки с расширением***,

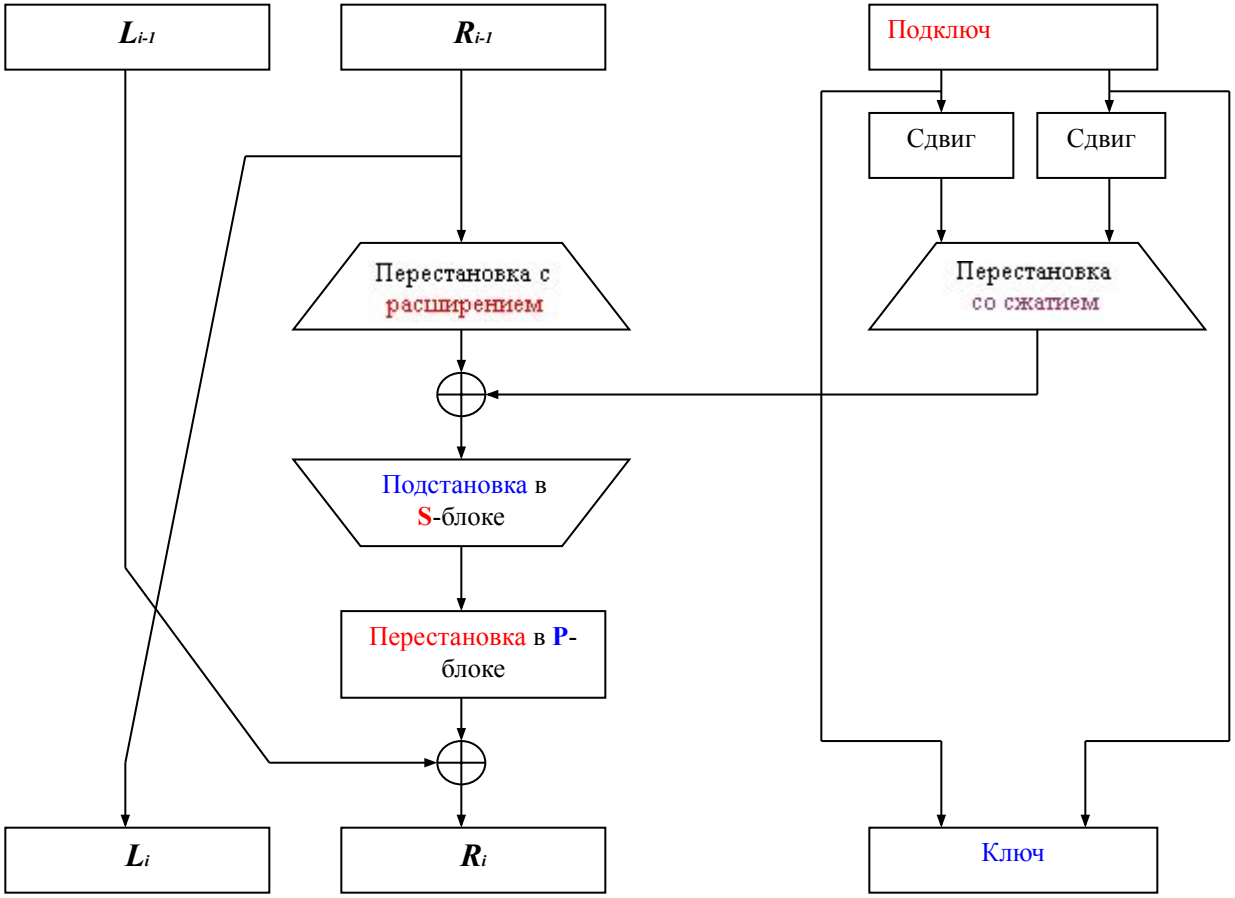
− ***объединяется*** посредством **XOR** (*«исключающее ИЛИ», суммирование по модулю 2*) с **48** битами смещенного и переставленного **ключа**,

− проходит через **8 S-блоков**, образуя **32** новых бита,

− ***переставляется*** снова.

Затем результат функции **f** *объединяется с* ***левой половиной*** с помощью **XOR**. В итоге этих действий появляется **новая правая половина**, а *старая правая* половина становится **новой левой**.

Эти действия повторяются **16** раз, образуя **16 этапов DES**.



Один этап DES

Если

L_i и R_i – левая и правая половины *кода* – **результата i-той итерации**,
K_i – 48-битовый ключ для этапа i,
f – функция, выполняющая все подстановки, перестановки и ***XOR*** с ключом,
то этап можно представить как:

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$$

Начальная перестановка

Начальная перестановка выполняется еще до **этапа 1**
и
служит для облегчения *побайтной загрузки* данных открытого текста и шифротекста в микросхему **DES**.

Начальная перестановка

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

Преобразования ключа

1. Сначала **64**-битовый ключ DES ***уменьшается*** до **56**-битового ключа *отбрасыванием каждого восьмого* бита. (Эти биты используются только для контроля четности, позволяя проверять правильность ключа).

2. После извлечения 56-битового ключа для каждого из 16 этапов DES генерируется новый 56-битовый **подключ**. Эти подключи **Ki** определяются следующим образом:

Перестановка ключа − формирование подключа

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

3. Далее 56-битовый подключ **делится** на две **28**-битовых половинки.

4. Затем эти половинки ***циклически сдвигаются*** **налево** на один или два бита в зависимости от этапа.

Число битов сдвига ключа в зависимости от этапа

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

5. После сдвига выбирается **48** из **56** битов. Так как при этом не только выбирается подмножество битов, но и ***изменяется их порядок***, эта операция называется **перестановкой со сжатием**.

Ее результатом является набор из **48** битов.

Перестановка со сжатием

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,	32,	37,	47,	55,	30,	40,	51,	45,	33,	48,
44,	49,	39,	56,	34,	53,	46,	42,	50,	36,	29,	32

Перестановка с расширением

Операция **расширяет** *правую половину* данных R_i от ***32*** до **48** битов.

Так как при этом не просто повторяются определенные биты, но и **изменяется их порядок**, эта операция называется **перестановкой с расширением**.

Операция решает две задачи: привести размер правой половины ***в соответствие с ключом*** для операции **XOR** и ***получить более длинный результат***, который можно будет *сжать* в ходе операции подстановки.

Криптографический смысл:

за счет влияния ***одного бита*** на две подстановки быстрее возрастает **зависимость** битов *результата* от битов *исходных данных*.

Это называется **лавинным эффектом**.

Перестановка с расширением иногда называется **E-блоком** (от ***expansion***).

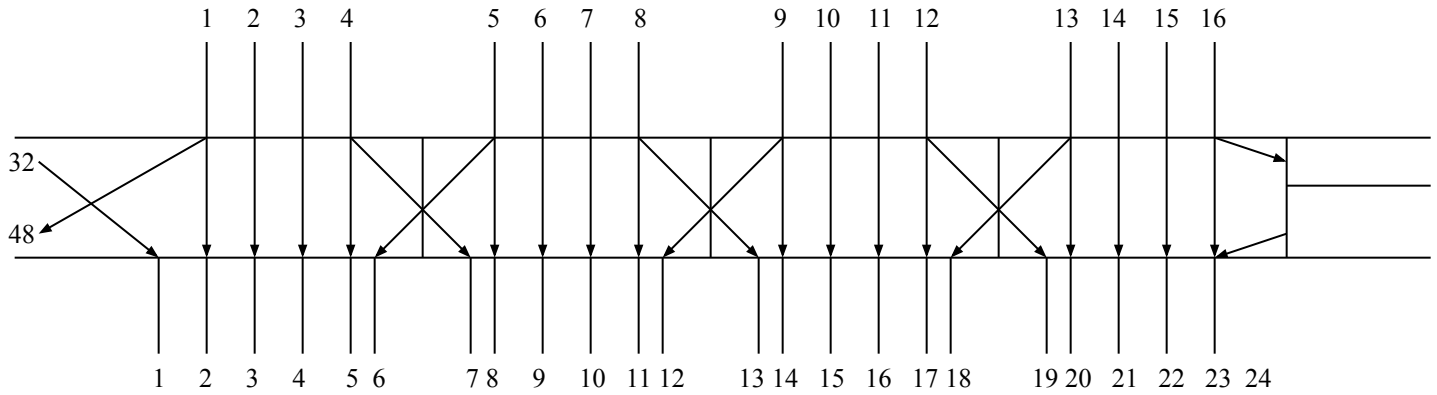


Схема перестановки с расширением

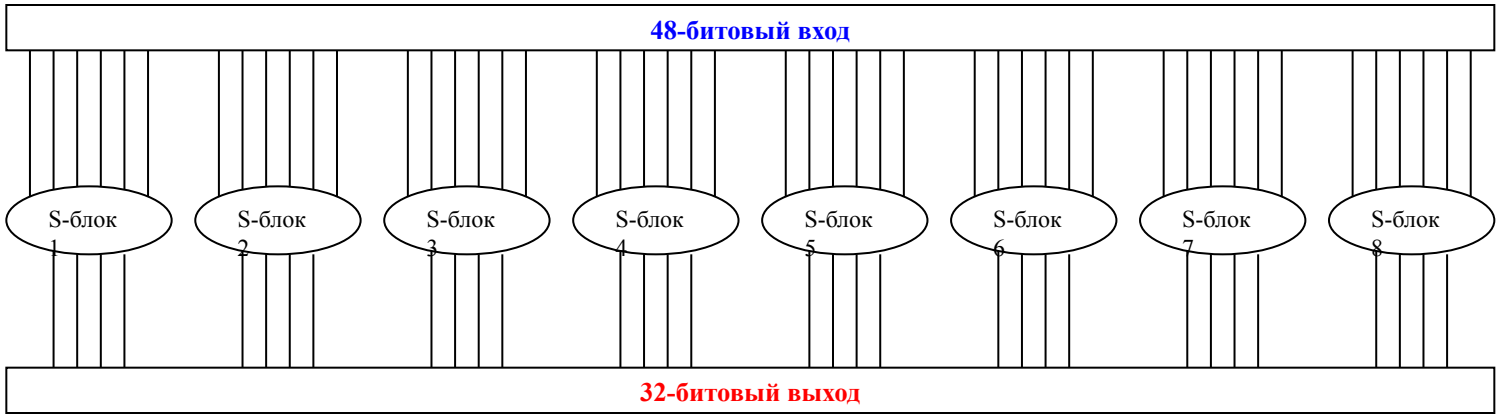
Перестановка с расширением

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

Подстановка с помощью S-блоков

Подстановки производятся в восьми блоках подстановки, или **S-блоках** (от ***substitution***).

1. Каждый S-блок имеет **6-битовый вход** и ***4-битовый выход*** .
2. **48** битов делятся на *восемь* **6-битовых** подблоков.
3. Каждый отдельный подблок обрабатывается отдельным S-блоком: ***первый*** подблок – S-блоком 1, ***второй*** – S-блоком 2, и так далее.



Подстановка с помощью S-блоков

S-блоки

S-блок 1:															
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
S-блок 2:															
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
S-блок 3:															
10,	0,	9,	14,	6,	3,	25,	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
S-блок 4:															
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

S-блок 5:															
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
S-блок 6:															
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
S-блок 7:															
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
S-блок 8:															
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

СТРУКТУРА ПОДСТАНОВКИ

1. Каждый **элемент** в блоке является ***4-битовым*** числом.

2. По **6** входным битам S-блока определяется, под какими номерами столбцов и строк искать выходное значение − входные биты *особым образом* определяют элемент S-блока.

6-битовый вход S-блока: $b_1, b_2, b_3, b_4, b_5, b_6$.

Биты b_1 и b_6 *объединяются*, образуя **2-битовое** число от **0 до 3,** соответствующее строке таблицы.

Средние четыре бита − с b_2 **по** b_5, *объединяются*, образуя **4-битовое** число от **0 до 15**, соответствующее столбцу таблицы.

3. Каждый S-блок можно рассматривать как ***функцию подстановки*** 4-битового элемента:

Биты b_2 **по** b_5 являются ***входом***, а некоторое 4-битовое число – ***результатом***.

Биты b_1 **и** b_6 определяют ***одну из четырех функций подстановки***, возможных в данном S-блоке.

Подстановка с помощью S-блоков является **ключевым этапом** DES.

S-блоки **нелинейны**, и именно они в большей степени, чем все остальное, обеспечивают **безопасность DES**.

В результате этого этапа подстановки получаются ***восемь 4-битовых*** блоков, которые вновь объединяются в единый **32-битовый блок,**

поступающий на вход следующего этапа –

перестановки с помощью P-блоков.

Перестановка с помощью Р-блоков

Эта перестановка перемещает каждый входной бит в **другую** позицию:

- ни один бит не используется дважды,
- ни один бит не игнорируется.

Этот процесс называется **прямой перестановкой** или просто **перестановкой**.

Перестановка с помощью Р-блоков

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

Результат перестановки с помощью Р-блока **объединяется** посредством **XOR** с *левой половиной первоначального 64-битового блока*.

Затем левая и правая половины **меняются местами**, и начинается следующий этап.

Заключительная перестановка

Заключительная перестановка является обратной по отношению к начальной.

Левая и правая половины не меняются местами после последнего этапа DES, вместо этого объединенный блок $R_{16}L_{16}$ используется как *вход заключительной перестановки*.

Заключительная перестановка

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

Дешифрование DES

DES позволяет использовать для шифрования и дешифрования блока ***одну и ту же функцию.***

Единственное отличие состоит в том, что ключи должны использоваться в *обратном порядке*.

(Если на этапах шифрования использовались ключи **К1, К2, К3,…,К16**, то ключами дешифрирования будут ***К16, К15, К14,…, К1)***.

Алгоритм, который создает **ключ для каждого этапа**, также *цикличен*.
Ключ сдвигается **вправо**, а число позиций сдвига равно

0,1,2,2,2,2,2,2,1,2,2,2,2,2,2,1.

Режимы DES

Определяют **четыре** режима работы:

ECB,
CBC,
OFB
CFB.

Банковские стандарты ANSI определяют для **шифрования** *ECB* и *CBC*, а для **проверки подлинности** – *CBC* и *n-битовый CFB*.

Аппаратные и программные реализации DES

Коммерческие микросхемы DES

Производитель	Микросхема	Год	Тактовая частота	Скорость данных	Доступность
AMD	Am9518	1981	3 МГц	1.3 Мбайт/с	Н
AMD	Am9518	?	4 МГц	1.5 Мбайт/с	Н
AMD	AmZ8068	1982	4 МГц	1.7 Мбайт/с	Н
AT&T	T7000A	1985	?	1.9 Мбайт/с	Н
CE-Infosys	SuperCrypt CE99C003	1992	20 МГц	12.5 Мбайт/с	Д
CE-Infosys	SuperCrypt CE99C003A	1994	30 МГц	20.0 Мбайт/с	Д
Cryptech	Cry12C102	1989	20 МГц	2.8 Мбайт/с	Д
Newbrige	CA20C03A	1991	25 МГц	3.85 Мбайт/с	Д
Newbrige	CA20C03W	1992	8 МГц	0.64 Мбайт/с	Д
Newbrige	CA95C68/18/09	1993	33 МГц	14.67 Мбайт/с	Д
Pijnenburg	PCC100	?	?	2.5 Мбайт/с	Д
Semaphore Communications	Roadrunner284	?	40 МГц	35.5 Мбайт/с	Д