Дәріс 3. Блокты шифрларды құру принциптері

Жоспар:

1. Блоктық шифрларды құрудың негізгі принциптері
2. DES шифры
3. DES алгоритмінің жұмыс режимдері

Блок шифрлары ашық мәтін блоктарын шифрлауға арналған [14, с. 96]. Әр блокты түрлендіру нәтижесінде бірдей ұзындықтағы шифрланған мәтін блогы алынады. Қазіргі заманғы блок шифрларында әдетте биттік деректер көрінісі қолданылады, яғни ашық және шифрланған мәтіндер нөлдер мен бекітілген ұзындық бірліктерінің тізбегі болып табылады.

Клод Шеннон блоктық шифрларды құрудың негізгі принциптерін тұжырымдады.

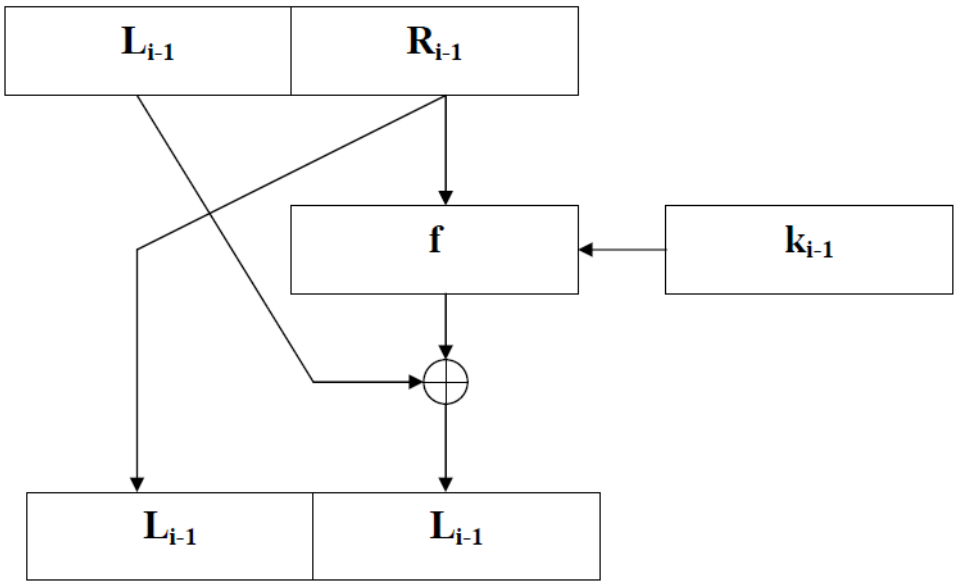
Біріншіден, ол бірнеше қарапайым карталардың суперпозициясы арқылы күрделі түрлендірулерді жүзеге асыруды ұсынды. Оның моделіне сәйкес, блоктық шифрлар қарапайым түрлендірулерді ашық мәтін блоктарына бірнеше рет қолдану арқылы жасалады, олар оңай жүзеге асырылады, бірақ жеткілікті қайталанулармен шифрлаудың жоғары күрделілігін қамтамасыз етеді.

Екіншіден, Шеннон бұл түрлендірулер екі маңызды криптографиялық принципке сәйкес келуі керек екенін атап өтті, «шашырау» және «араластыру».

Дисперсия принципі ашық мәтіннің бір таңбасының әсері шифрланған мәтіннің мүмкіндігінше көп таңбаларына таралуы. Бұл ашық мәтін таңбалары арасындағы статистикалық тәуелділіктерді жасыруға мүмкіндік береді.

Араластыру принципі кілт пен шифрланған мәтін арасындағы байланысты қиындатуға бағытталған. Басқаша айтқанда, түрлендіру ашық және шифрланған мәтін арасындағы байланысты анықтау мүмкін болмайтындай етіп ұйымдастырылуы керек.

Бұл талаптарды, атап айтқанда, Фейстель шифрының құрылымы қанағаттандырады. 2.1-суретте осы шифрдың бір раундының схемасы көрсетілген.



2.1 – сурет. Фейстель шифры (бір айналым)

Шашырау және араластыру қасиеттеріне ие болған бірінші блоктық шифрды өткен ғасырдың 60 - жылдарында IBM компаниясы жасаған, осы шифр негізінде Люцифер (*Lucifer*) шифрлау жүйесі құрастырылды [15, 78 б.]. Люцифер жүйесі алмастыру және ауыстыру әдістерінің комбинациясына (алмастыруына) негізделген. Ол ауыстыру және ауыстыру блоктарының күйлерін басқаратын 128 биттік кілтті пайдаланды. Люцифер жүйесі салыстырмалы түрде төмен шифрлау жылдамдығына байланысты практикалық іске асыру үшін өте қиын болды, яғни 2190 байт/сек. бағдарламалық қамтамасыз етуді енгізу және 96970 байт/сек. аппараттық қамтамасыз етумен жүзеге асырылады.

DES шифры

Ең алғашқы кеңінен қабылданған және қолданылатын деректерді шифрлау стандарттарының бірі DES (Data Encryption Standard) болды. Бұл стандарт Америкада 1977 жылы жарияланған. DES стандартының алдында IBM компаниясының Люцифер жүйесі болды. Оның үстіне DES шифрлау алгоритмі Люциферде қолданылатын алгоритмге өте ұқсас [16, 85 б.]. Сондай-ақ, DES стандарты Фейстель шифрінің өкілі, дәлірек айтқанда, жалпылама болып табылатынын ескеру қажет.

Жалпы DES алгоритмі блоктық шифр болып табылады, яғни белгіленген өлшемдегі мәліметтерді бір уақытта n битке түрлендіретін шифр (*DES жүйесі үшін n мәні 64*). Сондықтан ашық мәтінді DES алгоритмімен шифрлау алдында оны 64 биттік (8 байт) блоктарға бөлу керек. Әрі қарай әрбір i-ші ($i = 1,2,..,N$, мұндағы N ашық мәтін блоктарының саны) ашық мәтіндік блок екі тең бөлікке бөлінеді, яғни сол жақ L_i және оң жақ R_i бөліктері әрқайсысы 32 бит. Содан кейін әрбір блок үшін Фейстель шифрінің құрылымына сәйкес 16 түрлендіру айналымы орындалады. Трансформацияның әрбір раунды басқа кілтті пайдаланады.

Мәтінді кері шифрлаудың шифрлаудан еш айырмашылығы жоқ. Дәл

осындай алгоритм пайдаланылады, жалғыз айырмашылығы кіріс 64 бит шифрленген мәтін блогы, ал пернелер кері тәртіпте қолданылады. *DES* алгоритмін толығырақ көрсетуге көшейік. 2.2-суретте *DES* алгоритмінің жалпы диаграммасы көрсетілген.

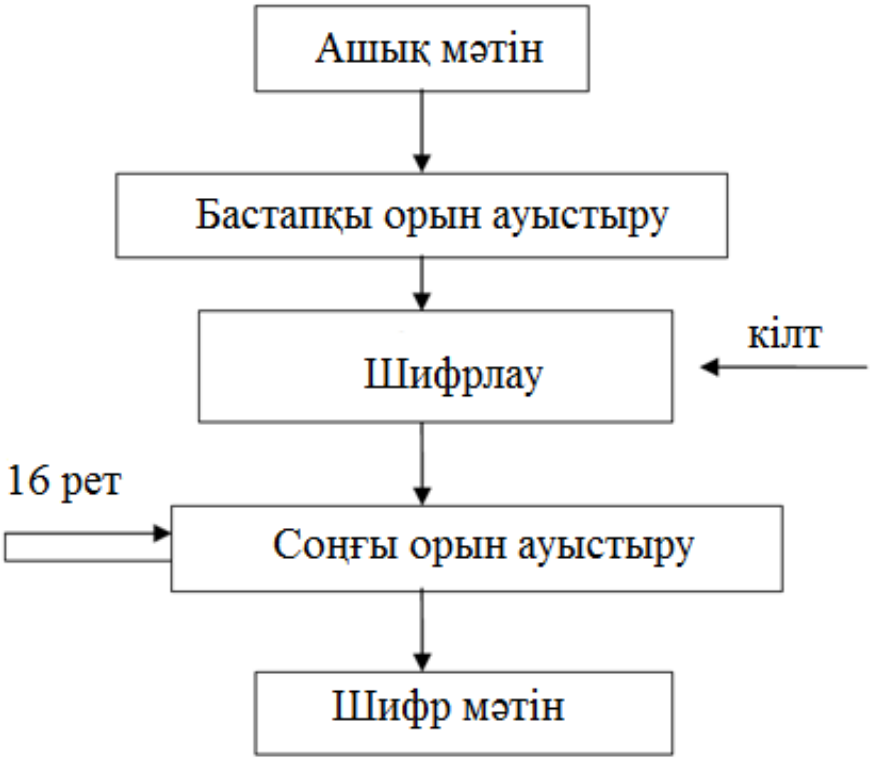
DES алгоритмінің негізгі кезеңдерін тізіп көрейік:

1) 56 биттен тұратын кездейсоқ Q тізбегі жасалады;

2) 8, 16, 24, ..., 64 позицияларындағы паритетті тексеру үшін Q тізбегіне сегіз тексеру биті қосылады. 64 биттен тұратын U блогы алынды;

3) U блогынан басқару биттерін алып тастау және K шифрлау кілтін қалыптастыру үшін U блогы $G(U)$ функциясының көмегімен түрлендіріледі. G функциясы стандартты кесте түрінде анықталған, ол өзгеріссіз қолданылуы керек. Түрлендіру нәтижесінде өлшемі 56 бит болатын $K = G(U)$ блогы алынады. K блогы әрқайсысы 28 биттен тұратын екі жартыға бөлінген C_0 және D_0;

4) C_0 және D_0 көмегімен C_i және D_i , i = 1, 2, …, дәйекті түрде анықталады. C_i және D_i қалыптастыру үшін бір немесе екі биттің циклдік солға жылжыту операциялары қолданылады. Ауыстыру мөлшері стандартты кестемен анықталады. C_i және D_i үшін ауыстыру операциялары тәуелсіз орындалады. C_5 тізбегі C_4-тен 2 бит солға айналдыру арқылы және D_5 2 бит солға бұру арқылы алынады (кілттік есептеу үшін жылжулардың 1–кестесін қараңыз). Төртінші кезеңнің нәтижесінде *DES* алгоритмінің 16 айналымына 16 кілт генерацияланады;



2.2 – сурет. DES алгоритмнің жалпы сұлбасы

5) Стандартты кестеде көрсетілген H ауыстыруды пайдалану арқылы

әрбір 56 биттік кілт және 48 биттік блокқа түрлендіріледі. Бұл түрлендіру негізгі генерациялау кезеңін аяқтайды;

6) L_0 және R_0 екі 32 биттік блоктар ретінде ұсынылған ағымдағы 64 биттік ашық мәтін блогы тіркелген стандартты кестемен көрсетілген бастапқы IP ауыстыру арқылы түрлендіріледі;

7) Келесі формулалар (2.1), (2.2), (2.3) арқылы түрлендірудің 16 айналымы орындалады

$$L_i = R_{i-1}, \quad (2.1)$$
$$R_i = L_i \oplus f(R_{i-1}, K_i), \quad (2.2)$$
$$i = 1, 2, \dots, 16. \quad (2.3)$$

$f(R_{i-1}, K_i)$ функциясы қарапайым түрлендірулердің белгілі бір суперпозициясы болып табылады, оның толық сипаттамасы төменде келтірілген. $f(R_{i-1}, K_i)$ функциясы сызықты емес екенін ескеріңіз. $f(R_{i-1}, K_i)$ функциясының сызықты еместігі S блоктармен қамтамасыз етіледі.

Он алтыншы раундтың нәтижесі R_{16}, L_{16} түрлендіруден өтеді, бұл IP^{-1} ауыстыру IP ауыстыруына кері, осы схеманың алтыншы кезеңі. Бұл түрлендіру DES алгоритмімен ашық мәтіндік блокты шифрлаудың соңғы кезеңі болады.

Назар аударыңыз. DES-те қолданылатын барлық кестелер стандартты болып табылады және DES алгоритмін жүзеге асыруда өзгеріссіз қосылуы керек. Кестелер кілтті болжау арқылы кері шифрлау процесін барынша қиындататындай етіп жасалған.

2.1 – кесте. Бастапқы кілтті дайындаудың G функциясы

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

DES алгоритмінің жеке кезеңдерін егжей-тегжейлі көрсетуге көшейік.

Алгоритмнің жалпы сұлбасында кілттерді генерациялау 5 қадамды қамтиды. 56 биттік кілтке биттер 8, 16, 24, ..., 64 позицияларына қосылады, осылайша әрбір байтта бірлердің тақ саны болады. Бұл факт кілт алмасу және сақтау кезіндегі қателерді анықтау үшін қолданылады [17]. Осы түрлендіру нәтижесінде өлшемі 64 бит болатын U блогы алынады. U блоктан басқару биттерін алып тастау және өлшемі 56 биттің K_0 бастапқы кілт мәнін қалыптастыру үшін G функциясы кілтті бастапқы дайындауға арналған U

блогына қолданылады. G функциясы 2.2 – кестеде көрсетілген.

2.2 – кесте. Кілттерді есептеу үшін ауысу кестесі (бірінші жол айналым саны, екінші жол ауысым саны)

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

G функциясына сәйкес $K_0 = G(U)$ кілтінің бірінші биттері сәйкесінше U блогының 57, 49, 41 және т.б. биттері болады. K_0 соңғы 56 биті U блогының төртінші биті болады. K_0 кілтін қалыптастыру кезінде U блогының 8, 16, 24, ..., 64 позицияларынан биттер пайдаланылмайтынын ескеріңіз, өйткені K_0 қалыптастыру кезінде басқару разрядтары жойылады. G функциясы U блогын түрлендіру кезінде осы блоктың тек 56 битін пайдаланады. Нәтижесінде түрлендіру көлемі 56 биттен тұратын K_0 бастапқы кілтін анықтайды, ол әрқайсысы 28 биттен екі тең бөлік ретінде ұсынылған. Бастапқы кілттің сол жағы C_0, ал оң жағы D_0 арқылы белгіленеді. Түрлендіру нәтижесін (2.4) формула түрінде жазуға болады

$$K_0 = G(U) = C_0 D_0 \qquad (2.4)$$

C_0 және D_0 біле отырып, біз C_iжәне D_i, $i = 1, \dots, 16$ он алты блокты ретімен анықтаймыз. Ол үшін кілтті есептеу үшін ауысым кестесі қолданылады. Циклдік солға ауыстыру операциялары C_i және D_iблоктарында дербес орындалады. Мысалы, C_9 тізбегі C_8–ден бір бит солға айналдыру арқылы, ал D_9 бір бит солға айналдыру арқылы D_8 алынады.

Әрбір $i, i = 1, \dots, 16$раунды үшін K_i кілтінің генерациясы әрбір 56–биттік C_i және D_i блогы стандартты кестеде көрсетілген H ауыстыру функциясы арқылы түрлендірілсе аяқталады. H функциясы C_i және D_i, $i = 1, \dots, 16$, 56 бит өлшемін, өлшемі бойынша $K_i, i = 1, \dots, 16$, 48 биттік кілтке түрлендіреді. Н функциясы C_i және D_i, $i = 1, \dots, 16$, өлшемі 56 бит блоктарын түрлендіру кезінде осы блоктың 9,18,22,25,34,35,38,43 биттерін пайдаланбайтынына назар аударыңыз. Бұл түрлендіру DES алгоритмінің он алты айналымы үшін он алты кілтті генерациялау кезеңін аяқтайды. 2.3 – кестеде кілт өңдеуді аяқтайтын H функциясы көрсетілген.

2.3 – кесте. Алгоритмнің H функциясы

Функция H					
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

Әрқайсысы 64 биттен тұратын N блокқа бөлінген ашық мәтінді шифрлау процесінің егжей–тегжейлі сипаттамасына тікелей көшейік. Әрбір ашық мәтін блогы T_i, $i = 1, \dots, N$, стандартты кестеде көрсетілген бастапқы IP ауыстыруынан өтеді. Кіріс блогының биттері IP матрицасына сәйкес қайта реттеледі. Бастапқы ауыстыру матрицасы 2.4–кестеде келтірілген.

2.4 – кесте. Бастапқы IP ауыстыру матрицасы

5 8	5 0	4 2	3 4	2 6	1 8	1 0	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

T_i кіріс блогының 58 биті түрлендірілетін блоктың бірінші биті, 50 биті екінші биті болады және т.б. Бұл ауыстыруды келесі 2.5 – формулада өрнек арқылы сипаттауға болады

$$W_i = IP(T_i). \qquad (2.5)$$

Нәтижесінде W_i бит тізбегі екі реттілікке бөлінеді

L_0 – сол немесе ең маңызды бит,

R_0 – оң немесе төменгі ретті бит.

L_0 және R_0 тізбегі әрқайсысы 32 биттен тұрады. Келесі W_i блогы үшін келесі (2.1), (2.2), (2.3) формулалар арқылы түрлендірудің 16 айналымы орындалады

$$L_i = R_{i-1},$$
$$R_i = L_i \oplus f(R_{i-1}, K_i),$$
$$i = 1, 2, \dots, 16.$$

Мұндағы:

i – дөңгелек сан;

K_i – дөңгелек кілт;

$\oplus$ – қосу модулі 2 (XOR операциясы);

$f()$ – шифрлау функциясы.

Шифрлау функциясын есептеу алгоритмі үшін ER_{i-1} кеңейту функциясы R_{i-1} блогына қолданылады, ол 32–биттік R_{i-1} блогын $R' = E(R_{i-1})$ блогына түрлендіреді, 48 бит өлшемді $E(R_{i-1})$ функциясы стандартты 5–кестемен анықталады. R′ блогын құру кезінде 1, 4, 5, 8, 9, 12, 13, 16, 17, 20, 21, 24, 25, 28, 29, 32 бит разрядтарын екі рет пайдаланылады.

2.5 – кесте. E кеңейту функциясы

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

48 биттік жаңа блок келесі (2.6) формуламен есептеледі:

$$B = E(R_{i-1}) \to K_i$$
$$B = B_1B_2B_3B_4B_5B_6B_7B_8 \qquad (2.6)$$

Әрбір алты биттік B_j блогы S_j функциясын пайдаланып төрт разрядты $S_j(B_j)$ блогына түрлендіріледі. Нәтижесінде біз өлшемі 32 бит болатын блокты аламыз

$$B = S_1(B_1)\, S_2(B_2)\ S_3(B_3)\ S_4(B_4)\ S_5(B_5)\ S_6(B_6)\ S_7(B_7)\ S_8(B_8)$$

Алты разрядты B_j блогын төрт разрядты $S_j(B_j)$ блогына түрлендіру алгоритмі келесідей. Әрбір S_j функциясы стандартты кесте болып табылады, ол 0,1,2,3 сандары бар төрт жолдан және 0,1,2,...,15 сандары бар он алты бағаннан тұрады. Мысалы, кейбір блоктың пішіні болсын

$$B_j = b_1\, b_2\, b_3\, b_4b_5\, b_6 = 110010$$
$$j = 1, \quad 2, \dots, 8.$$

Сонда $b_1b_6 = 10$ разрядтары (10 саны екілік жүйеде жазылады, ондық жүйеде 2 санымен сәйкес келеді) S_j кестесінің $10\ (= 2)$ бағанының санын, ал $b_2\, b_3\, b_4b_5 = 1001$ разрядтарын құрайды (1001 саны екілік жүйеде жазылады, ондық жүйеде 9 санымен сәйкес келеді) S_j кестесінің $1001\ (= 9)$ жолының

формасы. В$B_j = b_1\, b_2\, b_3\, b_4 b_5\, b_6 = 110010$ блогы $b_1 b_6 = 10$ (=2) саны бар жолдың бағанымен қиылысында орналасқан S_j кестесінің екілік мәнімен ауыстырылды. Саны $b_2\, b_3\, b_4 b_5 = 1001$ (=9) B_j блогынан әрбір B_j, j = 1,2,...,8 түрлендіру, біз өлшемі 32 бит болатын жаңа B′ блогын аламыз. 2.6 – кестеде $S_j, j = 1, 2, \ldots, 8$ түрлендіру функцияларын көрсетеді.

2.6 – кесте. $S_1, S_2, \ldots, S_8$ түрлендіру функциялары

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

Алынған блок

$B' = S_1(B_1)\, S_2(B_2)\ S_3(B_3)\ S_4(B_4)\ S_5(B_5)\ S_6(B_6)\ S_7(B_7)\ S_8(B_8)$

өлшемі 32 бит $P(B)$ ауыстыру функциясы блок өлшемі 32 бит арқылы түрлендіріледі. $P(B)$ функциясы стандартты 2.7 – кесте арқылы анықталады.

2.7 – кесте. Орын ауыстыру функциясы P

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

Шифрлау блоктағы (L_{16}, R_{16}) биттердің соңғы ауыстырылуымен аяқталады, бұл он алтыншы айналымдағы түрлендірудің нәтижесі болып табылады. Орын ауыстыру стандартты 2.8 – кестеде келтірілген IP^{-1} функциясымен анықталады.

2.8 – кесте. IP^{-1} кері алмастыру матрицасы

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

Шифрлау блоктағы биттердің соңғы ауыстырылуымен аяқталады (L_{16}, R_{16}), ол он алтыншы айналымдағы түрлендірудің нәтижесі болып табылады. Қайта реттеу тиісті 2.8 – кестеде көрсетілген қысқартылған IP^{-1} анықталған.

L_{16} және R_{16} үшін (2.7), (2.8) формулалар жарамды екенін ескеріңіз:

$$L_{16} = R_{15}, \quad (2.7)$$

$$R_{16} = L_{15} \oplus f(R_{15}, K_{16}). \quad (2.8)$$

(2.7), (2.8) формулаларға сүйене отырып, блоктың декодтауын ұйымдастыру оңай. R_{15} белгілі және L_{16} тең және K_{16} кілтін білетіндіктен, $f(R_{15}, K_{16})$ мәнін есептеу оңай. Осы жерден,

$$R_{16} = L_{15} \oplus f(R_{15}, K_{16}) \text{ и } f(R_{15}, K_{16}),$$

(2.9) формула арқылы L_{15} анықтау оңай

$$R_{16} \oplus f(R_{15}, K_{16}) = L_{15} \oplus f(R_{15}, K_{16}) \oplus f(R_{15}, K_{16}) = L_{15}. \qquad (2.9)$$

Екі бірдей мән үшін $\oplus$ (XOR) операциясының нәтижесі 0 болғандықтан, содан кейін қатынас сақталады

$$f(R_{15}, K_{16}) \oplus f(R_{15}, K_{16}) = 0,$$
$$L_{15} = R_{16} \oplus f(R_{15}, K_{16}).$$

$R_{14} = L_{15}$ анықтаңыз. Содан кейін $f(R_{14}, K_{15})$ және келесі теңдіктен есептейміз $R_{15} \oplus f(R_{14}, K_{15}) = L_{14} \oplus f(R_{14}, K_{15}) \oplus f(R_{14}, K_{15})) = L_{14}$ L_{14} анықтаймыз. Біз L_0 және R_0 блоктарына түскенше осылай жалғасады.

Белгілі бір уақыт ішінде DES коммерциялық ақпаратты қорғау жүйелерінде қолданылатын ең кең таралған алгоритм болды. Сонымен қатар, оны мұндай жүйелерде жүзеге асыру жақсы тон белгісі болып саналды [17, б.75]. DES алгоритмінің негізгі артықшылықтары

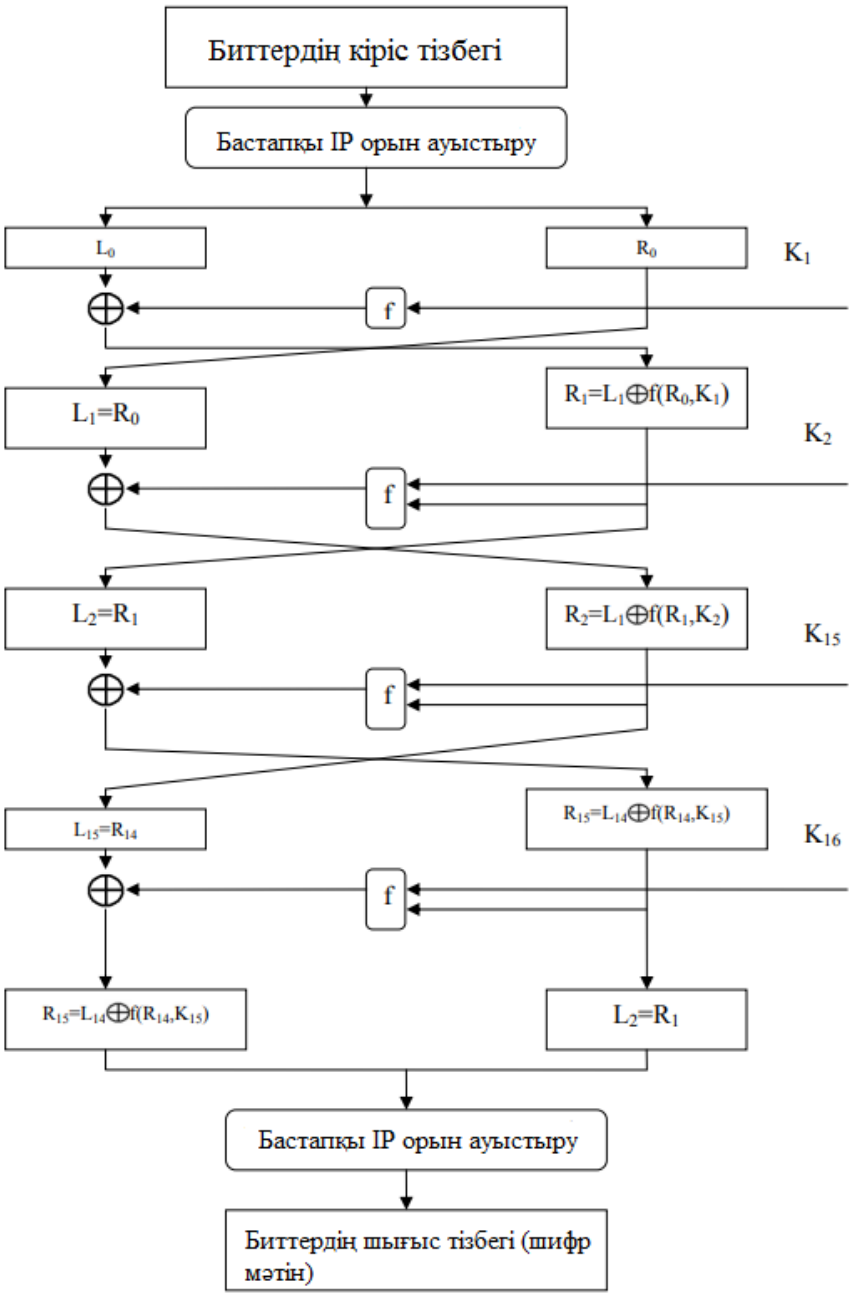
1) тек бір 56 биттік кілт пайдаланылады;

2) бір бағдарламалық пакетті пайдаланып хабарламаны шифрлаған соң, сіз шифрды шешу үшін DES стандартына сәйкес келетін кез келген басқа бағдарламалық пакетті пайдалана аласыз;

3) алгоритмнің салыстырмалы қарапайымдылығы өңдеудің жоғары жылдамдығын қамтамасыз етеді;

4) алгоритмнің жеткілікті жоғары беріктігі.

DES алгоритмінің жалпы диаграммасы 2.2 – суретте көрсетілген.



3.2 – сурет. *DES* алгоритмнің жалпы схемасы

DES алгоритмінің жұмыс режимдері

DES алгоритмі негізінде әртүрлі криптографиялық есептерді шешу үшін оның жұмысының бірнеше схемалары (режимдері) жасалды. DES стандартында келесі негізгі режимдерді қолдану ұсынылады:

Электрондық код кітабы (ECB, Electronic Code Book) әрбір деректер блогы дербес шифрланады, бұл режимді жылдам, бірақ деректердегі қайталанатын үлгілерге осал етеді.

Шифр блоктарының ілінісі (CBC, Cipher Block Chaining) – әрбір келесі блок алдыңғы нәтижені ескере отырып шифрланады, бұл қосымша төзімділікті қамтамасыз етеді, бірақ инициализация векторын (IV) қажет етеді.

Шифрланатын мәтін бойынша кері байланыс (CFB, Cipher Feed Back) – шифрлау биттік немесе байт бойынша орындалады, бұл режимді ағындық шифрлауға ыңғайлы етеді.

*Шығу кері байланысы (*OFB, Output Feed Back) – CFB-ге ұқсас, бірақ Шифр мәтінінің орнына шифрлау функциясының Шығыс мәндерін пайдаланады, бұл оны тасымалдау қателеріне төзімді етеді.

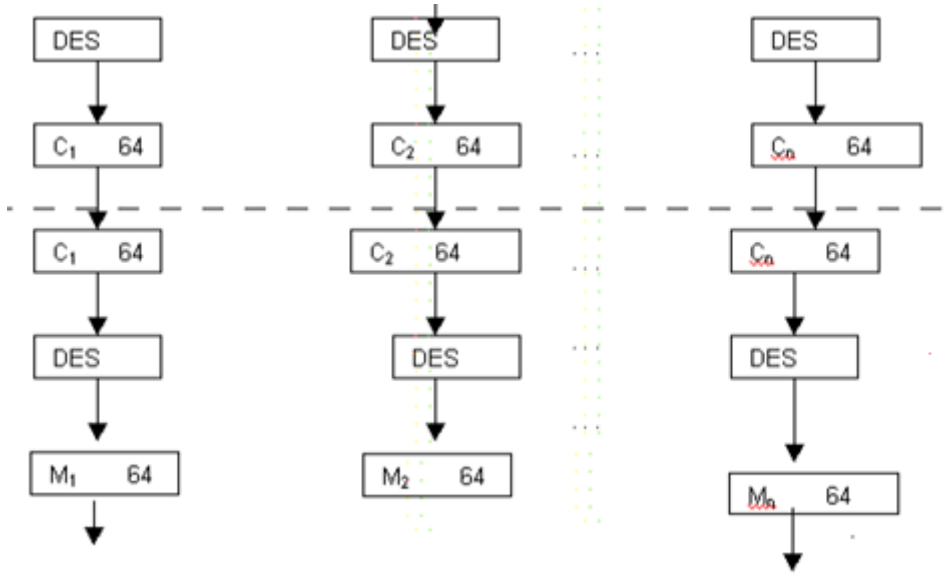
Режимді таңдау қауіпсіздік талаптарына, деректерді өңдеу жылдамдығына және шифрланған деректердің тұтастығына қарсы шабуылдарға төзімділікке байланысты.

«Электрондық код кітабы» режимі

«Электрондық код кітабы» режимі блоктық шифрлаудың қарапайым жұмыс режимдерінің бірі болып табылады. Бұл режимде хабарлама алдымен белгіленген ұзындықтағы блоктарға бөлінеді, әдетте 64 бит (8 байт), содан кейін әр блок бірдей кілтпен дербес шифрланады. Бұл режимді келесі формулалар түрінде ұсынуға болады (2.10)

$$C_i = E_k(M_i),$$
$$M_i = D_k(C_i), \qquad (2.10)$$

мұндағы M_i, C_i сәйкесінше кіріс және шифрланған ақпараттың i-ші блоктары. E_k, D_k сәйкесінше k пернесі бар DES шифрлау және дешифрлеу алгоритмі. «Электрондық код кітабы» режимінің ерекшелігі-бірдей ашық мәтін блоктары бірдей шифрланған блоктарға айналады, бұл деректерді талдау кезінде осалдыққа әкелуі мүмкін. Бұл кемшілік бұл режимді қайталанатын құрылымдармен деректерді қорғауға жарамсыз етеді. «Электрондық код кітабы» режимінің жұмыс схемасы 2.3 – суретте көрсетілген.



Сурет 2.3 – «Электрондық код кітабы» режимінің схемасы

«Шифрлық блоктарды тізбектеу» режимі

«Шифрлық блоктарды тізбектеу» режимі «Электрондық код кітабы» режимін жетілдіру болып табылады және блоктық шифрлау жүйелерінде кеңінен қолданылады. Оның басты айырмашылығы, шифрлау алдындағы әрбір ашық мәтін блогы (XOR операциясы арқылы) алдыңғы шифрланған блокпен біріктіріліп, крипто-төзімділікті айтарлықтай арттырады. Шифрлық блок тізбегі (CBC) режимін келесі (2.11) формула түрінде көрсетуге болады

$$\begin{aligned} C_i &= E_k(M_i \oplus C_{i-1}), \\ M_i &= C_{i-1} \oplus D_k(C_i), \end{aligned} \qquad (2.11)$$

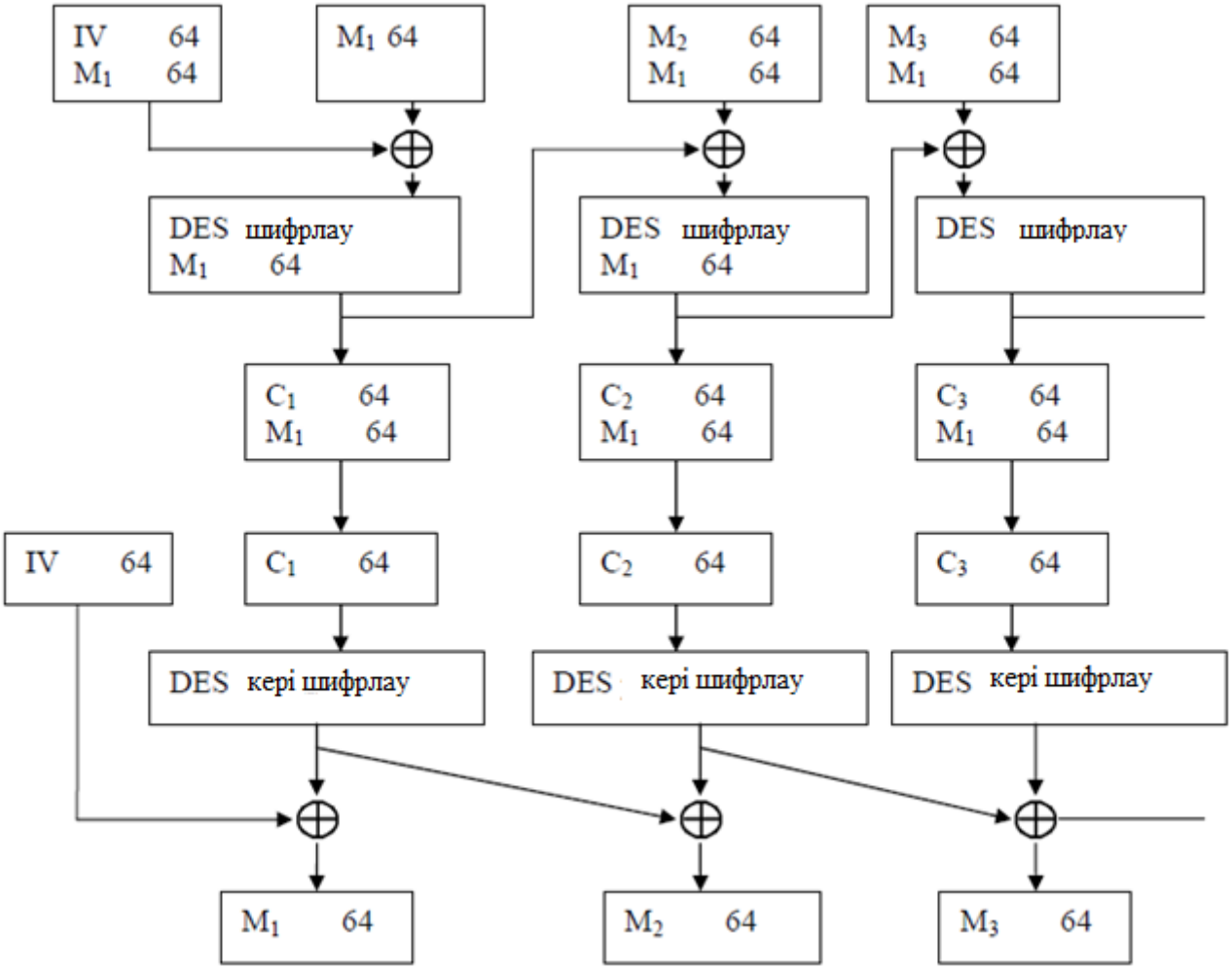
мұндағы M_i, C_i – сәйкесінше кірістің i–ші блоктары және шифрланған ақпарат;

E_k, D_k – сәйкесінше k кілтімен DES шифрлау және дешифрлеу алгоритмі;

C_0 – кейбір бастапқы вектор;

$\oplus$ – 2–ші модуль бойынша қосу операциясы.

Бұл режимде ECB режиміндегі сияқты – бастапқы файл M 64–биттік блоктарға бөлінеді $M_i, i = 1, 2, \ldots, n$, n – блоктар саны. Бірінші блок M_1 64–биттік C_0бастапқы блогына 2 модулі қосылған, оны күнделікті өзгертуге және құпия сақтауға болады. Содан кейін алынған сома k кілтінің көмегімен DES алгоритмімен шифрланады. Шифрлау нәтижесінде алынған 64–биттік C_1 блогына мәтіннің екінші блогымен модуль 2 қосылады, нәтиже шифрланады және т.б. Процедура мәтіннің барлық блоктары өңделгенше қайталанады. «Шифрлық блоктардың тізбегі» режим сұлбасы 2.4 – суретте көрсетілген.



2.4 – сурет. DES алгоритмінің Шифр блоктарының ілінісу режиміндегі сұлбасы

«Шифрланатын мәтін бойынша кері байланыс» режимі

Режимі шифрланған мәтін бойынша кері байланыс деректерді блоктық немесе байтты түрде шифрлауға мүмкіндік береді, бұл оны ағындық шифрлауға ыңғайлы етеді. Бұл режимде бастапқы вектор (IV) алдымен шифрлау кілті арқылы шифрланады, содан кейін нәтиже XOR операциясы арқылы ашық мәтінмен біріктіріліп, шифр мәтінін құрайды. Әрі қарай, шифрланған фрагмент келесі деректер блогын өңдеу үшін қолданылады. Шифр бойынша кері байланыс режимі келесі формулалармен сипатталады (2.12), (2.13):

Шифрмен кері байланыс (CFB) режимін келесі (2.12), (2.13) формулалармен көрсетуге болады:

$$C_i = C_i \oplus E_k(C_{i-1}), \quad (2.12)$$

$$M_i = C_i \oplus D_k(C_{i-1}), \quad (2.13)$$

мұндағы M_i, C_i сәйкесінше кіріс және шифрланған ақпараттың i–ші блоктары;

E_k, D_k – сәйкесінше k кілтімен DES шифрлау және дешифрлеу алгоритмі;

C_0 – кейбір бастапқы вектор;

$\oplus$ – қосу операциясының модулі 2.

Бұл режимде блок өлшемі 64 биттен өзгеше болуы мүмкін. Шифрланатын (шифрдан шығарылатын) файл ұзындығы $t,\ t = 1, \ldots, 64$ бит блоктарында оқылады. Кіріс блогы, 64–биттік аккумулятор, алдымен оңға тураланған инициализация векторын қамтиды.

«Шығыс бойынша кері байланыс» режимі

«Шығыс бойынша кері байланыс» *(OFB)* режимі айнымалы блок өлшемін және CFB режиміндегідей инициализацияланған ауысым регистрін қолданады. Бастапқы сәтте кіріс блогында оң жақ жиекке тураланған C0 инициализация векторы бар. Әрбір деректерді шифрлау сеансы үшін Ашық арна арқылы берілетін жаңа бастапқы Регистр күйін пайдалану қажет.

OFB режимін келесі (2.14), (2.15) формулалар түрінде көрсетуге болады:

$$C_i = M_i \oplus P_i, \quad (2.14)$$

$$P_i = E_k(C_{i-2}C_{i-1} >> t), \quad (2.15)$$

мұндағы M_i, C_i сәйкесінше кіріс және шифрланған ақпараттың і–ші блоктары;

E_k, D_k – сәйкесінше k кілтімен DES шифрлау және дешифрлеу алгоритмі;

C_0 – кейбір бастапқы вектор;

$\oplus$ – қосу операциясының модулі 2.

$C_{i-1} >> t$ битке солға жылжуын білдіреді, ең жоғары t биттері ең аз маңызды биттердің орнына орналастырылады.

Пайдаланылған әдебиеттер тізімі

1. Гольдвассер С., Беллар М. Достижения в криптологии / Гольдвассер С. – М.: Триумф, 2016. – 513 с.

2. Шнайер, Брюс. Криптографическая методы защиты информации, 26-я ежегодная международная конференция по криптологии, Санта-Барбара, Калифорния, США, 20–24 августа 2022 г.

3. Фергюсон Н., Шнайер Б., Коно Т. Криптографическая инженерия: принципы проектирования и практическое применение. / Фергюсон Н. – М.: Пресс, 2022. – 416 с.