Әл-Фараби атындағы Қазақ ұлттық университеті

ӘОЖ 004.421 Қолжазба құқығында

АЛҒАЗЫ КҮНБОЛАТ ТІЛЕУХАНҰЛЫ

Әртүрлі әдістерге негізделген шифрлау алгоритмдерін құру және зерттеу

6D100200 – Ақпараттық қауіпсіздік жүйелері

Философия докторы (PhD)
дәрежесін алу үшін дайындалған диссертация

Отандық ғылыми кеңесші:
Бияшев Р.Г.
т.ғ.д., профессор
Шетелдік ғылыми кеңесші:
Andrzej Smolarz
т.ғ.д., профессор
(Польша, Люблин техникалық университеті)

Қазақстан Республикасы
Алматы, 2021

МАЗМҰНЫ

НОРМАТИВТІК СІЛТЕМЕЛЕР

Бұл диссертацияда келесі стандарттарға сілтемелер қолданылды:

1 ҚР Үкіметінің 2017 жылғы 30 маусымдағы № 407 қаулысымен бекітілген Киберқауіпсіздік («Қазақстанның киберқалқаны») тұжырымдамасы».

2 Киберстратегия және ақпараттық қауіпсіздікті басқару ISO/IEC 27001:2013 сәйкес және сыртқы және ішкі шабуылдардан ақпаратты алдыңғы қатарлы қорғау тәжірибелерімен ақпараттық қауіпсіздікті басқаруды бағалау, енгізу және дамыту.

3 Қазақстан Республикасы Үкіметінің 2017 жылғы 12 желтоқсандағы № 827 қаулысымен бекітілген "Цифрлық Қазақстан" мемлекеттік бағдарламасы.

4 СТ РК 1073-2007 – Ақпаратты криптографиялық қорғау құралдары.

5 «Диссертацияларды және авторефераттарды рәсімдеу бойынша нұсқаулық», ҚР БҒМ, Жоғары аттестаттау комитеті, Алматы, 2004.

6 ГОСТ 7.32-2001 – Ғылыми-зерттеу жұмысының есебі.

БЕЛГІЛЕУЛЕР МЕН ҚЫСҚАРТУЛАР

АЕТИ	–	Ақпараттық есептеуіш технологиялар институты
АККҚ	–	Ақпаратты криптографиялық қорғау құралдары
АҚЖ	–	Ақпараттық қауіпсіздік жүйелері
АҚЗ	–	Ақпараттық қауіпсіздік зертханасы
ҒЗЖ	–	Ғылыми-зерттеу жұмыстары
ЖЖҚ	–	Жедел жадылық құрылғысы
ПЕПСЖ	–	Позициялық емес полиномды санау жүйесі
ПКТ	–	Псевдокездейсоқ тізбек
ЭЦҚ	–	Электрондық цифрлық қолтаңба
AES	–	Advanced Encryption Standard
DES	–	Data Encryption Standard, АҚШ-тың деректерді шифрлау стандарты
DSA	–	Digital Signature Algorithm, Цифрлы қолтаңба алгоритмі
IDEA	–	International Data Encryption Algorithm, Халықаралық деректерді шифрлау алгоритмі
MQ attack	–	Multivariate Quadratic attack, Көп айнымалылы квадраттық шабуыл
NIST	–	National Institute of Standards and Technology, АҚШ-тың Ұлттық стандарттар және технологиялар институты
RSA	–	Ашық кілтті криптографиялық алгоритм (аббревиатурасы - Rivest, Shamir және Adleman фамилияларының қысқартуы)
SP	–	Substitution-Permutation, ауыстыру-алмастыру
XL	–	eXtended Linearization, кеңейтілген сызықтандыру
XSL	–	eXtended Sparse Linearization, кеңейтілген сирек сызықтандыру

КІРІСПЕ

Ақпаратты криптографиялық қорғау кез-келген ақпараттық қауіпсіздік жүйесінің негізгі ішкі жүйелерінің бірі болып табылады. Ақпаратты өңдеу, сақтау, беру және пайдалану үдерістері заманауи қоғам өмірінің басым бағытына айналды. Криптографияның барлық нақты міндеттері технологияның даму деңгейіне, пайдаланылатын коммуникация құралдарына және ақпарат беру әдістеріне айтарлықтай дәрежеде байланысты [1 - 3].

Шифрлардың абсолюттік және теориялық беріктілігі туралы сұрақтарды Клод Шеннон алғаш рет математикалық түрде тұжырымдады. Атап айтқанда, шексіз ресурстары бар шабуылдаушы үшін шифр қаншалықты берік болып табылады. Абсолютті беріктілік үшін төменде келтірілген талаптардың әрқайсысы өте маңызды: 1) кілттің кездейсоқтығы (тең ықтималдықтылығы); 2) кілттіңі ұзындығы мен ашық мәтіннің ұзындығының теңдігі; 3) кілтті бір рет пайдалану. Осы шарттардың кем дегенде бірі бұзылған жағдайда, шифрдың абсолютті беріктілігі орындалмайды және оны заңсыз жолмен ашу мүмкіндіктері пайда болады. Бірақ, бұл шарттар абсолютті берік шифрды өте қымбатқа түсіреді және қолдануға тиімсіз жасайды. Мұндай шифрды пайдаланбас бұрын, біз барлық абоненттерді кездейсоқ кілттермен жеткілікті қамтамасыз етуге және оны қайта пайдалану мүмкіндігін болдырмауға тиіспіз. Ал, бұны жасау өте қиын және қымбат [4 - 7]. Сондықтан, абсолютті берік шифрларды тек аздаған ақпарат тасымалдайтын байланыс желілерінде ғана пайдаланылады. Әдетте бұл желілерді мемлекеттік маңызы бар ақпараттарды жіберуге қолданады.

Зерттеу тақырыбының өзектілігі ақпараттық қауіпсіздікті қамтамасыз ету мақсатында ақпараттық-коммуникациялық технологиялардың қарқынды дамуы мен ақпараттық қауіпсіздіктің қолданыстағы түрлерін жетілдіру қажеттілігіне байланысты. Ақпаратты өңдеу, сақтау, беру және пайдалану үдерістері заманауи қоғам өмірінің басым бағытына айналды және көбінесе байланыс құралдары мен ақпарат беру тәсілдерінің дамуы мен қолданылу деңгейіне тәуелді. Ақпаратты қорғаудың заманауи құралдарын құру арқылы оны қорғаудың қажетті деңгейін қамтамасыз ету, ақпараттың қауіпсіздігін қамтамасыз етудің өзекті мәселелерінің бірі.

Тәуелсіз мемлекет үшін де ақпараттық және коммуникациялық технологиялар дамуында үлкен рөл атқарады. Қазақстанда 2017 жылы Киберқауіпсіздік тұжырымдамасы («Қазақстанның киберқалқаны») қабылданды. Тұжырымдаманың мақсаты – жаһандық бәсекелестік жағдайда Қазақстан Республикасының орнықты дамуын қамтамасыз ету үшін, электрондық ақпараттық ресурстарды, ақпараттық жүйелер мен ақпараттық-коммуникациялық инфрақұрылымдарды сыртқы және ішкі қатерлерден қорғау деңгейіне қол жеткізу және ұстап тұру болып табылады [8]. Осыған байланысты халықаралық ақпаратты қорғауға қойылатын заманауи талаптарды қанағаттандыратын отандық ақпаратты қорғау жүйелерін құру өзекті болып табылады.

ҚР БҒМ ҒК Ақпараттық және есептеуіш технологиялар институтының Ақпараттық қауіпсіздік зертханасында отандық ақпаратты криптографиялық қорғау құралдарын құру саласында ғылыми-зерттеу жұмыстары жүргізілуде. Атап айтқанда, электронды хабарламаларды симметриялы блоктық шифрлау жүйелерін, соның ішінде позициялық емес полиномдық санау жүйелеріне негізделген модификациялар әзірленді [9-20].

Бүгінгі күні блоктық шифрлау алгоритмдері компьютерлерде сақталған немесе жалпыға ортақ ақпаратты-коммуникациялық желісі арқылы берілетін ақпаратты криптографиялық қорғаудың негізгі құралы болып отыр. Шифрлау алгоритмінің бұл түріне деген сұраныс оның практикалық қолданылуының артықшылықтарына байланысты. Заманауи аппаратты-бағдарламалық құрылғыларда тиімді бағдарламалық іске асыру мүмкіндігі мол, шифрлау жылдамдығының жоғарылығы және жоғары деңгейде беріктілікке кепілдік береді. Симметриялы блоктық шифрлар тек жеке криптографиялық алгоритм ретінде ғана қолданылмайды, сонымен бірге басқа да криптографиялық алгоритмдер мен хаттамалардың құрамына кіретін маңызды криптографиялық механизм. Оларды псевдокездейсоқ тізбек генераторының және криптографиялық хэш алгоритмдерінің құрамының негізгі бөлігі ретінде қолдану практикада жиі кездеседі.

Блоктық шифрлардың тағы бір артықшылығы – кілтінің қысқалығында. Ұзындығы көп жағдайда 128 – 256 бит аралығында жататын бір ғана қысқа кілтпен үлкен бірнеше файлды немесе деректерді шифрлауға болады. Бұл ағындық шифрлардан қарағандағы ең негізгі артықшылығы. Себебі, ағындық шифрларда кілтті бір реттен артық қолданбау ұсынылады. Ұзын кілттерді сақтау және қолданушылар арасында оларды алмасуда тағы да қосымша қорғанысты талап етеді. Жоғарыда аталғандарды ескере отырып, шифрлардың ішінде қолданысқа ең тиімдісі және лайықтысы блоктық шифрлар. Сондықтан симметриялы блоктық шифрлау алгоритмдері қазіргі уақытта, ақпаратты өңдеудегі құпиялылықты қамтамасыз етудің негізгі криптографиялық құралы болып табылады.

Криптографияның жетілуімен қатар криптографиялық шабуылдар және криптоталдау әдістері дамып отырды. Криптография және криптоталдау бір-бірінен ажырамастай ұғымдарға айналды: олар криптологияның екі құрамдас бөлігі. Берік криптографиялық жүйені құру үшін оған жасалынатын шабуылдың барлық мүмкін жолдарын ескеру қажет. Криптография мен криптоталдаудың құны уақыт өткен сайын тек қана өседі. Сондықтан, криптографиялық алгоритмдер құру, ғылыми зерттеу жұмыстарында да және практикада да **өзекті** болып табылады.

Қазақстанда электрондық ақпаратты қорғау үшін негізінен шетелдік криптографиялық құралдар және бағдарламалық жасақтамалар қолданылады, сондықтан отандық криптографиялық қорғау құралдарын құру сөзсіз өзекті және қажет.

Диссертациялық жұмыстың мақсаты. Итеративті блоктық шифрлау алгоритмін және позициялық емес полиномдық санау жүйесін пайдаланып

раундтық кілттер алгоритмін құру. Құрылған алгоритмдердің криптоберіктілігін зерттеу.

Зерттелу міндеттері:

- ақпаратты криптографиялық қорғаудың қолданыстағы симметриялық блоктық алгоритмдеріне сараптау жүргізу;
- криптографиялық шабуылдардың және криптоталдаудың белгілі әдістерін қарастыру және талдау;
- позициялық емес полиномдық санау жүйесін пайдаланып раундтық кілттер алу және ауыстыру-алмастыру желісі негізінде симметриялы блоктық шифрлау алгоритмдерін құру;
- құрылған шифрлау алгоритмдерін беріктілігін криптоталдау әдістері арқылы зерттеу;
- құрылған итеративті шифрлау алгоритмдерін бағдарламалық жүзеге асыру.

Зерттелу нысаны. Шифрлау жүйелері, позициялық емес полиномдық санау жүйесі, криптографиялық шабуылдар, криптоталдау әдістері.

Зерттеудің пәні. Симметриялы блоктық шифрлау алгоритмдері, оның ішінде позициялық емес полиномды санау жүйесі негізінде құрылған алгоритмдер.

Зерттелу құралы мен әдісі. Жұмыста бульдік функция теориясы, сызықтық алгебра, ықтималдықтар теориясы және математикалық статистика, сондай-ақ әртүрлі криптографиялық алгоритмдер және криптоталдау әдістері қолданылды.

Жұмыстың ғылыми жаңалығы:

- шифрлау алгоритмдеріне қойылатын жалпы талаптарға жауап беретін, ауыстыру-алмастыру жүйесі құрылымындағы жаңа симметриялы блоктық шифрлау алгоритмі құрылды;

дәстүрлі емес әдіске (ПЕПСЖ) негізделген симметриялы блоктық шифрлау алгоритмі құрылды, оны қолдану алгоритмнің криптографиялық беріктігін арттыруға мүмкіндік береді;

- дифференциалдық және сызықтық криптоталдауларға беріктілік көрсеткіштері жоғары, сызықты емес (S-блок) ауыстыру түйіндері құрылды.

Зерттеудің теориялық және практикалық құндылығы. Жүргізілген ғылыми зерттеулердің және алынған нәтижелердің практикалық мүмкіндігі жоғары және ақпараттық-коммуникациялық жүйелер мен желілерде құпия ақпараттарды сақтауға және алмасуда оларды қорғауға пайдалануға болады. Сонымен қатар, осы нәтижелер отандық ақпаратты қорғау құралдарын құруға және дамытуға ықпал етеді және ақпаратты шифрлаудың тиімді алгоритмдерін құру теориясын кеңейтеді. Әзірленген шифрлау алгоритмінің бағдарламалық жасақтамасы іске асырылып, ҚР ӘМ Ұлттық зияткерлік меншік институтынан «Qamal v 1.0.1» 2019 жылғы 6 қыркүйектегі № 5200 авторлық куәлігі алынды.

Қорғауға шығарылған негізгі тұжырым. Шифрлау алгоритміне қойылатын жалпы талаптарға жауап беретін жаңа симметриялық блоктық шифрлау алгоритмі құрылды. Алгоритмнің позициялық емес полиномдық санау

жүйесінде әзірленген екінші нұсқасы ұсынылды. Құрылған алгоритмдердің беріктілігі криптоталдаудың дифференциалдық, сызықтық, алгебралық және т.б. түрлері бойынша зерттелді.

Сенімділік дәрежесі мен апробациялау нәтижелері. Диссертациялық жұмыс бойынша жүргізілген зерттеулер мен нәтижелерінің сенімділігі үшінші бөлімде көрсетілген.

Зерттеулер нәтижесі төменде көрсетілген ғылыми-практикалық конференцияларда баяндалды және талқыланды.

1) «Информатика және қолданбалы математика» атты III Халықаралық ғылыми-тәжірибелік конференциясында (Алматы, 26-29 қыркүйек 2018).

2) International Conference on Wireless Communication, Network and Multimedia Engineering, WCNME-2019 (Гуйлин, Китай, 2019).

3) «Информатика және қолданбалы математика» атты IV Халықаралық ғылыми-тәжірибелік конференциясында (Алматы, 25-29 қыркүйек 2019).

4) International Conference on Security of Information and Networks (Sochi, Russia September, 2019).

5) «Қазақстандағы Ақпараттық қауіпсіздіктің өзекті мәселелері» атты Халықаралық ғылыми-тәжірибелік конференциясында (Алматы, 15 қаңтар 2020).

Диссертациялық тақырыптың ғылыми бағдарламалармен байланысы. Диссертациялық жұмыс Қазақстан Республикасының Білім және Ғылым министірлігі Ғылым комитетінің Ақпараттық және есептеуіш технологиялар институтында бекітілген PhD докторлық диссертациялар жоспарына және ЖТН – BR05236757-ОТ-20 «Жалпы мақсаттағы желілер мен инфокоммуникациялық жүйелерде ақпаратты жіберу және сақтау кезінде оны криптографиялық қорғау үшін бағдарламалық және бағдарламалық-аппараттық кешендерді құрастыру» бағдарламалық – нысаналы қаржыландыру жобасының ғылыми-зерттеу жұмыстарының аясында орындалды. Диссертациялық жұмыс бойынша жүргізілген зерттеу жұмыстарының нәтижесі аталған БНҚ жобасының 2018-2020 жылдарындағы есебіне енгізілген.

Жұмыс көлемі мен құрылымы. Диссертациялық жұмыс кіріспе, 4 бөлім, қорытынды және пайдаланылған әдебиеттерден тұрады. Диссертацияның толық көлемі: 118 бет жазба мәтіні, соның ішінде 23 сурет, 42 кесте, 94 пайдаланылған әдебиеттер тізімінен және 4 қосымшадан тұрады.

Нәтижелердің жарияланымдары. Ғылыми зерттеу жұмыстарын орындау барысында 21 ғылыми жұмыстар жазылды. Оның ішінде 3 мақала Scopus және Thomson Reuters базаларында индекстелінетін «Cogent Engineering» және «International journal of electronics and telecommunications» журналдарында, 8 мақала Қазақстан Республикасы Білім және ғылым министрлігінің білім және ғылым саласы бойынша бақылау комитетімен ұсынылған басылымдарда, 10 мақала халықаралық ғылыми-практикалық конференциялар жинақтарында жарық көрді.

Кіріспеде диссертациялық жұмыст тақырыбының өзектілігінің негіздемесі берілген. Ғылыми-зерттеу жұмысының мақсаты, нысаны және пәні

тұжырымдалған. Сонымен бірге, ғылыми жаңалығы және тәжірибелік маңызы көрсетілген. Зерттеу жұмысттарының нәтижелерінің апробациясы және жарияланымдары туралы мәліметтер келтірілген.

Бірінші бөлімде ақпаратты қорғауда қолданылатын алгоритмдердің түрлері және негізгі бағыттары сипатталған. Сонымен бірге, жалпы криптографияда және диссертациялық жұмыста пайдаланылған терминдерге түсініктеме берілген. Криптоалгоритмдердің қауіпсіздігінің дәрежесі бойынша бөлінген топтарына сипаттама беріліп, симметриялы блоктық шифрларға қойылатын талаптар аталған және шифрлауда қолданылатын режимдер сипатталған. Сондай-ақ заманауи симметриялы шифрлау алгоритмдеріне жүргізілетін криптоталдаулардың негізгі түрлері келтірілген.

Екінші бөлімде SP-жүйесі негізінде құрылған жаңа «Qamal» симметриялы блоктық шифрлау алгоритмі сипатталады және кілтті пайдаланудағы ерекшелігіне байланысты осы алгоритмнің «Qamal NPNS» екінші нұсқасы да ұсынылды. Әзірленген алгоритмде қолданылған түрлендірулер жеке-жеке сипатталған. Әртүрлі қауіпсіздік деңгейлеріне сәйкес алгоритмнің шифрлау блогының және кілтінің ұзындықтары да әртүрлі үш мән қабылдай алады. Әзірленген шифрлау алгоритмі үшін құрылған S-блок ауыстыруының құрылысы сипаттталған. Сонымен бірге, «Qamal NPNS» алгоритмі ПЕПСЖ негізделген алгоритм болғандықтан, ПЕПСЖ-лерінің құрылуы және оны шифрлауда, шифрды кері ашуда қалай қолданылатыны туралы мәліметтер берілген. Әзірленген алгоритм бойынша деректерді шифрлау мысалы келтірілген.

Үшінші бөлімде құрылған шифрлау алгоритмінің сенімділігі зерттеліп, оның нәтижелері берілген. Зерттеу жұмыстары шифрлау алгоритмінің көмегімен алынған шифрмәтіндердің статистикалық қауіпсіздігін тексеру жұмыстарымен басталады. Одан кейін, криптографияда қажетті шарттардың бірі – шифрдың лавиндік әсері тексерілген. Алгоритмнің беріктілігін бағалау үшін криптоталдаудың алгебралық, дифференциалдық, сызықтық және тағы басқа да әдістерімен тексерілген. Жүргізілген криптошабуылдардың теориялық бағасы ғана емес, нәтижелері нақты мысалдар арқылы да көрініс табады. Сонымен қатар, кілтті ПЕПСЖ-де пайдаланғанда алгоритмнің беріктілігіне әсері зерттелді және анықталды.

Төртінші бөлімде әзірленген шифрлау алгоритмі үшін құрылған бағдарламалық жасақтама туралы ақпараттар берілген. Мысалы, бағдарламалық тілі, жүйелік талаптар, жұмыс істеу түсіндірілімдемелері және т.б. Бағдарламаның есептеу істеу жылдамдығын арттыру мақсатында шифрлау алгоритмінде қолданылған Mixer2 түрлендіруін үш тәсіл бойынша жүзеге асырылуы қарастырылды. Алынған нәтижелерге салыстыру жүргізілді.

Қорытындыда жұмыстың негізгі қорытындылары мен нәтижелері тұжырымдалды.

Ғылыми тағылымдамалар. Люблин техникалық университеті, Люблин қаласы, Польша, 2019 жыл.

1 ЗАМАНАУИ СИММЕТРИЯЛЫ БЛОКТЫҚ ШИФРЛАУ АЛГОРИТМДЕРІ ЖОБАЛАУ ЖӘНЕ ОЛАРДЫ ЗЕРТТЕУ ӘДІСТЕРІ

Ақпаратты құпия түрде беру, қол жетімсіз немесе рұқсат етілмеген тұлғаларға түсініксіз жасау туралы ғылым ақпараттың қорғалуын қамтамасыз ету қажеттілігін адамзат түсінген кезде пайда болды және дами бастады. Адам өмірінің ақпараттық технологияларға тәуелділігінің артуына және ақпараттық қауіпсіздікті қамтамасыз ету қажеттілігіне байланысты криптографиялық әдістерді қолдану бүкіл қоғам үшін маңызды болды. Құпиялылықты қамтамасыз ету өзектілігі қауіпсіздіктің тұтастығы, түпнұсқалығы және басқа да аспектілерінен маңыздылығы кем емес. Криптографияның жаңа қағидаларын ойлап табу және ашық кілтті криптографияның пайда болуы азаматтық қоғам мен бизнестің қажеттілігі және банк қызметі үшін криптографияның кеңінен пайдалануға түрткі болды [21].

Жаңа ақпараттық технологиялардың дамуы өндірістік тыңшылық, компьютерлік қылмыстар және құпия ақпаратқа рұқсатсыз қол жеткізу сияқты жағымсыз құбылыстармен қатар жүреді. Сондықтан ақпаратты қорғау кез-келген елдегі ең маңызды мемлекеттік міндеті болып табылады.

Ақпаратты қорғау кез-келген түрдегі ақпаратты жоғалту (ұрлау, жоғалту, бұрмалау, қолдан жасау) нәтижесінде келетін залалдың алдын-алуды қамтамасыз етуі керек. Ақпаратты қорғау шараларын ұйымдастыру қолданыстағы заңнамаға және ақпараттық қауіпсіздіктің нормативті құжаттарына, ақпаратты пайдаланушылардың мүдделеріне сәйкес жүргізілуі керек. Ақпаратты қорғаудың жоғары деңгейінде кепілдік беру үшін оны қорғау құралдарын дамытудың ғылыми-техникалық мәселелерін үнемі шешіп отыру қажет және қорғау құралдарын жетілдіріп отыру керек. Қазіргі заманғы кәсіпорындардың көпшілігі қызмет түріне және меншік формасына қарамастан, автоматтандырылған жүйелерде (АЖ) ақпаратты өңдеу, сақтау және беру кезінде қауіпсіздікті бақылауды ұйымдастырушылық-реттеушілік шаралары мен техникалық құралдарын қамтитын өздерінің ақпараттарын қорғау жүйелерін қамтамасыз етпей басқа қызметтерді ойдағыдай жүргізе алмайды [22-25].

Қорғалатын ақпараттардың иелері әртүрлі салалар болуы мүмкін, мысалы мемлекеттік басқару органдары және олар құрған құрылымдар (мемлекеттік құпиялар, қызметтік құпиялар), бизнес және банктіктер, заңды тұлғалар (коммерциялық, заңдық, медициналық, аудиторлық құпиялар және т.б.); қоғамдық ұйымдар (партиялық т.б.), жеке азаматтары (жеке және отбасылық адвокаттық, медициналық құпияларға қатысты).

Ақпараттар пішініне (формасына), кодтау және сақтау әдістеріне байланысты графикалық, дыбыстық, мәтіндік, цифрлық, бейнелік ақпарат және т.б. болып келеді. Ақпараттың сенімділігі, толықтығы, әділдігі, жеделдігі және маңыздылығы оның ең елеулі қасиеттері. Құпия және құпия емес ақпаратты сақтау үшін бірдей тасымалдағыш құралдар қолданылады [26]. Жалпы жағдайда құпия болсын немесе құпия емес болсын ақпараттарды тасымалдаушыларды өз иесі қорғайды.

Деректерді шифрлау арқылы қорғау - қауіпсіздік мәселесін шешудің бір мүмкіндігі. Шифрланған деректер тек шифрды ашуды білетіндерге ғана қол жетімді, сондықтан шифрланған деректерді ұрлау рұқсатсыз пайдаланушылар үшін мүлдем мағынасыз.

1.1 Симметриялы блоктық шифрлау алгоритмдеріне қойылатын талаптар

Есептеу машиналары мен жүйелерін кеңінен практикаға енгізу оларды әр түрлі ақпараттық шабуылдар үшін пайдалануға түрткі болды. Бұған қосымша ақпаратты бұрынғыдай физикалық тасымалдау тұрғысынанда айырылғаны ықпал етті. Енді бұрынғыдай мәтінді қағазға жазып, астына автор қол қойған түрде емес, ақпарат электронды түрде тасымалданатын болды. Осыған байланысты құпиялылықты ғана емес, ақпараттың шынайылығы мен тұтастығын бақылауды қамтамасыз ету қажет болды. Сонымен қатар, ақпарат құндылығының өсуі және қоғамды ақпараттандыру ақпаратқа қол жетімділікті және компьютерлік терроризмнен қорғау мәселелерін шектеу мәселелерін көтерді. Бүгінгі күні мұндай қорғау криптографиялық құралдарды қолдану арқылы тиімді жүзеге асырылып отыр [27].

Криптографиялық әдістерді қолданудың негізгі бағыттары – бұл құпия ақпаратты байланыс арналары арқылы беру (мысалы, электрондық пошта), жіберілген хабарламалардың аутентификациясы, ақпаратты (құжаттарды, мәліметтер базасын) тасымалдаушыларда шифрланған түрде сақтау.

Заманауи криптография төрт негізгі бөлімнен тұрады: симметриялы криптожүйелер, ашық кілт жүйелері, электрондық қолтаңба жүйелері және кілттерді басқару.

Ашық кілттер (асимметриялық) жүйесінде бір-бірімен математикалық байланыста болатын екі: ашық және құпия кілттер қолданылады. Шифрланатын ақпарат барлығына қол жетімді ашық кілт арқылы шифрланады да, шифрды тек хабарлама алушыға белгілі құпия кілт арқылы ғана кері аша алады. Бұл деректерді «ортадағы шабуылдаушыдан» қорғауға кепіл бола алады. Жүз мыңдаған клиенттермен үнемі байланыста болатын веб-электрондық пошта серверлері үшін тек бір кілтті қорғау қажет. Асимметриялық шифрлаудың екінші маңызды ерекшелігі бұл – аутентификация, деректерді қабылдауы керек объекті ғана деректерді көре алуына және кері шифрлай алуына кепілдік етеді [28-29].

Компьютерлік ресурстарды көбірек қажет ететіндіктен асимметриялық шифрлау симметриялық шифрлауға қарағанда «ауыр» деп саналады. Сонымен бірге, кілт жасау процесінде шектеулер бар. Асимметриялық шифрлау жүйелеріне RSA, Эль-Гамаль, DSA, ECDSA, Rabin және т.б. жатады.

Кілттерді тарату және *кілттерді басқару* терминдері ақпаратты өңдеу жүйесінің үдерістерін білдіреді, оның мағынасы пайдаланушылар арасында кілттерді құру және тарату болып табылады.

Симметриялы криптожүйелерде ақпаратты шифрлау және кері шифрлау үшін бірдей кілт қолданылады. Яғни, қандай кілтпен шифрласақ, дәл сол кілтпен кері ашатын боламыз.

Криптологияда қолданылатын бірқатар терминдерге тоқтала кетейік.

Шифр дегеніміз – ашық немесе басқа да деректер жиынтығының, берілген криптографиялық түрлендіру алгоритмінің көмегімен шифрланған қайтымды түрлендірулер жиынтығы [30, 31].

Криптографиялық жүйе – ақпараттық үдерісті қорғаудың белгілі бір мәселесін шешу мақсатында криптографиялық түрлендірулер немесе алгоритмдер жиынтығы.

Кілт – криптографиялық деректерді түрлендіру алгоритмінің кейбір параметрлерінің ерекше құпия күйі және осы алгоритм үшін барлық мүмкін болатын жағдайлардың жиынтығынан бір нұсқаны таңдауды қамтамасыз етеді. Кілттің құпиялығы шифрланған мәтінді пайдаланып түпнұсқа мәтінді қалпына келтіру мүмкін еместігін қамтамасыз етуі қажет. Ақпаратты шифрлауда қолданылатын түрлендірулер осы кілтке айтарлықтай байланысты болып табылады.

Электрондық (цифрлық) қолтаңба деп мәтінді басқа пайдаланушы алған кезде хабарламаның авторлығы мен тұтастығын тексеруге мүмкіндік беретін криптографиялық түрлендіру аталады және ол мәтінге бекітілген, яғни бірге беріледі.

Деректерді *шифрлау* деп шифрды қолдану арқылы ашық деректерді шифрланған деректерге түрлендіру үдерісін айтады, ал *шифрды кері ашу* дегеніміз жабық деректерді шифр көмегімен ашық деректерге түрлендіру. «Шифрланған деректер» терминінің орнына «шифрмәтін» терминдері жиі қолданылады [32].

Дешифрлау деп кілт немесе шифрлау алгоритмі белгісіз болған жағдайларда криптоталдау әдістерінің көмегімен шифрмәтіндерден ашық деректерді қалпына келтіру үдерістерін айтамыз.

Қазіргі уақытта криптографиялық ақпаратты жабудың көптеген әдістері белгілі. Шифрлау әдістерін жіктеу келесі критерийлер бойынша жүзеге асырылуы мүмкін [33, 34]:

- кілт түрлері бойынша: симметриялық криптоалгоритмдер; асимметриялық криптоалгоритмдер;
- ақпараттық блоктың өлшеміне қарай: ағындық шифрлар; блоктық шифрлар;
- мәліметтерге әсер ету сипаты бойынша: ауыстыру әдісі, алмастыру әдісі; аналитикалық әдістер, аддитивті әдістер (гаммалау), аралас әдістер.

Жоғарыда аталғандай симметриялы криптографиялық жүйелерде ақпаратты шифрлау және шифрды кері ашу үшін бірдей кілттерді қолданады (сурет 1.1).

Сурет 1.1 – Симметриялы криптожүйелердің жұмысының әдістемесі

Бұл жүйені қолдануға кіріспес бұрын пайдаланушылардың бәрі ортақ құпия кілтті алдын-ала алып алу қажет және оған шабуылдаушының қолжетімділік жолдарын жою қажет дегенді білдіреді. Симметриялы криптоалгоритмдердің негізгі бір түрі блоктық шифрлар [35].

Блоктық шифрлар - бұл бастапқы мәтіннің блоктарының (бекітілген ұзындықтағы бөліктер) қайтымды түрлендірулерінің тобы. Қазіргі заманда осы блоктық шифрлар іс жүзінде кеңінен таралған. Шифрлаудың ресейлік және американдық стандарттары (ГОСТ Р 34.12-2015 және AES) дәл осы шифрлар класына жатады.

Блоктық шифрлардың негізгі түрлері болып Фейстель желісі мен алмастыру-ауыстыру желілері (SP-желі) болып табылады. 1971 жылы Хорст Фейстель әр түрлі шифрлау алгоритмдерін іске асыратын екі құрылғыны патенттеді, кейін олар Lucifer деп аталып кетті. Құрылғылардың бірі «Фейстель желісі» деп аталған дизайнды қолданды. Бұл жүйеде ашық мәтіннің блогы екі немесе бірнеше бөлікке бөлінеді де, бірінші жартысына кілтті пайдалану арқылы түрлендіру жүргізеді де екінші жартысына XOR операциясы арқылы қосылады. Ал екінші жартысы ешқандай өзгермейді. Келесі қадамда осы алынған бөліктер орындарын ауыстыратын болады. Осы аталған іс-әрекеттің бәрі бір раунд болып саналады да, әр алгоритм өзінің құрылымына қарай бірнеше (бекітілген) раундтан тұрады. Фейстель желісінің негізінде құрылған алгоритмдердің мысалдары ретінде DES, IDEA, Blowfish, Twofish, FEAL, Магма және т.б. көптеген алгоритмдерді атауға болады [34, б. 176].

SP желісі деп әр айналымның түрлендірулері алмастырулардың және ауыстырулардың комбинациясы болып табылатын шифрды атайды. AES (Rijndael), Serpent, Кузнечик шифрлары SP-желілерінің мысалдары болып табылады [36]. Күн өткен сайын мұндай мысалдар көбейіп келеді. Дәл осы криптографияның жаңа практикалық қосымшалары оның дамуының бір көзі болып отыр. Блоктық шифрлау алгоритмдерінің мәні ашық мәтіндердің блогына математикалық түрлендірулерді бірнеше қайталап қолдану болып табылады. Орындалатын түрлендірулердің негізгі мақсаты - шифрланған мәтіндер блогының әрбір биті, кілттің және ашық мәтіннің әрбір битіне тәуелділігін құру.

S-блок кіріс биттерінің кіші блогын басқа шығыс биттер блогымен ауыстырады және бұл ауыстыру қайтымдылықты қамтамасыз ету үшін өзара бірмәнді болуы керек. S-блокты қолданудағы мақсат сызықтық криптоталдау жүргізуге қарсы тұру үшін сызықтық емес түрлендіруді пайдалану.

Практикада шифрлаудың жалпы екі қағидасы қолданылады: шашырату және араластыру. Шашыратудың мақсаты – ашық мәтіндегі бір символдың өзгеруі шифр мәтіндегі көп символдың өзгеруіне алып келу. Бұл ашық мәтіннің статистикалық қасиеттерін жасыруға мүмкіндік береді. Осы қағиданы кілттің бір символының өзгерісінің шифр мәтінге әсерін де қадағалау қажет. Ал, бұл өз кезегінде кілтті бөлшектеп зерттеуге мүмкіндік бермейді. Араластыру ашық мәтін мен шифр мәтіннің статистикалық қасиеттері арасындағы байланысты қалпына келтіруді болдырмайтын түрлендірулерді қолданудан тұрады. Жақсы араластыруға қол жеткізудің қарапайым тәсілі – әрқайсысы аз-аздан араластыруға үлес қосатын қарапайым шифрлар тізбегі ретінде жүзеге асыру. Қарапайым шифрлар ретінде көбінесе алмастырулар мен ауыстырулар кестелерін қолданады [37].

Деректерді криптографиялық жолмен қорғау үдерісі бағдарламалық жасақтамада да, аппараттық құралдарда да жүзеге асырылуы мүмкін. Аппараттық іске асыру айтарлықтай қымбатқа түседі, бірақ сонымен қатар оның артықшылығы бар: жылдамдығы жоғары, қауіпсіздігі жоғары және т.б. Ал, бағдарламалық жасақтаманы іске асыру қолайлы болып табылады және пайдалануға икемдеуге мүмкіндік береді.

Іске асырылу түріне (бағдарламалық, аппараттық) қарамастан, ақпараттық қауіпсіздіктің заманауи криптографиялық жүйелеріне келесідей талаптар қойылған [38, 39]:

− шифрдың беріктілігі – оның криптоталдау әдістері бойынша жүргізілген шабуылдарға қарсы тұра алу мүмкіндігі кілттерді толық теріп шығу күшінен кем болмауы керек және қазіргі компьютерлердің есептеу мүмкіндіктерінен асып кетуі керек (желілік есептеуді ұйымдастыру мүмкіндігін ескере отырып);

− криптографиялық беріктілік алгоритмнің құпиялығымен емес, кілттің құпиялығымен ғана қамтамасыз етілуі керек, яғни алгоритм шабуылдаушыға қол жетімді деп саналады;

− шифрланған хабарлама кілт болған жағдайда ғана оқылатын болуы керек;

− шабуылдаушыға бастапқы ашық мәтіндер мен оларға сәйкес шифр мәтіндердің көп жұптары белгілі болса да, шифр берік болып қала беру керек;

− кілтке немесе ашық мәтінге енгізілген ең аз өзгерту шифр мәтінге айтарлықтай өзгерістер әкелуі керек;

− шифрлау алгоритмінің құрылымдық элементтері өзгеріссіз болуы керек;

− шифр мәтіннің көлемі бастапқы ақпараттан айтарлықтай аспауы керек; шифрлау үдерісінде енгізілген қосымша биттер шифр мәтінде сенімді жасырылуы керек;

− шифрлау кезінде пайда болатын қателіктер ақпаратты бұрмалауға және жоғалтуға әкелмеуі керек;

− шифрлау үдерісінде қолданылатын кілттер тізбегі арасында бір-біріне тәуелділіктері қарапайым байланыстар болмауы керек;

− кілттер жиынының ішінен кез-келгені криптографиялық беріктілікті қамтамасыз етуі керек, яғни әлсіз кілттер болмауы тиіс;

− шифрлау уақыты ұзақ болмауы керек;

− шифрлау құны жабылатын ақпарат құнымен сәйкес келуі керек;

− алгоритм шифрлау және шифрды кері ашу операцияларын орындауға арналған аппараттық құралдарда тиімді іске асырылуы керек;

− алгоритм қауіпсіздіктің әр түрлі деңгейлеріне сай жетілдіруге икемді болуы қажет.

Блоктық шифрларда ашық мәтіндердің бірдей блогын шифрлағанда бірдей шифрмәтінге түрленетіні түсінікті. Осы кемшіліктерді жасыру үшін блоктық шифрларды қолдану саласына, деректердің құпиялылық дәрежесіне, бастапқы мәтіннің көлеміне және қолданылатын кілттеріне байланысты әр түрлі режимдерді қолдануға болады [40-43].

Шифрлау режимі дегеніміз, бұл ашық мәтіндер блоктарының тізбегін шифр мәтіндер блоктарының тізбегіне түрлендіретін блоктық шифрлау әдісі. Ашық мәтіннің бір блогын шифрлағанда басқа блоктағы деректерді пайдалану мүмкін. Режимдердің әрқайсысының өзіне тән артықшылықтары мен кемшіліктері бар. Мысалы, кейбір режимдер шабуылдардың белгілі бір түріне берік болуы мүмкін, ал кейбіреулері байланыста үзіліс болып қалған жағдайларда қалпына келуі тезірек немесе жақсырақ болуы мүмкін. Блоктық шифрларды қолданудың барлық мүмкін режимдерінің тізімі шексіз болуы мүмкін, бірақ олардың арасынан ең жиі қолданылатын түрлерін бөліп қарауға болады [44]:

− қарапайым ауыстыру режимі;
− шифр мәтіннің блоктарын ілестірумен;
− шифр мәтін бойынша кері байланыспен;
− шығыс бойынша кері байланыспен;
− есептегіш бойынша;
− ашық мәтіннің блоктарын ілестірумен;
− ашық мәтін бойынша кері байланыспен;
− шифр мәтіндердің блоктарының күшейтілген ілуімен;
− шығыс бойынша кері байланыспен және сызықтық емес функциямен;
− сызықтық емес функциясы бар есептегіш бойынша.

Криптоалгоритмдерді олардың қауіпсіздігінің дәлелдену дәрежесі бойынша бірнеше топқа бөлу қалыптасқан. Олар абсолютті берік, дәлелденетін берік және болжамды берік криптоалгоритмдер. Абсолютті берік криптоалгоритмдердің қауіпсіздігі кілтті ашудың мүмкін еместігі туралы дәлелденген теоремаларға негізделген. Оған мысал ретінде, бір реттік блокнотты айтуға болады. Дәлелденетін берік криптоалгоритмдердің қауіпсіздігі көптеген математиктер шешуге тырысқан және олар қиын деп таныған белгілі математикалық есептерді шешудің күрделілігімен анықталады. Мысал ретінде, дискретті логарифмдер мен бүтін санды факторизациялаудың күрделілігіне негізделген Диффи-Хеллман немесе RSA жүйелерін келтіруге болады. Болжамды берік шифрлар дербес математикалық есептерді шешудің күрделілігіне негізделген. Бұл топқа блоктық алгоритмдердің басым көпшілігін жатқызуға болады [45].

1.2 Шифрлау алгоритмдердің сапасын бағалау критерийлері және криптоталдау әдістеріне шолу

Ақпараттық қауіпсіздік жүйелерінің (АҚЖ) басқа жүйелерден өзгешелігі, олар үшін ақпараттың сенімді қорғалатындығына көз жеткізетін қарапайым және бір мәнді тесттер жоқ. Сонымен қатар, АҚЖ-ның тиімділігі және олардың болуы тек негізгі жүйенің жұмысымен байланысты емес. Сондықтан, АҚЖ-нің тиімділігі мәселесін әдеттегі тестілеу арқылы шешу мүмкін емес. Мысалы, байланыс жүйесінің іске қабілетті екенін тексеру үшін оны сынақтан өткізу жеткілікті. Алайда, осы сынақтардың сәтті аяқталуы да оған енгізілген ақпаратты қорғаудың ішкі жүйесі де тиімді деген қорытынды жасауға мүмкіндік бермейді. Криптографиялық қорғау әдістері қолданылғанда АҚЖ құрудан қарағанда, оның тиімділігін анықтау міндеті көп еңбек, арнайы білімді қажет етеді. Көбінесе жаңа шифрды талдау техникалық емес жаңа ғылыми мәселе болып табылады.

Криптоталдауда негізгі әрекет етуші тұлға шабуылдаушы (немесе криптоталдаушы) деп аталады. Бұл криптографиялық әдістермен қорғалған хабарламаларды оқу немесе жалған ақпарат жасаушы болып табылатын адам немесе адамдар тобы [46].

Шабуылдаушының іс-әрекетіне қатысты бірқатар болжамдар жасалады:

1. Шабуылдаушы шифрлау алгоритмін (немесе ЭЦҚ генерациясын) және оны жүзеге асырудың ерекшеліктерін біледі, бірақ құпия кілтті білмейді.

2. Шифр мәтіндердің бәріне шабуылдаушы қол жеткізе алады және тиісті шифр мәтінге сәйкес бастапқы ашық мәтінге де қол жеткізе алуы мүмкін.

3. Шабуылдаушының қолында талдау жүргізуге қажетті адамдар саны, компьютерлік және басқа да ресурстар бар.

Шифрланған хабарламаны оқу немесе қолдан жасау, криптоталдау әдістерінің көмегімен кілтті есептеу әрекеті *криптошабуыл* немесе *шифрға шабуыл* деп аталады. Сәтті жасалған криптографиялық шабуыл *шифрдың бұзылуы* деп аталады.

Криптоберіктілік - бұл кілтті білмей-ақ шифрды бұзуға беріктілігін анықтайтын шифрдың сипаттамасы. Криптографиялық беріктікліктің көрсеткіші кез-келген криптожүйенің негізгі параметрі болып табылады. Беріктіліктің көрсеткіші ретінде келесілерді таңдауға болады:

- барлық мүмкін кілттердің саны немесе белгілі бір уақыт аралығында берілген ресурстармен кілт таңдау ықтималдығы;
- берілген ықтималдықпен шифрды бұзуға қажет операциялардың саны немесе уақыты;
- бастапқы ашық мәтіннің немесе басты ақпаратты есептеу құны.

Осы көрсеткіштер барлық мүмкін болатын криптошабуылдардың деңгейін ескеруі керек. Алайда, ақпаратты криптографиялық әдістермен қорғаудың тиімділігі шифрдың криптографиялық беріктілігіне ғана емес, сонымен қатар басқа да көптеген факторларға, соның ішінде криптожүйелерді құрылғылар немесе бағдарламалар түрінде іске асыруға байланысты екенін түсіну керек. Шифрдың криптографиялық беріктілігін талдағанда адам факторын ескеру

қажет. Мысалы, қолында қажетті ақпараттар шоғырланған белгілі бір адамды сатып алу, шифрды бұзу үшін суперкомпьютер жасағаннан гөрі бірнеше есе арзанға түсуі мүмкін [47].

Симметриялы блоктық шифрлауда қарастырастырылатын криптошабуылдардың және криптоталдаудың түрлерін қарастырайық.

Шифрмәтінге негізделген шабуыл: криптоталдаушыға тек $y_1, y_2, \dots, y_n$ шифрмәтіндер ғана белгілі, ал оған сәйкес бастапқы $x_1, x_2, \dots, x_n$ ашық мәтіндер белгісіз. Шабуылдаушыға ең болмаса $x_i, i = \overline{1,n}$ – нің біреуін немесе соған сәйкес кілтті $\left(k_i,\ i = \overline{1,n}\right)$ табу керек, болмаса мұны іске асыру мүмкін еместігіне көз жеткізуі.

Белгілі ашық мәтінге негізделген шабуыл: Криптоталдаушыға ашық мәтіннің және оған тиісті шифрмәтіннің жұптары $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ белгілі. Жұптардың кем дегенде біреуі үшін кілтті анықтау қажет. Кейбір жеке жағдайда, мысалы $k_1 = k_2 = \cdots = k_n$ болса, k – ні табуға тырысу керек. Бұл мүмкін еместігіне көз жеткізген жағдайда, сол кілтпен шифрланған басқа криптограмманың ашық мәтінін анықтау қажет.

Таңдалған ашық мәтінге негізделген шабуыл: алдыңғы түрден айырмашылығы тек криптоталдаушының $x_1, x_2, \dots, x_n$ ашық мәтінді таңдау мүмкіндігіне ие болуында. Шабуылдың да мақсаты алдыңғысымен бірдей. Мұндай шабуыл жүргізу криптоталдаушының ақпаратты жіберуші жақтың шифрлау аппаратына қол жеткізген жағдайда жүзеге асырылуы мүмкін.

Таңдалған шифр мәтінге негізделген шабуылдың екінші шабуылдан айырмашылығы тек криптоталдаушыға $y_1, y_2, \dots, y_n$ шифр мәтіндерін таңдау мүмкіндігі бар. Шабуылдың мақсаты да дәл екінші жағдайдағыдай. Мұндай шабуыл криптоталдаушыға қабылдаушы тараптың аппаратына қол жетімді болған жағдайда мүмкін [24, с. 169]. Таңдалған шифр мәтіндерге негізделген шабуылдар ең қауіпті болып саналады. Кейде бұл шабуылдарға басқалады да қосады. Барлық шабуылдарға төтеп бере алатын шифрды берік немесе сенімді деп санауға болады.

Криптошабуылдардың айырмашылықтарын және олардың тиімділігін қарапайым ауыстыру шифрына криптоталдау жүргізе отырып түсіндіруге болады. Бұл шифрды тек шифр мәтінге негізделген шабуылмен бұзу оңай болғанымен де, оны іске асыруға біраз тырысу керек. Белгілі ашық мәтінге негізделген шабуыл жасаған кезде, тек қана шифрмәтінге негізделген шабуылдан қарағанда жұмыс жеңілдейді.

Қазіргі заманғы криптоталдау ықтималдықтар теориясы және математикалық статистика, алгебра, сандар теориясы, алгоритмдер теориясы және басқа да бірқатар математикалық ғылымдарға негізделген. Криптоталдаудың барлық әдістерін негізгі төрт бағытқа жатқызуға болады [36, с. 19].

1. Статистикалық криптоталдау - ашық және шифр мәтендердің статистикалық заңдылықтарын зерттеу негізінде криптожүйелерді бұзу мүмкіндіктерін зерттейді. Көп жағдайларда, ақпаратты шифрлауға дейін оны сығымдайтын (архивтейтін) болғандықтан статистикалық талдауды нақты

практикада қолдану қиындайды. Ал гаммалауды қолданған жағдайда ұзын псевдокездейсоқ тізбектер қолданылады.

2. Алгебралық криптоталдау – шифрлау алгоритмдерінің математикалық әлсіз бөліктерін іздейді. Мысалы, 1997 жылы криптоталдауды едәуір жеңілдететін эллиптикалық жүйелерде кілттер классы анықталды. Алгебралық криптоталдауларды заманауи берік шифрларғада қолдануға болады. Әдістің негізі болып ашық мәтін, шифрмәтін және кілттің элементтерін байланыстыратын теңдеулер жүйесін құру және оны шешу болып табылады. Қазіргі кезде оның әртүрлі әдістері бар, мысалы XL, XLS және т.б. Бұл әдістің басқа әдістерден артықшылығы ашық мәтіндер мен шифрмәтіндер жұптарының саны аз болған кезде де шифрлау кілтін таба алу мүмкіндігінде.

3. Дифференциалдық криптоталдау шифр мәтіндегі өзгерістердің ашық мәтініндегі өзгерістерге тәуелділігін талдау негізінде жүргізіледі. Бұл әдісті ең бірінші болып ағылшын крптографы Шон Мерфи қолданды. Кейінірек Израиль ғалымдары Эли Бихам мен Ади Шамир DES алгоритміне шабуыл жасау үшін осы әдісті одан әрі дамытты. Осы әдіс DES алгоритмін бұзуда есептеу қиындығын 2^{55}-не төмендеткен бірінші криптографиялық әдіс болды. Дифференциалдық криптоталдау әдісі белгілі бір айырмашылығы бар мәтіндер жұптарының көптеген жиынтығын талдауды қажет етеді, бұл жоғары ықтималдықпен кілттің белгілі бір бөлігін (немесе толық кілтін) табуға мүмкіндік береді.

4. Сызықтық криптоталдау – бастапқы ашық мәтін мен шифрмәтін арасындағы сызықтық жуықтауды іздеуге негізделген әдіс. Бұл әдісте бірінші болып DES шифрын бұзуға қолданылды және оны жапон ғалымы Мацуи 1993 жылы жариялады. Бұл әдіс, бүгінгі күні дифференциалдық криптоталдаумен қатар, блоктық шифрларды зерттеудің кең таралған әдістерінің бірі. Блоктық шифрлау алгоритмдерінде ауыстыру блогының кіріс және шығыс биттері арасындағы статистикалық сызықтық қатынасты құруды қолданады.

Алгоритмдердің беріктілігін талдаудың негізгі әдістеріне толық теру әдісі де жатады. Бұл әдіс, барлық мүмкін болатын кілттердің көмегімен шифр мәтінді кері ашып ашық мәтін алынғанын немесе алынбағанын тексеруден тұрады. Егер алгоритм есептеу кезінде бұл шабуылға тұрақсыз болса, онда оны әрі қарай қолдану қорғалатын ақпараттың құпиялылығын қамтамасыз етпейді. Толық теру әдісі шифрлау алгоритмдерінің беріктігін талдаудағы әмбебап әдісі болып табылады және басқа әдістердің нәтижесін салыстыратын өлшем болып есептеледі. Алгоритмнің бұл әдіске беріктілігі кілттің ұзындығы мен түрлендірулеру операцияларының жылдамдығына байланысты.

Жоғарыда аталған әдістерді негізге ала отырып жасалған басқа да криптоталдаудың әдістері бар. Оларға бумеранг әдісі, дифференциалды-сызықты, мүмкін емес дифференциал, интегралдық және тағы да басқа әдістер жатады.

2 АУЫСТЫРУ – АЛМАСТЫРУ ЖҮЙЕСІ НЕГІЗІНДЕ СИММЕТРИЯЛЫ БЛОКТЫҚ ШИФРЛАУ АЛГОРИТМІН ҚҰРУ

Симметриялы блоктық шифрлау алгоритмдері, қазіргі уақытта, ақпараттық және телекоммуникациялық жүйелерде ақпаратты өңдеу барысында, құпиялылықты қамтамасыз етудің негізгі криптографиялық құралдары болып табылады [38, с. 48]. Кілтті беру механизмінің әлсіздігіне қарамастан симметриялы шифрлау жүйесі бүгінде, асимметриялы шифрлаудан жылдамдығының жоғарылығының арқасында өзекті болып отыр. Мемлекеттік дәрежедегі құпияларды сақтауда қолданылатын стандарттар алгоритмі, негізінен симметриялық блоктық шифрлауды қолданады. Заманауи симметриялы блоктық шифрлар негізінен екі тәсілге негізделген: Фейстель желісі және ауыстыру-алмастыру желісі (SP-желісі). Шифрлар ашық мәтіндерді қайтарылымды түрлендірулерге негізделген. Алгоритмдерді әзірлеу кезінде операциялардың әрқайсысы криптографиялық сенімді және кілтті білген жағдайда қайтарылымды екенін қадағалау қажет. Заманауи шифрлар Керкхофф қағидасына негізделген [28, с. 120], оған сәйкес, шифрдың құпиялығы алгоритмінің құпиялылығымен емес кілтінің құпиялылығымен ғана қамтамасыз етілуі керек.

Қазақстанда ақпараттық технологияларды және ақпаратты криптографиялық қорғау құралдарын қолдану туралы бірқатар заңдар қабылданды [48-53]. ҚР Қауіпсіздік Кеңесі қабылдаған Ақпараттық қауіпсіздік тұжырымдамасы, отандық ақпараттық қауіпсіздік жүйесін құру, оның әртүрлі деңгейдегі құпиялылығы үшін ақпаратты қорғаудың әдістері, модельдері мен алгоритмдерін, содан кейін оларды аппараттық және бағдарламалық қамтамасыз етуді құру қажеттілігін көрсетеді.

ҚР БҒМ Ақпараттық-есептеуіш технологиялар институтының АҚЗ-да жаңа шифрлау жүйелерін, ЭЦҚ әзірлеу және талдау, криптографиялық кілттерді аутентификациялау, сондай-ақ солардың негізінде АККҚ бағдарламалық жасақтамасын құруда позициялық емес полиномдық санау жүйесін пайдалану жұмыстары бойынша ҒЗЖ жүргізіліп келеді.

Бұл саладағы зерттеулер бұрынғы Кеңес өкіметінің ғалымдары И.Я.Акушский, Д.И.Юдицкий және В.М.Амербаевтардың зерттеу еңбектері арқылы кең тарады [54, 55]. Модульдік арифметиканы одан әрі дамыту бағыттарының бірі, отандық ғалым Р.Г.Бияшевтің қателіктерді табу және түзетуге, сонымен бірге, өзін-өзі түзететін кодтар жасау үшін қолданылатын позициялық емес полиномдық санау жүйелерін құру және қолдану бойынша еңбегі [56]. Ол симметриялы блоктық шифрлау жүйесін құруда ПЕПСЖ-ның алгебрасының негізгі ережелерін негіздеді. Бұл жүйені әр түрлі қосымшаларда іс жүзінде қолдану үшін оны іске асыруға арналған ұсыныстар жиынтығын әзірлеу қажет. Мұндай ұсыныстарды алу бірнеше бағытта зерттеуді қажет етеді. Оның бірі, позициялық емес жүйелерді симметриялы блоктық шифрлау жүйесінде практикада қолдану мақсатында модельдеу.

2.1 Позициалық емес полиномдық санау жүйесін құру

ПЕПСЖ құру – бұл жұмыс негіздері деп аталатын оның көпмүшеліктерін таңдап алу. Мұндай жұмыс негіздері ретінде бірнеше келтірілмейтін көпмүшеліктер таңдап алынсын:

$$p_1(x), p_2(x), \ldots, p_S(x) \tag{2.1}$$

Олардың дәрежелерін сәйкесінше $m_1, m_2, \ldots, m_S$ деп белгілейік. (2.1) көпмүшелері олардың орналасу ретін ескере отырып, бір негіздер жүйесін құрайды. ПЕПСЖ-нің негізгі жұмыс ауқымы – бұл дәрежесі $m = \sum_{i=1}^{S} m_i$ болатын $P^m(x) = p_1(x)\, p_2(x)\ \cdots\ p_S(x)$ көпмүшелігі.

ПЕПСЖ-да дәрежесі m-нен кіші болатын кез-келген $F(x)$ көпмүшелігі, оны (2.1) жұмыс негіздері бойынша қалдықтар тізбегі түрінде жалғыз позициялық емес түрге ие:

$$F(x) = \big(\alpha_1(x), \alpha_2(x), \ldots, \alpha_S(x)\big) \tag{2.2}$$

мұндағы $\alpha_i \equiv F(x)\big(mod p_i(x)\big), i = 1, \ldots, S.$ $F(x)$ – ті (2.2) түрдегі позициялық емес көрінісі арқылы бастапқы қалпына келтіруге болады [13, с. 84; 57, 58]:

$$F(x) = \sum_{i=1}^{S} \alpha_i(x) B_i(x), \text{ мұндағы } B_i(x) = \frac{P^m(x)}{p_i(x)} M_i(x) \equiv 1\big(mod p_i(x)\big). \tag{2.3}$$

$M_i(x)$ көпмүшеліктері (2.3) салыстырымы орындалатындай таңдап алынады. Тек ақпаратты беру және сақтау жағдайында $F(x)$ полиномын позициялық қалпына келтіру, келесі формула бойынша жүзеге асырылады:

$$F(x) = \sum_{i=1}^{S} \alpha_i(x) P_i(x), \quad \text{мұндағы} \quad P_i(x) = \frac{P^m(x)}{p_i(x)}. \tag{2.4}$$

Әрбір жұмыс негіздерінің дәрежесі L - дің мәнінен аспайтын болуы керек (біздің жағдайда 128, 192 және 256). Жұмыс негіздері ретінде (2.4) теңдігінің шарты орындалатындай дәрежелері m_1-ден m_S -ке дейінгі барлық келтірілмейтін көпмүшеліктер ішінен таңдалына алады:

$$k_1 m_1 + k_2 m_2 + \cdots + k_S m_S = L \tag{2.5}$$

(2.5) теңдеуде $0 < k_i < n_i, i = 1, \ldots, S$ - белгісіз коэффициенттер және дәрежесі m_i болатын таңдалған келтірілмейтін көпмүшеліктердің саны, осы коэффициенттердің бір жиынтығы (2.5)-тің шешімдерінің бірі болып табылады және жұмыс негіздерінің бір жүйесін береді, ал n_i дәрежесі m_i болатын барлық келтірілмейтін полиномдардың саны, $1 \le m_i \le L, \quad S = k_1 + k_2 + \cdots + k_S$ таңдалған жұмыс негіздерінің саны. m_i дәрежелі көпмүшеліктер модульдері

бойынша толық қалдықтар жүйелеріне, оларды жазу үшін m_i қажет болатын, яғни дәрежесі $m_i - 1$-ден артпайтын барлық полиномдары кіреді.

Шифрлау. Деректерді шифрлауда жылдамырақ жұмыс істеу үшін, ұзындығы L биттен тұратын пайдаланылып отырған кілт тізбегін келтірілмейтін көпмүшеліктердің (жұмыс негіздерінің) дәрежесіне қарай $k_1(x), k_2(x), \ldots, k_S(x)$ түріндегі қалдықтарының тізбегі ретінде кесіп аламыз.

$$K(x) = \big(k_1(x), k_2(x), \ldots, k_S(x)\big) \tag{2.6}$$

Онда $\omega_1(x), \omega_2(x), \ldots, \omega_S(x)$ криптограммасы ретінде $H(F(x), K(x))$ шифрлау функциясының мәнін қарастыруға болады:

$$H(x) = \big(\omega_1(x), \omega_2(x), \ldots, \omega_S(x)\big), \tag{2.7}$$

$F(x)$, $K(x)$, $H(x)$ функциялары ПЕПСЖ-де орындалатын операцияларға сәйкес, жұмыс негіздері ретінде таңдалған (2.1) полиномдардың модулі бойынша параллель орындалады.

Егер $H(F(x), K(x))$ функциясы ретінде көбейту операциясы қолданылатын болса, онда шифрдің $\omega_1(x), \omega_2(x), \ldots, \omega_S(x)$ элементтерін алу үшін $\alpha_i(x) \cdot k_i(x)$ көбейтіндісін сәйкес $p_i(x)$ жұмыс негіздеріне бөлгендегі қалдықтар алынады [14 с. 14; 15 с. 393]:

$$\alpha_i(x)\, k_i(x) \equiv \omega_i(x)\big(mod p_i(x)\big), i = 1, \ldots, S. \tag{2.8}$$

Шифрді кері ашу. $H(x)$ криптограммасын белгілі $K(x)$ кілті бойынша кері ашу үшін $k_i(x)$-дің әрбір мәніне сәйкес келесі салыстырылым шарты орындалатындай, кері $k_i^{-1}(x)$ көпмүшеліктерін есептеулер жүргізу арқылы табамыз:

$$k(x)k_i^{-1}(x) = 1\big(mod p_i(x)\big), i = 1, \ldots, S \tag{2.9}$$

Нәтижесінде, $K(x)$ көпмүшелігінің керісі болып табылатын $K^{-1}(x) = \big(k_1^{-1}(x), k_2^{-1}(x), \ldots, k_S^{-1}(x)\big)$ көпмүшелігін аламыз. Онда (2.2) қалдықтар тізбегінің элементтерін келесідей формуламен қалыпқа келтіруге (табуға) болады:

$$\alpha_i(x) = k_i^{-1}(x)\omega_i(x)\big(mod p_i(x)\big), i = 1, \ldots, S$$

Осылайша, берілген L ұзындықты электрондық хабарламаларды шифрлау алгоритмінің қарастырып отырған ПЕПСЖ-ға негізделген модификациясында толық кілт ретінде таңдалған $p_1(x), p_2(x), \ldots, p_S(x)$ полиномдық жұмыс негіздері жүйесі, $K(x) = \big(k_1(x), k_2(x), \ldots, k_S(x)\big)$ кілті және шифрді ашу үшін қажет оған кері $K^{-1}(x) = \big(k_1^{-1}, k_2^{-1}(x), \ldots, k_S^{-1}(x)\big)$ кілттері болып табылады.

2.2 «Qamal» және «Qamal NPNS» шифрлау алгоритмдерін құру

Берік блоктық шифрлар белгілі бір шарттарды қанағаттандыруы тиіс. Бұл шарттарды К.Шеннонның өзінің шифрлау теориясы бойынша жазған бірқатар іргелі жұмыстарында тұжырымдаған [5, с. 11]. Берік шифрларда шашырату және араластыру қасиеттері болуы қажет.

Шашырату: бұл бастапқы мәтіннің бір таңбасы (биті) шифр мәтінінің бірнеше символына (биттеріне), оңтайлы жағдайда - бір блок шеңберіндегі барлық символдарына әсер ететін шифр қасиеті. Егер бұл шарт орындалған жағдайда мәтіндер айырмашылығы ең аз болатындай екі блоктарды шифрлау барысында бір-біріне мүлде ұқсамайтын шифрмәтіндер алатын боламыз. Дәл осы шарт шифрмәтіннің кілттен тәуелділігіне де орындалуы қажет, яғни кілттің бір таңбасы (биті) шифр мәтінінің бірнеше таңбасына (биттеріне) әсер етуі керек. Сонымен, шашырату - шифрмәтін мен ашық мәтіннің арасындағы байланысты жасырады немесе бүркемелейді.

Араластыру: шифрдың бұл қасиеті ашық мәтіннің символдары мен шифр мәтіннің арасындағы тәуелділікті жасырады. Егер шифр бастапқы мәтіннің биттерін жеткілікті түрде «араластыратын» болса, онда сәйкес шифр мәтінінде ешқандай статистикалық және функционалдық заңдылықтар болмайды. Яғни, араластыру - шифр мәтін мен кілт арасындағы қатынасты жасырады.

Жоғарыда айтылғандарды және бірінші бөлімде көрсетілген симметриялы блоктық алгоритмдерге қойылатын талаптарды ескере отырып, SP желісіне негізделген симметриялы блоктық шифрлау алгоритмі әзірленді және кілтті пайдалану түріне қарай екі нұсқасы ұсынылды. Жаңа алгоритм шартты түрде «Qamal» және «Qamal NPNS» деп аталады.

Әзірленген «Qamal» шифрлау алгоритмінің құрылымдық сұлбасы сурет 2.1 де көрсетілген. Алгоритмнің 128, 192 және 256 битті ұзындықтағы блоктармен және кілттермен жұмыс жасайтын мүмкіндіктері бар. Шифрлау раундтарының саны блоктың және кілттің ұзындығына байланысты. Ұзындықтары 128, 192 және 256 бит болатын кілттердің раунд сандары сәйкесінше 8, 10 және 12. Барлық раундтар, раундттық кілттерді модуль 2 бойынша қосумен аяқталады [58, p. 198; 59, 60, 61].

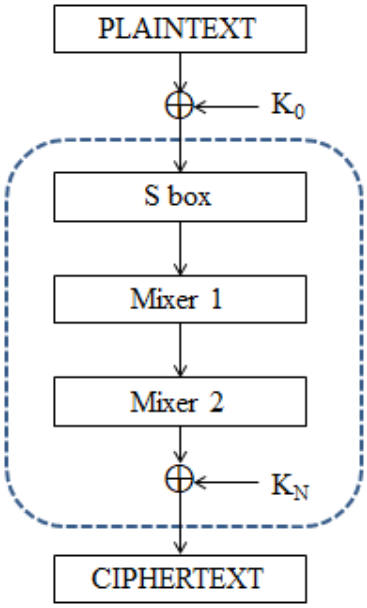
Шифрлау алгоритмінде кілттерді биттік қосу (XOR) операциясы, S-блок ауыстыруы, Mixer1 және Mixer2 араластыру процедуралары пайдаланылған.

Бірінші процедура – кілтті ашық мәтіннің блогына модуль 2 бойынша қосу операциясы (XOR) орындалады.

Екінші процедура – байттарды сызықты емес түрлендіретін S-блок ауыстыру кестесін құру, әрбір байтқа сызықты емес биективтік ауыстыру қолданылады.

Қазіргі заманда стандарт ретінде қолданылатын симметриялы блоктық алгоритмдердің көпшілігінде S-блоктар қолданылады. Шифрлау алгоритмдерінің беріктілігін арттыруда S-блоктарды пайдаланудың өте үлкен маңыздылығы бар. S-блоктардың қасиеттерін зерттеу және оларды жетілдіру

жолдарын қарастыру блоктық шифрларды құрудағы бірден-бір басты есептердің бірі [36, с. 84].



Сурет 2.1 – «Qamal» шифрлау алгоритмінің сұлбасы

S-блоктың негізгі сипаттамаларының бірі - оның мөлшері болып саналады. Кірісі n-биттен тұратын мәндер және шығысы m-биттен тұратын мәндерді сәйкестендіретін $n \times m$ мөлшердегі S-блок (немесе кесте) дейді. S-блоктың мөлшері қаншалықты үлкен болса, криптоталдаудың сызықтық және дифференциалдық әдістеріне қатысты алгоритмнің беріктілігі соншалықты артады. Бірақ, мөлшері үлкен болған сайын оны практикалық жобалау қиынға соғады. Сондықтан, S-блоктың мөлшері практикалық себептері бойынша 8-ден 10-ға дейінгі аралықта (диапазонында) болу керектігін В. Столлингс пен Л.К. Бабенконың кітаптарында айтады [38, с. 117; 47, с. 19].

Құрылған алгоритмде қолданылған S-блокты алу жолы Rijndael шифрлау алгоритміне ұқсас [62], айырмашылығы таңдалған келтірілмейтін көпмүшеліктері мен көбейтілетін матрицасында. Шифрлау алгоритміне арнап екі S-блок құрылды, біріншісі негізгі алгоритмнің құрамына кірсе, екіншісі раундтық кілттерді алуда қолданылды. Зерттеу жұмыстары көрсеткендей басқа да таңдалған көпмүшеліктер арқылы алынған S-блоктың балама нұсқаларының криптобериктілігі төмендемейтінін көрсетті. Зерттеу нәтижелері үшінші бөлімде толықтай келтірілген.

S-блок құруға пайдаланылған келтірілмейтін көпмүшеліктер $z_i(x)$, векторлар v_i (мұндағы $i = 1,2$) және матрица төменде келтірілген:

$$z_1(x) = x^8 + x^5 + x^4 + x^3 + 1,\ v_1 = (11001001)$$
$$z_2(x) = x^8 + x^5 + x^3 + x^2 + 1,\ v_2 = (01010101)$$

S-блоктың әрбір кіріс элементіне сәйкес шығыс элементін алу жолын көрсетейік:

1) $GF(2^8)$ өрісінде әрбір элементтің кері элементін аламыз;
2) Екінші түрлендіру келесідей орындалады:

$$\begin{bmatrix} s'_0 \\ s'_1 \\ s'_2 \\ s'_3 \\ s'_4 \\ s'_5 \\ s'_6 \\ s'_7 \end{bmatrix} = \begin{bmatrix} 1\,0\,1\,1\,1\,1\,1\,1 \\ 1\,1\,0\,1\,1\,1\,1\,1 \\ 1\,1\,1\,0\,1\,1\,1\,1 \\ 1\,1\,1\,1\,0\,1\,1\,1 \\ 1\,1\,1\,1\,1\,0\,1\,1 \\ 1\,1\,1\,1\,1\,1\,0\,1 \\ 1\,1\,1\,1\,1\,1\,1\,0 \\ 0\,1\,1\,1\,1\,1\,1\,1 \end{bmatrix} \text{x} \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \end{bmatrix} + \begin{bmatrix} v_0 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{bmatrix}$$

мұндағы, s_i, $i = 0, \ldots, 7$ – алынған кері элементтердің $GF(2)$ өрісіндегі коэффициенттері.

Шифрлау алгоритмінде қолданылған S1-блок кесте 2.1 - де көрсетілген.

Кесте 2.1 – S1-блок ауыстыру кестесі

Ауыстыру кестесінің мәндері																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	C9	34	F0	18	55	86	21	6B	87	D2	6E	99	BD	31	98	89
1	29	73	83	8B	1A	19	E1	E4	F3	5B	72	3F	A6	F9	2E	A3
2	7E	10	94	07	EC	AD	2F	26	20	93	66	3D	DD	64	5F	C1
3	13	E0	80	25	D3	08	75	6A	B9	2D	D1	CC	FD	CA	3B	FC
4	D5	DA	E2	CE	A0	7F	AE	C8	9C	09	3C	95	BA	35	3E	7B
5	FA	8D	23	AB	D9	E8	74	2A	C3	A8	D8	52	45	B5	0A	0C
6	A4	61	9A	FB	AA	F6	78	84	C4	E9	EE	54	50	81	DF	90
7	36	B4	BB	44	C5	96	4B	28	14	E6	8F	FF	B0	1F	53	47
8	00	4C	40	2C	9B	9F	4A	01	7D	AF	92	56	7A	DB	8E	16
9	63	24	A9	1D	33	4D	E7	1C	70	69	B7	C6	32	E5	57	03
A	97	A5	EB	D4	BC	5D	F8	85	06	F2	59	F4	17	22	38	DC
B	0B	FE	BE	CD	41	82	04	0E	48	71	30	AC	EF	C7	2B	CB
C	B8	8C	5A	42	A7	4E	D0	46	BF	B3	91	E3	11	7C	6F	DE
D	88	58	1E	5C	9D	60	C0	62	05	79	ED	76	C2	02	65	D7
E	F1	8A	77	F7	37	B1	0F	67	CF	0D	A1	6C	4F	3A	39	1B
F	27	B6	5E	F5	EA	6D	15	9E	B2	12	A2	68	43	51	49	D6

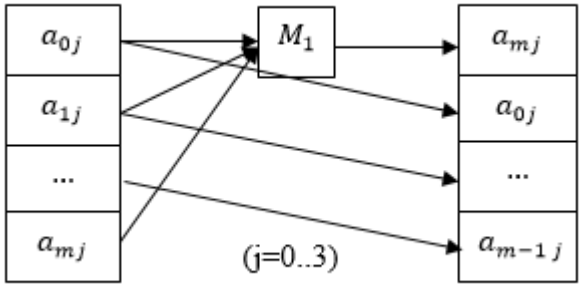
Үшінші процедура – Mixer1 түрлендіруі. Блоктың байттары өлшемі $m \times 4$ болатындай екі өлшемді A массиві түрінде жазылады, мұндағы m бастапқы блоктың ұзындығына байланысты 4, 6 және 8 мәндерін қабылдайды:

$$A = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ . & . & . & . \\ a_{m0} & a_{m1} & a_{m2} & a_{m3} \end{bmatrix}.$$

Әрбір бағанның байттары модуль 256 бойынша бір-бірімен қосылады:

$$M_1(b_{ij}) = \sum_{i=0}^{m} a_{ij} \; mod \; 256 \, , j = \overline{0,3}.$$

Содан кейін бірінші бағанда алынған жаңа байт жоғарғы a_{00} байтының орнына жазылады, ал бастапқы байт бір позицияға төмен жылжытылады. Бұл әрекет m рет қайталанады. Нәтижесінде бірінші бағандағы жаңа m байт аламыз. Ары қарай, бұл операция қалған үш баған үшін де орындалады (сурет 2.2).



Сурет 2.2 – Mixer1 түрлендіруінің сұлбасы

Төртінші процедура - Mixer2 түрлендіруі. Mixer1 блогін іске асыру нәтижесінде өлшемі $m \times 4$ болатындай жаңа B массиві алынады, мұндағы m бастапқы блоктың ұзындығына байланысты 4, 6 және 8 мәндерін:

$$B = \begin{bmatrix} b_{00} & b_{01} & b_{02} & b_{03} \\ b_{10} & b_{11} & b_{12} & b_{13} \\ . & . & . & . \\ b_{m0} & b_{m1} & b_{m2} & b_{m3} \end{bmatrix}.$$

Массивтің әр жолын коэффициенттері $GF(2^8)$ ақырлы өрісінде жататындай үшінші дәрежелі полином түрінде қарастырамыз. Бұл көпмүшеліктер келесідей түрде болады:

$$b_i(x) = b_{i0}x^3 + b_{i1}x^2 + b_{i2}x + b_{i3}, i = 0, \ldots, m \, .$$

Әрбір $b_i(x)$ көпмүшелігі алдын-ала таңдалған $m_i(x)$ бектітілген көпмүшеліктеріне $p(x)$ модулі бойынша көбейтіледі:

$$c_i(x) = b_i(x) \cdot m_i(x) \left(mod\ p(x)\right)$$

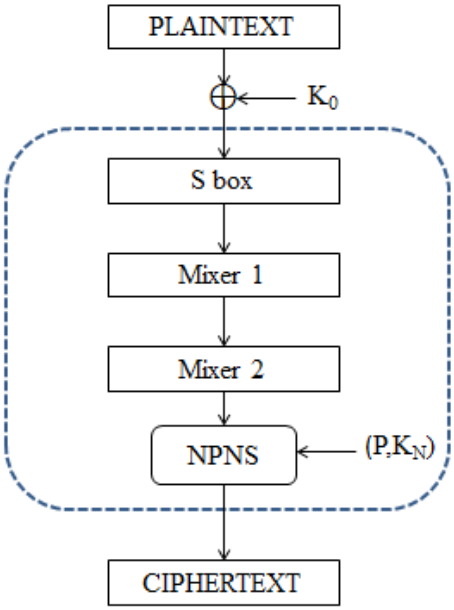
$m_0(x) = 168\ x^3 + 34x^2 + 187x + 186$, $m_1(x) = 210x^3 + 53x^2 + 210x + 101$,
$m_2(x) = 218x^3 + 25x^2 + 150x + 210$, $m_3(x) = 144x^3 + 75x^2 + 158x + 27$,
$m_4(x) = 163x^3 + 4x^2 + 111x + 106$, $m_5(x) = 150x^3 + 237x^2 + 13x + 53$,
$m_6(x) = 99x^3 + 59x^2 + 104x + 205$, $m_7(x) = 167x^3 + 49x^2 + 241x + 154$,
$p(x) = x^4 + x + 55$.

$m_i(x)$ көпмүшеліктері келесідей қолданылады. Ашық блоктың ұзындығы 128 бит болған жағдайда, алғашқы төрт $m_0(x)$, $m_1(x)$, $m_2(x)$, $m_3(x)$ көпмүшеліктері пайдаланылады. Блок ұзындығы 192 бит үшін алғашқы 6 алты көпмүшелік $m_0(x)$, $m_1(x)$, $m_2(x)$, $m_3(x), m_4(x), m_5(x)$ алынады. Үшінші ықтимал блок ұзындығы үшін барлық сегіз көпмүшелік қолданылады.

«Qamal NPNS» шифрлау алгоритмі.

Шифрлау алгоритмінің екінші нұсқасы ұсынылады. Жоғарыда айтылғандай негізгі шифрлау алгоритмінде кілттерді биттік қосу (XOR) операциясы, S-блок ауыстыруы, Mixer1 және Mixer2 араластыру процедуралары пайдаланылған [58, p. 198].

Қарастырылғалы отырған алгоритмнің құрылысында позициялық емес полиномдық санау жүйесі қолданылған және жоғарыда айтылған Qamal алгоритмінің модификациясы болып табылады (сурет 2.3). Негізгі айырмашылығы - кілтті ашық мәтіннің блогына модуль 2 бойынша қосу операциясының (XOR) орнына ПЕПСЖ бойынша көбейту орындалады. Осыған байланысты алгоритм «Qamal NPNS» (NPNS – ПЕПСЖ-нің ағылшынша аббревиатурасы) деп аталды. Әзірленген алгоритм ұзындығы 128 бит болатын бекітілген ұзындықтағы блокпен және кілтпен жұмыс істейді. Бұл да негізгі алгоритмнен өзгешелігінің бірі болып табылады. Сонымен қатар, раунд саны 4-ке тең. Шифрлауда ПЕПСЖ-ні қолдану жолы 2.1- бөлімде баяндалған.



Сурет 2.3 – «Qamal NPNS» шифрлау алгоритмінің сұлбасы

2.3 Шифрді кері ашу және раундтық кілттерді алу алгоритмі

Шифрмәтінді кері ашу үшін, шифрлау үшін қолданылатын барлық криптографиялық түрлендірулер инверцияланады және шифрді ашу алгоритмінде кері ретпен пайдаланылады. Сондай-ақ, раундтық кілттер де кері ретпен қолданылады. Шифрді кері ашуда әрбір көрсетілген блок ұзындықтары үшін сәйкесінше 8, 10 және 12 раундтар орындалады. Олардың әрқайсысында *InvS1*, *InvMixer1*және *InvMixer2* кері операциялары іске асырылады.

InvS1 операциясы S1-блок блогінен элементтерді алуға кері операция. S1-блоктың байттары кері алмастыру жолымен жаңа байттарға алмасады. Нәтижесінде инверцияланған S1-блогы алынады (кесте 2.2).

Кесте 2.2 – Inv*S1* (S1-блокқа инверсия) ауыстыру кестесі

Кері ауыстыру кестесінің мәндері															
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	80	87	DD	9F	B6	D8	A8	23	35	49	5E	B0	5F	E9	B7	E6
1	21	CC	F9	30	78	F6	8F	AC	03	15	14	EF	97	93	D2	7D
2	28	06	AD	52	91	33	27	F0	77	10	57	BE	83	39	1E	26
3	BA	0D	9C	94	01	4D	70	E4	AE	EE	ED	3E	4A	2B	4E	1B
4	82	B4	C3	FC	73	5C	C7	7F	B8	FE	86	76	81	95	C5	EC
5	6C	FD	5B	7E	6B	04	8B	9E	D1	AA	C2	19	D3	A5	F2	2E
6	D5	61	D7	90	2D	DE	2A	E7	FB	99	37	07	EB	F5	0A	CE
7	98	B9	1A	11	56	36	DB	E2	66	D9	8C	4F	CD	88	20	45
8	32	6D	B5	12	67	A7	05	08	D0	0F	E1	13	C1	51	8E	7A
9	6F	CA	8A	29	22	4B	75	A0	0E	0B	62	84	48	D4	F7	85
A	44	EA	FA	1F	60	A1	1C	C4	59	92	64	53	BB	25	46	89
B	7C	E5	F8	C9	71	5D	F1	9A	C0	38	4C	72	A4	0C	B2	C8
C	D6	2F	DC	58	68	74	9B	BD	47	00	3D	BF	3B	B3	43	E8
D	C6	3A	09	34	A3	40	FF	DF	5A	54	41	8D	AF	2C	CF	6E
E	31	16	42	CB	17	9D	79	96	55	69	F4	A2	24	DA	6A	BC
F	02	E0	A9	18	AB	F3	65	E3	A6	1D	50	63	3F	3C	B1	7B

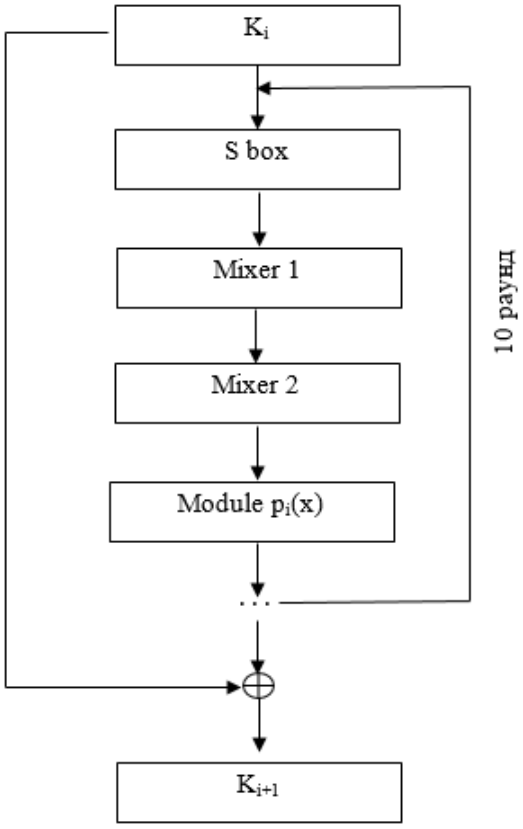
InvMixer1 операциясы $M_1(b_{ij})$-ге кері түрлендіру.

InvMixer2 операциясы - Mixer2 блогын алу процедурасына кері процедура болып табылады. Инверсияланған Mixer2 блогын алу үшін, массивтің әрбір жолы коэфициенттері $GF(2^8)$ өрісіндегі төрт мүшесі бар көпмүшелік ретінде қарастырылады. Бұл көпмүшеліктер $p(x)$ модулі бойынша бекітілген келесі $m_0^{-1}(x), m_1^{-1}(x), m_2^{-1}(x), m_3^{-1}(x), m_4^{-1}(x), m_5^{-1}(x), m_6^{-1}(x), m_7^{-1}(x)$ көпмүшеліктеріне көбейтіледі. Мұндағы,

$$m_0^{-1}(x) = 130x^3 + 142x^2 + 229x + 216, m_1^{-1}(x) = 60x^3 + 216x^2 + 18x + 81,$$
$$m_2^{-1}(x) = 133x^3 + 255x^2 + 246x + 210, m_3^{-1}(x) = 236x^3 + 90x^2 + 225x + 244,$$

$m_4^{-1}(x) = 232x^3 + 177x^2 + 164x + 110$, $m_5^{-1}(x) = 122x^3 + 68x^2 + 237x + 71$,
$m_6^{-1}(x) = 93x^3 + 150x^2 + 242x + 186$, $m_7^{-1}(x) = 66x^3 + 157x^2 + 115x + 197$

Раундтық кілттерді құру алгоритмі. K_i раундтық кілттері – кілтерді кеңейту процедураларын қолдану арқылы шифрдың бастапқы K_0 кілтінен жасалынады. Нәтижесінде, раундтық кілттердің тізбегі құрылады. Осы кілттердің ішінен тікелей қажетті раундтық кілттер таңдалады. Кілттерді алу сұлбасы сурет 2.4 - те көрсетілген.



Сурет 2.4 – K_i кілтін кеңейту сұлбасы, мұндағы i =0,1,…,6 (8,10)

Раундтық кілттерді алу процедурасы шифрлау үдерісінде қолданылатын барлық түрлендірулерді қамтиды, сонымен қатар, жоғарыда ескертілгендей өзіндік басқа ауыстыру кестесі (кесте 2.3) қолданылады және қосымша жаңа *Module $p_i(x)$* түрлендіруін пайдаланылады.

Module $p_i(x)$. $p_1(x)$, $p_2(x), \dots, p_S(x)$ – жұмыс негіздері ретінде қолданылатын коэффициенттері екілік жүйедегі келтірілмейтін көпмүшеліктер болсын.

Ескерте кететін бір жағдай, бұл жерде пайдаланылып отырған $p_i(x)$, $\overline{\iota = 1, s}$ - лердің Mixer2 түрлендіруінде қолданылған $p(x)$ көпмүшелігіне қатысы жоқ және $P(x) = p_1(x)\, p_2(x) \dots p_S(x)$. $P(x)$ көпмүшелігінің дәрежесі $N = m_1 + m_2 + \dots + m_S$ блоктың ұзындығына тең (яғни, 128, 192, 256).

Mixer2 блогынан шыққан мәндерді коэффициенттері екілік жүйеде болатын $N(x)$ көпмүшелігі ретінде қарастырамыз. Мұнда $k_1(x), k_2(x), \dots, k_S(x)$ - $N(x)$

көпмүшелігін сәйкес $p_i(x), i = 1, \dots, s$ жұмыс негіздеріне бөлгендегі қалдықтар деп қарастырамыз. Бұл жерде $p_i(x), i = 1, \dots, s$ кілттерді түрлендіру процедурасындағы құпия элемент болып табылады және оны жарияламаймыз.

Раундтық кілттерді құру алгоритмінде жоғарыдағы аталған қосымша түрлендіруді қосудағы мақсатымыз: криптоберіктілігі арта түседі, әлсіз кілттер болмайды және әртүрлі пайдаланушы топтар өзіндік $p_i(x), i = 1, \dots, s$ - лердің жиынын пайдалануға мүмкіндік береді.

Кесте 2.4 - те құрылған шифрлау алгоритмдерінің параметрлері, басқа да белгілі шифрлау алгоритмдерінің параметрлерімен қатар келтірілген.

Кесте 2.3 – Раундтық кілттерді құруда пайдаланылатын S2-блок

Раундтық кілттер алу алгоритміне қолданылған ауыстыру кесте мәндері																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	55	A8	78	9C	C3	ED	B1	DE	CD	2C	09	51	27	2D	43	C2
1	CA	45	3A	CE	7B	79	84	7D	BF	E6	69	1F	5E	CB	9E	E2
2	49	38	8E	7C	31	DF	98	42	91	57	90	A6	BD	F1	41	AC
3	20	96	8C	C7	4B	BE	70	E9	D0	4D	1A	A1	B0	DA	5D	D3
4	88	B5	30	47	6B	35	12	B2	B4	17	10	A2	60	9B	0D	FD
5	E4	C6	54	EB	B7	B9	7F	AF	21	5C	D4	99	5F	3E	A9	F3
6	3C	C0	67	13	6A	2F	1C	29	89	58	73	EC	14	39	D8	4E
7	44	02	59	23	F2	0C	FC	AB	74	87	92	36	82	04	16	0E
8	BB	01	F6	15	E7	DC	8F	07	4A	FF	65	1B	25	8B	75	D7
9	A5	7A	A7	FA	24	E5	AE	61	CF	9D	32	66	AA	05	D2	62
A	8D	C4	4F	26	06	0A	D9	7E	F7	E3	F0	34	40	0F	FB	1E
B	6F	A3	D1	BA	95	3D	33	71	83	18	E0	CC	2B	A0	D5	28
C	E1	64	9F	97	4C	A4	76	B3	19	08	68	C1	22	1D	B8	8A
D	E8	50	00	C9	46	56	5A	72	F5	3B	63	94	93	9A	0B	AD
E	DD	C8	FE	5B	53	85	6E	EE	86	80	F9	52	81	11	2A	48
F	C5	EA	EF	DB	B6	3F	37	77	6D	03	2E	D6	F4	BC	F8	6C

Кесте 2.4 – Белгілі шифрлау алгоритмдерінің кейбір параметрлері

Шифралгоритм атауы / Параметрлері және операциялар	DES	AES	IDEA	Кузнечик	Qamal	Qamal NPNS
1	2	3	4	5	6	7
Кілт ұзындығы	56	128 192 256	128	256	128/ 192/ 256	256

2.4-кестенің жалғасы

1	2	3	4	5	6	7
Блок ұзындығы	64	128	64	128	128/ 192 256	128
Раунд саны	16	10/12/14	8	9	8/10/12	4
Жылжыту	+	+	-	+	+	+
Орын ауыстыру	+	-	+	-	-	-
Алмастыру кестесі	+	+	-	+	+	+
2^n модулі бойынша қосу	-	-	+	-	+	+
2^n+1 модулі бойынша көбейту	-	-	+	-	-	-
Полиномдарды $GF(2^n)$-де көбейту	-	+	-	+	+	+

2.4 Деректерді шифрлау мысалы

Бастапқы ашық мәтін *X*-ті негізгі құпия *K* шифрлау кілтімен қалай шифрланатындығын қарастырайық, мұндағы:

X = 0x81754b8c671be306adee86fc52174dcd

K = 0x904b9e1bd6eaa64db9a9c168a5e5f92d

Жұмысты раундтық кілттерді алу алгоритмі арқылы ішкі кілттерді жасап алудан бастайық. Бірінші раундтың кілтін жасау үшін түрлендірудің он раунды орындалуы керек. Түрлендірулерде қолданылатын бастапқы мән ретінде негізгі құпия шифрлау кілтінің мәні болып табылады, яғни *K*= =0x904b9e1bd6eaa64db9a9c168a5e5f92d.

Бірінші түрлендіру кесте 2.3 - ке сәйкес байтты ауыстыру болып табылады. Сонымен, 0x90 байты 0xa5-ке, келесі 0x4b байт 0xa2-ге және т.с.с. Нәтижесінде K = 0x a5a2d21f5af9d99b18e364890a8503f1 мәні алынатын болады. Келесі түрлендіру - Mixer 1 түрлендіруін қолданып мәліметтерді араластыру. Оны орындау үшін K =0x a5a2d21f5af9d99b18e364890a8503f1 мәнін кесте 2.5 - те көрсетілгендей квадраттық матрица ретінде қарастырамыз [59, p. 2].

Кесте 2.5 – Мәліметтерді матрица ретінде көрсету

Кілт мәндері			
a5	a2	d2	1f
5a	f9	d9	9b
18	e3	64	89
0a	85	03	f1

Кесте 2.5 - тің бірінші (сол жақ) бағанымен жұмыс істейміз. Барлық байттарды 256 модулімен қосып, оларды ең жоғарғы ұяшыққа жазып, ұяшықтың

қалған мәндерін біреуге төмен жылжытамыз. Осы әрекетті барлық ұяшықтардағы мәндер өзгергенше төрт рет қайталау керек (кесте 2.6).

Кесте 2.6 – Бірінші бағанды Mixer1 көмегімен түрлендіру нәтижесі

Бастапқы күйі	Бірінші өзгеріс	Екінші өзгеріс	Үшінші өзгеріс	Төртінші өзгеріс
a5	(a5+5a+18+0a) mod 256 = 21	(21+a5+5a+18) mod 256 = 38	(38+21+a5+5a) mod 256 = 58	(58+38+21+a5) mod 256 = 56
5a	a5	21	38	58
18	5a	a5	21	38
0a	18	5a	a5	21

Қалған үш бағанғада дәл осыған ұқсас әрекеттерді қайталап орындап, жаңа түрленген жағдайын аламыз (кесте 2.7).

Кесте 2.7 – Mixer1 түрлендіруінің нәтижесі

Кілттің Mixer1 түрлендіруінен кейінгі нәтижесі			
56	45	e3	2f
58	1f	de	65
38	81	21	77
21	03	12	34

Mixer2 түрлендіруінің нәтижесінде келесі түрленген жағдайды аламыз:
(0x5645e32f*0xa822bbba) mod 0x100000137 =0x3c0d7d49
(0x581fde65*0xd235d265) mod 0x100000137 =0xafeeb61e
(0x38812177*0xda1996d2) mod 0x100000137 =0x6627b56f
(0x21031234*0x904b9e1b) mod 0x100000137 =0x24c7ed8f

Алынған нәтижелерді бір блокқа жинап, Mixer2 түрлендіруінің нәтижесін алатын боламыз: K = 0x3c0d7d49afeeb61e6627b56f24c7ed8f.

Бірінші раундттағы соңғы түрлендіру ModuleP болып табылады. Оны орындау үшін K = 0x3c0d7d49afeeb61e6627b56f24c7ed8f мәнін әртүрлі сегіз модульде қалдықтарын алып және алынған мәндерден жаңа блок құрастыру керек:
0x3c0d7d49afeeb61e6627b56f24c7ed8fmod 0x1002b =0x0ccf
0x3c0d7d49afeeb61e6627b56f24c7ed8fmod 0x1002d =0x5463
0x3c0d7d49afeeb61e6627b56f24c7ed8fmod 0x10039 =0xb236
0x3c0d7d49afeeb61e6627b56f24c7ed8fmod 0x1003f =0xa02f
0x3c0d7d49afeeb61e6627b56f24c7ed8fmod 0x10047 =0x338f
0x3c0d7d49afeeb61e6627b56f24c7ed8fmod 0x10053 =0xd5c2
0x3c0d7d49afeeb61e6627b56f24c7ed8fmod 0x1008d =0x7714
0x3c0d7d49afeeb61e6627b56f24c7ed8fmod 0x100bd =0xf02f.

Алынған нәтижелерді біріктіріп жалғай отырып, ModuleP түрлендіруінің нәтижесін: K = 0x0ccf5463b236a02f338fd5c27714f02f.

Осымен бірінші раундтың түрлендірулері аяқталды. Ішкі раундтық кілтті алу үшін осындай айналымды тоғыз рет қайталау қажет. Әрбір айналым төрт операциядан тұрады: S-блок арқылы ауыстыру, Mixer1, Mixer2 және ModuleP түрлендірулері. Екінші айналымнан оныншы айналымға дейінгі түрлендірулер нәтижесі кесте 2.8 - де келтірілген.

Кесте 2.8 – Бірінші раундтық ішкі кілтті жасау (2-10 айналымдар)

Операциялар	Операция нәтижелері
1	2
S-box:	K = 0x278ab713d1708dacc7d7569fab7bc5ac
Mixer1:	K = 0x4556abb68b639c31291df9686a4c5f0a
Mixer2:	K = 0x7c6df73a33debe79be69fa62595adff8
Module P 1	K = 0x4f83630ba1bb2d74bab883a5adc0532f
S-box:	K = 0xfd151351c4ccf1f2e083150a0fe1ebac
Mixer1:	K = 0xc0d25912c2cf258251a91d46b04504f9
Mixer2:	K = 0x23520af35fb22e94774387edd0e2e37d
Module P 2	K = 0x2dffe89209aae42b90f41363fc93e67e
S-box	K = 0xf16c86a72cf053a6a5b6ce13f4fa6e16
Mixer1	K = 0x6a1c018c4b86aa99781ebcd6b60c1576
Mixer2	K = 0x5c4c8ca13c893231108af639d3a61920
Module P 3	K = 0x9f42de86ee6a1de488ff40fc656ea045
S-box:	K = 0x62300b8f2a73cb534a6c88f42fd88d35
Mixer1	K = 0xae8d49496c800acedbf649e105e7eb0b
Mixer2	K = 0x807c17bc323f3a64b0a07f48d23bc9e9
Module P 4	K = 0xc46a252a7f8eda5748168ee6a3c4f418
S-box	K = 0x4c73df900e7563afb484756e264cb6bf
Mixer1	K = 0x921343d9d0c4d3c44224241934b86d6c
Mixer2	K = 0x4a3278ae78d112bf629d456ff0dbf334
Module P 5	K = 0x18a1f69e77e8cb29739948b9f636282b
S-box	K = 0xbfc437d2ab86c157239db418377091a6
Mixer1	K = 0x53387b197fdf1e38513ee928c4573de7
Mixer2	K = 0x8190f1574c00ce5a5019f844d04a43a9
Module P 6	K = 0xbe2e0f5e481937a89d323c37ee83904c
S-box	K = 0xd541c2a9b4e6e9f758cb0e92a15a560
Mixer1	K = 0x5aee23ff876a06fb467b5b72b8c800e9
Mixer2	K = 0xeb45c1be3db33e4590ef823d3e0770db
Module P 7	K = 0xdc5165e3ecfbc1e18157f4b19704365f
S-box	K = 0x93c62f5b81d664c81afb6a361c370f3
Mixer1	K = 0xa93038ee15034e5b8b59027f760eb9b9

2.8-кестенің жалғасы

1	2
Mixer2	K = 0xccca3a6555d21719fd3d640d52286c23
Module P 8	K = 0x8406a687d232289ad32fefec9d0360a
S-box	K = 0xe7b1d907008c91322d8cf8f808e87009
Mixer1	K = 0x66444f8a336870de307a346b1cb1d23a
Mixer2	K = 0x3f342d47be4ec33d5f8367efad24ff02
Module P 9	K = 0x270d01b891fbd94d7433cad069ba874d

Бірінші раундтық ішкі кілттің соңғы нәтижесін алу үшін, кесте 2.8 - дегі нәтижені негізгі кілтке модуль 2 бойынша қосу керек K = =0x904b9e1bd6eaa64db9a9c168a5e5f92d. Нәтижесінде:

K_2=0x270d01b891fbd94d7433cad069ba874d⊕

⊕0x904b9e1bd6eaa64db9a9c168a5e5f92d=

= 0xb7469fa347117f00cd9a0bb8cc5f7e60

Осылай жұмысты жалғастыру арқылы кесте 2.9 - да көрсетілгендей қажетті раундтық кілттерді алатын боламыз.

Кесте 2.9 – Алынған рандтық кілттер

Раундтық кілттің нөмері	Кілттің мәні
K_1	0xb7469fa347117f00cd9a0bb8cc5f7e60
K_2	0x756012f7ba128da67efba084be1b78cf
K_3	0x411ded5410a74f8489d5a891ff54f91f
K_4	0x7a574831d813d72a3360f34b30b9dc06
K_5	0xe808de737d501937f397539aa152bbdf
K_6	0xbf674c851c0bc3ddf5c043d0ca87b4b
K_7	0x72a34ba12595f9cd1b629a163e546836
K_8	0x3d173b7e2961a26293b178bd70c77460

Раундтық ішкі кілттер дайындалғаннан кейін мәтінді шифрлауға кірісуге болады. Шифрлауға арналған ашық мәтін ретінде Text= =0x81754b8c671be306adee86fc52174dcd мәні таңдалды. Раундтың бірінші түрлендіруі болып бастапқы негізгі кілтпен ашық деректі XOR бойынша қосу:

Text=0x81754b8c671be306adee86fc52174dcd⊕

⊕0x904b9e1bd6eaa64db9a9c168a5e5f92d=

= 0x113ed597b1f1454b14474794f7f2b4e0.

Келесі түрлендіру S-блоктың (кесте 2.1) көмегімен байтты ауыстыру болып табылады. Нәтижесінде мәліметтер блогы келесідей болады:

Text = 0x733b601cfeb67f951ac8c8339e5e41f1.

Раундтық кілттерді алу алгоритмі мен шифрлау алгоритмі ұқсас болғандықтан жоғарыда істелген әрекеттерді қайталайтын боламыз. Яғни, келесі кезекте Mixer1 көмегімен түрлендіретін боламыз. Оны орындау үшін тағы да

Text = 0x733b601cfeb67f951ac8c8339e5e41f1 мәнін кесте 2.10 - де көрсетілгендей квадраттық матрица түрінде жазып аламыз. Mixer1 түрлендіруінің нәтижесі кесте 2.11 - де берілген.

Кесте 2.10 – Text мәнінің матрицалық түрдегі көрінісі

Text мәндері			
73	3b	60	1c
fe	b6	7f	95
1a	c8	c8	33
9e	5e	41	f1

Кесте 2.11 – Mixer1 түрлендіруінің нәтижесі

Text-тің Mixer1 түрлендіруінен кейінгі нәтижесі			
9e	fa	2d	e9
4e	d8	56	3f
b4	d0	8f	b9
29	17	e8	d5

Егер кесте 2.11 - дің нәтижесін бір жолға жазатын болсақ, Mixer1 нәтижесі келесідей болады: Text = 0x9efa2de94ed8563fb4d08fb92917e8d5.

Ары қарай Mixer2 операциясын қолдану арқылы келесі нәтижелерге қол жеткіземіз:

(0x9efa2de9*0xa822bbba) mod 0x100000137 =0xe5d8e3a5
(0x4ed8563f*0xd235d265) mod 0x100000137 =0x5c899c10
(0xb4d08fb9*0xda1996d2)mod 0x100000137=0x1a5323de
(0x2917e8d5*0x904b9e1b) mod 0x100000137 = 0xad001adf

Түрлендірулердің нәтижелерін бір бөлікке жинау арқылы Mixer2-ден келесі нәтижені аламыз: Text = 0xe5d8e3a55c899c101a5323dead001adf. Осымен шифрлаудағы бірінші раундтағы түрлендірулер аяқталады.

Ұзындығы 128 бит болатын блокты шифрлау үшін сегіз раундтық шифрлау қолданылады, яғни осы үдерісті қосымша жеті рет қайталайтын боламыз. Екінші раундтан бастап сегізінші раундтқа дейінгі аралық мәндер кесте 2.12 - де берілген. Шифрлау нәтижесінде пайда болған шифрмәтінді көруге болады: Cipher = 0x2040844e82689d9279fd3bce5c67541.

Кесте 2.12 – 2-8 раундтардағы шифрлау нәтижелері

Раунд нөмірі	Операция	Операция нәтижесі
1	2	3
2	XOR K_1	Text= 0x529e7c061b98e310d7c92866615f64bf
	S-box	Text = 0x2357b0213f70f72962b32078610caacb
	Mixer1	Text= 0xa12aa923704d5026e900384f2586718d
	Mixer2	Text = 0xa16ce2f0d6b462f612ab4b6ab5d8f998

2.12-кестенің жалғасы

1	2	3
3	XOR $К_2$	Text = 0xd40cf0076ca6ef506c50ebee0bc38157
	S-box	Text= 0x9dbd276b50f81bfa50fa6c3999424c2a
	Mixer1	Text= 0x5c94ad2cd646e49313a0a866d6f1fac8
	Mixer2	Text = 0x95d27d43bf26a0c7b4ea6b2b072cd3e
4	XOR $К_3$	Text = 0xd4cf9017af81ef43829b0e234f263421
	S-box	Text= 0x9dde63e4dc4c1bce40c698077b2fd310
	Mixer1	Text = 0x5864b12c9a5866fded0fff82341fe9c9
	Mixer2	Text = 0x8f572d43da96805539997736ae66389d
5	XOR $К_4$	Text = 0xf50065720285577faf9847d9edfe49b
	S-box	Text = 0x6dc9f6bbf09f2a476e129b1f57d737c6
	Mixer1	Text= 0xe869549b6c84bff1edcbad082251f2e7
	Mixer2	Text= 0x8df913275ec121504f553da5389c1cc2
6	XOR $К_5$	Text= 0x65f1cd5423913867bcc26e3f99cea71d
	S-box	Text= 0xf6b67cd90724b984ef5adffc696f85f9
	Mixer1	Text= 0x1f843d3093547b5a41d7adab55a39952
	Mixer2	Text= 0x9cb06a2ac7ea32eb92d5498464e0a5c4
7	XOR $К_6$	Text = 0x97461ee2962a8ed64d894db96848de8f
	S-box	Text= 0x1cae2e77e7668ec035af3571c49c6516
	Mixer1	Text = 0x7fc424f63395595b34224766fc5f56be
	Mixer2	Text = 0xca6fc89095a99a73fc4df640e48cf1b3
8	XOR $К_7$	Text= 0xb8cc8331b03c63bee72f6c56dad89985
	S-box	Text = 0x48112ce00bfdfb2b67c15074ed05699f
	Mixer1	Text= 0xab0dc1615b855ec661a3579da7d4e01e
	Mixer2	Text = 0x3f13333ac1472bbbb42eab0195010121
	XOR $К_8$	Text = 0x2040844e82689d9279fd3bce5c67541

3 ҚҰРЫЛҒАН СИММЕТРИЯЛЫ БЛОКТЫҚ ШИФРЛАУ АЛГОРИТМІНІҢ СЕНІМДІЛІГІН ЗЕРТТЕУ

Кез-келген шифрлау алгоритмге сенімді болу үшін жан-жақты зерттеу жұмыстарын жүргізуді талап етеді. Бірінші кезекте құрылған шифр алгоритмді пайдаланып ашық мәтіндерді шифрмәтінге түрлендіреміз. Алынған шифрмәтіндерді белгілі статистикалық тесттер арқылы кездейсоқ тізбектердің сипаттарына тексереміз. Неғұрлым көбірек сынақ жұмыстарын жүргізсек, нәтиже соғұрлым дәлірек болады. Келесі кезекте ашық мәтіннің немесе кілттің элементтеріне аз ғана өзгерістер енгізіп алынған шифрмәтіндердің мәндеріндегі өзгерістер зерттеледі және бұл лавиндік әсері деп аталады. Лавиндік әсерде шифрлау үшін маңызды криптографиялық қасиет болып табылады [38, с. 118].

Алгоритмдердің беріктігін талдаудың негізгі әдістеріне толық теру әдістері, дифференциалдық, сызықтық және алгебралық криптоталдау әдістері жатады. Сонымен қатар, әртүрлі криптошабуылдар жүргізу арқылы тексереді. Толық теру әдісінде, кілттердің барлық мүмкін жағдайларын шифрмәтінді кері ашуға пайдаланып, нәтижесінде ашық мәтін алынғанын тексеріп көруден тұрады. Статистикалық әдістерде талдау барысында жалған кілтке қарағанда дұрыс кілттер үшін жиірек орындалатын алгоритмнің кейбір статистикалық тәуелділігі пайдаланылады. Ал алгебралық әдістердің негізі болып, айнымалылар ретінде ашық мәтін мен кілттердің элементтері таңдалып сызықтық теңдеулер жүйесін құру болып табылады. Ал, сызықтық емес теңдеулер жүйесін шешуде сызықтандыру жүргізу әдісін қолданып, кілттің элементтерін табу мүмкіндігі қарастырылады.

3.1 Шифрмәтіндердің статистикалық қауіпсіздігін бағалау

Криптография мен кездейсоқтық тығыз байланысты. Оның бір көрінісі ретінде, шифрмәтіннің статистикалық қасиеттері кездейсоқ үдерістердің қасиеттеріне белгілі бір статистикалық критерийлер бойынша «жақын» болуы керек екендігін айтқан жөн.

Шифрлау алгоритмдерін құру жұмыстарын жүргізу барысында статистикалық тестілеу немесе сынақтан өткізу деп аталатын, олардың криптографиялық қасиеттерін талдау есебі туындайды. Блоктық шифрларды талдауда автоматтандыруға болатын жалғыз кезең көрінуі мүмкін. Бірақ, бұл жерде де статистикалық тестілеу әдістемесін қажет етеді. Сонымен қатар, әртүрлі шифрлардың сынақ нәтижелерін салыстыру үшін стандартталған әдістеме болған жөн.

Тізбектің статистикалық қасиеттерін тексеру тәжірибеде жиі кездесетін міндет. Бұл міндет криптография саласында ерекше маңызға ие. Оны іс жүзінде шешу үшін статистикалық тесттер жиынтығы қолданылады. Мұндай жиынтықтардың алғашқыларының бірін 1969 жылы Д.Кнут ұсынды [63 - 65].

Әрбір статистикалық тест – кіріс тізбегінің кездейсоқтығы туралы гипотезаны тексеру мәселесін шешеді. Сонымен бірге, жалған теріс нәтиженің ықтималдығы, яғни мәндік деңгейі белгіленеді. Әрбір статистикалық тест

деректерге сүйене отырып, тексеріліп отырған тізбекті α ықтималдықпен кездейсоқ тізбекпен салыстыратын P-мәні (P-value) деп аталатын параметр есептеледі. Егер $\mathrm{P-value} \geq \alpha$ болса кездейсоқтық туралы гипотеза қабылданады, ал кері жағдайда ол қабылданбайды.

Қазіргі уақытта екілік (0 мен 1 ден тұратын) тізбекті статистикалық тексеруге бағытталған және бағдарламалық қамтамасыз ету түрінде іске асырылған ең танымал тесттер жинақтары ретінде келесілерді атауға болады:

− DieHard – АҚШ-тың математигі Дж. Марсальей әзірлеген статистикалық тесттер жинағы. Сонымен бірге бұл тесттердің DieHarder іске асырған кеңейтілген нұсқасыда ашық жарияланымдарда бар;

− NIST статистикалық тесттер жинағы;

− TestU01 – Монреаль университеті әзірлеген статистикалық тесттер жинағы;

− RaBiGeTe – Windows үшін ыңғайлы GUI-мен жасақталған, Италияда әзірленген статистикалық тесттер жинағы;

− PractRand АҚШ-тық Доти-Хамфри әзірлеген статистикалық тесттер жинағы.

Аталған статистикалық тесттер жинақтарынан басқада қазіргі уақытта әртүрлі жағдайларға байланысты көп қолданыла бермейтін тесттер жинақтары бар, мысалы Crypt-X, Ent және т.б. Көптеген тесттердің өздеріне тән кемшіліктері бар. Мысалы, кейбір тесттердің жан-жақты сиппатамасы және әдістемелік түсіндірімдемелері жоқ.

ПКТ тесттердің қажетті және жеткілікті жиынтығын анықтау шешілмеген мәселе болып табылады [66]. Тізбекті тексеру кезінде барлық мүмкін болатын заңдылықтарды іздей отырып, мүмкіндігінше белгілі статистикалық тесттерді қолдану қажет. Егер қолданылған статистикалық критерийлердің ешқайсысы жарамсыз деп таппаса, зерттеп отырған шифрлау алгоритмін немесе тізбекті генерациялау алгоритмін пайдалануға болады. Қандай да бір ықтималдықпен бір немесе басқа тұжырым қате болып шығуы мүмкін, әдетте неғұрлым таңдама үлкен болған сайын қателік ықтималдығы соғұрлым аз болады. Алайда, жиі қолданылатын таңдамалар салыстырмалы түрде үлкен емес болса, бұл жағдайда қателік ықтималдығы айтарлықтай болуы мүмкін.

Статистикалық бағалау жүргізу барысында жоғарыда аталған P-мәні деп аталатын қателік ықтималдығы есептеледі. Басқаша айтқанда, бұл бірінші түрдегі статистикалық қателікке жол берудің ықтималдығы, яғни шындығында кездейсоқ тізбекті, ауытқуы ескерілетіндей деп саналатын қателік. Әдетте 0,05, 0,01 және 0,001 мәндері қолданылады. Мысалы, 0,05 мәні 5 пайыздан аспайтын қателікке жол берілетіндігін білдіреді.

Кездейсоқ және псевдокездейсоқ тізбектер сапасын бағалау әдістерін екі топқа бөлуге болады: графикалық және бағалау.

Графикалық тесттерде тізбектердің қасиеттері графикалық тәуелділіктер түрінде көрсетіледі, олардың түріне қарап зерттелетін тізбектің қасиеттері туралы қорытындылар жасалады. Бұл санатқа келесі сынақтарды жатқызуға

болады: элементтердің үлестірім гистограммасы, жазықтықта үлестірілуі, монотондылыққа тексеру және т.б.

Бағалау тестілерінің ішінде келесілер қолданылды: 0 мен 1-ді тексеру; байланыстырылмаған сериялады тексеру; символ бойынша тексеру; интервалдарды тексеру; комбинацияларды тексеру; купондар жинаушы тесті; алмастыруларды тексеру; монотондылықты тексеру; корреляцияны тексеру; сызықтық күрделілікті тексеру; спектрлік тест; қиылысатын ауыстыруларды тексеру. Тест нәтижелерін талдау кезінде хи-квадрат критерийі қолданылды.

Құрылған шифрлау алгоритмін статистикалық қауіпсіздігін тексерудегі бірінші мақсатымыз, осы алгоритм бойынша алынған шифрмәтіндердің кездейсоқ тізбекке жақындығын тексеру. Бұл жұмыстар жоғарыда аталған бағалау және графикалық тесттерді қолдану арқылы бағаланды.

Әрбір шифрлау алгоритмдері (Qamal және Qamal NPNS) үшін графикалық және бағалау тестілерінен 20 түрлі форматтағы 1000 шифрланған файл (әр форматтағы 50 файл) үшін өткізілді (кесте 3.1).

Кесте 3.1 – Статистикалық талдауға арналған бастапқы файлдар

Файл	Типі	Графиалық тест	Бағалау тесті
.docx	Microsoft Word	50	50
.xls	Microsoft Excel	50	50
.pptx	Microsoft PowerPoint	50	50
.pdf	HTML	50	50
.rar	Архивтелген файл	50	50
.zip	Архивтелген ZIP-папкасы	50	50
.jpg	Jpg форматтағы сурет	50	50
.png	Png форматтағы сурет	50	50
.txt	Текстік	50	50
.gif	Gif форматтағы сурет	50	50
.html	HTML Document	50	50
.cat	Қауіпсіздік каталогы	50	50
.mp4	Бейнефайл	50	50
.wmz	Windows Media	50	50
.dll	Кеңейту қосымшасы	50	50
.log	Файлды тіркеу, журнал	50	50
.lex	Dictionary File	50	50
.djvu	Файл djvu	50	50
.xml	XML Document	50	50
.mp3	Дыбыстық файл	50	50

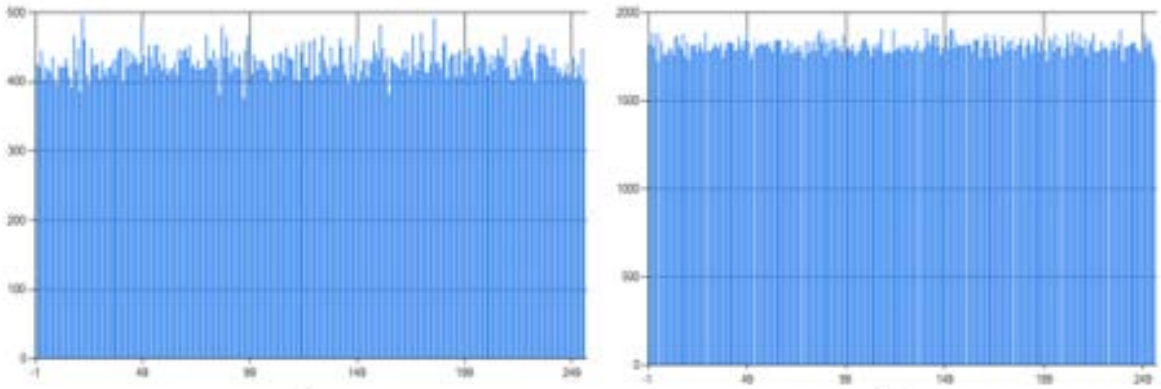
χ^2 критерийінің көмегімен сенімділік интервалын 0,05 деп алып шифр мәтіндерге статистикалық бағалау жүргізілді. Еркіндік дәрежесі 255 болған жағдайда χ^2 - тың критикалық мәні $\chi^2_{0,05;255} = 293,2478$. Кесте 3.2 - де зерттеу

нәтижелері көрсетілген. Бұл кестеде әрбір тесттер үшін сәтті өткен файлдар саны берілген.

Зерттеліп отырған алгоритмдерге жүргізілген статистикалық зерттеулер, осы алгоритмдер бойынша алынған шифрмәтіндердің статистикалық қасиеттері талап етілген деңгейге сай екендігін көрсетті. Графикалық тесттердің нәтижелері сурет 3.1-3.8-дерде көрсетілген.

Кесте 3.2 – Шифрмәтіндерді статистикалық зерттеу нәтижелері

Типі	Тесттер	Qamal	Qamal NPNS
Графикалық тесттер	1. Элементтердің үлестірім гистограммасы	995	990
	2.Жазықтықта үлестірілуі	1000	1000
	3.Сериялады тексеру	968	980
	4.Монотондылыққа тексеру	980	965
	5.Байттық автокорреляциялық функция (АКФ)	969	955
	6. Биттік автокорреляциялық функция	972	970
	7.Графикалық спектральдық тест	999	976
	8. Сызықтық күрделілік профилі	1000	1000
Бағалау тесттер	1. 0 мен 1-ді тексеру	1000	1000
	2. Байланыстырылмаған сериялады тексеру	998	991
	3. Символ бойынша тексеру	987	983
	4. Интервалдарды тексеру	983	988
	5. Комбинацияларды тексеру	970	975
	6. Купондар жинаушы тесті	956	968
	7. Алмастыруларды тексеру	960	971
	8. Монотондылықты тексеру	959	962
	9. Корреляцияны тексеру	980	962
	10. Сызықтық күрделілікті тексеру	1000	1000
	11.Спектрлік тест	998	1000
	12. Қиылысатын ауыстыруларды тексеру	1000	995

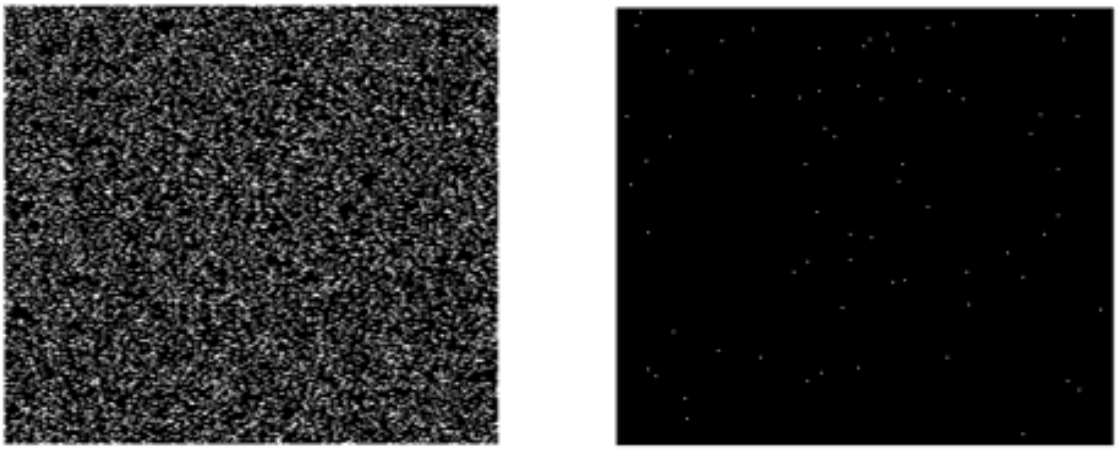


а)

ә)

а – «Qamal» алгоритмі үшін; ә – «Qamal NPNS» алгоритмі үшін

Сурет 3.1 – Элементтердің үлестірім гистограммасы тесті нәтижесі

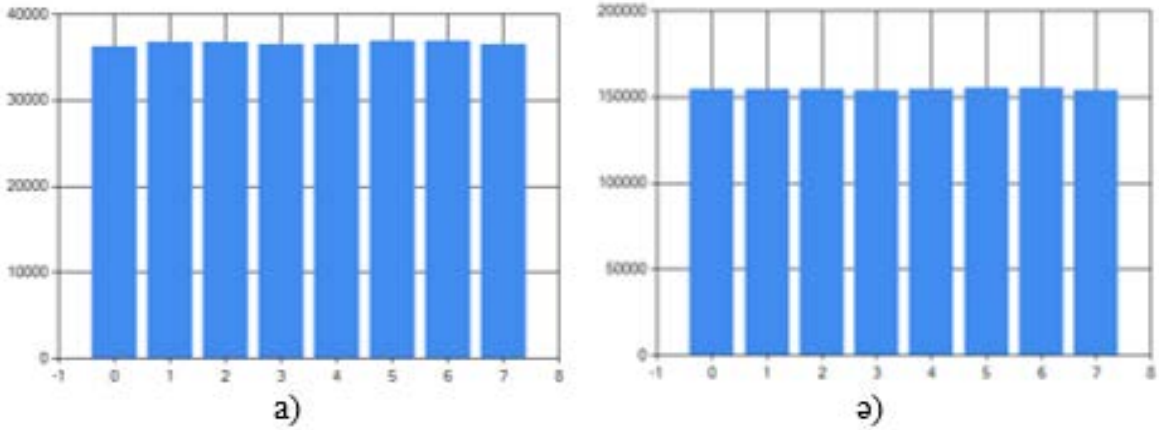


а)

ә)

а – «Qamal» алгоритмі үшін; ә – «Qamal NPNS» алгоритмі үшін

Сурет 3.2 – Жазықтықта үлестірім тесті нәтижесі

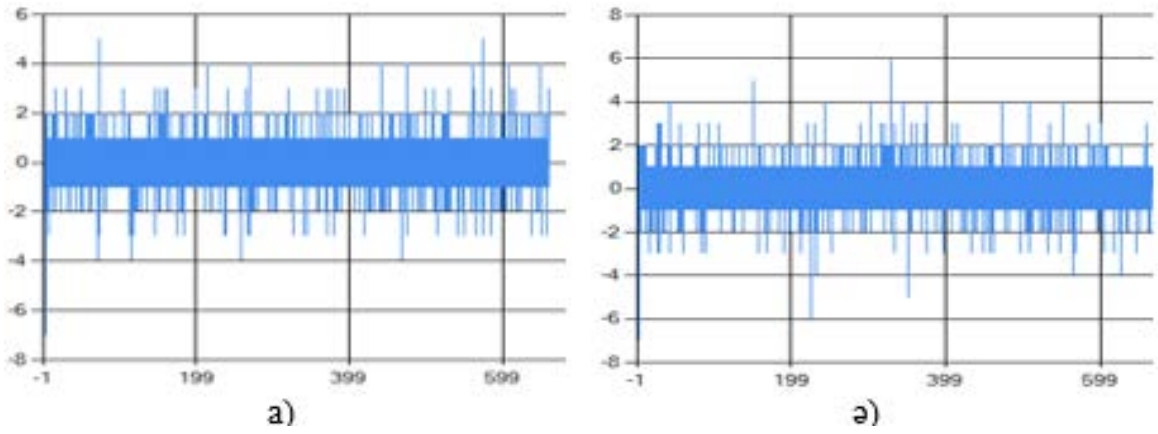


а)

ә)

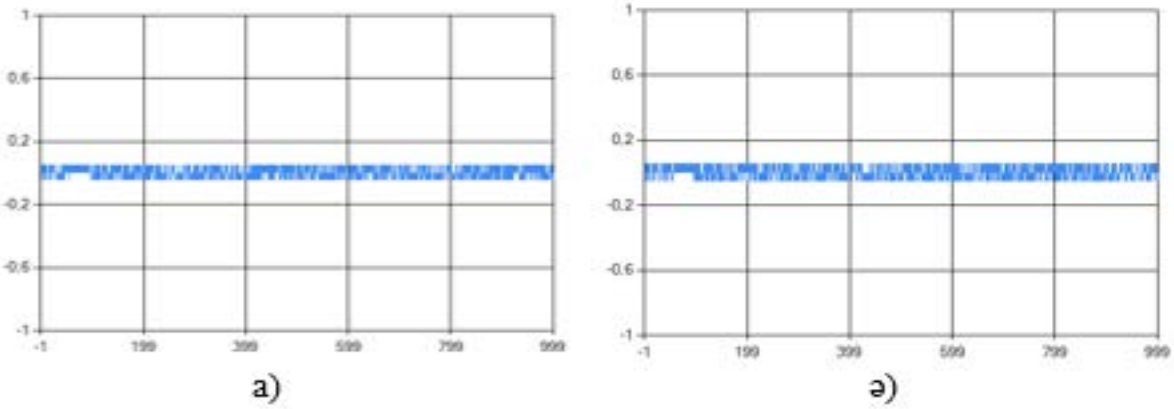
а – «Qamal» алгоритмі үшін; ә – «Qamal NPNS» алгоритмі үшін

Сурет 3.3 – Сериялардың тексеру тесті нәтижесі



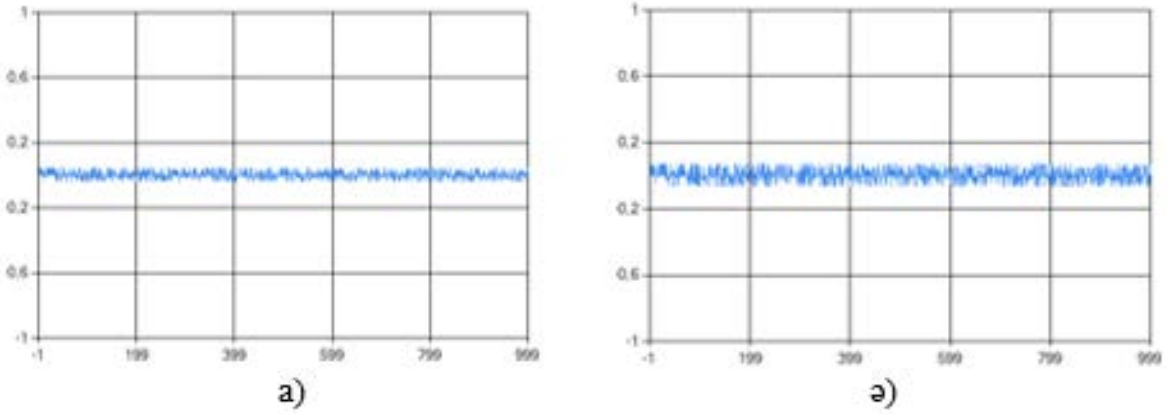
а – «Qamal» алгоритмі үшін; ә – «Qamal NPNS» алгоритмі үшін

Сурет 3.4 – Монотондылыққа тексеру тесті нәтижесі



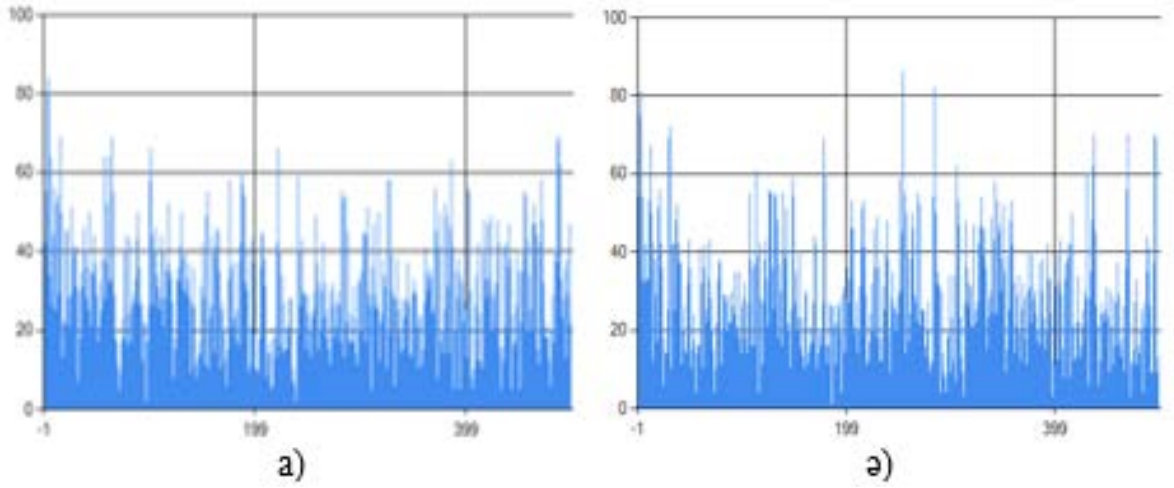
а – «Qamal» алгоритмі үшін; ә – «Qamal NPNS» алгоритмі үшін

Сурет 3.5 – Биттік автокорреляциялық функция тесті нәтижесі



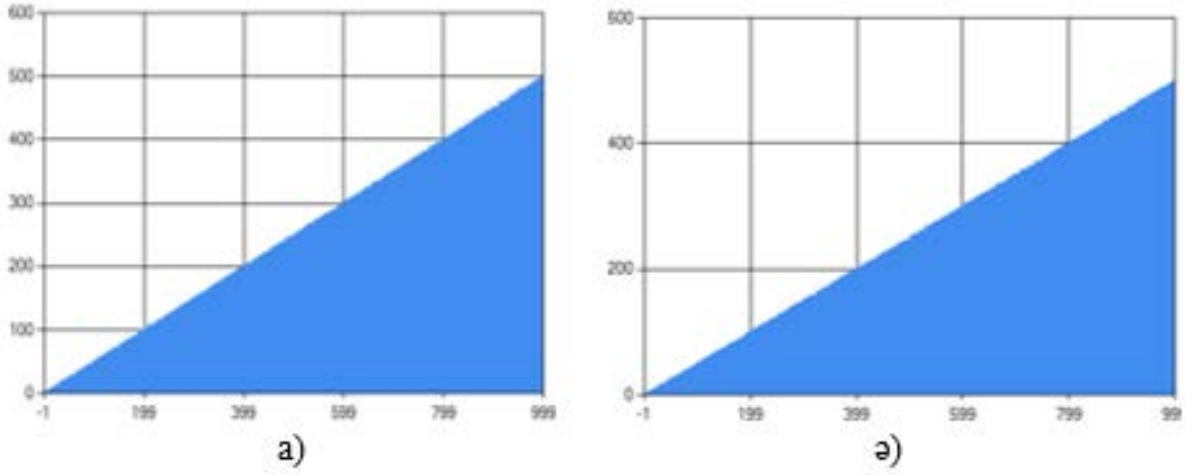
а – «Qamal» алгоритмі үшін; ә – «Qamal NPNS» алгоритмі үшін

Сурет 3.6 – Символдық автокорреляциялық функция тесті нәтижесі



а – «Qamal» алгоритмі үшін; ә – «Qamal NPNS» алгоритмі үшін

Сурет 3.7 – Графикалық спектральдық тест нәтижесі



а – «Qamal» алгоритмі үшін; ә – «Qamal NPNS» алгоритмі үшін

Сурет 3.8 – Сызықтық күрделілік профиль тесті нәтижесі

Критоалгоритмде шифрлауға пайдаланылатын кілттер тізбегі псевдокездейсоқ тізбек болуы қажеттігіне байланысты, раундтық кілттерді құру алгоритмін ПКТ алу генераторы ретінде қарастырылуы керек. ПКТ генераторлары бірнеше шартты қанағаттандыруы қажет. Осы генератор арқылы алынатын тізбек үлестірімі бірқалыпты үлестірімге жақын, яғни генерацияланған екілік тізбекте нөл немесе бірлердің саны жалпы санның жартысына жақын болуы шарт [67, 68]. Бұдан басқа, алынған тізбекті құрайтын кездейсоқ мәндер статистикалық тәуелсіз болуы қажет. Бұл жекелеген биттер арасында да, биттер топтары арасында да ешқандай байланыс болмауы керек дегенді білдіреді. ПКТ генераторын сынау кезінде белгілі бір үлкен ұзындықтағы тізбектер тексеріледі (мысалы, NIST тесттер жиынында 1 000 000 бит ұзындықтағы тізбекті тексеруді ұсынады). Егер тестілеуден өткен тізбектің үлесі

жеткілікті үлкен болса және P-мәні бірқалыпты үлестірілген болса онда генератор осы сынақтан өтеді деген қорытынды жасалады.

Құрылған шифрлау алгоритмінің раундтық кілттер тізбегіне статистикалық зерттеулер жүргізілді. Сынаққа саны 10 000, 20 000, 50 000, 100 000, 200 000 500 000, 1 000 000, 1 200 000 болатын әртүрлі кілттер топтары қатысты (әрбір кілттің ұзындығы 128 бит). Зерттеу нәтижесі – кілттер тізбегі жоғарыдағы 12 статистикалық бағалау тесттерінің бәрінен сәтті өтетіндігін көрсетті. Бұл кілттер тізбегінің элементтерінің арасында статистикалық байланыс жоқ екендігін білдіреді.

СТ ҚР 1073-2007 стандартында АККҚ ең жоғарғы төртінші деңгейіне қойылатын талаптардың ішінде, кілттерді құруда кілттің әрбір битінің 0 мен 1-ді қабылдау ықтималдығы (0,5 ± 0,001) интервалында жатуын қамтамасыз етуі керек. Ұзындықтары 128 биттен тұратын 10 000 кілт тобының 0 мен 1-ді қабылдау ықтималдығын тексергенде 0,5-тен ауытқуы 0,0003-ті көрсетті. Зерттеуімізді одан да нақтылай түсу үшін, шифрлау алгоритмінде қолданылатын кілттің ұзындығы 128 болғандықтан, 128 позицияның әрқайсысына есептеулер жүргізілді. 10 000 кілтті тексергенде қауіпсіздіктің бірінші деңгейіне қойылытын талаптан 100% барлығы, ал екіншіден, үшіншіден және төртінші деңгейлерден сәйкесінше 97%, 45% және 18% өтті. Басқа кілттер үшін аталған стандартқа сәйкес қауіпсіздік деңгейінің талаптарынан өту көрсеткіштері кесте 3.3 - те көрсетілген.

Кесте 3.3 – Кілттер тізбегінің қауіпсіздік деңгейі талаптарынан өту көрсеткіштері

Қауіп. деңгейі/ кілт саны, мың	10	20	50	100	200	500	1000	1200
1	100%	100%	100%	100%	100%	100%	100%	100%
2	97%	100%	100%	100%	100%	100%	100%	100%
3	45%	61%	78%	93%	99%	100%	100%	100%
4	18%	21%	27%	49%	81%	87%	98%	100%

3.2 Әзірленген шифрлау алгоритмінің лавиндік әсерін эксперименттік зерттеу

Лавиндік әсер ұғымы криптографияда блоктық шифрлар мен криптографиялық хэш функцияларды зерттеуде қолданылады және маңыздылығы жағынан басқа да жүргізілетін талдаулардан кем түспейді. Егер алгоритмде қажетті деңгейде лавиндік әсері болмаса, онда криптоталдаушылар алгоритмге кіретін ашық деректерге қарап шифрлау нәтижесінде шығатын мәліметтер туралы болжам немесе қорытынды шығара алады. Түрлендірулер жүргізу кезіндегі лавиндік әсер ету дәрежесін сипаттау үшін лавиндік параметр анықталады. Ол кіріс биттер тізбегінің шығыс биттер тізбегіне өзгеру ықтималдығының 0,5 мәнінен ауытқуы [38, с. 119; 69, 70, 71].

Ашық мәтінге енгізілген азғана өзгерістер алынған шифр мәтінде де аз ғана өзгерістерге алып келсе, бұл шабуылдаушыға кілт кеңістігін немесе ашық мәтінді іздеу аймағын тарылтуға мүмкіндік береді. Лавиндік әсер критерийін ашық мәтін бойынша ғана емес, осыған ұқсас әдіспен кілт бойынша бағалау үшін анықтауға болады, яғни кілттер тізбегінен бір бит өзгерткенде, орта есеппен шыққан биттердің жартысы өзгеруі керек.

Лавиндік критерий үшін лавиндік параметр келесі формуламен анықталады:

$$\varepsilon_i = |2k_i - 1| \tag{3.1}$$

мұндағы, i – кіріс мәндердегі өзгеретін биттің номері, k_i – кірістегі i-ші бит өзгерген жағдайдағы шығыс биттердің, бастапқы өзгеріссіз шифрланған мәтінмен салыстырғанда жартысының өзгеру ықтималдығы.

Qamal шифрлау алгоритмінің құрылымы кілтті модуль 2 бойынша қосу, *S*-блок ауыстыруы, Mixer1 және Mixer2 араластыру түрлендірулерінен тұратыны жоғарыда айтылды. Алгоритмде қолданылған осы түрлендірулердің лавиндік әсерге ықпалын мысал арқылы көрелік.

Кіріс дерегі ретінде бір-бірінен айырмашылығы тек бір битте болатын екі ашық мәтін алайық. Осы мәтіндерді шифрлауға бірдей кілтті қолданамыз. Енгізілген өзгерістің бір раунд ішінде қалай таралатынын келесі мысал арқылы анықтайық.

Мысал 3.1

Ашық мәтін 1 (T_1)	`00000000000000000000000000000000`
Ашық мәтін 2 (T_2)	`01000000000000000000000000000000`
Кілт	`CDBF03369EAD5EF3E98F2F2FDF8AB4B1`
$T_1 \oplus K$	`CDBF03369EAD5EF3E98F2F2FDF8AB4B1`
$T_2 \oplus K$	`CCBF03369EAD5EF3E98F2F2FDF8AB4B1`
$S(T_1 \oplus K)$	`7CCB187557220AF50D16C1C1D79241FE`
$S(T_2 \oplus K)$	`11CB187557220AF50D16C1C1D79241FE`
$M_1S(T_1 \oplus K)$	`EB1290D9211A4DE797980754B7952429`
$M_1S(T_2 \oplus K)$	`931290D9751A4DE7C19807544C952429`
$M_2M_1S(T_1 \oplus K)$	`B8558B3E22C350383BB3815B6EAAA2B1`
$M_2M_1S(T_2 \oplus K)$	`40F993A8163D55C08137E4C53C7B7606`

Алғашқы таңдалған екілік жүйедегі ашық мәтін тек нөлдерден тұрады. Екінші ашық мәтін де тек сегізінші биттен басқасы нөлден тұрады, яғни таңдалып отырған екі мәтіннің айырмашылығы тек бір ғана битте. Биттік қосу (xor) операциясы өзгерістердің таралуына әсер етпейді. S-блок ауыстыруында бір биттің өзгерісі тек бір байтқа, ал Mixer1 операциясында әрбір төртінші байтқа әсер етеді. Осы аталған операциялардан кейін Mixer2 операциясы орындалады, нәтижесінде өзгеріс бүкіл шифр мәтінге таралады. Нақты сандық сипаттамалар төменде көрсетілген.

Лавиндік әсерді одан ары қарай тексеру үшін ұзындығы 128 бит болатын кездейсоқ ашық мәтін таңдалды. Әрбір орындағы биттерді инверциялау, яғни 0

болса 1-ге, 1 болса 0-ге айналдыру арқылы жаңа 128 ашық мәтін алынды. Осы алынған ашық мәтіндердің бәрі шифрланды. Шифрмәтіндер мен ашық мәтін арасында әр раунд үшін k_i ықтималдығы анықталды. Кесте 3.4 - 3.5 - терде бірінші раундтан кейінгі және толық шифрлау алгоритмдер бойынша талдау нәтижелері келтірілген. Олардың әрқайсысы үшін ε орташа мәні сәйкесінше 0,07 және 0,062 құрайды. Лавиндік әсер параметрінің мәні неғұрлым кіші болған сайын, шифрлаудың лавиндік әсері соғұрлым күшті болады.

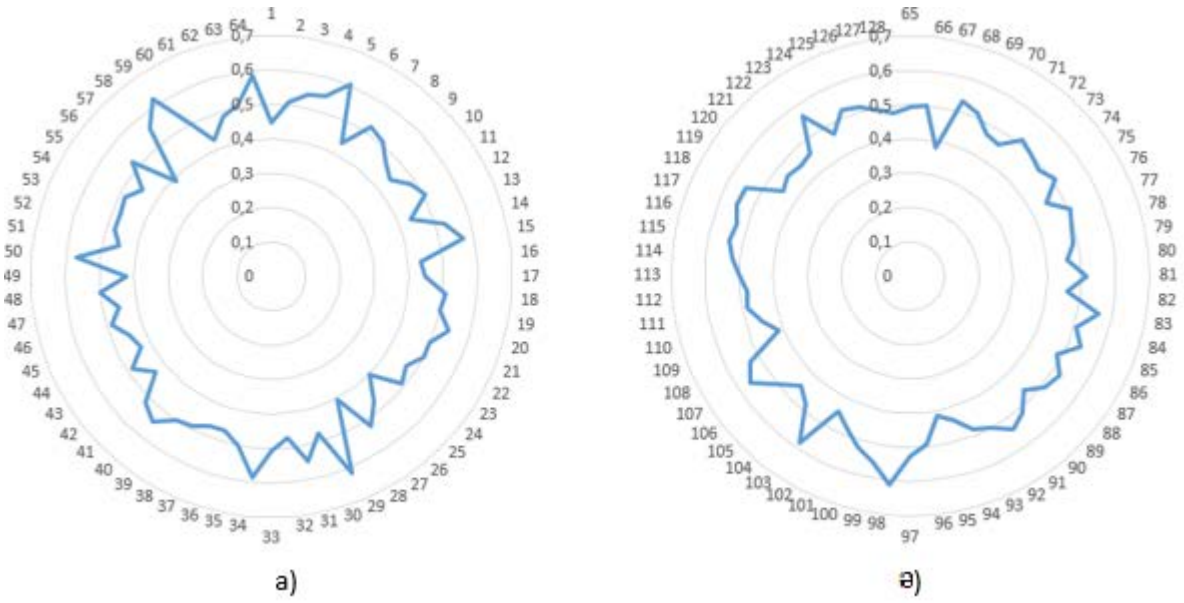
Кесте 3.4 – Бірінші раундтан кейінгі лавиндік әсер параметрінің мәндері

i	k_i	i	k_i	i	k_i	I	k_i	i	k_i	i	k_i	i	k_i	i	k_i
1	0,48	17	0,40	33	0,40	49	0,55	65	0,47	81	0,53	97	0,51	113	0,44
2	0,46	18	0,51	34	0,45	50	0,46	66	0,44	82	0,48	98	0,49	114	0,52
3	0,50	19	0,51	35	0,47	51	0,49	67	0,46	83	0,51	99	0,44	115	0,55
4	0,53	20	0,43	36	0,48	52	0,51	68	0,45	84	0,50	100	0,55	116	0,55
5	0,62	21	0,42	37	0,49	53	0,48	69	0,45	85	0,45	101	0,55	117	0,52
6	0,48	22	0,45	38	0,53	54	0,48	70	0,48	86	0,53	102	0,41	118	0,47
7	0,48	23	0,45	39	0,44	55	0,44	71	0,50	87	0,41	103	0,47	119	0,52
8	0,47	24	0,58	40	0,46	56	0,47	72	0,56	88	0,54	104	0,48	120	0,42
9	0,46	25	0,47	41	0,57	57	0,52	73	0,46	89	0,50	105	0,49	121	0,49
10	0,48	26	0,52	42	0,50	58	0,50	74	0,57	90	0,47	106	0,52	122	0,50
11	0,55	27	0,45	43	0,46	59	0,53	75	0,44	91	0,54	107	0,57	123	0,45
12	0,44	28	0,52	44	0,55	60	0,51	76	0,49	92	0,51	108	0,48	124	0,49
13	0,48	29	0,54	45	0,49	61	0,63	77	0,49	93	0,52	109	0,57	125	0,49
14	0,44	30	0,52	46	0,52	62	0,51	78	0,48	94	0,51	110	0,44	126	0,53
15	0,55	31	0,52	47	0,48	63	0,54	79	0,48	95	0,53	111	0,51	127	0,59
16	0,52	32	0,51	48	0,56	64	0,48	80	0,47	96	0,48	112	0,45	128	0,54

Кесте 3.5 – Сегізінші раундтан кейінгі лавиндік әсер параметрінің мәндері

i	ki	i	ki	i	ki	I	ki	i	ki	i	ki	i	ki	i	ki
1	0,48	17	0,47	33	0,50	49	0,52	65	0,46	81	0,45	97	0,52	113	0,54
2	0,52	18	0,49	34	0,54	50	0,52	66	0,50	82	0,52	98	0,54	114	0,57
3	0,43	19	0,49	35	0,48	51	0,48	67	0,51	83	0,55	99	0,46	115	0,58
4	0,48	20	0,53	36	0,52	52	0,49	68	0,51	84	0,53	100	0,41	116	0,50
5	0,44	21	0,56	37	0,44	53	0,50	69	0,42	85	0,52	101	0,54	117	0,50
6	0,48	22	0,48	38	0,49	54	0,54	70	0,46	86	0,45	102	0,46	118	0,45
7	0,48	23	0,51	39	0,50	55	0,48	71	0,50	87	0,56	103	0,51	119	0,55
8	0,50	24	0,50	40	0,56	56	0,47	72	0,41	88	0,53	104	0,52	120	0,58
9	0,49	25	0,55	41	0,48	57	0,47	73	0,52	89	0,52	105	0,56	121	0,45
10	0,48	26	0,51	42	0,48	58	0,48	74	0,50	90	0,46	106	0,55	122	0,46
11	0,52	27	0,50	43	0,48	59	0,47	75	0,45	91	0,54	107	0,51	123	0,43
12	0,45	28	0,49	44	0,55	60	0,55	76	0,41	92	0,52	108	0,45	124	0,48
13	0,52	29	0,50	45	0,49	61	0,49	77	0,41	93	0,48	109	0,48	125	0,49
14	0,52	30	0,43	46	0,45	62	0,49	78	0,54	94	0,48	110	0,50	126	0,45
15	0,50	31	0,45	47	0,48	63	0,48	79	0,55	95	0,56	111	0,52	127	0,60
16	0,50	32	0,54	48	0,54	64	0,47	80	0,52	96	0,47	112	0,53	128	0,49

Әрі қарай, алгоритмнің лавиндік әсерін практикалық бағалау үшін критерийді кілт үшін қолданамыз. Бұл жағдайда да алдыңғы ашық мәтінге енгізгендей өзгерісті кілтке енгізу арқылы жүзеге асырылды. Алынған нәтижелердің мәндері кесте 3.4 - 3.5 - тердегі мәндерге жақын. Талдау нәтижелері сурет 3.9 - да көрсетілген және осы суретте көрініп тұрғандай, k_i -дің мәндері (0.4; 0.6) аралығында жатыр.



а) і-дің мәні 1-ден 64-ке дейін; ә) і-дің мәні 65-тен 128-ге дейін

Сурет 3.9 – Кілт бойынша лавиндік әсерге талдау

Егер алгоритмнің кірісінде бір бит өзгергенде, шығыстағы биттердің жартысы өзгеретін болса, криптоалгоритм лавиндік критерийді қанағаттандырады. Qamal алгоритмі лавиндік әсер критерийінің шартын қанағаттандырады.

3.3 Кілттер кеңістігінің көлемін есептеу

Толық теру әдісі – шифрлау алгоритмдерінің беріктігін талдауға арналған әмбебап және басқа талдаулардың нәтижесін осы мәнмен салыстыратын әдіс. Ұзақ уақыттан бері пайдаланылып келе жатқанына қарамастан, қазіргі кезде компьютерлік технологияның қарқынды дамуына байланысты бұл әдіс кеңінен қолданылады.

ПЕПСЖ – де кілт бір-біріне тәуелсіз алынған екі бөліктен тұрады. Әрбір кілттердің ұзындықтары 128 биттен тұрады. Олардың бірі биттік қосу және позициялық емес шифрлау жүйесі үшін генерациялау арқылы алынған псевдокездейсоқ тізбек. Ал, таңдап алынған $p_1(x), p_2(x), \dots, p_S(x)$ полиномдық жұмыс негіздерінің жүйесі ПЕПСЖ-де кілттің екінші бөлігі болып табылады. Ұзындығы 128 бит болатын кілтті толық теру саны 2^{128}-ге тең екендігі белгілі. ПЕПСЖ-ға негізделген шифрлау алгоритмінің криптографиялық беріктігі бір-

бірінен өзгеше болатын толық кілттерді таңдауға арналған барлық мүмкін санымен анықталады. Берілген L биттік ұзындықтағы хабарламаны шифрлау кілттерінің толық теру саны келесі формуламен табылады [17, с. 126; 58, p. 198]:

$$Q_k = 2^L \cdot \sum_{k_1,\dots,k_S}(k_1 + \cdots + k_S)!\ C_{n_1}^{k_1} \dots C_{n_S}^{k_S} \qquad (3.2)$$

L –дің әрбір мәніне сәйкес Q_k -ның нақты мәнін анықтау үшін, L-ші дәрежеге дейінгі келтірілмейтін көпмүшеліктердің санын және L санының композициясын есептеу қажет.

Дәрежесі L–ге тең екілік коэффициентті келтірілмейтін көпмүшеліктердің саны келесі формула бойынша есептеледі [17, с. 126]:

$$I_L = \frac{1}{L}\sum_{d\backslash L}\mu(d)\,2^{L/d} = \frac{1}{n}\sum_{d\backslash L}\mu(L/d)\,2^d$$

мұндағы, d – L-дің бөлгіштері, $\mu(x)$ – келесідей анықталған Мёбиус функциясы:

$$\mu(x) := \begin{cases} 0, & \text{егер } x \text{ квадраттан таза болмаса} \\ (-1)^k, & \text{егер } x \text{ әртүрлі } k \text{ санның көбейтіндісі болса} \\ +1, & \text{егер } x = 1 \end{cases}$$

L -дің мәні 1 ден 32-ге дейін болғандағы I_L-дің мәндері кесте 3.6 - да берілген. Егер $L = 128$ болса, онда $I_L \approx 2^{122}$.

Кесте 3.6 – I_L -дің мәні 1 ден 32-ге дейінгі мәні

L-ші дәрежеге дейінгі келтірілмейтін көпмүшеліктердің саны							
L	I_L	L	I_L	L	I_L	L	I_L
1	2	9	56	17	7710	25	1342176
2	1	10	99	18	14532	26	2580795
3	2	11	186	19	27594	27	4971008
4	3	12	335	20	52377	28	9586395
5	6	13	630	21	99858	29	18512790
6	9	14	1161	22	190557	30	35790267
7	18	15	2182	23	364722	31	69273666
8	30	16	4080	24	698870	32	134215680

Жалпы жағдайда L саны үшін 2^{L-1} композиция бар екені сандар теориясынан белгілі. Олардың ішінен ұзындығы k-ға тең болатындарының нақты саны C_{L-1}^{k-1}.

Осы есептеулердің көмегімен L-дің әр түрлі мәндері үшін толық кілттердің жалпы санын есептедік. L - 16, 32 және 64 сандарын қабылдағанда, толық терулер

саны сәйкесінше 2^{34}, 2^{69} және 2^{138} болатынына көз жеткіздік. Осы есептеулерді ескере отырып L=128 мәнін қабылдаған жағдайда толық терулер мөлшері 2^{276} санына жақын екендігін болжауға болады.

3.4 Шифрлау алгоритміне дифференциалдық криптоталдау әдісін қолдану

Шифрлау алгоритмін дифференциалдық криптоталдауға кіріспес бұрын, алдымен шифрлауда қолданылатын әрбір операциялардың дифференциалдық қасиеттерін бөлек қарастырған жөн. Дифференциалдық критоталдау әдісінің толық сипаттамасын [72 - 75] әдебиеттерден табуға болады. Бұл әдісті қолданудың негізгі төрт кезеңі бар екенін атап кетейік.

1-кезең. Шифрлау алгоритмін құрайтын барлық түрлендірулердің дифференциалдық қасиеттерін талдау.

2-кезең. Дифференциалдың ең ықтимал мәнін іздеу, яғни табылуы ең ықтимал кіріс айырымы мен шығыс айырымдарының жұбы.

3-кезең. Мәтіндердің дұрыс жұптарын табу. Яғни, шифрлау алгоритміне кірісіндегі модуль 2 бойынша қосындысы кіріс айырымымен сәйкес келетін, ал шифрлау алгоритмінің шығысындағы мәндердің қосындысы шығыс айырымымен сәйкес келетін мәтіндерді табу.

4-кезең. Құпия кілттің биттерін анықтау үшін мәтіндердің дұрыс жұптарын талдау.

Дифференциалдық криптоталдаудың негізгі қиындығы дұрыс мәтін жұптарын табуда, ал ол кезегінде қарастыратын айырымдардың ықтималдықтарының мәніне тікелей байланысты. Сондықтан да, ықтималдығы ең үлкен айырымдарды тауып алу бірінші кезектегі жұмыс болып табылады. Оны білген жағдайда, шифрлау алгоритміне немесе оның қысқарған түріне жүргізілетін криптоталдаудың нәтижесін болжауға болады. Басқаша айтқанда, дифференциалдық криптоталдау қолдану мүмкін болатын шифрдың раундтар санын алдын-ала шамалап білуге болады [65, с. 179; 66, с. 35; 76].

Модуль 2 бойынша қосу операциясының дифференциалдық қасиеті. Дифференциалдық криптоталдауда түрлендірілетін мәтіндер жекелей қарастырылмайды, бірге қарастырылады. Мәтіндердің модуль 2 бойынша қосындыларының нәтижесі ретінденде анықталатын олардың айырымдары қарастырылады:

$$\Delta X = X_1 \oplus X_2.$$

Бұл жағдайда ΔX айырымының мәні бастапқы мәтіндер бірдей болатын позицияларда нөл болады және әртүрлі болатын позицияларда бір болады.

Құпия кілттерді қосу операциясы мәтіндердің айырымындағы өзгеріске әсер етпейтіні белгілі. Бұл шифрлау барысында бірдей құпия кілтті пайдаланамыз деп алуымызға байланысты. Осылайша, мәтіндер бірдей K кілтінің мәнімен қосылады, ал бұл өз кезегінде бір-біріне қосылып, нөлге тең мән құрайды:

$$\Delta X = X_1 \oplus K_I \oplus X_2 \oplus K_I = X_1 \oplus X_2.$$

S-блокты қолдана отырып, биттерді ауыстыру операциясының дифференциалдық қасиеттері. S-блок кірісі n-биттен және шығысы m-биттен тұратын функция.

S-блоктың сызықтық және дифференциалдық криптоталдау әдістеріне беріктілігі жоғары болуы келесі шарттарты қанағаттандыруы керек:

- сызықтық криптоталдау үшін берілген мөлшердегі S-блокқа сәйкес құрылатын матрицаның (кестенің) барлық элементтері толық мүмкін болатын мәндердің санының жартысы болатындай бірдей үлесте таратылуы тиіс.
- дифференциалдық криптоталдау үшін берілген мөлшердегі S-блокқа сәйкес құрылатын айырма матрицасының элементтері бірдей үлесте таралуы тиіс.

Зерттеліп отырған алгоритмнің S-блогы 8 битті басқа 8 битке өзгерткендіктен, кіріс айырымдарының диапазоны шығыс айырымдарының диапазонымен сәйкес келеді және 0-ден 255 аралығында болады. S-блоктың ΔC шығыс айырымдарының ΔA кіріс айырымдарының мәніне тәуелділік кестесі құрылды (өлшемі үлкен болуына байланысты, толық кестені келтірмедік) және келесідей қасиеттері анықталды:

1 қасиет. Түрлендірулердің шығыс мәндерінде $\Delta C = 0$ мәнін $\Delta A = 0$ болған кезде ғана алуға болады. Бұл жағдайда шығыс айырымдарының пайда болу ықтималдығы 1-ге тең болады.

2 қасиет. Құрылған дифференциалдық талдау кестесінде ықтималдықтың максималды мәні $6/256 = 3/128$ құрайды.

3 қасиет. Ауыстыру S-блогынан өткеннен кейін өзгеріссіз қалатын ΔA кіріс айырымдары бар. Мәндерін ондық санау жүйесінде жазсақ, бұларға ΔA = 2, 3, 4, 6, 15, 16, 17, 18 және басқалары жатады.

4 қасиет. Кіріс айырмасының $\Delta A = 254$ ($\Delta A = 0x fe$) мәні $p = 4/256 = 1/64$ ықтималдықпен $\Delta C = 128$ ($\Delta C = 0x80$) шығыс айырымының мәніне түрленеді.

Mixer1 түрлендіруінің дифференциалдық қасиеттері.

Mixer1 түрлендіруі сызықты түрлендіру болғанына қарамастан, модуль 256 бойынша қосу операциясы қолданылған кезде айырымдардың мәндерінің қалай өзгеретінін анықтау қажет. 2^n модулі бойынша қосу амалын орындау кезінде, кіріс айырымдарында бір ғана, ең жоғарғы разрядтағы битте нөлдік емес болған жағдайда ғана $p = 1$ ықтималдығымен өзгеріссіз қалатыны ғана белгілі. Сонымен, егер Mixer1 түрлендіруінде 0x80-ге тең айырмашылық мәні қолданылса, онда онымен қандай түрлендірулер жасасақта, ықтималдық әрқашан 1-ге тең болады. Mixer1 түрлендіруі бір бағанның төрт байтына тәуелді. Сондықтан шығыс мәндерінің қалай өзгеретінін қарастыру маңызды. Сонымен қатар, дифференциалдық криптоталдау тұрғысынан қарағанда белсенді байттарға әсер ететін нұсқаларға қызығушылық танытамыз. Mixer1 операциясында қосу 256 модулі бойынша орындалатындықтан, 0x80

айырымының мәні әрқашан өзгеріссіз қалады. Сонымен, 0x80 және 0x80 бірдей айырымдарды модуль 256 (0x100) бойынша қосу нөлге әкеледі. Осылайша, Mixer1 түрлендіруінің байт мәндері тек 0x00 немесе 0x80 болуы мүмкін, бағанын толтырудың 15 нұсқасын қарастыра аламыз. Осындай түрлендірудің мысалы кесте 3.7 - де көрсетілген.

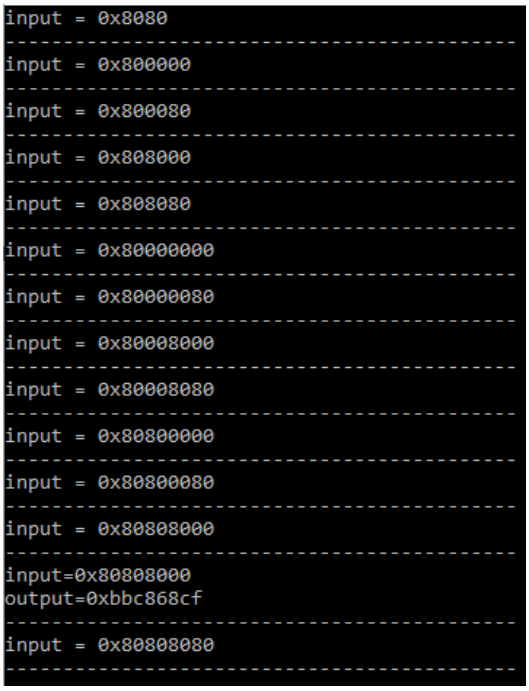
Кесте 3.7 – Mixer1 түрлендіруіндегі айырымдарды түрлендіру нәтижесі, 1 нұсқа

Бастапқы жағдайы	Бірінші өзгеріс	Екінші өзгеріс	Үшінші өзгеріс	Төртінші өзгеріс
0x80	0x80	0	0	0
0	0x80	0x80	0	0
0	0	0x80	0x80	0
0	0	0	0x80	0x80

Mixer2 түрлендіруінің дифференциалдық қасиеттері.

Mixer2 түрлендіруі сызықтық түрлендіру болып табылады. Бұл айырымдардың өзгерісінің ықтималдығына әсер етпейді. Дегенмен, көп раундты сипаттамаларды құру үшін, Mixer1 түрлендіруінен кейін алынған байттардың бірінде 0x80 мәні бар жолдардың мәні қалай түрленетінін анықтау өте маңызды. Көбейтуді орындау үшін әр жолға өзінің жеке көпмүшелігі қолданатынын ұмытпаған жөн. Әр жолда 4 байттан бар. Егер әр байтта айырымдарының 0-ге тең немесе 0x80-ге тең мәндері болуы мүмкін екенін ескеретін болсақ, онда 0x00000080-ден 0x80808080-ге дейінгі әр жолға барлығы 15 мүмкін толтырулар алынады. Полиномдарды қолданғандағы айырымдардың мәндері қалай түрленетінін қарастырамыз. 128-биттік блок нұсқасы үшін тек 60 нұсқа бар: 15 толтыру нұсқасы және төрт полиномдар. Бізді Mixer2 түрлендіруінің шығысындағы байттар, S ауыстыру блогынан өткеннен кейін, 0x80 мәндеріне айналдыруға болатын жағдайлар қызықтырады. Яғни, Mixer2 түрлендіруінің шығыс байтынан қалыптасқан, ΔA және ΔC тәуелділіктер кестесінде ΔA мен $\Delta C = 0x80$ қиылысында 0-ден басқа мән болуы керек. Барлық мүмкін нұсқаларды есептеп шығуға арналған бағдарлама жасалды.

Осы бағдарламаны қолдану нәтижесінде қарастырылған 60 комбинацияның тек бір мәні ғана берілген шартты қанағаттандыратыны анықталды. 0x80808000-ге тең кіріс айырымы 0xbbc868cf айырымына айналады және S-блок ауыстыруынан өткеннен кейін 0x80808080 айырымының мәніне түрленеді (сурет 3.10). Дәл осы комбинацияны көп раундтық сипаттамаларды құру үшін қолданатын боламыз.



Сурет 3.10 – Mixer2 операциясын талдау нәтижесі

Көпраундтық сипаттамаларды құру. Жұмыстың келесі сатысында, Qamal шифрлау алгоритмінің негізгі операцияларының дифференциалдық қасиеттеріне сүйене отырып, оның көпраундтық сипаттамасын құрамыз және оның ықтималдығын анықтайтын боламыз. Бұл жердегі міндетіміз – белсенді S-блоктарға мүмкіндігінше аз әсер ететіндей сипаттаманы құру. Берілген сипаттама үшін мәтіндердің дұрыс жұбын табу ықтималдығы осыған тікелей байланысты. Біздің міндетіміз - Qamal шифры үшін толық теру әдісінен гөрі қанша раундта жылдам талдауға болатындығын анықтау. Басқаша айтқанда, шифрлау алгоритмінің неше раундында дифференциалдық криптоталдау қиындығы толық теру қиындығымен сәйкес келетінін табу. 128 биттік мәліметтер блогы үшін ұзындығы 128 бит болатын құпия кілт қолданылады, ендеше толық іздеудің күрделілігі 2^{128} құрайды.

Шифрлаудың бірінші раундын қарастырайық. Мәтіннің айырымдарының өзгеруіне әсер етпейтіндіктен, раундтық кілтті қосуды талдауда ескермейміз, ол туралы жоғарыда да айтылды. Бізге Mixer1 түрлендіру кірісінде мәні 0x80 болатын байт пайда болуы керек. Жоғарыда айтылған 4-қасиетке сәйкес 0xfe мәні $4/256 = 1/64 = 2^{-6}$ ықтималдықпен 0x80 мәніне түрленеді. Бұл ретте бірінші раундтың кірісін Mixer1 түрлендірінен кейін нөлдік емес айырмашылық массивтің үшінші жолында болатындай етіп құруымыз керек. Егер кіріс айырымы алғашқы үш бағанының бірінші және төртінші байттарына әсер етсе, онда S-блок ауыстыруынан кейін нөлдік емес байттардың барлығы $(2^{-6})^6 = 2^{-36}$ ықтималдықпен 0x80 байтына айналады. Көріп отырғанымыздай, шифрлаудың бірінші айналымынан бастабақ раундтық сипаттаманы алу ықтималдығы өте аз. Mixer1 түрлендіруі жалпы ықтималдыққа әсер етпей массивтің элементтерін өзгертеді. Нәтижесінде нөлдік емес байттар тек үшінші жолдың алғашқы үш

позициясында пайда болады. Қалған барлық мәндер нөлге тең болады. Бірінші раундтқа арналған түрлендірулердің толық сұлбасы кесте 3.8 - де келтірілген.

Кесте 3.8 – Бірінші раунд үшін айырымдардың өзгерісі

Бірінші раунд кірісі				Mixer1 түрлендіруінің кірісі				Mixer2 түрлендіруінің кірісі			
0xfe	0xfe	0xfe	0	0x80	0x80	0x80	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0x80	0x80	0x80	0
0xfe	0xfe	0xfe	0	0x80	0x80	0x80	0	0	0	0	0

Егер Mixer2 түрлендіруіндегі үшінші жолдың кірісіне 0x80808000 мәні келсе, онда шығысында 0xbbc868cf мәні болатыны жоғарыда көрсетілген. Сонымен бірге, 0xbbc868cf айырымының әрбір байтын 0x80 байтына айналдыруға болады. Төрт байттың барлығының 0x80 мәніне айналу ықтималдығы $(2^{-7})^4 = 2^{-28}$ –ке тең болады. Сонымен, шифрлаудың екі раундының жиынтық ықтималдығы 2^{-64} құрайды. Mixer1 түрлендіруінен кейін екінші және төртінші жолдар 0x80 байттың орналасуы кесте 3.9 - да көрсетілген.

Кесте 3.9 – Екінші раунд үшін айырымдардың өзгерісі

S-блоктың кірісі				Mixer1 түрлендіруінің кірісі				Mixer2 түрлендіруінің кірісі			
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0x80	0x80	0x80	0x80
0xbb	0xc8	0x68	0xcf	0x80	0x80	0x80	0x80	0	0	0	0
0	0	0	0	0	0	0	0	0x80	0x80	0x80	0x80

Mixer2 түрлендіруінің дифференциалдық қасиеттерін талдау нәтижесінде, екінші және төртінші жолдар үшін 0x80808080 кіріс айырымын келесі раундтарда S-блок түрлендіруінен кейінгі айырымдары 0x80-ге тең барлық байттарын құрайтындай етіп түрлендіруге болмайтындығы анықталды. Сондықтан, түрлендірудің басқа нұсқаларын қарастырдық. Mixer2 түрлендіруінен кейін алынған 0x95d14821 айырымының мәнін қамтитын екінші және төртінші жолдар үшінші раундтың S-блогына кіріс элемент болып келеді (кесте 3.10). Дифференциалдық қасиеттер кестесіне сәйкес 0x95 байтын 0x80 байтымен ауыстыруға болатындығын анықтадық. Қалған байттар үшін Mixer1 түрлендіруінен кейін бағанның нөлдік емес үш байтына (төртеуінен) әсер ететін ауыстыру мүмкіндігі таңдалды. 0xd1 байты талдау кестесіне сәйкес 0x40 байтына және 0xc0 байтына айналу мүмкіндігі бар. Бұл жағдайда Mixer1 түрлендіруі кесте 3.11 - ге сәйкес орындалады. 0x48 және 0x21 байттарын 0xd1 байты сияқты түрлендіру мүмкін емес, сондықтан олар 0x10 және 0xf0 байттарына ауыстырылатындығы анықталды. Бұл жағдайда Mixer1 түрлендіруі кесте 3.12 - ге сәйкес орындалады.

Кесте 3.10 – Үшінші раунд үшін айырымдардың өзгерісі

S-блоктың кірісі				Mixer1 түрлендіруінің кірісі				Mixer2 түрлендіруінің кірісі			
0	0	0	0	0	0	0	0	0x80	0xc0	0x30	0x30
0x95	0xd1	0x48	0x21	0x80	0x40	0x10	0x10	0	0x80	0x20	0x20
0	0	0	0	0	0	0	0	0x80	0x40	0x10	0x10
0x95	0xd1	0x48	0x21	0x80	0xc0	0x1f	0x1f	0	0	0	0

Кесте 3.11 – 0x40 және 0xc0 байттары үшін Mixer1 түрлендіруі

Кіріс айырымдары	Түрлендірудің 1-қадамы	Түрлендірудің 2-қадамы	Түрлендірудің 3-қадамы	Шығыс айырымдары
0x00	0x00	0x40	0x80	0xc0
0x40	0x00	0x00	0x40	0x80
0x00	0x40	0x00	0x00	0x40
0xc0	0x00	0x40	0x00	0x00

Кесте 3.12 – 0x10 и 0xf0байттары үшін Mixer1 түрлендіруі

Кіріс айырымдары	Түрлендірудің 1-қадамы	Түрлендірудің 2-қадамы	Түрлендірудің 3-қадамы	Шығыс айырымдары
0x00	0x00	0x10	0x20	0x30
0x10	0x00	0x00	0x10	0x20
0x00	0x10	0x00	0x00	0x10
0xf0	0x00	0x10	0x00	0x00

Үшінші раунд үшін әрбір байтты S-блок бойынша түрлендіру ықтималдығы 2^{-7} құрайды. Үшінші раундта барлығы 8 нөлдік емес блок қолданылады. Сондықтан, үшінші раундта түрлендірілу ықтималдығы $(2^{-7})^8 = 2^{-56}$ -ке тең болады. Шифрлау алгоритмінің үш раунды үшін ықтималдықтың мәні 2^{-120}-не тең болады, бұл кілтті толық теру (2^{-128}) ықтималдығының мәніне өте жақын. Сондықтан айырмалардың өзгеруін одан әрі қарастырудың мағынасы жоқ. Шифрлаудың үшінші раундының шығысындағы айырымдардың мәнін анықтауымыз керек. Бұны табу үшін үшінші раундтағы Mixer2 түрлендірілуінің кіріс айырымдарын қарастырамыз (кесте 3.10). Оған Mixer1 және Mixer2 түрлендірулерін қолдана отырып, кесте 3.13 - те көрсетілгендей айырымдардың түрін аламыз.

Кесте 3.13 – Шифрлаудың 3-раундынан кейінгі айырымдардың күйі

Шифрмәтіннің үшінші раунд үшін мәндері			
0x4c	0x6b	0x94	0xea
0xad	0xde	0x47	0x5b
0xe1	0xb2	0xd3	0xb1
0x00	0x00	0x00	0x00

Шифрлау алгоритмінде блоктың және кілттің ұзындығы 128 биттен тұратын жағдай үшін, алгоритм дифференциалдық криптоталдау үшінші раундтан кейін берік екендігін көрсеттік. Шифрдің сенімділігін толығымен тексеру үшін оның басқада криптоталдау әдістерінің көмегімен мұқият зерттеу қажет. Ондай талдаулардың нәтижелерін келесі бөлімдерде көрсететін боламыз.

3.5 Шифрлау алгоритміне сызықтық криптоталдау әдісін қолдану

Сызықтық криптоталдауды жапондық криптолог Мицуру Мацуи ойлап тапты. Оның 1993 жылы ұсынған алгоритмі DES шифрін ашуға бағытталған. Кейіннен сызықтық криптоталдау басқа да алгоритмдерге қолданылды. Бұл әдіс, бүгінгі күні дифференциалдық криптоталдаумен қатар, блоктық шифрларды ашудың кең таралған әдістерінің бірі [47, с. 109]. Сызықтық криптоталдау жүргізу нәтижесінде алынатын мәндер, сызықтық емес S-блок түрлендіруіне тікелей қатысты. Сондықтан, шифрды бағалағанда, алынған шифрдың кірісінен шығысына дейінгі аралықтағы шифрдың орындалу тізбегіне қатынасатын S-блоктардың санына қатысты статистикалық мән есептеледі. Ол осы тізбек арқылы кілттердің кейбір элементтерін анықтау ықтималдығын көрсетеді.

Сызықтық криптоталдаудың негізгі мақсаты, берілген шифр алгоритм үшін келесі тиімді сызықты өрнек табу:

$$A[i_1, i_2, \dots, i_a] \oplus C[j_1, j_2, \dots, j_b] = K[k_1, k_2, \dots, k_c] \quad (3.3)$$

мұндағы, $i_1, i_2, \dots, i_a, j_1, j_2, \dots, j_b$ және $k_1, k_2, \dots, k_c$ – берілген A ашық мәтін және оған сәйкес C шифр мәтіннің және K кілттің биттерінің бекітілген орындарын білдіреді.

Сызықтық криптоталдау екі қадам арқылы іске асады. Біріншісі - ашық мәтін, шифрмәтін және кілт арасында жоғары ықтималдықпен теңдеу құрып алу. Екіншісі – кілттердің биттерін табу үшін осы теңдеулерді белгілі ашық мәтін жұптарымен бірге пайдалану [77, 78]. Сызықтық талдауды бағалау үшін келесі екі лемманы қолданамыз.

Лемма 1. $X_i (1 \le i \le n)$ – мәндері p_i ықтималдықпен 0-ге тең немесе $1 - p_i$ ықтималдықпен 1-ге тең тәуелсіз кездейсоқ шамалар болсын. Онда $X_1 \oplus X_2 \oplus \dots \oplus X_n = 0$ болуының ықтималдығы келесідей анықталады:

$$\frac{1}{2} + 2^{n-1} \prod_{i=1}^{n} (p_i - 1/2) \quad (3.4)$$

Лемма 2. N – кездейсоқ берілген ашық мәтіндер саны және p – (3.3) теңдеуінің орындалуының ықтималдығы болсын, және $\left|p - {}^{1}/_{2}\right|$ өте кіші болсын. Онда сәттілікке жету(біздің жағдайда, кілтті табу) ықтималдығы келесі формула бойынша есептеледі:

$$\int_{-2\sqrt{N}|p-\frac{1}{2}|}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx.$$

1 кезеңде алгоритмге толықтай криптоталдау жүргізбес бұрын, пайдаланылған түрлендірулерді жеке-жеке қарастырып алайық. Шифрлау алгоритмінде кілттерді биттік қосу (XOR) операциясы, S-блок ауыстыруы, Mixer1 және Mixer2 түрлендірулері пайдаланылған. Алдымен, белгілеулерге түсіндірме бере кетейік: ашық мәтін биттері - $a[i,j]$, шифрмәтін биттері - $c[i,j]$, S-блок ауыстыруының шығыс биттері - $x[i,j]$, Mixer1 түрлендіруінің шығыс биттері - $y[i,j]$, Mixer2 түрлендіруінің шығыс биттері - $z[i,j]$. Мұндағы, i – раундты білдіреді, j – биттің орнын білдіреді.

Кілтті *XOR* операциясы арқылы кіріс мәндерге қосуды сызықты криптоталдауда жеке қарастырмасада болады. Себебі $z[i,j] \oplus k[i,j] = c[i,j]$ өзі сызықты функция.

S-блок. Қазіргі заманда стандарт ретінде ұсынылған симметриялы блокты шифрлардың басым көпшілігінде *S*-блоктар қолданылады. Оларды қолдану шифрлау алгоритмдерінің беріктілігін арттыруда өте үлкен маңызға ие. Көп жағдайда сызықтылықты бұзатын да осы *S*-блоктар болып табылады.

Криптоталдау барысында *S*-блоктың кірісі және шығысы болатын екілік векторлардың барлық комбинацияларын қадағалап отырамыз. Векторлардың әрбір жұбын орынтірек (маска) ретінде пайдаланамыз. Барлық мүмкін болатын кіріс және шығыс элементтерін екілік жүйеде қосамыз. Одан алынған нәтижелердегі 0 мен 1–ді санап, оларды матрицаның сәйкес элементі ретінде аламыз. Қарастырып отырған шифрлау алгоритмінің S-блогын зерттеу кезіндегі алынған нәтижелер кестесі тым үлкен болғандықтан, кесте 3.14 - те тек олардың ең кіші және ең үлкен мәндері ғана келтірілген. Салыстыру үшін бұл кестеде, басқада белгілі шифр алгоритмдерінде қолданған S–блоктарға жүргізілген сызықтық және дифференциалдық криптоталдаудың нәтижелері көрсетілген.

Кесте 3.14 – S-блоктарға жүргізілген сызықтық криптоталдау нәтижелері

Шифрлау алгоритмінің атауы	Ең аз кездесу саны	Ең көп кездесу саны
DES	12	48
ГОСТ 28147-89 (құрастырушылар көрсеткен S-блок үшін)	2	14
ГОСТ Р 34.13-2015	100	156
AES-128 (Rijndael)	111	145
Калина	104	152
EM cipher	100	156
Qamal	112	144

Mixer1 түрлендіруінде әрбір бағанның байттары модуль 256 бойынша бір-бірімен қосылады. Содан кейін бірінші бағанда алынған жаңа байт жоғарғы x_{00} байтының орнына жазылады, ал бастапқы байт бір позицияға төмен жылжытылады. Бұл әрекет *4* рет қайталанады. Нәтижесінде бірінші бағандағы жаңа *4* байт аламыз. Ары қарай, бұл операция қалған үш баған үшін де орындалады. Кесте 3.15 - де әр қадам сайын бағандардың әрбір жолындағы қосындылар келтіріліп отыр. Бұл жерде алынған қосындыдан модуль алынбаған.

Кесте 3.15 – Mixer1 түрлендіруіндегі әр қадамдағы қосындылар өрнегі

Қадам/ жол	0	1	2	3	4
1	X_0	$X_0+X_1+X_2+X_3$	$2X_0+2X_1+2X_2+X_3$	$4X_0+4X_1+3X_2+2X_3$	$8X_0+7X_1+6X_2+4X_3$
2	X_1	X_0	$X_0+X_1+X_2+X_3$	$2X_0+2X_1+2X_2+X_3$	$4X_0+4X_1+3X_2+2X_3$
3	X_2	X_1	X_0	$X_0+X_1+X_2+X_3$	$2X_0+2X_1+2X_2+X_3$
4	X_3	X_2	X_1	X_0	$X_0+X_1+X_2+X_3$

Mixer1 түрлендіруінде әрбір байттың кіші разрядтағы биті келесідей өрнектелетініне көз жеткізу қиын емес: $y[i,120]=x[i,88]$, $y[i,112]=x[i,80]$, $y[i,104]=x[i,72]$, $y[i,96]=x[i,64]$, $y[i,88]=x[i,56]$, $y[i,80]=x[i,48]$, $y[i,72]=x[i,40]$, $y[i,64]=x[i,32]$, $y[i,56]=x[i,24]$, $y[i,48]=x[i,16]$, $y[i,40]=x[i,8]$, $y[i,32]=x[i,0]$, $y[i,24]=x[i,120]\oplus x[i,88]\oplus x[i,56]\oplus x[i,25]$, $y[i,16]=x[i,112]\oplus x[i,80]\oplus x[i,48]\oplus x[i,17]$, $y[i,8]=x[i,104]\oplus x[i,72]\oplus x[i,40]\oplus x[i,9]$, $y[i,0]=x[i,96]\oplus x[i,64]\oplus x[i,32]\oplus x[i,1]$.

Сандық қосынды орындалатындықтан қалған орындағы биттер кіші разрядтардың қосындысынан тәуелді. Сондықтан, бұл түрлендіруде әрбір байттың кіші разрядтағы биттері арқылы шабуыл жасау тиімді.

Mixer2 түрлендіруі көпмүшеліктерді модуль бойынша көбейтуден тұрады. Яғни,

$$z(x)=y(x)\times m(x) mod\big(p(x)\big)$$

мұндағы, $m(x)$ және $p(x)$ бектітілген, белгілі көпмүшеліктер болғандықтан шығыс өрнекті кірістегі y айнымалысы арқылы сызықты өрнектеуге болады.

$z[i,127]=y[i,96]\oplus y[i,98]\oplus y[i,100]\oplus y[i,106]\oplus y[i,110]\oplus y[i,112]\oplus$
$\oplus y[i,114]\oplus y[i,115]\oplus y[i,116]\oplus y[i,118]\oplus y[i,119]\oplus y[i,126]\oplus$
$\oplus y[i,125]\oplus y[i,124]$
$z[i,126]=y[i,97]\oplus y[i,99]\oplus y[i,105]\oplus y[i,109]\oplus y[i,111]\oplus y[i,113]\oplus$
$\oplus y[i,114]\oplus y[i,115]\oplus y[i,117]\oplus y[i,118]\oplus y[i,125]\oplus y[i,127]\oplus$
$\oplus y[i,124]\oplus y[i,123]$
…
$z[i,1]=y[i,1]\oplus y[i,3]\oplus y[i,7]\oplus y[i,10]\oplus y[i,12]\oplus y[i,16]\oplus y[i,20]\oplus$
$\oplus y[i,22]\oplus y[i,24]\oplus y[i,30]\oplus y[i,2]\oplus y[i,5]\oplus y[i,8]\oplus y[i,11]\oplus y[i,14]\oplus\oplus$
$y[i,17]\oplus y[i,21]\oplus y[i,23]\oplus y[i,29]$
$z[i,0]=y[i,0]\oplus y[i,2]\oplus y[i,5]\oplus y[i,6]\oplus y[i,8]\oplus y[i,9]\oplus y[i,11]\oplus y[i,14]$

$\oplus y[i,15] \oplus y[i,17] \oplus y[i,18] \oplus y[i,19] \oplus y[i,21] \oplus y[i,23] \oplus y[i,29]$. (3.5)

(3.5) формула барлық шығыс биттер үшін (В.1) формуласында келтірілген.

2 кезең. Алынған нәтижелерді ескере отырып раундтар бойынша толық алгоритмге талдау жүргізейік. Жоғарыда айтылғандай Mixer1 түрлендіруінде әрбір байттың кіші разрядтағы биті сызықтық криптоталдау жүргізуге тиімді болғандықтан солардың ішінен біреуін таңдап алайық.

$y[1,48] = x[1,16]$

Ал $x[1,16]$ өз кезегінде S-блок ауыстыруының шығыс биті болып табылады. Сондықтан 11111111x000000001 орынтірегін (маскасын) қарастырған тиімді.

Ықтималдықтың ең үлкен ауытқу кесте 3.14-те көрсетілгендей 0,0625. Өйткені, сызықтық криптоталдау бойынша алынған мәндердің кездесу саны 112-ден 144-ге дейінгі аралық, ал ықтималдықтары 0,4375-тен 0,5625-ке дейін аралығында. Ең үлкен және ең кіші мәндердің 128 ден қашықтықтары бірдей 16-ға тең болғандықтан, жоғарыда көрсетілген орынтіректен 112-ні таңдасақта 144-ті таңдасақта болады. Сондықтан, кестені пайдаланып ықтималдығы 144/256-ға тең келесі орынтіректерді таңдап алдық:
00011101x000000001, 10101000x000000001, 10101100x000000001,
10110001x000000001, 10110101x000000001.

Яғни, 9/16 ықтималдылықпен төмендегі теңдеулердің әрқайсысы ақиқат:

$a[1,16] \oplus a[1,18] \oplus a[1,19] \oplus a[1,20] \oplus k[0,16] \oplus k[0,18]k[0,19] \oplus k[1,20] = = x[1,16]$,
$a[1,19] \oplus a[1,21] \oplus a[1,23] \oplus k[0,19] \oplus k[0,21] \oplus k[0,23] = x[1,16]$,
$a[1,18] \oplus a[1,19] \oplus a[1,21] \oplus a[1,23] \oplus k[0,18] \oplus k[0,19] \oplus k[1,21] \oplus k[0,23] = = x[1,16]$,
$a[1,16] \oplus a[1,20] \oplus a[1,21] \oplus a[1,23] \oplus k[0,16] \oplus k[0,20] \oplus k[1,21] \oplus k[0,23] = = x[1,16]$,
$a[1,16] \oplus a[1,18] \oplus a[1,20] \oplus a[1,21] \oplus a[1,23] \oplus k[0,16] \oplus k[0,18] \oplus k[1,20] \oplus k[0,21] \oplus k[0,23] = x[1,16]$.

Барлық теңдеудің ықтималдығы бірдей болғандықтан, айнымалысының саны ең азы 2-ші теңдеуді таңдап алайық және $y[1,48] = x[1,16]$ болғандықтан,

$$a[1, 19] \oplus a[1, 21] \oplus a[1, 23] \oplus k[0, 19] \oplus k[0, 21] \oplus k[0, 23] = y[1,48]$$

$y[1,48]$ Mixer1 түрлендіруінің шығыс биті, ал Mixer2 түрлендіруіне кіріс бит болғандықтан (3.5) теңдеулер жүйесінің ішінен осы бит қатысатын теңдеулерді бөліп аламыз. Ондай теңдеулердің жалпы саны он үш. Солардың ішінен айнымалысының саны ең аз болатын келесі теңдеуді таңдап алдық:

$z[1,35] = y[1,41] \oplus y[1,42] \oplus y[1,45] \oplus y[1,48] \oplus y[1,49] \oplus y[1,52] \oplus y[1,53] \oplus \oplus y[1,58] \oplus y[1,54] \oplus y[1,61] \oplus y[1,63]$

мұндағы, $y[1,j],\ j = 41,42,45,48,49,52,53,58\ 54,61,63$ өрнектерінің ықтималдықтары S-блоктың статистикалық мәніне байланысты жоғарыда айтылғандай 0,4375-тен 0,5625-ке дейін аралығында. Сызықтық криптоталдаудың мақсаты, ықтималдығының ауытқуы 0,5-тен ең үлкен теңдеулерді табу болғандықтан, алгоритмді бағалау үшін әрбір $y[1,j]$ өрнектерін ең үлкен ауытқумен алып, жоғарыда келтірілген леммалардың көмегімен $z[1,35]$ теңдеуінің ықтималдығын бағалауға болады. Бұл әдісті төменде баяндайтын боламыз. Ал, қазір ең төменнен бағалау үшін $z[1,35]$ теңдеуіндегі $y[1,42]$ айнымалысынан басқасының бәрі 0-ге тең немесе қосындысының ықтималдығы 1-ге тең болатын жағдайды таба аламыз деген болжам жасайық. Яғни, $y[1,41] \oplus y[1,42] \oplus y[1,45] \oplus y[1,49] \oplus y[1,52] \oplus y[1,53] \oplus y[1,58] \oplus$ $\oplus\, y[1,54] \oplus y[1,61] \oplus y[1,63] = 0$ болсын. Онда Mixer2 түрлендіруінен кейін раундтық кілт қосылатындығын ескерсек,

$$z[1,35] = a[1,19] \oplus a[1,21] \oplus a[1,23] \oplus k[0,19] \oplus k[0,21] \oplus k[0,23] \oplus k[1,35]$$

теңдеуінің ақиқат болуының ықтималдығы 9/16-ға тең.

2-ші раундтқа өткенде $z[1,35]$ биті өзі орналасқан сәйкес байтта 4-ші орында тұрғандықтан, S-блоктың кестесінен 00001000x11111111 орынтірегін қараймыз. 112 кездесетін орынтіректер 00001000x 00101000, 00001000x 11101110, ал 144 кездесетіндері 00001000x 00010000, 00001000x 11000110, 00001000x 11010110. Алынған мәндерге қарап, ең тиімдісі $z[1,35] \oplus k[1,35] = x[2,36]$ теңдеуі екенін көру қиын емес. Бұл теңдеудің ақиқат болу ықтималдығы $(9/16)^2$.

Енді, кірісте $x[2,36]$ қатысатын Mixer1-ді қарастырамыз. Ол үшін Mixer1-ді 2-ші раунд төртінші баған үшін жазып алайық.

Кесте 3.15 - тегі мәндер кесте 3.16 - 3.19 - дерде 4-ші баған үшін жазылған және “*”-қойылған ұяшықтардағы мәндер өзінен төменгі орындағы мәндерден келетін разрядтарға тәуелді болғандықтан нақты көрсете алмаймыз. Сонымен бірге мәні аса маңызды емес ұяшықтарға да «*» белгісі қойылды.

Кесте 3.16 – Mixer1 түрлендіруі 1-ші жол 4-ші баған үшін

Бірінші жол, төртінші баған мәндері								
1	2	3	4	5	6	7	8	9
1-ші жол биттері	$y_{[2,103]}$	$y_{[2,102]}$	$y_{[2,101]}$	$y_{[2,100]}$	$y_{[2,99]}$	$y_{[2,98]}$	$y_{[2,97]}$	$y_{[2,96]}$
$8Y_0$	$x_{[2,100]}$	$x_{[2,99]}$	$x_{[2,98]}$	$x_{[2,97]}$	$x_{[2,96]}$	0	0	0
$7Y_1$	*	*	*	*	*	*	$x_{[2,66]}$ + $x_{[2,65]}$	$x_{[2,64]}$
$6Y_2$	*	* $x_{[2,36]}$ + $x_{[2,37]}$	* $x_{[2,35]}$ + $x_{[2,36]}$	* $x_{[2,34]}$ + $x_{[2,35]}$	* $x_{[2,33}$ + $x_{[2,34]}$	$x_{[2,32]}$ + $x_{[2,33]}$	$x_{[2,32]}$	0

3.16-кестенің жалғасы

1	2	3	4	5	6	7	8	9
$4Y_3$	x_5	x_4	x_3	x_2	x_1	x_0	0	0
$8Y_0+7Y_1+$ $+6Y_2+4Y_3$	*	*	*	*	*	*	$x_{[2,65]}+x_{[2,66]}+x_{[2,32]}$	$x_{[2,64]}$

Кесте 3.17 – Mixer1 түрлендіруі 2-ші жол 4-ші баған үшін

Екінші жол, төртінші баған мәндері								
2-ші жол биттері	$y_{[2,71]}$	$y_{[2,70]}$	$y_{[2,69]}$	$y_{[2,68]}$	$y_{[2,67]}$	$y_{[2,66]}$	$y_{[2,65]}$	$y_{[2,64]}$
$4Y_0$	$x_{[2,101]}$	$x_{[2,100]}$	$x_{[2,99]}$	$x_{[2,98]}$	$x_{[2,97]}$	$x_{[2,96]}$	0	0
$4Y_1$	$x_{[2,69]}$	$x_{[2,68]}$	$x_{[2,67]}$	$x_{[2,66]}$	$x_{[2,65]}$	$x_{[2,64]}$	0	0
$3Y_2$	$*\ x_{[2,38]}+x_{[2,39]}$	$*\ x_{[2,37]}+x_{[2,38]}$	$*\ x_{[2,36]}+x_{[2,37]}$	$x_{[2,35]}+x_{[2,36]}$	$x_{[2,4]}+x_{[2,5]}$	$x_{[2,33]}+x_{[2,34]}$	$x_{[2,32]}+x_{[2,33]}$	$x_{[2,32]}$
$2Y_3$	$x_{[2,6]}$	$x_{[2,5]}$	$x_{[2,4]}$	$x_{[2,3]}$	$x_{[2,2]}$	$x_{[2,1]}$	$x_{[2,0]}$	0
$4Y_0+4Y_1+$ $+3Y_2+2Y_3$	*	*	*	*	*	*	$x_{[2,32]}+x_{[2,33]}+x_{[2,0]}$	$x_{[2,32]}$

Кесте 3.18 – Mixer1 түрлендіруі 3-ші жол 4-ші баған үшін

Үшінші жол, төртінші баған мәндері								
3-ші жол биттері	$y_{[2,39]}$	$y_{[2,38]}$	$y_{[2,37]}$	$y_{[2,36]}$	$y_{[2,35]}$	$y_{[2,34]}$	$y_{[2,33]}$	$y_{[2,32]}$
$2Y_0$	$x_{[2,102]}$	$x_{[2,101]}$	$x_{[2,100]}$	$x_{[2,99]}$	$x_{[2,98]}$	$x_{[2,97]}$	$x_{[2,96]}$	0
$2Y_1$	$x_{[2,70]}$	$x_{[2,69]}$	$x_{[2,68]}$	$x_{[2,67]}$	$x_{[2,66]}$	$x_{[2,65]}$	$x_{[2,64]}$	0
$2Y_2$	$x_{[2,38]}$	$x_{[2,37]}$	$x_{[2,36]}$	$x_{[2,35]}$	$x_{[2,34]}$	$x_{[2,33]}$	$x_{[2,32]}$	0
$1Y_3$	$x_{[2,7]}$	$x_{[2,6]}$	$x_{[2,5]}$	$x_{[2,4]}$	$x_{[2,3]}$	$x_{[2,2]}$	$x_{[2,1]}$	$x_{[2,0]}$
$2Y_0+2Y_1+$ $+2Y_2+Y_3$	*	*	*	*	*	*	*	$x_{[2,0]}$

Кесте 3.19 – Mixer1 түрлендіруі 4-ші жол 4-ші баған үшін

Төртінші жол, төртінші баған мәндері								
4-ші жол биттері	$y_{[2,7]}$	$y_{[2,6]}$	$y_{[2,5]}$	$y_{[2,4]}$	$y_{[2,3]}$	$y_{[2,2]}$	$y_{[2,1]}$	$y_{[2,0]}$
Y_0	$x_{[2,103]}$	$x_{[2,102]}$	$x_{[2,101]}$	$x_{[2,100]}$	$x_{[2,99]}$	$x_{[2,98]}$	$x_{[2,97]}$	$x_{[2,96]}$
Y_1	$x_{[2,71]}$	$x_{[2,70]}$	$x_{[2,69]}$	$x_{[2,68]}$	$x_{[2,67]}$	$x_{[2,66]}$	$x_{[2,65]}$	$x_{[2,64]}$
Y_2	$x_{[2,39]}$	$x_{[2,38]}$	$x_{[2,37]}$	$x_{[2,36]}$	$x_{[2,35]}$	$x_{[2,34]}$	$x_{[2,33]}$	$x_{[2,32]}$
Y_3	$x_{[2,7]}$	$x_{[2,6]}$	$x_{[2,5]}$	$x_{[2,4]}$	$x_{[2,3]}$	$x_{[2,2]}$	$x_{[2,1]}$	$x_{[2,0]}$
Y_0+Y_1+ $+Y_2+Y_3$	*	*	*	*	*	*	*	*

$x[2,36]$ қатысатын mixer1 түрлендіруінің шығыс биттері: $y[2,103]$, $y[2,102], y[2,101], y[2,101], y[2,71], y[2,70], y[2,69], y[2,68], y[2,67], y[2,39], y[2,38],$ $y[2,37], y[2,7], y[2,6], y[2,5], y[2,4]$. Осы биттердің ішінде ең кіші разрядттарда тұрғандары 5-ші разрядта тұр. Неғұрлым кіші разрядта тұрса, соғұрлым алынатын мәннің ықтималдығын оңай есептей аламыз. Сонымен қатар, неғұрлым жоғарғы разрядтарға төменгі разрядтардан 1 келу ықтималдығы 0,5-ке соғұрлым жақындай түседі [65]. Талдау жасау үшін алынатын теңдеудің ықтималдығы жоғарыда айтқандай, 0,5-тен алшағырақ болғаны жақсы. Сондықтан, алынған биттердің ішінен $y[2,68]$-ні таңдап алдық.

$y[2,68] = x[2,98] \oplus x[2,66] \oplus x[2,35] \oplus x[2,36] \oplus x[2,3] \oplus 1$ теңдеуінің ықтималдығын есептейік. 1 қосылмауының ықтималдығы оған қарсы оқиға болғандықтан, қайсысын тапсақ та жеткілікті. Өзінен төмен разрядтардан 1 келу ықтималдығын табайық. y[2,68]-ге разрядттар y[2,67] ден және y[2,66] дан келеді. Ал өз кезегінде y[2,66]-ға y[2,65] тен келуі мүмкін. Осы мүмкіндіктерді есептеу үшін алдымен теңдеулерді жазып алайық.

$y[2,67] = x[2,97] \oplus x[2,65] \oplus x[2,35] \oplus x[2,34] \oplus x[2,2] \oplus r[67],$
$y[2,66] = x[2,96] \oplus x[2,64] \oplus x[2,34] \oplus x[2,33] \oplus x[2,1] \oplus r[66],$
$y[2,65] = x[2,33] \oplus x[2,32] \oplus x[2,1]$

мұндағы, $r[66]$ және $r[67]$ келетін разрядттарға байланысты 0-ге немесе 1-ге тең. Қажет ықтималдықты есептелік.

$$P = P\left(y[2,67] \overset{+}{\rightarrow} y[2,68]\right) \times P\left(y[2,66] \overset{-}{\rightarrow} y[2,68]\right) + P\left(y[2,67] \overset{-}{\rightarrow} y[2,68]\right) \times$$
$$\times P\left(y[2,66] \overset{+}{\rightarrow} y[2,68]\right)$$

мұндағы $P\left(y[2,i] \overset{+}{\rightarrow} y[2,j]\right)$ және $P\left(y[2,i] \overset{-}{\rightarrow} y[2,j]\right)$ ықтималдықтары $y[2,j]$ өрнегіне $y[2,i]$-ші өрнектен сәйкесінше разряд келу және келмеу ықтималдықтары. Ол ықтималдықтарды, $y[2,66]$, $y[2,67]$ өрнектері 5

айнымалының қосындысынан тұрғандықтан кесте 3.20 - дан оңай көруге болады. Мысалы,

$\left(y[2,66] \overset{+}{\rightarrow} y[2,67]\right) = P\left(y[2,67] \overset{+}{\rightarrow} y[2,68]\right) = \frac{20}{32}$, $P\left(y[2,66] \overset{+}{\rightarrow} y[2,68]\right) = \frac{6}{32}$,

$P\left(y[2,67] \overset{-}{\rightarrow} y[2,68]\right) = \frac{12}{32}$, $P\left(y[2,66] \overset{-}{\rightarrow} y[2,68]\right) = \frac{26}{32}$ т.с.с.

Кесте 3.20 – 5 қосынды жағдайындағы разряд берілу мүмкіндіктері

5 орын үшін биттердің орналасу мүмкіндіктері	Неше орын разряд кетеді	5 орын үшін биттердің орналасу мүмкіндіктері	Неше орын разряд кетеді	5 орын үшін биттердің орналасу мүмкіндіктері	Неше орын разряд кетеді	5 орын үшін биттердің орналасу мүмкіндіктері	Неше орын разряд кетеді
00000	-	01000	-	10000	-	11000	1
00001	-	01001	1	10001	1	11001	1
00010	-	01010	1	10010	1	11010	1
00011	1	01011	1	10011	1	11011	2
00100	-	01100	1	10100	1	11100	1
00101	1	01101	1	10101	1	11101	2
00110	1	01110	1	10110	1	11110	2
00111	1	01111	2	10111	2	11111	2

$y_{[2,67]}$ –өрнегі өз кезегінде $y_{[2,66]}$-дан, ал $y_{[2,66]}$ өрнегі $y_{[2,65]}$ өрнегінен келетін разрядтарға тәуелді. $y_{[2,65]}$ теңдеуі 3 айнымалының қосындысынан тұрғандықтан, одан $y_{[2,66]}$ теңдеуіне разряд қосылу және қосылмау ықтималдықтары бірдей $^1/_2$ –ге тең болатындығына оңай көз жеткізуге болатындықтан, жұмыста келтірмедік. Осыларды ескеретін болсақ, іздеп отырған ықтималдығымыз шартты (толық) ықтималдық формуласы бойынша:

$P = P\left(y[2,67] \overset{+}{\rightarrow} y[2,68]\right) \times P\left(y[2,66] \overset{-}{\rightarrow} y[2,68]\right) + P\left(y[2,67] \overset{-}{\rightarrow} y[2,68]\right) \times$
$P\left(y[2,66] \overset{+}{\rightarrow} y[2,68]\right) = \left[P\left(y[2,65] \overset{-}{\rightarrow} y[2,66]\right) \times P\left(y[2,66] \overset{+}{\rightarrow} y[2,67]\right) \times P\left(y[2,67]\right.\right.$
$\left.\left. \overset{+}{\rightarrow} y[2,68]\right) + P\left(y[2,65] \overset{-}{\rightarrow} y[2,66]\right) \times P\left(y[2,66] \overset{-}{\rightarrow} y[2,67]\right) \times P\left(y[2,67] \overset{+}{\rightarrow} y[2,68]\right) +\right.$
$P\left(y[2,65] \overset{+}{\rightarrow} y[2,66]\right) \times P\left(y[2,66] \overset{+}{\rightarrow} y[2,67]\right) \times P\left(y[2,67] \overset{+}{\rightarrow} y[2,68]\right) + P\left(y[2,65]\right.$
$\left. \overset{+}{\rightarrow} y[2,66]\right) \times P\left(y[2,66] \overset{-}{\rightarrow} y[2,67]\right) \times P\left(y[2,67] \overset{+}{\rightarrow} y[2,68]\right)\Big] \times \Big[P\left(y[2,65]\right.$
$\left. \overset{-}{\rightarrow} y[2,66]\right) \times P\left(y[2,66] \overset{-}{\rightarrow} y[2,68]\right) + P\left(y[2,65] \overset{+}{\rightarrow} y[2,66]\right) \times P\left(y[2,66]\right.$
$\left. \overset{-}{\rightarrow} y[2,67]\right)\Big] + \Big[P\left(y[2,65] \overset{-}{\rightarrow} y[2,66]\right) \times P\left(y[2,66] \overset{+}{\rightarrow} y[2,67]\right) \times P\left(y[2,67]\right.$
$\left. \overset{-}{\rightarrow} y[2,68]\right) + P\left(y[2,65] \overset{-}{\rightarrow} y[2,66]\right) \times P\left(y[2,66] \overset{-}{\rightarrow} y[2,67]\right) \times P\left(y[2,67] \overset{-}{\rightarrow} y[2,68]\right) +$
$P\left(y[2,65] \overset{+}{\rightarrow} y[2,66]\right) \times P\left(y[2,66] \overset{+}{\rightarrow} y[2,67]\right) \times P\left(y[2,67] \overset{-}{\rightarrow} y[2,68]\right) + P\left(y[2,65]\right.$

$\overset{+}{\rightarrow} y[2,66]\Big) \times P\big(y[2,66] \overset{-}{\rightarrow} y[2,67]\big) \times P\big(y[2,67] \overset{-}{\rightarrow} y[2,68]\big)\Big] \times \Big[P\big(y[2,65]$
$\overset{-}{\rightarrow} y[2,66]\big) \times P\left(y[2,66] \overset{+}{\rightarrow} y[2,68]\right) + P\left(y[2,65] \overset{+}{\rightarrow} y[2,66]\right) \times P\left(y[2,66]\right.$
$\left.\overset{+}{\rightarrow} y[2,67]\right)\Big] == \left[\frac{1}{2} \times \frac{20}{32} \times \frac{20}{32} + \frac{1}{2} \times \frac{6}{32} \times \frac{20}{32}\right] \times 2 \times \left[\frac{1}{2} \times \frac{6}{32} + \frac{1}{2} \times \frac{6}{32}\right] + \left[\frac{1}{2} \times \frac{20}{32} \times \frac{12}{32} + \frac{1}{2} \times\right.$
$\left.\frac{6}{32} \times \frac{12}{32}\right] \times 2 \times \left[\frac{1}{2} \times \frac{26}{32} + \frac{1}{2} \times \frac{26}{32}\right] = \frac{351}{2^{10}}$.

2-ші раундта Mixer2 түрлендіруіне келейік. Кірісте $y[2,68]$ қатысатын барлық теңдеулердің жалпы саны 13. Ешқандай теңдеудің ерекшелігі жоқ болғандықтан кез-келген бір теңдеуді аламыз да, алдыңғы раундтағыдай $y[2,68]$-ден басқасының қосындысы 0-ге тең болсын деп алып талдауды жалғастыра береміз.

2 раунд бойынша талдаудың нәтижесін қорытындылай кетсек, лемма1 бойынша, алынған теңдеудің ықтималдығы

$$p = \frac{1}{2} + 2^{2-1} \times \left(\frac{9}{16} - \frac{1}{2}\right) \times \left(\frac{9}{16} - \frac{1}{2}\right) \times \left(\frac{351}{2^{10}} - \frac{1}{2}\right) \approx \frac{1}{2} + 2 \times \frac{1}{2^4} \times \frac{1}{2^4} \times \left(-\frac{1}{2^{2,5}}\right) = \frac{1}{2} -$$
$$- \frac{1}{2^{9,5}} \approx 0{,}4986$$

$N = \left(\frac{1}{0{,}5+2^{-9{,}5}-0{,}5}\right)^2 = 2^{19}$, мұндағы N - 0,977 ықтималдықпен теңдеуді шешуге қежетті ашық мәтіндер жұбы. Демек, 2 раундтан кейін кілттің кейбір элементтерін табу үшін 2^{19} жұп ашық мәтін қажет және алынатын теңдеудің ықтималдығы 0,4986 –ге тең.

Сызықтық криптоталдауды дәл осылай жалғастыра береміз деп алып, осы алынған нәтижелерді толық 8 раундтқа пайдаланайық.

$$p = \frac{1}{2} + 2^{9-1} \times \left(\frac{9}{16} - \frac{1}{2}\right) \times \left[\left(\frac{9}{16} - \frac{1}{2}\right) \times \left(\frac{351}{2^{10}} - \frac{1}{2}\right)\right]^9 \approx \frac{1}{2} + 2^8 \times \frac{1}{2^4} \times \frac{1}{2^{36}} \times \left(-\frac{1}{2^{22,5}}\right) = \frac{1}{2} -$$
$$- \frac{1}{2^{54,,5}} \approx 0{,}4999$$

$N = \left(\frac{1}{0{,}5+2^{-54{,}5}-0{,}5}\right)^2 = 2^{109}$, сонымен - 0,977 ықтималдықпен кілттердің 8 элементін (әр раунд сайын 1 биті қосылып отырады) табу үшін 2^{109} ашық мәтіндер жұбы керек. Бірақ, бұл жерде Mixer 2 түрлендіруін мүлде ескермедік десек те болады. Өзіміз алдын-ала қойған талабымыздай Mixer2-нің шығысында барлық раундттар бойынша таңдалған элементтерден басқаларының қосындысы 0 болатындай теңдеулерді табу өте қиын (ықтималдығы 0-ге жуық). Соның өзінде жалпы терулер саны 2^{109}-ден кем түспейді.

Келесі кезекте 1-ші раундтан кейін Mixer2 түрлендіруінде ең аз айнымалылар қатысатын теңдеулерді әр раунд сайын табамыз деп есептейік. Барлығы сызықты өрнектер болғаннан кейін талдау барысында тиімді өрнектер таңдап алуымыз ғана қажет. Ондай теңдеулерде кемінде 10 айнымалы қатысады. Мысалы, $z[i,016] = y[i,1] \oplus y[i,2] \oplus y[i,3] \oplus y[i,5] \oplus y[i,7] \oplus y[i,13]y[i,19] \oplus$ $\oplus\, y[i,21] \oplus y[i,27] \oplus y[i,29]$ ең айнымалысы аз өрнек. Кіріс айнымалы болып қатысып отырған әрбір $y[i,j]$-дің ықтималдығы $\frac{9}{16}$ –ден артық болатындығын

жоғарыда айттық. Олай болса, 4 раунд үшін алынатын теңдеудің ықтималдығын есептейік.

$$p = \frac{1}{2} + 2^{2-1} \times \left(\frac{9}{16} - \frac{1}{2}\right) \times \left[\left(\frac{9}{16} - \frac{1}{2}\right)^{10}\right]^2 = \frac{1}{2} + \frac{1}{2^{83}} \approx 0{,}5,\ N = \left(\frac{1}{0{,}5 + 2^{83} - 0{,}5}\right)^2 = 2^{186}$$

Демек, алгоритм сызықтық криптоталдауға 4 раундтан кейін-ақ берік деген қорытынды жасауға болады.

3.6 Алгебралық критоталдау негізіндегі зерттеулер

Алгебралық криптоталдау булдік теңдеулер жүйесін қолдана отырып, шифр жұмысын сипаттаудан тұрады, криптоталдаудың жетістігі шешімнің тиімділігіне байланысты. Блоктық және ағындық шифрларға шабуылдарды, булдік алгебралық теңдеулердің үлкен жүйесін шешу есебі ретінде ұсынуға болады. Практикада алгебралық шабуылдар блоктық шифрлардың раундтар саны өте аз жағдайларына қарсы жүзеге асырылды.

Барлық алгебралық шабуылдар симметриялық шифрлаудың құпия кілтін және криптоталдаушыға белгілі деректерді байланыстыратын теңдеулер жүйесі түрінде сипаттауға негізделген. Шеннонның белгілі қағидаларына сәйкес мұндай алгоритмдерде араластыруға арналған сызықтық емес операцияларды және шашыратуға арналған сызықтық түрлендірулерді қолданады. Араластыру мен шашыратуды үсті-үстіне бірнеше рет қолдану арқылы криптографиялық беріктіктің жоғары деңгейіне қол жеткізуге болады. Заманауи симметриялық примитивтердің сызықтық емес буындары әдетте ауыстыру кестелері немесе S-блок ретінде жүзеге асырылады. Сызықтық түрлендірулер арқылы құрылған теңдеулерде келесі раундтарға өткенде айнымалылар саны көбеймейді [79-81]. Басқаша айтқанда, бірінші раундтан кейін неше айнымалы болса, n-ші раундтан кейінде сонша айнымалы болады. Ал, сызықты емес түрлендірулер үшін айнымалылар саны раунд саны артқан сайын еселеп өседі.Теңдеулер жүйесіндегі теңдеулер мен айнымалылар санына байланысты шешімді сызықтандыру [82], eXtended Linearization (XL) [83], eXtended Sparse Linearization (XSL) [84] әдістерін қолдану арқылы табуға болады. Алгебралық шабуылдардың маңызды ерекшелігінің бірі – бұл криптоталдауда қажет болатын ашық мәтін және шифрмәтін жұптарының аздығы.

Алгебралық криптоталдауға деген қызығушылық алгебралық операцияларға негізделген AES шифрын бұзу техникасын жасауға талпынудан туындады. Алайда, бұл мәселені шешуде авторлар күрделі қиындықтарға тап болды және осы уақытқа дейін қарапайым шифрларға, оның ішінде басынан алгебралық құрылымы болмаған шифрларға сәтті сынап жатыр. 2000 жылы Н.Куртуа, А.Климов, Дж.Патарин және А.Шамир кеңейтілген сызықтандыру әдісін ұсынды (eXtended Linearization). 2002 жылы Н.Куртуа мен Д.Пейпжик кеңейтілген сирек сызықтандыру алгоритмін жасады (eXtended Sparse Linearization).

XSL алгоритмі [84, p. 267; 85] еңбектерде келтірілген және бұл XL [83, p. 392] деп аталатын алгоритмнің жетілдірілген түрі болып табылады. XL алгоритмі және оның көптеген нұсқалары көп өлшемді полиномдық теңдеулер жүйелерін шешудің белгілі технологиясына негізделген сызықтандыру әдісіне негізделген [86, 87]. Бұл әдісте теңдеулер жүйесіндегі барлық бірмүшеліктерді тәуелсіз айнымалылар ретінде қарастырады және оны сызықтық алгебра әдістерінің көмегімен шешуге тырысады. Айта кететін жағдай, сызықтық тәуелсіз теңдеулер саны, теңдеулер жүйесіндегі бірмүшеліктер санынымен шамалас болған жағдайда ғана сызықтықтандыру әдісі тиімді болады. Бұл орындалмаған жағдайда, XL алгоритмі және оның басқа нұсқаларында жеткілікті теңдеулер жасап алуға тырысады.

XL әдісі қарапайым алгоритмнен тұрады: егер, ақырлы K өрісінде n айнымалыдан тұратын m квадраттық теңдеулер жүйесін қарастырсақ,

$$f_1(x_1, \dots, x_n) = 0, \dots, f_m(x_1, \dots, x_n) = 0, \tag{3.6}$$

бұл алгоритм бастапқы теңдеулерді M_i бірмүшеліктеріне берілген *D-2* дәрежесіне дейін көбейтеді және келесі теңдеуді *D* -ден жоғары емес дәрежеге дейін теңдеулер жүйесін сызықтандыру жүргізу әдісімен шешуге тырысады:

$$M_i \cdot f_j(x_1, \dots, x_n) = 0.$$

XSL алгоритмі бұдан басқаша жұмыс істейді. XL алгоритмі теңдеулер бірмүшеліктерге берілген дәрежеге дейін көбейтілсе, XSL алгоритмі теңдеулерді тек «таңдалған бірмүшеліктерге» көбейтеді. Мұндағы мақсат - жаңа теңдеулер алған кезде жаңа бірмүшеліктердің азырақ санын құру. Сонымен қатар, кез-келген жаңа бірмүшелік құрмай, сызықтық тәуелсіз теңдеулер алуға тырысатын соңғы (T_0 деп аталатын) қадам бар.

XSL алгоритмін талдау оңай емес екенін айттық және қазіргі уақытта оның алдағы бет алысы немесе бағыты туралы көп айтылмаған. Мұның бірнеше себептері бар. Біріншіден, XSL-ді мамандандырылған әдіс деп санауға болады және бұл алгоритм белгілі бір формадағы түсінік жүйесіне сүйенеді. Мысалы, сызықтық теңдеулер қабаттарын қайталай отырып, S-блоктардың асыра анықталған теңдеулер жүйесі және т.б. Екіншіден, алгоритмнің әртүрлі нұсқалары бар. [84, p. 267-282] әдебиетте ұсынылған шабуылдан айтарлықтай ерекшеленетін екі шабуыл [86, p. 201-217] әдебиетте келтірілген. Сонымен қатар, қатысатын жүйелердің көлемін ескере отырып, [84, p. 267-282] әдебиетте келтірілген тәсілдерді тексеру үшін, шағын мысалдарда да тәжірибе жасау және жүргізу өте қиын.

Сызықты емес түрлендірулер үшін булдік теңдеулер жүйесін құру. Қауіпсіздікті бағалаудың алғашқы қадамы ашық мәтін, шифрмәтін және шифрлау кілтін байланыстыратын теңдеулер жүйесін құру болып табылады. Ақпаратты қорғау түрлендірулерінің көпшілігінде теңдеулер жүйесі алмастыру

блоктары (S-блогы) үшін құрылады, өйткені бұл көбінесе оларда қолданылатын жалғыз сызықты емес түрлендіру болып табылады.

Алдымен, берілген S-блок үшін ақиқаттар кестесін құрамыз және осы кестенің көмегімен тәуелсіз квадраттық теңдеулер жүйесін табамыз. Жалпы алғанда, ауыстыру блогындағы, әр элементі 8 биттен тұратын түрлендірулерді сипаттайтын теңдеулерді келесі формуламен көрсетуге болады [88]:

$$\sum_{i,j=0}^{7} \alpha_{i,j} x_i x_j \oplus \sum_{i,j=0}^{7} \beta_{i,j} y_i y_j \oplus \sum_{i,j=0}^{7} \gamma_{i,j} x_i y_j \oplus \sum_{i=0}^{7} \delta_i x_i \oplus \sum_{i=0}^{7} \varepsilon_i y_i \oplus \tau = 0 \qquad (3.7)$$

мұндағы, $x_i, y_i,\ i = 1, \dots, 7$ – S-блоктың сәйкесінше кіріс және шығыс биттері, $x_i x_j$ – S-блоктың кіріс биттерінің көбейтіндісі, $y_i y_j$ – S-блоктың шығыс биттерінің көбейтіндісі, $x_i y_j$ – кіріс биті иен шығыс битінің көбейтіндісі, $\alpha_{i,j}, \beta_{i,j}, \gamma_{i,j}, \delta_i, \varepsilon_i, \tau -$ 0 немесе 1 мәндерін қабылдайтын коэффициенттері. Зерттеу жұмыстары шеңберінде екі айнымалының көбейтіндісін қарастыру жеткілікті.

Мысал ретінде кіріс және шығыс векторларының ұзындығы 3 битке тең S-блок үшін сызықтық тәуелсіз теңдеулер құру әдісін қарастырайық (кесте 3.21).

Кесте 3.21 – 3 биттік ауыстыру кестесі

S-блоктың кіріс және шығыс мәндері								
X	0	1	2	3	4	5	6	7
S(X)	3	7	4	6	1	0	5	2

Осы S-блок үшін құрылған ақиқаттар кестесі кесте 3.22 - де көрсетілгендей болады.

Кесте 3.22 – Берілген S-блок үшін ақиқаттар кестесі

Ауыстыру кестесінің ақиқаттар кесткінің мәндері																					
1	x_0	x_1	x_2	y_0	y_1	y_2	x_0x_1	x_0x_2	x_1x_2	y_0y_1	y_0y_2	y_1y_2	x_0y_0	x_0y_1	x_0y_2	x_1y_0	x_1y_1	x_1y_2	x_2y_0	x_2y_1	x_2y_2
1	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	1	1	1	1	0	0	0	1	1	1	0	0	0	0	0	0	1	1	1
1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
1	0	1	1	1	1	0	0	0	1	1	0	0	0	0	0	1	1	0	1	1	0
1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
1	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	1	1	0	1	0	0	0
1	1	1	1	0	1	0	1	1	1	0	0	0	0	1	0	0	1	0	0	1	0

Осы кестеге бағандар бойынша Гаустың біртіндеп жою әдісін қолданайық. Түрлендіруден кейінгі алынған нәтиже кесте 3.23 - те көрсетілген.

Кесте 3.23–Гаустың біртіндеп жою әдісін қолданғаннан кейінгі ақиқаттар кестесі

1	1	1	1	1	1	1	1	1		1	0	0	0	0	0	0	0	$1\oplus x_0\oplus y_0\oplus x_0y_0$
0	0	0	0	1	1	1	1	x_0		0	1	0	0	0	0	0	0	$x_1\oplus y_0\oplus x_0x_1\oplus x_0y_0$
0	0	1	1	0	0	1	1	x_1		0	0	1	0	0	0	0	0	$1\oplus x_0\oplus y_1\oplus x_0x_1\oplus x_0y_0$
0	1	0	1	0	1	0	1	x_2		0	0	0	1	0	0	0	0	$1\oplus x_0\oplus x_1\oplus y_1\oplus x_0y_0$
0	1	1	1	0	0	1	0	y_0		0	0	0	0	1	0	0	0	$1\oplus x_2\oplus y_0\oplus y_1\oplus x_0x_1\oplus x_0y_0$
1	1	0	1	0	0	0	1	y_1		0	0	0	0	0	1	0	0	$1\oplus x_0\oplus x_2\oplus y_0\oplus y_1\oplus x_0y_0$
1	1	0	0	1	0	1	0	y_2		0	0	0	0	0	0	1	0	x_0y_0
0	0	0	0	0	0	1	1	x_0x_1		0	0	0	0	0	0	0	1	$x_0x_1\oplus x_0y_0$
0	0	0	0	0	1	0	1	x_0x_2		0	0	0	0	0	0	0	0	$1\oplus x_0\oplus x_1\oplus x_2\oplus y_0\oplus y_1\oplus \oplus y_2$
0	0	0	1	0	0	0	1	x_1x_2	→	0	0	0	0	0	0	0	0	$1\oplus x_0\oplus x_2\oplus y_0\oplus y_1\oplus \oplus x_0x_1\oplus x_0x_2$
0	1	0	1	0	0	0	0	y_0y_1		0	0	0	0	0	0	0	0	$1\oplus x_0\oplus x_1\oplus y_1\oplus x_0x_1\oplus \oplus x_1x_2$
0	1	0	0	0	0	1	0	y_0y_2		0	0	0	0	0	0	0	0	$1\oplus x_0\oplus y_0\oplus y_1\oplus x_0x_1\oplus \oplus y_0y_1$
1	1	0	0	0	0	0	0	y_1y_2		0	0	0	0	0	0	0	0	$x_1\oplus y_0\oplus x_0x_1\oplus y_0y_2$
0	0	0	0	0	0	1	0	x_0y_0		0	0	0	0	0	0	0	0	$1\oplus x_0\oplus x_1\oplus x_0x_1\oplus y_1y_2$
0	0	0	0	0	0	0	1	x_0y_1		0	0	0	0	0	0	0	0	$x_0x_1\oplus x_0y_0\oplus x_0y_1$
0	0	0	0	1	0	1	0	x_0y_2		0	0	0	0	0	0	0	0	$1\oplus x_2\oplus y_0\oplus y_1\oplus x_0x_1\oplus x_0 \oplus y_2$
0	0	1	1	0	0	1	0	x_1y_0		0	0	0	0	0	0	0	0	$x_1\oplus x_0x_1\oplus x_0y_0\oplus x_1y_0$
0	0	0	1	0	0	0	1	x_1y_1		0	0	0	0	0	0	0	0	$1\oplus x_0\oplus x_1\oplus y_1\oplus x_0x_1\oplus \oplus x_1y_1$
0	0	0	0	0	0	1	0	x_1y_2		0	0	0	0	0	0	0	0	$x_0y_0\oplus x_1y_2$
0	1	0	1	0	0	0	0	x_2y_0		0	0	0	0	0	0	0	0	$1\oplus x_0\oplus y_0\oplus y_1\oplus x_0x_1\oplus \oplus x_2y_0$
0	1	0	1	0	0	0	1	x_2y_1		0	0	0	0	0	0	0	0	$1\oplus x_0\oplus y_0\oplus y_1\oplus x_0y_0\oplus \oplus x_2y_1$
0	1	0	0	0	0	0	0	x_2y_2		0	0	0	0	0	0	0	0	$x_1\oplus y_0\oplus x_0x_1\oplus x_0y_0\oplus x_2y_2$

Ақиқат кестесіндегі қадамдарға сәйкес келетін теңдеулер сызықты тәуелсіз және блоктық шифрлардың алгебралық анализін жасауда әрі қарай қолдануға жарамды.

Шифрды XL әдісімен алгебралық криптоталдау. (3.6) квадраттық теңдеулер жүйесін бекітілген дейік. Болжам бойынша, оның берілген K өрісінде шешімі бар, ал кейбір $D \in N$ саны XL алгоритмінің параметрі болып табылады деп алайық. XL-алгоритмінің негізгі идеясы (3.6) теңдеулер жүйесін барлық полиномдық теңдеулерді сызықтандыру жүргізу жолмен шешуге тырысу.

$$\prod_{l=1}^{k} X_{il} \cdot f_j(X_1, \dots, X_n)$$

мұндағы, $k \leq D - 2$.

$U_D - k \leq D-2$ үшін $\prod_{l=1}^{k} X_{il} \cdot f_j$ көпмүшеліктерімен туындаған K – векторлық кеңістігі болсын. [83, p. 398-413] әдебиетке сәйкес, XL-алгоритмі келесідей төрт қадамнан тұрады:

1. Көбейту: $k \leq D-2$ үшін $\prod_{l=1}^{k} X_{il} \cdot f_j$ –дің барлық көбейтіндісін алу.
2. Сызықтандыру жүргізу: дәрежесі D-дан кіші немесе тең болатын барлық X_i бірмүшеліктерін тәуелсіз айнымалы ретінде қарастырамыз және 1-қадамда алынған теңдеулерге Гаустың жою әдісін орындаймыз. Барлық мүшелері бір (арнайы) айнымалыдан тұратындары (мысалы, X_1) ең соңында жойылатындай бірмүшеліктер реттелуі керек.
3. Шешім қабылдау: 2-қадамда кем дегенде X_1 –дің дәрежесі бойынша бірөлшемді бір теңдеу береді деп болжаймыз. Осы теңдеуді ақырлы өрісте шешеміз (мысалы, Berlekamp алгоритмін қолдану арқылы).
4. Қайталау: теңдеулерді қысқартып және басқа айнымалыларды табу үшін осы үрдісті қайталау.

Есептеу қиындығын тексеру. $GF(2)$ өрісінде m квадраттық теңдеулер жүйесін шешуді қарастырайық. Жалпы жағдайда, бұл теңдеулердегі квадраттық мүшелердің саны $t \approx n^2/2$ шамасында [83, p. 410]. $D = 2, 3, \ldots$ XL алгоритмінің параметрі болсын. Алгоритм жүйедегі барлық теңдеулерді дәрежесі $D-2$ болатын айнымалыларға көбейтуге негізделген. Осылайша жаңа $R \approx \binom{n}{D-2} m$ теңдеулер аламыз. Осы теңдеулердегі бірмүшеліктердің жалпы саны шамамен $T = \binom{n}{D}$ құрайды. Осы теңдеулердің көпшілігі сызықты тәуелсіз болады деп күтіледі.

Келесі кезекте, $R = \binom{n}{D-2} m \geq \binom{n}{D} = T$ болатындай үлкен D санын таңдап аламыз. Сызықты тәуелсіз теңдеулер саны, мүшелерінің саны болатын T-дан аспайтыны анық. Егер теңдеулер жүйесінің бір шешімі болса [84, p. 267-287], онда $R \geq T$ үшін D табылады және сызықты тәуелсіз теңдеулер саны T-ге жақын болады деп күтіледі. Егер бірмүшеліктер мен сызықты тәуелсіз теңдеулер саны $(T - Free)$ арасындағы айырма үлкен болмаса, онда теңдеулер жүйесі шешілетін болады. Бірмүшеліктер саны мен сызықты тәуелсіз теңдеулер арасындағы айырма өте аз болғанда теңдеулер жүйесі оңай шешіледі. XL әдісінде қолданылатын D мәні, R ≥ T үшін D параметрінің теориялық мәніне тең немесе жақын болады деп күтіледі. Сондықтан келесі шарт орындалғанда, XL алгоритмі сәтті жүргізіледі деп болжанады:

$$R \geq T \Rightarrow m \geq \binom{n}{D} / \binom{n}{D-2} \approx n^2/D^2.$$

Бұдан, $D \approx \frac{n}{\sqrt{m}}$ және криптошабуыл қиындығы шамамен $T^\omega \approx \binom{n}{D}^\omega \approx \binom{n}{n/\sqrt{m}}^\omega$ болатынын аламыз, мұндағы ω ≤ 3 – гаусстық редукция көрсеткіші.

Жоғарыда келтірілген формуладан XL экспоненциалды болып көрінеді, дегенмен, XL-дің өте үлкен теңдеулер жүйесін шешудің нақты көрінісі туралы мәлімет аз.

Зерттеліп отырған алгоритмге талдау жасарда, берілген ауыстыру кестесін (кесте 2.1) пайдаланып, ақиқаттар кестесін құрдық. Осы кестені пайдаланып (3.7) түрдегі тәуелсіз квадраттық теңдеулер жүйесін құрдық. Ескерте кететін тағы бір жағдай, егер алынған (3.7) жүйенің теңдеулері көпөлшемді квадраттық болса, онда бұл шабуыл «MQ шабуыл» деп аталады.

«Qamal» шифрлау алгоритмінде қолданылған S-блок үшін тәуелсіз квадраттық теңдеулер үшін $r = 39$ және бірмүшеліктердің саны $t = 137$ (В.2).

XL шабуылының қиындығын есептеу үшін қажетті параметрлерді табайық. $m = r \cdot B \cdot N_r + r \cdot \frac{L_k - H_k}{s} - n$ айнымалысы бар квадраттық теңдеулер саны. Мұндағы, L_k – сызықты тәуелсіз биттердің саны; $H_k = 128$ – бастапқы кілттің биттерінің саны.

$L_k = H_k + 16 \cdot N_r \cdot s = 128 + 16 \cdot 8 \cdot 8 = 1152,$

$m = 39 \cdot 16 \cdot 8 + 39 \cdot \frac{1152-128}{8} = 9984,$

$n = s \cdot B \cdot (N_r - 1) + L_k = 8 \cdot 16 \cdot 7 + 1152 = 2048$– бірмүшеліктер саны;

$D = \frac{n}{\sqrt{m}} = \frac{2048}{\sqrt{9984}} \approx 20.$

Онда, «Qamal» шифрлау алгоритміне жүргізілетін XL шабуылының есептеу қиындығы $T^{\omega} = \binom{n}{D}^{\omega} = \binom{2048}{20}^{2,376} \approx 2^{375}$-не тең.

Алгоритмге XSL әдісімен криптоталдау жүргізу. XSL алгоритмі де негізгі төрт кезеңнен тұрады:

1. Алгоритмнің алдағы қадамдарында қолданылатын бірмүшеліктер мен теңдеулердің нақты жиынтығын таңдау арқылы өңделеді.

2. P параметрінің мәнін еркін таңдау және алдыңғы кезеңде таңдалған теңдеулерді $(P - 1)$ бірмүшеліктерге көбейту. Бұл XSL шабуылының негізі болып табылады және элементтері алдында таңдалған бірмүшеліктердің көбейтіндісі болатын көп теңдеулер алу керек.

3. Кейбір таңдалған теңдеулерді айнымалыларға көбейтетін T әдісін қолдану. Мұндағы мақсат – жаңа бірмүшеліктер алмай-ақ жаңа теңдеулер құру. Бұл қадам сызықтандыру жүргізу үшін қажетті айнымалылар санымен, жүйеде жеткілікті сызықты тәуелсіз теңдеулер болғанға дейін орындалады.

4. Әрбір бірмүшелікті жаңа айнымалы ретінде қарастыра отырып сызықтандыру жүргізу және Гаусс әдісін қолдану нәтижесінде теңдеулер жүйесінің шешімі алынуы керек.

r теңдеулері мен t мүшелері бар шифрдың әрбір S-блогы үшін бастапқы теңдеулерден бастап, шифрдың құпия кілтін толығымен анықтайтын квадраттық теңдеулер жүйесін құрамыз.

Жоғарыда айтылғандай, XL алгоритмінде дәрежесі (D-2)-ден аспайтын барлық мүмкін болатын бірмүшеліктер, жүйедегі әрбір теңдеуге көбейтіледі. XSL әдісінде оның орнына теңдеулер жүйесі тек таңдалған бірмүшеліктерге көбейту жоспарланады және басқа теңдеулерде пайда болған бірмүшеліктерді көбейту қолданған жөн. R ≥ T үшін теңдеулер саны бірмүшеліктер санымен сәйкес келеді және әрбір мүшесіне жаңа айнымалы қосу арқылы теңдеулер жүйесі шешіледі деп күтіледі [83, p. 409].

Әрбір «Белсенді S-блок» үшін келесі түрдегі r теңдеу құруға болады:

$$0 = \sum \alpha_{ijk} X_{ij} Y_{ik} + \sum \beta_{ij} X_{ij} + \sum \gamma_{ij} Y_{ij} + \delta$$

Бұл теңдеулерде пайда болатын бірмүшеліктердің саны аз, оны – t-деп белгілейік. Олардың көпшілігі $XX_{ij}Y_{ik}$ түрінде болады. Осы себепті X_{ij} және Y_{ik} айнымалылары сақталады.

S – осы криптошабуылға қатысатын S-блоктың саны болсын. Егер раундтық кілттерді алуды ескермейтін болсақ, онда $N_r + 1$ раундтық шифрлау үшін S-тің мәні $B \cdot N_r \cdot (N_r + 1)$-ке тең болады [85, p. 267-287]. Бұл шабуылдың критикалық параметрі P болады. Осы әдіспен құрылған теңдеулердің жалпы саны жуықтап:

$$R \approx r \cdot S \cdot t^{P-1} \cdot \binom{S-1}{P-1}.$$

Ал, теңдеулердегі айнымалылардың жалпы саны шамамен: $R \approx t^P \cdot \binom{S}{P}$.

XSL шабуылының есептеу қиындығы келесі формуламен анықталады [84, p. 279]:

$$T^\omega \approx t^{\omega P} \cdot \binom{S}{P}^\omega \approx (tS)^{\omega P} \approx (t \cdot B \cdot N_r^2)^{\omega P} \approx \left(\frac{t}{s} \cdot Bs \cdot N_r^2\right)^{\omega P} \approx \left(\frac{t}{s}\right)^{\omega P} \cdot (B \cdot s \cdot N_r^2)^{\omega P} \approx$$
$$\approx (t/s)^{\omega P} \cdot (\text{Блок өлшемі})^{\omega P} \cdot (\text{Раунд саны})^{\omega P}.$$

Зерттеліп отырған алгоритм үшін есептеу жүргізейік. «Qamal» шифрлау алгоритмі үшін (В.3) - ші формуладан:

1) бірмүшеліктер саны – $t = 137$,
2) S-блоктың ұзындығы – $s = 8$,
3) тәуелсіз теңдеулер саны – $r = 23$,
4) бір раундта қатысатын S-блоктар саны $B = 16$,
5) раунд саны $N_r = 8$.

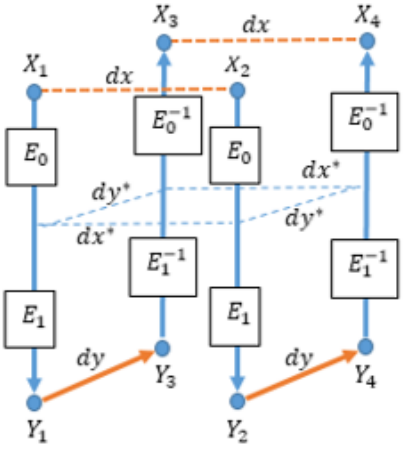
$$T^\omega \approx \left(\frac{t}{s}\right)^{\omega\left[\frac{t}{r}\right]} \cdot (B \cdot s \cdot N_r^2)^{\omega\left[\frac{t}{r}\right]} = \left(\frac{137}{8}\right)^{2{,}376 \cdot \left[\frac{137}{39}\right]} \cdot (16 \cdot 8 \cdot 8^2)^{2{,}376 \cdot \left[\frac{137}{39}\right]} \approx 2^{165}.$$

XSLшабуылының есептеу қиындығы $\omega = 2{,}376$ параметрі үшін бағаланды.

3.7 Бумеранг шабуыл нәтижелері

Бумеранг әдісі дифференциалдық криптоталдаудың жетілдірілген түрі болып табылады. Одан негізгі айырмашылығы: а) талдау ашық мәтіндер және оларға сәйкес шифрмәтіндердің квартеттер жиыны арқылы жүзеге асырылады; б) екі сатылы E_0 және E_1 шифрлау үдерісін пайдалануына байланысты ашық мәтіндегі өзгерістер шифрдың тек бір бөлігін қамтуы мүмкін. Нәтижесінде, белгілі бір айырмашылығы бар ашық және шифрмәтіндердің алынған квартеттерінің жиынтығын талдау арқылы кілтті немесе жоғары ықтималдықпен

оның бөлігін таңдауға болады [89]. Бумеранг шабуылын жүргізудің сұлбасы Сурет 3.11 - де көрсетілген.



Сурет 3.11 – Бумеранг шабуылын жүргізу сұлбасы

Криптотошабуыл жүргізу келесі қадамдардан тұрады:

1. N раундтық алгоритм тең екі $N/2$ раундтық бөлікке бөлінеді.

2. E_0 – алгоритмдегі шифрлаудың бірінші бөлігі. Квартет алу үшін, XOR биттік операциясы арқылы, айырымдары қандайда бір dx болатын X_1 және X_2 екі ашық мәтіндер таңдалады. Осы X_1 және X_2 мәтіндерге E_0 шифрлауын қолдану арқылы, шығыстарында $dx^* = E_0(X_1) \oplus E_0(X_2)$ айырымын аламыз.

3. Келесі кезекте X_1 және X_2 мәтіндеріне шифрлаудың E_1 бөлігін қолдану арқылы түрлендіру жүргіземіз. Нәтижесінде екі жаңа $Y_1 = E_1\big(E_0(X_1)\big)$ және $Y_2 = E_1\big(E_0(X_1)\big)$ шифрмәтіндері анықталады.

4. Y_1 және Y_2 шифрмәтіндердің көмегімен басқа екі Y_3 және Y_4 шифрмәтіндері анықталады. Олардың айырымдары болатын dy арасындағы байланыс келесідей болады: $Y_3 = Y_1 \oplus dy$ және $Y_4 = Y_2 \oplus dy$.

5. Әрі қарай, ашық мәтіндер квартетін құру үшін шифрлау үдерісі кері бағытта жүзеге асырылады: Y_3 және Y_4 шифрмәтіндеріне кері E_1^{-1} функциясы қолданылады, әрі $E_1^{-1}(Y_1) = E_0(X_1)$ және $E_1^{-1}(Y_2) = E_0(X_2)$. Олай болатын болса, $E_1^{-1}(\mathrm{Y}_3) \oplus E_0(X_1) = E_1^{-1}(\mathrm{Y}_4) \oplus E_0(X_2) = dy^*$.

6. Соңында Y_3 және Y_4 шифрмәтіндері X_3 және X_4 ашық мәтіндеріне келесідей кері шифрланатын болады: $X_3 = E_0^{-1}\big(E_1^{-1}(Y_3)\big)$ және $X_4 = E_0^{-1}\big(E_1^{-1}(Y_4)\big)$, сонымен бірге $X_1 \oplus X_2 = X_3 \oplus X_4 = dx$.

Ашық және жабық мәтіндердің жарамды квартеті деп, X_1 және X_2 кіріс мәтіндерінің берілген dx айырымы үшін бумеранг сұлбасы бойынша алынған X_3 және X_4 ашық мәтіндердің айырымымен бірдей болатын $(\mathrm{X}_1, \mathrm{Y}_1), (\mathrm{X}_2, \mathrm{Y}_2), (\mathrm{X}_3, \mathrm{Y}_3)$ және $(\mathrm{X}_4, \mathrm{Y}_4)$ квартеті аталады [90, 91].

Зерттеудің негізгі түйіні - ашық квартеттер жиынтығын және оларға сәйкес шифрмәтіндерді табу. Шифрлау алгоритмінің сипаттамасында келтірілгендей алгоритмінің мәліметтерді шифрлау жүйесі келесі түрлендірулерден тұрады: бит бойынша қосу (XOR), S-блок ауыстыруды, Mixer1 және Mixer2 араластыру түрлендірулері.

Бастапқыда шифрда қолданылған әрбір түрлендірулерге бумеранг шабуылын жүргізу үшін қажетті квартеттерді табу мүмкіндігін қарастырамыз, яғни бір-біріне тәуелсіз жұмыс істеген жағдайды қараймыз. Шифрлау блогының стандартты ұзындығы - 128 бит (16 байт). Бумерангтың E_0 және E_1 кезеңдеріне бөлу, түрлендірулерді есептеудің күрделілігіне байланысты. Әрбір түрлендіруге қатысты бумеранг шабуылының квартеттерінің болу ықтималдығын анықтау үшін есептеу алгоритмі құрылып, бағдарламалық іске асырылды. Осы бағдарламаның көмегімен табылған мәліметтерді төменде келтіретін боламыз.

Бит бойынша қосу операциясы. Талдау үшін шифрлаудың барлық 8 айналымы қарастырылған, квартеттерді қалыптастыру кезеңдері E_0 және E_1 симметриялы түрде төрт раундтан бөлінген. Бумеранг әдісінің алгоритмі бойынша келесі операциялар орындалады: ашық X_1 және X_2 мәтіндері алынады. E_j-дің құрылымдық кезеңдері ретінде ашық мәтінді құпия немесе раундтық кілтпен қосу әрекеті қарастырылады: $E_j(X_j) = X_j \oplus k_j$, мұндағы $j = 1,2$. X_1 мен X_2 арасындағы $X_1 \oplus X_2$ айырмы dx арқылы белгіленеді. Ары қарай, сұлбаға сәйкес $Y_1 = E_1(E_0(X_1)) = X_1 \oplus k_j$ және $Y_2 = E_1(E_0(X_2)) = X_2 \oplus k_j$ есептеледі, мұндағы k_j - раундтық кілттерді бит бойынша қосу арқылы алынған мән. Демек, Y_3 және Y_4 шифрмәтіндері келесі түрде болады: $Y_3 = X_1 \oplus k_j \oplus dy$, $Y_4 = X_2 \oplus k_j \oplus dy$. $E_0 = E_0^{-1}$ екенін ескере отырып, квартеттің компоненттері келесідей есептеледі: $X_3 = X_1 \oplus k_j \oplus dy \oplus k_j$; $X_4 = X_2 \oplus k_j \oplus dy \oplus k_j$. Ал олардың айырымдары тең болады:

$X_3 \oplus X_4 = X_1 \oplus k_j \oplus dy \oplus k_j \oplus X_2 \oplus k_j \oplus dy \oplus k_j = X_1 \oplus X_2$.

Мысалдар.

Төмендегі мысалда қажетті квартеттерді құру толығырақ көрсетіледі (деректер он алтылық санау жүйесінде 0x-белгілеуінсіз берілген):

X_1 = 87, E9, 5E, 62, 5E, AA, 78, AC, CA, 54, 8C, 58, 92, 0C, 5B, 0F.

X_2 = 9E, 8E, AA, A3, 09, 4C, 96, 7D, DD, 87, 9D, DA, 01, 2D, 32, 3D.

dx= $X_1 \oplus X_2$=19, 67, F4, C1, 57, E6, EE, D1, 17, D3, 11, 82, 93, 21, 69, 32.

Сәйкесінше шифрмәтіндер:

Y_1=41, 22, DF, 8D, 94, A2, CB, C0, 62, BB, BE, 50, DC, 4E, F3, 90.

Y_2=58, 45, 2B, 4C, C3, 44, 25, 11, 75, 68, AF, D2, 4F, 6F, 9A, A2.

dy айырымдары кездейсоқ түрде таңдалады:

dy = AB, CD, EF, 98, 76, 54, 32, 10, FE, DC, BA, 01, 23, 45, 67, 89.

Бұдан:

$Y_3 = Y_1 \oplus dy$ = EA, EF, 30, 15, E2, F6, F9, D0, 9C, 67, 04, 51, FF, 0B, 94, 19;

$Y_4 = Y_2 \oplus dy$= F3, 88, C4, D4, B5, 10, 17, 01, 8B, B4, 15, D3, 6C, 2A, FD, 2B;

Келесі кезекте X_3 және X_4 табылады:

X_3 =2C, 24, B1, FA, 28, FE, 4A, BC, 34, 88, 36, 59, B1, 49, 3C, 86.

X_4 =35, 43, 45, 3B, 7F, 18, A4, 6D, 23, 5B, 27, DB, 22, 68, 55, B4.

Соңында олардың айырымдары есептеліп X_1 және X_2 ашық мәтіндерінің айырымымен салыстырылады: *$X_3 \oplus X_4$= 19, 67, F4, C1, 57, E6, EE, D1, 17, D3, 11, 82, 93, 21, 69, 32.*

Бұдан, *$X_1 \oplus X_2 = X_3 \oplus X_4$* екендігіне көз жеткіздік.

Осылайша, «Qamal» шифрлау алгоритмінің биттік қосу (XOR) операциясына қатысты, $p = 1$ ықтималдықпен раундтар санына қарамастан, ашық және жабық мәтіндердің квартеттерін құруға болады.

S-блок түрлендіруі. Шифрлау алгоритмінде қолданылған S-блок ауыстыруына (кесте 2.1) қатысты квартеттерді алу мүмкіндігіне талдау жүргізілді. Бұл жағдайда, квартеттерді құруға жеңілірек болу үшін шифрлаудың тек екі раунды қолданылады. Бумеранг шабуылының бірінші E_0 кезеңінде түрлендірудің 1-ші раундын және E_1 - 2-ші раундын қамтиды. Бумеранг шабуылының шарттарын қанағаттандыратын квартеттердің санын дәл анықтау үшін, кіріс деректері X_1, X_2 және dy болатын және параметрлері 0-ден 255 аралығында өзгеретін мәліметтерді жоғарыда аталған бағдарламалық жасақтаманың көмегімен тексерілді. Мүмкін болатын таңдаулардың жалпы саны 256^3-ға немесе 2^{24}-не тең. Бағдарламаның көмегімен жүргізілген есептеу нәтижелері бойынша тек 63 960 квартет табылғандығы анықталды, бұл 2^{16}санына жақын.

Осылайша, екінші раундтан кейін S-блок ауыстыру үшін квартеттің табылу ықтималдығы 2^{-8}-ге тең. Осы квартеттерді анықтау кезінде келесі шарттар ескерілді:

а) квартетті құраушы барлық X_1, X_2, X_3 және X_4 үшін: $X_i \neq X_j$, мұндағы $i, j = 1,2,3,4$.

ә) dy айырымы үшін: dy ≠ 0.

Келесі үш мысалда квартеттер құру үдерісі көрсетілген (кесте 3.24):

Кесте 3.24 – S-блок үшін бумеранг әдісінің мысалы

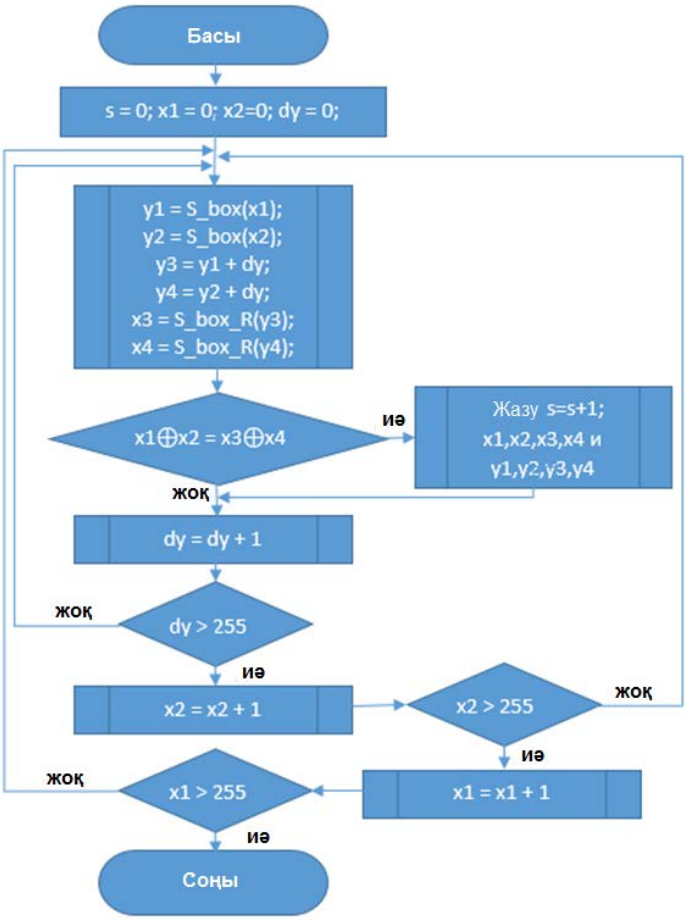
Параметрлері		1-мысал	2-мысал	3-мысал
Бастапқы параметрлері	X_1	04	6E	5E
	X_2	AE	9D	10
	dy	82	9B	EF
Есептелетін параметрлері	$Y_1=E_1\left(E_0(X_1)\right)$	E8	D7	6E
	$Y_2= E_1\left(E_0(X_2)\right)$	B9	B1	93
	$Y_3= Y_1 \oplus dy$	6A	4C	81
	$Y_4= Y_2 \oplus dy$	3B	2A	7C
	$X_3=E_0^{-1}\left(E_1^{-1}(Y_3)\right)$	E4	6D	F5
	$X_4=E_0^{-1}\left(E_1^{-1}(Y_4)\right)$	4E	9E	B3
Тексерілетін параметрлері	$dx=X_1 \oplus X_2$	AA	F3	4E
	$X_3 \oplus X_4$	AA	F3	46
Шарттарды тексеру	$X_1 \oplus X_2 = X_3 \oplus X_4$	+	+	-

Ескерту: E_i және E_i^{-1} түрлендіруге сәйкесінше S-блок және кері S-блок түрлендірулері орындалады, i - бұл раунд нөмірі (бумеранг кезеңдері).

Қарастырылған мысалдарда тексеру нәтижелері бойынша келесі квартеттер құрылды:

1-мысалда: ашық мәтіндер - 04, AE, E4, 4E және шифрмәтіндер - E8, B9, E4, 4E;
2-мысалда: ашық мәтіндер - 6E, 9D, 6D, 9E және шифр мәтіндер - D7, B1, 4C, 2A. Үшінші мысалда берілген бастапқы параметрлер бойынша квартеттер құру мүмкін болмады.

S-блок ауыстыруына қатысты рұқсат етілген квартеттер санын анықтауға арналған бағдарламалық жасақтама сурет 3.12 - де көрсетілген болк-схемаға сәйкес жүзеге асырылады.



Сурет 3.12 – S-блок бойынша квартеттер іздеу алгоритмінің блок-схемасы

Mixer2 араластыру операциясы. Талдау жүргізу үшін шифрлаудың барлық 8 раунды қарастырылады. E_0 және E_1 кезеңдері төрт раундтан бөлінген. $E_i, i = 1,2$ –дің кезеңдері ретінде Mixer2-нің берілген массивтің жолдық түрлендірулері қарастырылады. Mixer2 түрлендіруіне қатысты, алгоритмге сәйкес бумеранг әдісінің қабылданатын квартеттердің санын анықтайтын бағдарлама жасалды. Бұл түрлендіруді іске асыруда массив жолының барлық төрт элементіде қолданылатындықтан, толық теру саны 2^{96} –ге сәйкес келеді. Олардың әрбірінде төрт элементтен бар: $X_1 = \{x_{11}, x_{12}, x_{13}, x_{14}\}$, $X_2 = \{x_{21}, x_{22}, x_{23}, x_{24}\}$ және $dy = \{dy_1, dy_2, dy_3, dy_4\}$, мұндағы әр элемент 0-ден 255-ке дейінгі 256 мән қабылдайды.

Мәліметтерді эксперименттік талдау нәтижелері бойынша, $p = 1$ ықтималдықпен раундтар санына қарамастан, «Qamal» шифрлау алгоритмінің Mixer2 түрленуіне қатысты, бумеранг шабуылы үшін ашық және жабық

мәтіндердің қажетті квартеттерін құруға болатындығы анықталды және оның мысалдары кесте 3.25 - те келтірілген.

Кесте 3.25 – Mixer2 түрлендіруі үшін бумеранг әдісінің мысалдары

Параметрлері		1-мысал	2-мысал	3-мысал	4-мысал
Бастапқы параметрлері	X_1	*87,5e,ca,92*	*e9,aa,54,0c*	*5e,78,8c,5b*	*62,ac,58,0f*
	X_2	*9e,09,dd,01*	*8e,4c,87,2d*	*aa,96,9d,32*	*a3,7d,da,3d*
	dy	*ab,76,fe,23*	*cd,54,dc,45*	*ef,32,ba,67*	*98,10,01,89*
Есептелетін параметрлері	$Y_1=E_1(E_0(X_1))$	*90,8f,88,b3*	*13,87,b0,31*	*c1,48,d0,95*	*42,89,f7,e2*
	$Y_2= E_1(E_0(X_2))$	*00,a3,49,dd*	*fe,06,27,df*	*0e,f8,85,f2*	*ca,5b,fe,81*
	$Y_3= Y_1 \oplus dy$	*3b,f9,76,90*	*de,d3,6c,74*	*2e,7a,6a,f2*	*da,99,f6,6b*
	$Y_4= Y_2 \oplus dy$	*ab,d5,b7,fe*	*33,52,fb,9a*	*e1,ca,3f,95*	*52,4b,ff,08*
	$X_3= E_0^{-1}(E_1^{-1}(Y_3))$	*dc,76,46,d6*	*a9,9e,b5,42*	*4f,68,24,a8*	*01,d6,d1,02*
	$X_4= E_0^{-1}(E_1^{-1}(Y_4))$	*c5,21,51,45*	*ce,78,66,63*	*bb,86,35,c1*	*c0,07,53,30*
Тексерілетін параметрлері	$dx=X_1 \oplus X_2$	*19,57,17,93*	*67,e6,d3,21*	*f4,ee,11,69*	*c1,d1,82,32*
	$X_3 \oplus X_4$	*19,57,17,93*	*67,e6,d3,21*	*f4,ee,11,69*	*c1,d1,82,32*
Шарттарды тексеру	$X_1 \oplus X_2 = X_3 \oplus X_4$	+	+	+	+

Mixer1 араластыру операциясы. Талдау жүргізу үшін шифрлаудың 2 раунды қарастырылады. E_0 және E_1 кезеңдері бір раундтан бөлінген. Жоғарыда сипатталғандай, Mixer1 түрлендіруі 4x4, 6x4 немесе 8x4 матрицалық түрде жазылған деректерді баған бойынша түрлендіруді жүзеге асырылады. Ашық және шифрланған мәтіндердің қажет квартеттерін құру ықтималдығы жоғарыда аталған бағдарламаның көмегімен анықталды. Шифрлау блогының ұзындығы 16 байт болған жағдайда, бумеранг шабуылын құраушылардың әрқайсысында төрт элемент болады: $X_1 = \{x_{11}, x_{12}, x_{13}, x_{14}\}$, $X_2 = \{x_{21}, x_{22}, x_{23}, x_{24}\}$ және $dy = \{dy_1, dy_2, dy_3, dy_4\}$, әрқайсысы [0..255] аралығында өзгереді. Нәтижесінде нақты сандық мәнін анықтау үшін 2^{96} операция орындау қажет. Бірақ бұл есептеулер өте көп уақыт қажет ететіндіктен, X_2 мен dy-тің бектіліген параметрлері үшін 2^{24} операцияға дейін жүргізілді.

Сонымен, екі раундтан кейін бекітілген сегіз мән үшін және X_1 мен dy –тің параметрінің, әрқайсысы 256 мән қабылдайтын барлық мүмкін болатын сегіз нұсқалары үшін, бумеранг шабуылына қажетті квартеттерді құру ықтималдығы $1/2^8$-ге тең болады. Бекітілген компоненттер санының азаюымен квартеттердің пайда болу ықтималдығы да азаятыны анық. Сондықтан компоненттерді толық теру жағдайында квартеттерді құру ықтималдығы тым аз болады.

«Qamal» алгоритмінде қолданылған негізгі операциялардың жоғарыда аталған қасиеттеріне сүйене отырып, шифрлаудың екі раунды үшін бумеранг әдісіне қажетті квартеттерінің санын алудың ықтималдығын анықтайық. Егер алгоритмнің операцияларын тұтастай қарастырсақ және Mixer2 түрлендіруінде $p = 1$ ықтималдықпен қажетті квартет құруға болатын, сонымен қатар оның бір

байтындағы өзгеріс матрица қатарының барлық төрт байтына тарататындығын ескерсек, онда Mixer1 түрлендіруі бойынша квартет құру ықтималдығы азаяды. Төрт байттың бәрінің шабуылға қажетті мәнге айналу ықтималдығы $\left({}^{1}/_{2^8}\right)^4 = {}^{1}/_{2^{32}}$. Нәтижесінде екінші раундтан кейін жарамды квартетті алудың жалпы ықтималдығы $p = {}^{1}/_{2^{32}} \cdot {}^{1}/_{2^8} = 2^{-40}$ болады.

Екінші раундтан кейін қажетті квартеттерді алу ықтималдығы Mixer1 түрлендіруі үшін мүмкін болатын 2^{96}-ден тек 2^{24}операцияны есептеу арқылы алынғанын ескерсек, операциялар санының одан әрі өсуі p ықтималдығын тез төмендетеді. Нәтижесінде, раундтар санын сегізге дейін арттырғанда және Mixer1-дегі операциялар саны 2^{96}-ға көбейген кезде, мәтіндердің қажетті жұптарын табу ықтималдығы өте төмен, яғни толық терудің ықтималдығынан (2^{-128}) төмен болды. Бұдан шығатын қорытынды, зерттеліп отырған алгоритмге бумеранг шабуылын қолдану тиімсіз немесе шифрлау алгоритмі бумеранг шабуылына берік.

Алгоритмде қолданылған түрлендірулердің 3.4 бөлімде көрсетілген дифференциалдық қасиеттерін пайдаланып бумеранг шабуылының бағалауын көрсетуге де болады. Бумеранг әдісі бойынша жүргізілген шабуылдың ықтималдығын келесідей бағағаланады: $p_0 \geq P(a \to b) * P(c \to d) = pq$ [90, p. 1-15]. Мұнда, $E_0 - p$-ға тең ықтималдықпен $a \to b$ дифференциалдық сипаттамаға ие, ал $E_1 - q$-ға тең ықтималдықпен $c \to d$ дифференциалдық сипаттамаға ие. Жоғарыда көрсеткендей бұл ықтималдықтардың әрқайсысы үшінші раундтан кейін 2^{-120}-не тең. Олай болса, $p_0 \geq 2^{-240}$.

Жоғарыда айтылғандай, бумеранг шабуылын практикада қолдану, мәліметтердің үлкен көлемін және ұзақ уақыттық есептеулер жүргізуді қажет етеді. Қазіргі кезде іс жүзінде бумеранг шабуылы негізінен раунд саны аз шифрларға қолданылады, сондықтан алгоритм теориялық жетістік болып табылады.

3.8 ПЕПСЖ негізінде құрылған шифрлау алгоритміне криптоталдау

Позициялық емес полиномдық санау жүйесі бойынша кілтті пайдаланудағы криптоталдауды жеке қарастырайық. Ең бірінші, алгебралық талдау жүргізейік. Алгебралық әдістер ақпаратты түрлендіру алгоритмінің алгебралық қасиеттерін пайдалануға негізделген. Алгоритмдердің статистикалық әдістерге беріктігі ашық мәтіндер мен тиісті шифр мәтіндер туралы жинақталған ақпарат көлеміне байланысты екендігін көрсеттік. Әдетте алгебралық әдістер бірдей кілтті қолданған жағдайда да көп статистикалық мәндерді қажет етпейді. Бұл әдістердің негізі мақсаты, ашық мәтін мен кілт элементтерін айнымалы ретінде алып сызықтық теңдеулер жүйесін құру болып табылады.

Позициялық емес полиномдық санау жүйесіндегі көбейтуге арналған алгебралық криптоталдау жеке қарастырылды. ПЕПСЖ-дегі көбейту үшін бөліктеп шабуылдау қолданылды [12, p. 250-258; 58, p. 198-208]. ПЕПСЖ негізіндегі шифрлау сұлбасында бір ғана келтірілмейтін көпмүшелік үшін кілт,

ашық мәтін және шифрмәтіндерді байланыстыратын теңдеулер жүйесі келесідей түрде болады:

$$\begin{cases} y_{n-1}d_{n-1} \oplus k_n s_{n-2} = 0 \\ y_{n-1}d_{n-2} \oplus y_{n-2}d_{n-1} \oplus k_n s_{n-3} \oplus k_{n-1}s_{n-2} = 0 \\ \dots \\ y_{n-1}d_1 \oplus y_{n-2}d_2 \oplus \dots \oplus y_1 d_{n-1} \oplus k_n s_0 \oplus k_{n-1}s_1 \oplus \dots \oplus k_2 s_{n-2} = 0 \\ y_{n-1}d_0 \oplus y_{n-2}d_1 \oplus \dots \oplus y_0 d_{n-1} \oplus k_{n-1}s_0 \oplus \dots \oplus k_1 s_{n-2} = x_{n-1} \\ y_{n-2}d_0 \oplus y_{n-3}d_1 \dots \oplus y_0 d_{n-2} \oplus k_{n-2}s_0 \oplus k_{n-3}s_1 \oplus \dots \oplus k_0 s_{n-2} = x_{n-2} \\ \dots \\ y_2 d_0 \oplus y_1 d_1 \oplus y_0 d_2 \oplus k_2 s_0 \oplus k_1 s_1 \oplus k_0 s_2 = x_2 \\ y_1 d_0 \oplus y_0 d_1 \oplus k_1 s_0 \oplus k_0 s_1 = x_1 \\ y_0 d_0 \oplus k_0 s_0 = x_0 \end{cases} \quad (3.8)$$

Мұндағы $y = (y_{n-1}, y_{n-2}, \dots, y_2, y_1, y_0)$ – осы шифрмәтін тізбегі, $x = (x_{n-1}, x_{n-2}, \dots, x_2, x_1, x_0)$ – ашық мәтін тізбегі, $k = (k_n, k_{n-1}, \dots, k_2, k_1, k_0)$, $d = (d_{n-1}, d_{n-2}, \dots, d_2, d, d_0)$ және $s = (s_{n-2}, s_{n-3}, \dots, s_2, s_1, s_0)$ – белгісіз айнымалылар тізбегі.

Бұл жағдайда алдыңғы түрлендірулердің нәтижесі ПЕПСЖ-нің кіріс деректері ретіндегі кездейсоқ тізбектер болып табылады. [12, p. 250-258] жұмыста бір циклден кейінгі аралық нәтиженің әрбір биті ашық мәтін мен кілттің әрбір мәндеріне тәуелді екендігі көрсетілген. Ашық мәтінге немесе кілтке енгізген ең аз өзгерістердің өзі шамамен 50% өзгеруіне әкеледі (лавиндік әсер). Бұл жағдайда бөліктеп шабуылдау жүргізе алмаймыз.

Криптоталдаушылар алдымен шифрмәтіннің әлсіз жақтарын іздейді, яғни модуль бойынша көбейтілген көпмүшеліктердің көбейтіндісінің дәрежесі жұмыс негіздерінің дәрежесінен кем болатын жағдайлар қарастырады. Бұл жағдайда (3.8) теңдеулер жүйесінде k_i және s_i айнымалылары жойылып кетіп, тек d_i және x_i айнымалылары қалады. Бұл криптоталдаушының жұмысын жеңілдетеді. Сондықтан кілт таңдағанда бұл фактілерді ескеру қажет.

Егер келтірілмейтін көпмүшелердің коэффициенттері k_n және k_0 әрқашан 1-ге тең болатындығын ескерсек, онда (3.8) теңдеулер жүйесіндегі s_i айнымалыларын келесі түрге өрнектеуге болады:

$$s_{n-2} = y_{n-1} * d_{n-1},\ s_{n-3} = y_{n-1} * d_{n-2} \oplus y_{n-2} * d_{n-1} \oplus k_{n-1} * y_{n-1} * d_{n-1},$$

Формуланы жалпы түрде, келесідей рекурсивті түрде жазуға болады:

$$s_i = S_i(c_{n-1}, c_{n-2}, \dots, c_{i+1}, d_{n-1}, d_{n-2}, \dots, d_{i+1}, k_{n-1}, k_{n-2}, \dots, k_{i+2}) = S_i \quad (3.9)$$

Егер (3.9) теңдеуін (3.8) теңдеулер жүйесіне апарып қойсақ, онда $2n-1$ теңдеудің орнына n теңдеу аламыз. Осылайша, d_i, k_i және x_i айнымалылардан тұратын n теңдеу аламыз. Сонымен қатар, келтірілмейтін көпмүшеліктердің коэффициенттерінің қосындысы 1-ге тең екенін де ескереміз.

$$\begin{cases} f(y,d,k) = x_{n-1} \\ f_2(y,d,k) = x_{n-2} \\ \dots \\ f_n(y,d,k) = x_0 \\ k_{n-1} \oplus \dots \oplus k_1 = 1 \end{cases} \quad (3.10)$$

Бұл теңдеулер жүйесінің көптеген шешімдері бар. Солардың ішінен бір ақиқат шешімді табу керек. Бұл шешімді табу үшін әртүрлі әдістерді қолдануға болады. Ашық мәтінді пайдаланып, теңдеулер жүйесінің шешімін табудың екі әдісін қарастырамыз.

Бірінші әдісте, егер (3.10) теңдеулер жүйесінде белгілі заңдылықтары бар k және x айнымалыларын таңдап алсақ (ашық мәтін), онда айнымалылары $d = (d_{n-1}, d_{n-2}, \dots, d_2, d_1, d_0)$ болатын сәйкес сызықтық теңдеулер жүйесін аламыз. Мұндағы теңдеулер саны айнымалылар санына тең. Егер бұл мүмкін болмаса, келесі қадамға көшеміз. Осы әдістің күрделілігі келтірілмейтін көпмүшеліктердің дәрежесіне сәйкес өсетін болады (кесте 3.26).

Кесте 3.26 – Кілт ұзындығына байланысты нұсқаларды таңдау саны

Ұзындығы	Келтірілмейтін көпмүшеліктер саны	Ашық мәтін таңдау саны	Барлық мүмкін болатын нұсқалар саны	Ықтималдық
3	2	8	16	0,0625
4	3	16	64	0,015625
5	6	32	256	0,003906
6	9	64	832	0,001202
7	18	116	2920	0,000342
8	30	116	6400	0,0000156
9	56	232	19392	5,16E-05
10	99	464	65328	1,53E-05
11	186	928	237936	4,2E-06
12	335	1856	859696	1,16E-06
13	630	3712	3198256	3,13E-07
14	1161	7424	11817520	8,46E-08
15	2182	13456	41175812	2,428E-08
16	4080	13456	96078992	1,041E-08

$m = \sum_{i=1}^{s} m_i$ – кілт ұзындығы болса, яғни келтірілмейтін көпмүшеліктердің дәрежесі m_i болса, онда тиісінше шифрмәтіндегі m-биттік қадамдар арқылы тексереміз. Белгілі бір ұзындыққа дейін келтірілмейтін көпмүшелерді теру қажет.

Егер ашық мәтін ASCII кодтарында ұсынылған ағылшын тіліндегі мәтін, онда d_i және k_iайнымалыларынан тұратын сызықтық емес теңдеулер жүйесін құрамыз. Жүйедегі теңдеулер саны ашық мәтіннің ұзындығына байланысты. Бұдан әрі ашық мәтінді қолданбай, келтірілмейтін көпмүшеліктерді теру ғана қалады және нақты мәндері кесте 3.27 - де көрсетілген.

Кесте 3.27 – Келтірілмейтін көпмүшеліктердің нұсқаларын таңдау саны

Кілт ұзындығы	Келтірілмейтін көпмүшеліктер саны	Барлық мүмкін болатын нұсқалар	Ықтималдық
3	2	4	0,25
4	3	7	0,1429
5	6	13	0,0769
6	9	22	0,0455
7	18	40	0,025
8	30	70	0,0142
9	56	126	0,0079
10	99	225	0,0044
11	186	411	0,0024
12	335	746	0,0013
13	630	1376	0,0007
14	1161	2537	0,0004
15	2182	4719	0,0002
16	4080	8799	0,0001

Ашық мәтінде және шифрмәтінде $\sum_{i=1}^{i-1} m_i$ позициясынан бастап m_i жұмыс негіздерінің дәрежесіне тең ұзындықтағы бөліктердегі статистика (пайда болу жиілігі), ұзындығы m болатын әр блокта бірдей, тек кілтке байланысты m_i ұзындықтағы тізбектің барлық мүмкін болатын нұсқаларының біріне түрлендірілген. Сондықтан криптоталдаушы тиісті ұзындықтағы ашық мәтіннің бөліктеріне жиілік талдауын жүргізеді. Ары қарай, шифрмәтін бойынша дәл осындай талдау жасай отырып, негізгі бөліктердің ұзындықтарын анықтауға болады. m_i ұзындықтарын анықтауға болатын жағдайда, m_i дәрежелеріне сәйкес келтірілмейтін көпмүшелерді толық теру арқылы (3.8) теңдеулер жүйесін шешу керек.

Алгебралық шабуыл барысында, шифрмәтін мен ашық мәтін белгілі болған жағдайда, кілтті толық терудің саны келесі аралықта болады:

$$\sum_{i=1}^{s} I(m_i) \leq J(m) < \prod_{i=1}^{s} I(m_i),$$

мұндағы $I(m_i)$ - m_i-ші дәрежеге дейінгі келтірілмейтін көпмүшеліктер саны, $J(m)$ - ұзындығы m-ге тең кілттердің толық теруінің нұсқаларының саны.

3.6 бөлімде сипатталғандай жолмен XL және XSL шабуылдарының есептеу қиындығы төрт раундтық Qamal NPNS алгоритмі үшін сәйкесінше 2^{325} және 2^{165} тең. Яғни, негізгі алгоритмнің 8 раунды мен ПЕПСЖ негізделген шифрлау алгоритмінің 4 раунды алгебралық шабуылдарға беріктілігі шамамен бірдей.

ПЕПСЖ-не сызықтық және дифференциалдық талдауларды қолдану. (3.8) теңдеулер жүйесінде ақиқаттар кестесін пайдалана отырып, k_i және s_j айнымалаларының көбейтіндісін олардың қосындысына айырбастау арқылы сызықты түрге келтіруге болады. Біздің жағдайда мұны орындау мүмкіндігін қарастырайық. (3.8) теңдеулер жүйесінде $k_i \cdot s_j$ көбейтінділерін $k_i \oplus s_j$ қосындысына айырбастап және s_j айнымалыларын жою арқылы сызықты түрге келтіруге болады. Атай кететін бір жағдай, (3.8) жүйесінде барлық теңдеулер сызықты түрге келе бермейді .Криптоталдау жүргізу үшін сызықты теңдеулердің саны неғұрлым көбірек болғаны жақсы. Бұған келесідей қол жеткізуге болады.

Ақиқаттар кестесінен $xy \oplus x \oplus y$ өрнегі $3/4$ ықтималдықпен 1-ді, $1/4$ ықтималдықпен 0-ді қабылдайтынына оңай көз жеткізуге болады. Егер (3.8) теңдеулер жүйесінде x және y айнымалыларының көбейтіндісін $x \oplus y \oplus 1$-ге ауыстырсақ сызықты теңдеулер жүйесін аламыз. Егер теңдеулерде көбейту операцияларының саны өскен сайын теңдеудің ақиқат болуының ықтималдығы 0,5-ке жақындайды (кесте 3.28).

Кесте 3.28 – Көбейту операцияларының санына сәйкес теңдеудің ақиқат болуының ықтималдығы

Көбейтулер саны	Дұрыс ауыстырылу ықтималдығы	Қате ауыстырылу ықтималдығы
1	0,75	0,25
2	0,375	0,625
3	0,5625	0,4375
4	0,46875	0,53125
5	0,515625	0,484375
6	0,4921875	0,5078125
7	0,50390625	0,49609375
8	0,498046875	0,501953125
9	0,500976563	0,499023438
10	0,499511719	0,500488281
11	0,500244141	0,499755859
12	0,49987793	0,50012207
13	0,500061035	0,499938965
14	0,499969482	0,500030518
15	0,500015259	0,499984741
16	0,499992371	0,500007629

Егер теңдеудегі көбейткіштердің бірі 1-ге тең болса немесе үлкен ықтималдықпен 1-ге жақындаса, онда ықтимал сызықтық теңдеу құруға болады. Зерттеу нәтижесі көрсеткендей, келтірілмейтін көпмүшеліктердегі «0» мен «1» коэффициенттерінің саны бір-біріне жақын болғандықтан, ықтимал сызықтық

теңдеуді алу қиын. Осылайша, зерттелген алгоритм үшін сызықтық криптоталдауды қолдану тиімсіз [92].

Зерттеліп отырған алгоритмде позициялық емес полиномдық санау жүйесінің негіздері болып $GF(2)$ өрісіндегі келтірілмейтін көпмүшеліктер алынады. Бұл алгоритмде шифрлау үдерісі ашық мәтінді кілтпен $GF(2)$ өрісінде ПЕПСЖ негіздер модулі бойынша көбейту арқылы орындалады.

Талдау барысы келесі кезеңдерден тұрады:

- қажетті айырманы беретін ашық мәтін жұптарын алу;
- таңдалған ашық мәтіндер шифрлау;
- сәйкес шифрмәтіндерде де қандайда бір айырма болады, соны анықтау;
- ашық және шифрмәтіндердің жұптарынан статистика жинау;
- ашық мәтін жұптарының және шифрмәтін жұптарының айырмаларын сәйкестендіру;
- осы жиналған деректерге негіздеп, жүйенің кілті туралы жорамал жасау.

Классикалық дифференциалдық талдауларда жоғарыда айтылған кезеңдерді орындамас бұрын, S-блоктарға сәйкес айырмалар матрицалары құрылады. Айырмалар матрицасы дегеніміз – S-блоктың кірісіндегі барлық мүмкін мәндердің және оған сәйкес шығысындағы мәндердің айырмаларының қатынасынан құрылған матрица [47, с. 111]. Айырмалар матрицасын құру дифференциалдық талдаудың негізгі бөлігі болып саналады. Егер айырмалар матрицасының элементтері теңдей таралған болса, онда криптоталдаушы дифференциалдық шабуылды тиімсіз деп есептейді. Өйткені, айырмалар матрицасының стохастикалық матрицасы дәлме-дәл сәйкес келеді. Сондықтан, шифрлау алгоритміне статистикалық талдау жүргізілгенде, бұл матрицалардың пайдасы зор.

Ал, біздің қарастырып отырған шифрлау алгоритмінде, ПЕПСЖ-дегі модулі n дәрежелі $p(x)$ келтірілмейтін көпмүшелік болатын, көбейту амалына қатысты айырмалар матрицасын табамыз (кесте 3.29). Айырмалар матрицасының жолдары ашық мәтіндердің айырмалары (кіріс) және бағандары шифр мәтіндердің айырмалары (шығыс) болады.

Кесте 3.29 – Модулі n дәрежелі $p(x)$ келтірілмейтін көпмүшелік болатын көбейту амалының дифференциалдық талдаудағы айырымдардың матрицасы

		Сандық түрдегі шифр мәтінің айырмасы										
		0	1	2	3	4	5	6	...	2^n-3	2^n-2	2^n-1
Сандық түрдегі ашық мәтінің айырмасы	1	1	1	1	1	1	1	1	...	1	1	1
	2	1	1	1	1	1	1	1	...	1	1	1
	3	1	1	1	1	1	1	1	...	1	1	1
	4	1	1	1	1	1	1	1	...	1	1	1
	5	1	1	1	1	1	1	1	...	1	1	1
	6	1	1	1	1	1	1	1	...	1	1	1
	...	…	…	…	…	…	…	…	…	…	…	…
	2^n-1	1	1	1	1	1	1	1	...	1	1	1

Бұл айырымдық матрицасының барлық элементтері бірге тең. Яғни, шифрмәтіндердің айырымдары теңдей үлестірілген. Олай болса, ПЕПСЖ негізделген алгоритмге осы тұрғыда жүргізілген криптоталдау тиімсіз болады.

Екінші жағдайда, ашық мәтіндердің де, шифрмәтіндердің де айырмалары белгілі деп есептейміз. $\Delta\alpha(x) = \alpha(x) \oplus \alpha'(x)$ ашық мәтіндердің айырмасы, $\Delta\omega(x) = \omega(x) \oplus \omega'(x)$ шифр мәтіндердің айырмасы және $\alpha(x) \times \beta(x) = \omega(x) (mod\ p(x)$, $\alpha'(x) \times \beta(x) = \omega'(x) (mod\ p(x)$ болса, онда қарастырып отырған өрісте дистрибутивтік заң орындалғандықтан, келесі тұжырым орындалады:

$$\Delta\alpha(x) \times \beta(x) = \big(\alpha(x) \oplus \alpha'(x)\big) \times \beta(x) =$$
$$= (\alpha(x) \times \beta(x)) \oplus (\alpha'(x) \times \beta(x)) = \omega(x) \oplus \omega'(x) (mod\ p(x)) \qquad (3.11)$$

Егер, ашық мәтіндердің соңғы биті ғана өзгеше болатындай таңдап алатын болсақ, яғни $\Delta\alpha(x) = \alpha(x) \oplus \alpha'(x) = 1$ болсын, және осы мәнді (3.11) теңдігіне апарып қойсақ, онда келесі теңдікті аламыз:

$$\Delta\omega(x) = 1 \times \beta(x) (mod\ p(x)).$$

$\Delta\omega(x)$ және $\beta(x)$ көпмүшеліктерінің дәрежесі $p(x)$-тің дәрежесінен кіші болғандықтан $\Delta\omega(x) = \beta(x)$. Яғни, кілт шифрмәтіннің айырмасына тең болып шығады.

Шифрмәтінді ашу кезінде де дәл осылай болатынына көз жеткізу қиын емес. $\omega(x) \times \beta^{-1} = \alpha(x) (mod\ p(x))$ және $\Delta\omega(x) = \omega(x) \oplus \omega'(x) = 1$ болса, онда $\Delta\alpha(x) = \Delta\ \omega(x) \oplus \beta^{-1} (mod\ p(x))$ теңдігінен $\Delta\alpha(x) = \beta^{-1} (mod\ p(x))$ шығады. Олай болса, шифрды кері ашу кілті ашық мәтіндердің айырымына тең. Демек, осы тұрғыдан қарағанда қарастырып отырған шифрлеу алгоритмі дифференциалдық криптоталдауға берік емес.

Ал, айырымдары бірден өзге болғанда $\Delta\alpha \times \beta \oplus p \times \Delta s = \Delta\omega$ теңдеуді шешуге алып келеді. Бұл теңдеу сызықтық криптоталдауда қарастырылған теңдеулер жүйесіне алып келеді [92, с. 387].

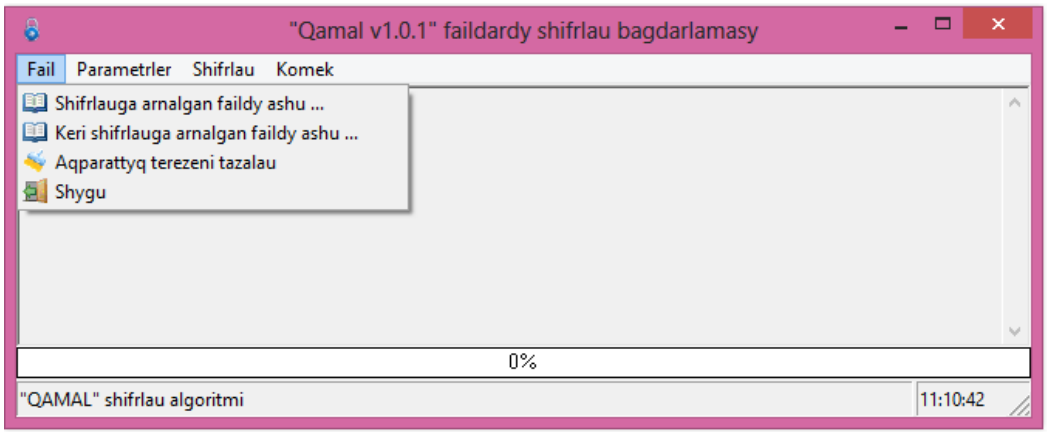
Қорыта айтқанда, егер ашық мәтіннің және жабық мәтіннің айырымдары белгілі болса кілтті анықтау мүмкіндігі туады. Ал, шифрмәтіндердің айырымдары ғана белгілі болған жағдайда кілтті және ашық мәтінді болжау мүмкін емес.

4 ШИФРЛАУ АЛГОРИТМІНЕ АРНАЛҒАН БАҒДАРЛАМАЛЫҚ ЖАСАҚТАМА МОДУЛЬДЕРІН ҚҰРУ

4.1 Құрылған шифрлау алгоритмін бағдарламалық іске асыру

Ұсынылған шифрлау алгоритмі Delphi v.7 бағдарламалық тілінде жүзеге асырылды. Жұмыс файлдарының көлемі - 1,13 МБ. Жүйелік талаптар: тактылық жиілік – 2 ГГц-тен төмен емес, ЖЖҚ – 1 МБ кем емес, қатқыл дискідегі бос орын көлемі – 10 ГБ-дан кем емес, операциялық жүйе - Windows Me/2000/XP/7/8. Бағдарламалық жасақтаманы әзірлеу кезінде аралық және қорытынды деректерді есептеу мен шығарудың әртүрлі нұсқалары қарастырылды.

«Qamal» шифрлау алгоритмінің бағдарламасының негізгі жұмыс терезесі сурет 4.1 - де келтірілген.

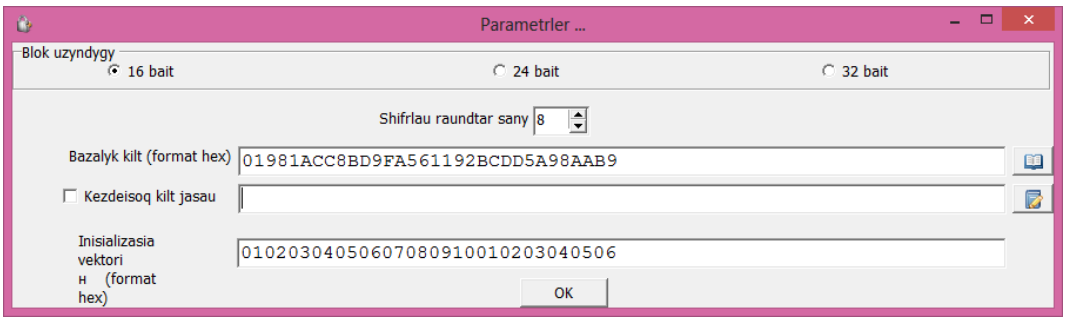


Сурет 4.1 – «Qamal» алгоритмінің бағдарламасының негізгі терезесі

Бағдарламаны іске қосу «Qamal_RW.exe» арқылы жүзеге асырылады және сурет 4.1 - дегідей диалогтық терезесі ашылады. Мәзір терезесі төрт бөлімнен тұрады: «Файл», «Параметрлер», «Шифрлау» и «Көмек».

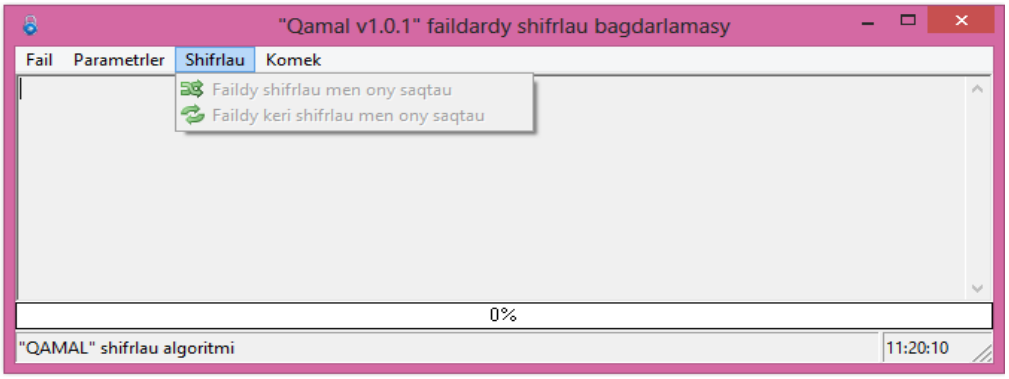
«Файл» бөлімі – бұл шифрлау немесе шифрды кері ашу үшін файлдарды енгізуге арналған диалогтық терезе.

«Параметрлер» бөлімі – блок ұзындығын, раундтар санын, негізгі кілт ұзындығын және шифрлаудағы бастапқы жүктеме векторын енгізуге және орнатуға арналған терезе. Шифрлау блогының белгіленген ұзындығына байланысты бағдарламаның өзі шифрлау раундтарының санын, негізгі кілттің ұзындығын және жүктеме векторын анықтайды (сурет 4.2).



Сурет 4.2 – «Параметрлер» бөлімі

«Шифрлау» бөлімі – шифрлау бағытын және оны іске қосуды таңдауға арналған терезе (сурет 4.3). Файлдан блокты оқу, шифрлау үдерісі және нәтижесінде алынған шифрланған блокты файлға сақтау, блок бойынша орындалады.



Сурет 4.3 – «Шифрлау» бөлімі

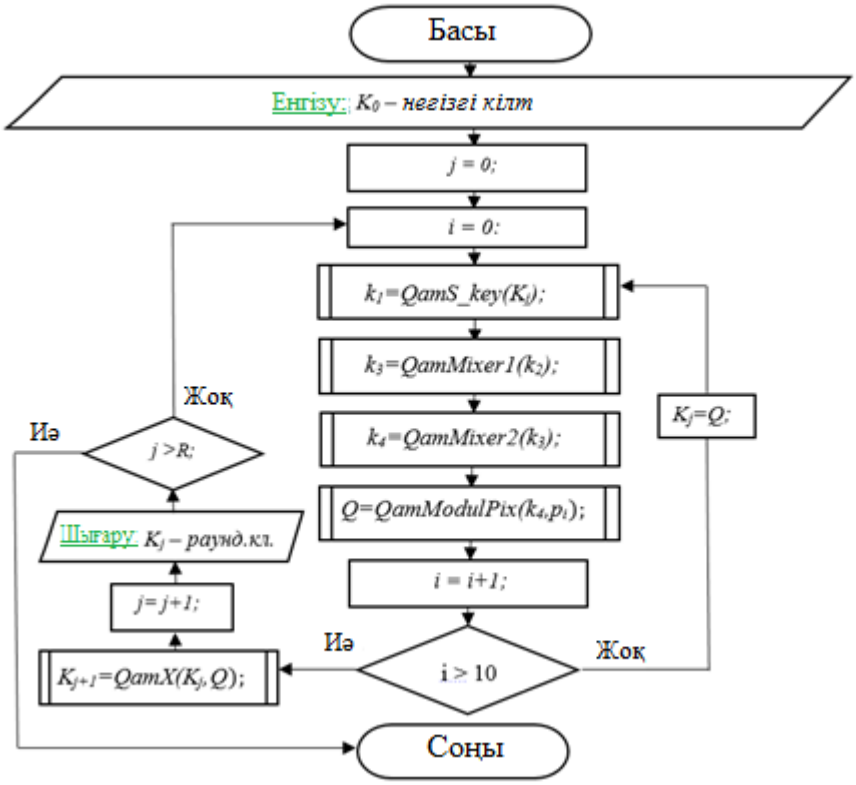
Шифрлау үдерісі:

1. Бастапқы және тұрақты деректерді енгізу. Бағдарламалық жасақтаманың кіріс мәліметтері болып келесілер табылады:

- шифрлауға арналған файлдарды енгізу;
- бастапқы параметрлерді енгізу, K_0 – ұзындықтары 16, 24 немесе 32 байт болатын негізгі кілт, V_0 - бастапқы жүктеме векторы.

2. Шифрлау блогының ұзындығына байланысты раундтық кілттер саны анықталады. Шифрлау алгоритмінің сипаттамасында келтірілгендей блоктың ұзындығы 128, 194 және 256 бит болуы мүмкін, раундтық кілттердің саны осыларға сәйкес 8, 10 және 12 мәндерін қабылдайды.

Келесі K_{i+1} раундтық кілтін алу үдерісі келесі түрлендірулер тізбегін орындау арқылы жүзеге асырылады: QamS_key сызықтық емес түрлендіруі, Mixer1 және Mixer2 процедуралары және *Module* $p_i(x)$ түрлендіруі. Бұл жұмыстар тізбегі 10 рет қайталанады және соңында алынған блокты K_i кілтімен биттік қосу (xor) орындалады. Раундтық кілттерді құрудың (QamKeyGen ішкі бағдарламасының) блок-схемасы төменде көрсетілген (сурет 4.4).



Сурет 4.4 – Раундтық кілттерді құру блок-схемасы
(QamKeyGen ішкі бағдарламасы)

3. Шифрлау блогы. Мәліметтерді шифрлау блок-схемасы сурет 4.5 - те көрсетілген. Мәліметтерді шифрлау үдерісі келесі түрлендіру блоктарынан тұрады:

3.1. Шифрларды тізбектеу режимі инициализация векторын және ашық мәтін блогына модуль 2 бойынша биттік қосу арқылы жүзеге асырылады (*QamX* процедурасы).

3.2. K_0 кілтімен 3.1-пунктте алынған блок арасында *xor* операциясы орындалады.

3.3. Сызықты емес биективті түрлендіру орындалады (*QamS* процедурасы).

3.4. А матрицасының бағандары бойынша түрлендіру (Mixer1 процедурасы). А матрицасын құру кезінде 3.3-пунктінде алынған блоктың байттары блоктың өлшеміне байланысты екі өлшемді массив түрінде ұсынылады.

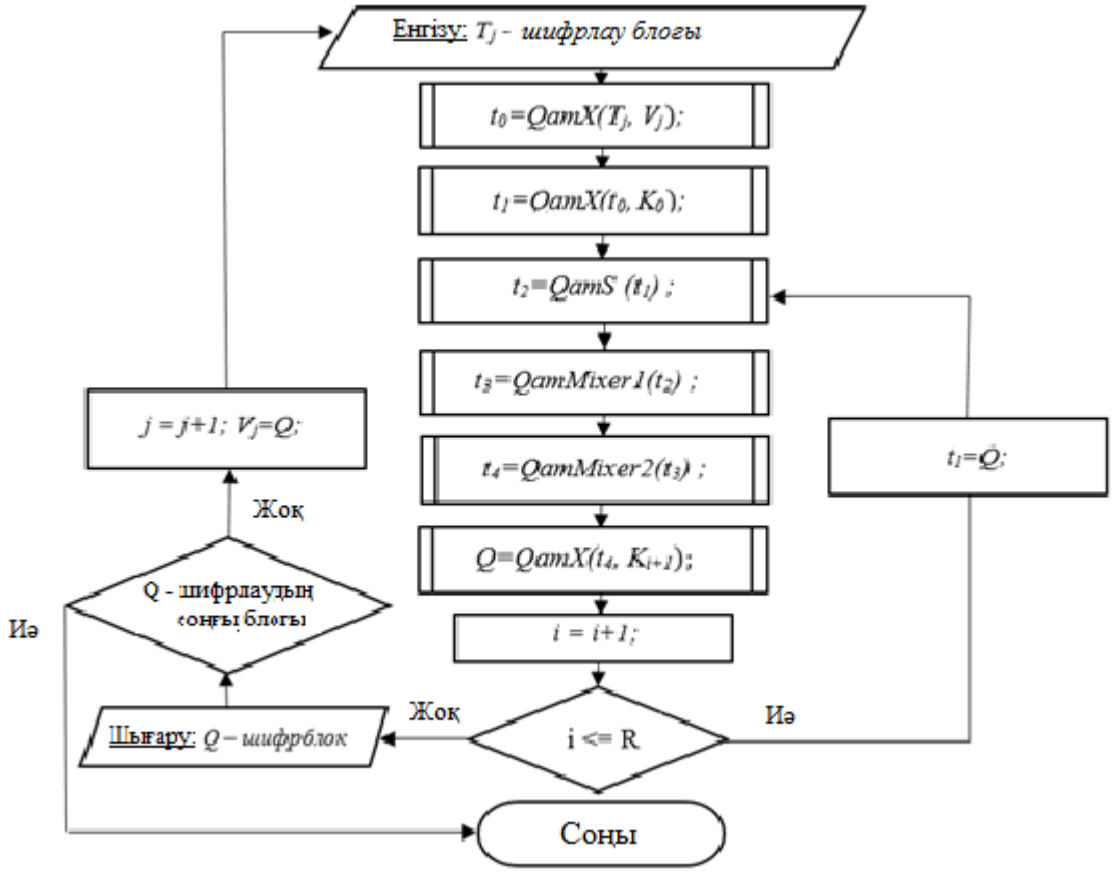
3.5. В матрицасының жолдары бойынша түрлендіру (Mixer2 процедурасы). В матрицасының құрылымы мен элементтері А матрицасы (Mixer1 түрлендіруі нәтижесінде алынған мәліметтер) сияқты анықталады.

3.6. K_{i+1} кілтімен 3.5-пунктте алынған блок арасында *xor* операциясы орындалады (*QamX* процедурасs).

3.7. Блок ұзындығына байланысты 3.3 – 3.6 пункттері K_{i+1} кілтін қолдану арқылы 8, 10 немесе 12 рет қайталанады.

3.8. 3.1 - 3.8 пункттері файлдың соңғы блогы шифрланғанға дейін қайталанады. Соңғы шифрлау блогы толық болмаған жағдайда, жетіспейтін

байттар нөлдік байттармен бүтін санға дейін толтырылады, ал соңғы байттың орнына - толтырылған байттар санының коды жазылады.



Сурет 4.5 – Блокты шифрлау блок-схемасы

Шифрды кері ашу үдерісінде қолданылған барлық QamS, Mixer1 және Mixer2 криптографиялық түрлендірулері кері ретпен және инверсияланған QamSR, MixerR1 және MixerR2 түрлендірулері қолданылады. K_i раундтық кілттеріде кері ретпен пайдаланылады. S-блок түрлендіруінің орнына оған кері SR-блок түрлендіруі сәйкес келеді, блок-схемасы сурет 4.6 - да көрсетілген.

4.1. K_i кілтімен кері шифрлау блогы арасында *xor* операциясы орындалады (*QamXR* процедурасы).

4.2. C матрицасының жолдары бойынша түрлендіруді орындау (*MixerR2* процедурасы).

4.3. *D* матрицасының бағандары бойынша түрлендіруді орындау (*MixerR1* процедурасы).

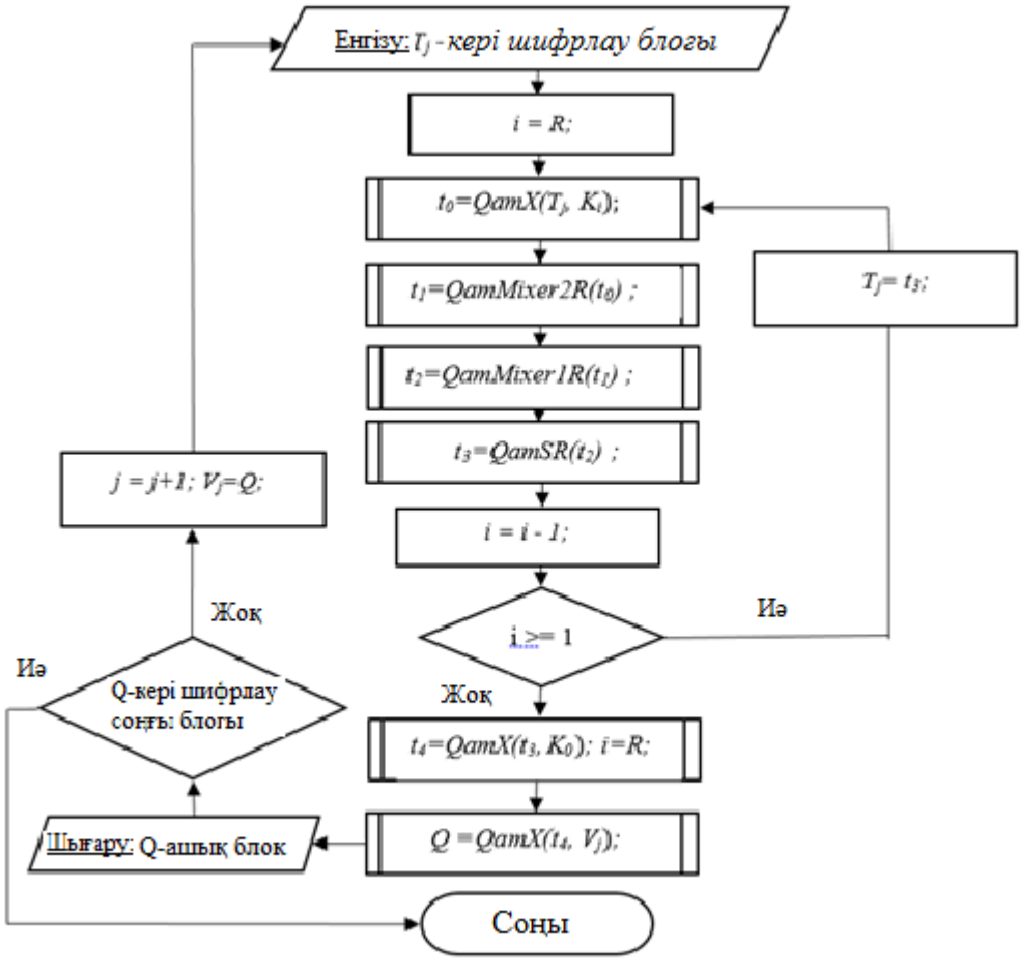
4.4. Бұдан кейін кері Sблок түрлендіруі орындалады (*QamSR* процедурасы).

4.5. 4.1 – 4.4 бөлімдері K_i, мұндағы $i = R, \dots, 1$ раундтық кілттерін пайдалана отырып, блоктың ұзындығына байланысты 8, 10 немесе 12 рет қайталанады.

4.6. 4.4 пункттің нәтижесімен K_0 кілттерін *xor* операциясы арқылы қосу (*QamX* процедурасы).

4.7.Шифрлаудың тіркелу режимі *QamX* процедурасы арқылы жүзеге асырылады.

4.8. Барлық пункттер шифрланған мәтіннің соңғы блогы кері шифрланып болғанға дейін орындалады.



Сурет 4.6 – Кері ашу шифрлауының блок-схемасы

Құрылған ««Qamal v1.0.1» - Файлдарды шифрлау бағдарламасына» 2019 жылдың 6 қыркүйегінде ҚР Әділет министрлігінің Ұлттық зияткерлік меншік институтынан №5200 Авторлық куәлік алынды (Қосымша Ә).

4.2 Шифрлау алгоритміндегі Mixer2 түрлендіруінің есептеу жылдамдығын арттырудың әдістері

Шифрлау алгоритмінің Mixer2 бөлігінің жұмыс істеу сұлбасын кеңірек қарастырайық. Mixer1 түрлендіруі орындалғаннан кейін блоктың ұзындығына байланысты өлшемі $k \times 4$, мұндағы k=4, 6 немесе 8 болатын екіөлшемді B массивін аламыз:

$$B = \begin{bmatrix} b_{00} & b_{01} & b_{02} & b_{03} \\ b_{10} & b_{11} & b_{12} & b_{13} \\ . & . & . & . \\ b_{k-1\,0} & b_{k-1\,1} & b_{k-1\,2} & b_{k-1\,3} \end{bmatrix}.$$

В массивінің әрбір i-ші қатары коэффициенттері $GF(2^8)$ ақырлы өрісінде жататын үшінші дәрежелі көпмүшелік түрінде ұсынылған. Бұл көпмүшеліктер келесідей түрге ие:

$$b_i(x) = b_{i0}x^3 + b_{i1}x^2 + b_{i2}x + b_{i3}, i = 0, \ldots, k-1 \quad (4.1)$$

Әрбір $b_i(x)$ көпмүшелігі бекітілген $m_i(x)$ көпмүшеліктеріне келтірілмейтін $p(x)$ көпмүшелігі модулінде көбейтіледі:

$$r_i(x) = \left(b_i(x) * m_i(x)\right) mod\ p(x) \quad (4.2)$$

Алгоритмнің сипаттамасында келтірілгендей бекітілген көпмүшеліктер келесідей болады:

$p(x) = x^4 + x + 55,$
$m_0(x) = 168x^3 + 34x^2 + 187x + 186,\ m_1(x) = 210x^3 + 53x^2 + 210x + 101,$
$m_2(x) = 218x^3 + 25x^2 + 150x + 210,\ m_3(x) = 144x^3 + 75x^2 + 158x + 27,$
$m_4(x) = 163x^3 + 4x^2 + 111x + 106,\ \ m_5(x) = 150x^3 + 237x^2 + 13x + 53,$
$m_6(x) = 99x^3 + 59x^2 + 104x + 205,\ m_7(x) = 167x^3 + 49x^2 + 241x + 154.$

Шифрлау барысында келесіні ескереміз: блок ұзындығы 128 бит болған жағдайда $m_i(x)$, мұндағы $i = \overline{0,3}$ көпмүшеліктерінен бірінші төрт көпмүшелік, 192 бит болғанда $m_i(x)$, мұндағы $i = \overline{0,5}$ алты көпмүшелік және 256 бит болғанда барлық сегіз $m_i(x)$, мұндағы $i = \overline{0,7}$ көпмүшеліктері қолданылады.

Шифрды кері ашу үдерісінде $r_i(x)$ көпмүшелігі келесідей анықталады:

$$r_i(x) = \left(b_i(x) * m_i^{-1}(x)\right) mod\ p(x) \quad (4.3)$$

мұнда, $m_i^{-1}(x)$ – кері элементтер және олардың түрі келесідей болады:

$m_0^{-1}(x) = 243x^3 + 72x^2 + 137x + 213,\quad m_1^{-1}(x) = \ 22x^3 + 115x^2 + 208x + 215,$
$m_2^{-1}(x) = 138x^3 + 46x^2 + 139x + 186,\quad m_3^{-1}(x) = 192x^3 + 162x^2 + 60x + 176,$
$m_4^{-1}(x) = 253x^3 + 165x^2 + 100x + 82,\quad m_5^{-1}(x) = 127x^3 + 156x^2 + 48x + 34,$
$m_6^{-1}(x) = 152x^3 + \ 75x^2 + 157x + 62,\quad m_7^{-1}(x) = \ 30x^3 + 115x^2 + 31x + 136.$

Бағдарламалық іске асыру барысында жоғарыда аталған көпмүшеліктерді коэффициенттері $GF(2)$ өрісінде болатындай келесі түрде жазып алуға болады:

168 34 187 186

m_0=10101000001000101011101110111010;
m_1=11010010001101011101001001100101;
m_2=11011010000110011001011011010010;
m_3=10010000010010111001111000011011;
m_4=10100011000001000110111101101010;

m_5=10010110111011010000110100110101;
m_6=01100011001110110110100011001101;
m_7=10100111001100011111000110011010;
m_0^{-1}=11110011010010001000100111010101;
m_1^{-1}=00010110011100111101000011010111;
m_2^{-1}=10001010001011101000101110111010;
m_3^{-1}=11000000101000100011110010110000;
m_4^{-1}=11111101101001010110010001010010;
m_5^{-1}=01111111100111000011000000100010;
m_6^{-1}=10011000010010111001110100111110;
m_7^{-1}=00011110011100110001111110001000;
p= 10000000000000000000000000100110111.

Mixer2 түрлендіруінде барлық операциялар ПЕПСЖ-де орындалатынын атай кетейік. Осы түрлендіруді бағдарламалық іске асырудың үш түрлі тәсілін қарастырайық [93].

1-есептеу тәсілі (тікелей іске асыру).

(4.2) өрнекті есептеу келесі екі кезең арқылы жүзеге асырылады. Бірінші – $b_i(x)$ және $m_i(x)$ көпмүшеліктері көбейтіледі, екінші – алынған көбейтіндіні $p(x)$ көпмүшелігіне бөліп, қалдығы алынады.

1-кезең. В массивіндегі әрбір жолдағы байттық түрдегі мәндер GF(2) өрісінде жазылып алынады және конкатенация операциясы арқылы 32 биттік тізбек құрайды, $B_i\left(b_{i,31}, b_{i,30}, b_{i,29}, \dots, b_{i,1}, b_{i,0}\right)$. B_i көпмүшелігі сәйкесінше $M_i\left(m_{i,31}, m_{i,30}, m_{i,29}, \dots, m_{i,1}, m_{i,0}\right)$ көпмүшелігіне көбейтіліп, ұзындығы 64 бит болатын $Q_i\left(q_{i,62}, q_{i,61}, q_{i,60}, \dots, q_{i,1}, q_{i,0}\right)$ екілік тізбегі анықталады. Q_i – дің элементтері келесі формула бойынша есептеледі:

$$q_{i,l} = \sum_{k=0}^{l} b_{i,l-k} * m_{i,k}, \quad l = \overline{0,62}; \tag{4.4}$$

мұндағы, $l > 31$ болғанда $b_{i,l} = 0$ және $m_{i,l} = 0$, $\sum$ – белгілеуі біздің жағдайда модуль 2 бойынша сумманы білдіреді.

2-кезең. (4.2) өрнегінің нәтижелік мәнін анықтау үшін $Q_i(x) = t_i(x) * p(x) + r_i(x)$ орындалатындай $r_i(x)$ қалдығын табу керек. $p(x)$ көпмүшелігі де GF(2) өрісінде $P = (p_{32}, p_{31}, p_{30}, \dots, p_1, p_0)$ түріндегі тізбек түрінде өрнектеледі және $\left(q_{i,62}, q_{i,61}, q_{i,60}, \dots, q_{i,1}, q_{i,0}\right) / \left(p_{32}, p_{31}, p_{30}, \dots, p_1, p_0\right)$ есептеуі баған бойынша бөлуге әдісіне сәйкес, қалдық $R_i = \left(r_{i,l}, r_{i,l-1}, \dots, r_{i,0}\right)$-дің ұзындығы 33 биттен кіші болғанға дейін жүргізіледі. Алынған 32 биттік тізбек $R_i = \left(r_{i,31}, r_{i,30}, \dots, r_{i,0}\right)$ сегіз биттен бөлінгеннен кейін R массивінің i-ші жолының төрт байтын (элементін) анықтайды.

[94] жұмыста қалдықпен бөлудің алгоритмі мен процедурасы туралы кеңірек ақпарат келтірілген.

Мысал. Айталық, шифрлау блогының ұзындығы 16 байт болғанда, түрлендіруден кейін келесі B массивін алдық делік:

$$B = \begin{bmatrix} 241 & 23 & 86 & 74 \\ 43 & 26 & 59 & 147 \\ 237 & 177 & 108 & 169 \\ 44 & 53 & 196 & 170 \end{bmatrix}$$

Конкатенация арқылы біріктірілгеннен кейінгі 1-жол элементтерінің екілік көрінісі $B_0 = 11110001000101110101011001001010$ болады. Сәйкес $M_0 = 10101000001000101011101110111010$ көпмүшелігіне көбейткеннен кейін, $Q_0 = 11000010011111001010100110101100001000010110100111$ 100000100100 аламыз.

Q_0-ді p=100000000000000000000000100110111 модуліне бөлгеннен кейін $R_0 = 00111101011101000011011101010000$ қалдығын анықтаймыз. 8 биттен топтау арқылы 61, 116 ,55, 80 мәндерін аламыз. 2-ші, 3-ші және 4-ші жолдарға дәл осындай процедураларды және өздерінің сәйкесінше M_i көпмүшеліктерін қолдану арқылы Mixer2 түрлендіруінің нәтижесін аламыз:

$$R = \begin{bmatrix} 61 & 116 & 55 & 80 \\ 250 & 22 & 111 & 176 \\ 212 & 27 & 15 & 194 \\ 242 & 231 & 169 & 232 \end{bmatrix}$$

2-есептеу тәсілі **(**коэффициенттері айнымалы болатын көпмүшеліктерді модуль бойынша көбейту әдісі).

Бұл әдісте, $m_i(x)$ және $p(x)$, мұндағы $i = \overline{0,7}$ бекітілген көпмүшеліктер екенін ескере отырып, коэффициенттері айнымалы болатын $z(x)$ көпмүшелігінің $r_i(x)$ қалдықтарын сызықты тәуелді айнымалылардың өрнегі ретінде қарастырылады.

Бағдарламалық жасақтамада есептеулер $GF(2)$ өрісінде жүргізіледі. Бағдарламаның бастапқы деректері – $M_i = (m_{i,31}, m_{i,30}, m_{i,29}, \dots, m_{i,1}, m_{i,0})$, $P = (p_{32}, p_{31}, p_{30}, \dots, p_1, p_0)$ және $Z(z_{31}, z_{30}, z_{29}, \dots, z_1, z_0)$ екілік тізбектері. Z - анықталмаған екілік айнымалылар тізбегі.

Көмекші бағдарламаның көмегімен Mixer2 түрлендіруін сызықты байланысты өрнектерге алмастырып аламыз. Ол туралы 3.5 бөлімде кеңірек жазылған. Төменде бекітілген p және m_0 үшін $R_i = (r_{i,31}, r_{i,30}, \dots, r_{i,0})$-дің $Z = (z_{31}, z_{30}, z_{29}, \dots, z_1, z_0)$-тен сызықты тәуелді өрнектері келтірілген. Бұл өрнектер (B.1) формуласында да көрсетілген, тек ол жерде айнымалылар басқаша белгіленген.

$r_{0,31} = z_0 \oplus z_2 \oplus z_4 \oplus z_{10} \oplus z_{14} \oplus z_{16} \oplus z_{18} \oplus z_{19} \oplus z_{20} \oplus z_{22} \oplus z_{23} \oplus z_{28} \oplus z_{29} \oplus z_{30}$;
$r_{0,30} = z_1 \oplus z_3 \oplus z_9 \oplus z_{13} \oplus z_{15} \oplus z_{17} \oplus z_{18} \oplus z_{19} \oplus z_{21} \oplus z_{22} \oplus z_{27} \oplus z_{28} \oplus z_{29} \oplus z_{31}$;

$r_{0,29}= z_0 \oplus z_2 \oplus z_8 \oplus z_{12} \oplus z_{14} \oplus z_{16} \oplus z_{17} \oplus z_{18} \oplus z_{20} \oplus z_{21} \oplus z_{26} \oplus z_{27} \oplus z_{28} \oplus z_{30} \oplus z_{31};$
…
$r_{0,0}= z_1 \oplus z_3 \oplus z_5 \oplus z_{11} \oplus z_{15} \oplus z_{17} \oplus z_{19} \oplus z_{20} \oplus z_{21} \oplus z_{23} \oplus z_{24} \oplus z_{29} \oplus z_{30} \oplus z_{31};$
$r_{1,31}= z_0 \oplus z_1 \oplus z_3 \oplus z_6 \oplus z_{10} \oplus z_{11} \oplus z_{13} \oplus z_{15} \oplus z_{16} \oplus z_{17} \oplus z_{19} \oplus z_{22} \oplus z_{24} \oplus z_{26} \oplus z_{30};$
…
$r_{7,31}= z_0 \oplus z_2 \oplus z_5 \oplus z_6 \oplus z_7 \oplus z_{10} \oplus z_{11} \oplus z_{15} \oplus z_{16} \oplus z_{17} \oplus z_{18} \oplus z_{19} \oplus z_{23} \oplus z_{26};$
$r_{7,30}= z_1 \oplus z_4 \oplus z_5 \oplus z_6 \oplus z_9 \oplus z_{10} \oplus z_{14} \oplus z_{15} \oplus z_{16} \oplus z_{17} \oplus z_{18} \oplus z_{22} \oplus z_{25} \oplus z_{31};$
$r_{7,29}= z_0 \oplus z_3 \oplus z_4 \oplus z_5 \oplus z_8 \oplus z_9 \oplus z_{13} \oplus z_{14} \oplus z_{15} \oplus z_{16} \oplus z_{17} \oplus z_{21} \oplus z_{24} \oplus z_{30} \oplus z_{31};$
…
$r_{7,0}= z_1 \oplus z_3 \oplus z_6 \oplus z_7 \oplus z_8 \oplus z_{11} \oplus z_{12} \oplus z_{16} \oplus z_{17} \oplus z_{18} \oplus z_{18} \oplus z_{20} \oplus z_{24} \oplus z_{27};$ (4.5)

Mixer2 түрлендіруінің нәтижесін алу үшін, әрбір жол үшін $Z = (z_{31}, z_{30}, z_{29}, \dots, z_1, z_0)$ айнымалысы ретінде B массивінің – $B_i = \left(b_{i,31}, b_{i,30}, b_{i,29}, \dots, b_{i,1}, b_{i,0}\right)$, мұндағы $i = \overline{0,k}$ – жолдың номері, k – 4, 6 немесе 8 i – ші жолының екілік тізбегі алынады.

Соңында ізделінді нәтиженің түрі келесідей болады:

$$R = \begin{bmatrix} r_{0,31}r_{0,30}r_{0,29}..r_{0,24} & r_{0,23}r_{0,22}r_{0,21}\dots r_{0,16} & r_{0,15}r_{0,14}r_{0,13}\dots r_{0,8} & r_{0,7}r_{0,6}r_{0,5}\dots r_{0,0} \\ r_{1,31}r_{1,30}r_{1,29}..r_{1,24} & r_{1,23}r_{1,22}r_{1,21}\dots r_{1,16} & r_{1,15}r_{1,14}r_{1,13}\dots r_{1,8} & r_{1,7}r_{1,6}r_{1,5}\dots r_{1,0} \\ \dots & \dots & \dots & \dots \\ r_{i,31}r_{i,30}r_{i,29}..r_{i,24} & r_{i,23}r_{i,22}r_{i,21}\dots r_{i,16} & r_{i,15}r_{i,14}r_{i,13}\dots r_{i,8} & r_{i,7}r_{i,6}r_{i,5}\dots r_{i,0} \end{bmatrix}$$

мұндағы, $r_{i,j}$ –дің биттік мәні жоғарыдағы формуланың көмегімен есептеледі.

Мысал. Жоғарыда көрсетілген мысалды қарастырайық:

$$B = \begin{bmatrix} 241 & 23 & 86 & 74 \\ 43 & 26 & 59 & 147 \\ 237 & 177 & 108 & 169 \\ 44 & 53 & 196 & 170 \end{bmatrix}$$

1-ші жолды екілік тізбек ретінде жазып алайық:

$B_0 = 11110001000101110101011001001010.$

(4.5) формулада z_l айнымалысы ретінде B_0 жолының $b_{0,l}$, $l = \overline{0,31}$ элементтері алынады да келесідей есептеу жүргіземіз.

$r_{0,31} = b_{0,0} \oplus b_{0,2} \oplus b_{0,4} \oplus b_{0,10} \oplus b_{0,14} \oplus b_{0,16} \oplus b_{0,18} \oplus b_{0,19} \oplus b_{0,20} \oplus b_{0,22} \oplus b_{0,23} \oplus b_{0,28} \oplus$
$\oplus b_{0,29} \oplus z_{0,30} = 0;$
…
$r_{0,29}= b_{0,0} \oplus b_{0,2} \oplus b_{0,8} \oplus b_{0,12} \oplus b_{0,14} \oplus b_{0,16} \oplus b_{0,17} \oplus b_{0,18} \oplus b_{0,20} \oplus b_{0,21} \oplus b_{0,26} \oplus b_{0,27} \oplus$
$\oplus b_{0,28} \oplus b_{0,30} \oplus b_{0,31} = 1;$
…
$r_{0,0}=b_{0,1} \oplus b_{0,3} \oplus b_{0,5} \oplus b_{0,11} \oplus b_{0,15} \oplus b_{0,17} \oplus b_{0,19} \oplus b_{0,20} \oplus b_{0,21} \oplus b_{0,23} \oplus b_{0,24} \oplus b_{0,29} \oplus$
$\oplus b_{0,30} \oplus b_{0,31} = 0;$

Нәтижесінде $R_0 = 00111101011101000011011101010000$ мәнін аламыз. Басқа жолдарға да дәл осылай есептеулер жүргізу арқылы және оларды байттармен өрнектей отырып,келесі нәтижеге қол жеткіземіз.

$$R = \begin{bmatrix} 61 & 116 & 55 & 80 \\ 250 & 22 & 111 & 176 \\ 212 & 27 & 15 & 194 \\ 242 & 231 & 169 & 232 \end{bmatrix}$$

3-есептеу тәсілі (үлкен сандар әдісі).

Бұл әдістің ерекшелігі – модуль бойынша көбейту операциясы ондық негізібен позициялық емес санау жүйесінде орындалады. Бекітілген деректердің ондық санау жүйесінде жазып аламыз:

$p = 4\,294\,967\,607,\quad m_0 = 2\,820\,848\,570,\quad m_1 = 3\,526\,742\,629,$
$m_2 = 3\,659\,110\,098,\ m_3 = 2\,420\,874\,779,\quad m_4 = 2\,734\,976\,874,$
$m_5 = 2\,532\,117\,813,\ m_6 = 1\,664\,837\,837,\ m_7 = 2\,805\,068\,186.$

В массивінің әрбір жолындағы барлық төрт байт келесі формула арқылы бір сандық мән құрайды:

$$B_i = 2^{24}b_{i0} + 2^{16}b_{i1} + 2^8 b_{i2} + b_{i3},\ i = \overline{0, k-1}. \quad (4.6)$$

Бағдарламалық жасақтамада келесі өрнекті қолдануға болады:

$$B_i = (b_{i0} \ll 24) + (b_{i1} \ll 16) + (b_{i2} \ll 8) + b_{i3},\ i = \overline{0, k-1},$$

мұндағы, «≪» – b_{ij} байтының екілік түрінің оң жағына нөлдерді қоса отырып солға жылжыту операциясы, $j = \overline{0,3}$. B_i және R_i – 64 биттік бүтін мәндер (Int64).

Mixer2 түрлендіруі (4.7) формула арқылы есептеледі:

$$R_i = (B_i * m_i)\ mod\ p, i = \overline{0, k-1} \quad (4.7)$$

(4.7)-ні есептеудің ең қарапайым тәсілі: B_i-дің m_i –ге көбейтіндісін есептеп және p модулі бойынша қалдықты табу. Үлкен мәндер үшін бұл тәсілде операциялар саны өте көп және оны іске асыру өте күрделі. Сондықтан бағдарламалық жасақтаманы жүзеге асыру үшін келесі сандық алгоритм оңтайлы әдіс болып саналады [28, с. 189]:

1. $R_i = 0$ болсын.
2. Егер $m_i = 0$, онда 7 қадамға көш.
3. Егер m_i- тақ болса, онда $R_i = R_i \oplus B_i$ болсын.
4. $B_i = 2 * B_i$ болсын.
5. Егер $B_i \geq 2^{32}$, онда $B_i = B_i \oplus p$ болсын.
6. $m_i = [m_i/2]$.
   $m_i \neq 0$ болса 3 қадамға қайт.
7. Нәтиже: R_i.

6-шы қадамда және бұдан ары қарай «[]» символы бөліндінің бүтін бөлігін білдіреді.

Осы есептелген R_i –ден R массивінің төрт байтын алу үшін келесі формуланы пайдаланамыз:

$$
\begin{aligned}
r_{i,0} &= [R_i/2^{24}];\\
r_{i,1} &= \left[\left(R_i - r_{i,0}\right)/2^{16}\right];\\
r_{i,2} &= \left[\left(R_i - r_{i,0} - r_{i,1}\right)/2^{8}\right];\\
r_{i,3} &= R_i - r_{i,0} - r_{i,1} - r_{i,2}.
\end{aligned}
\tag{4.8}
$$

Бағдарламалық жасақтаманы іске асыру үшін келесі операцияларды қолдануға болады:

$r_{i,0} = R_i \gg 24;$

$r_{i,1} = (R_i \ll 8) \gg 24;$

$r_{i,2} = (R_i \ll 16) \gg 24;$

$r_{i,3} = (R_i \ll 24) \gg 24;$

$i = \overline{0, k-1}$, мұндағы k – блоктың ұзындығына байланысты 4, 6 немесе 8 үшін дәл осылай жалғастыру арқылы R – ді анықтаймыз:

$$
R = \begin{bmatrix}
r_{0,0} & r_{0,1} & r_{0,2} & r_{0,3}\\
r_{1,0} & r_{1,1} & r_{1,2} & r_{1,3}\\
\dots & \dots & \dots & \dots\\
r_{k-1,0} & r_{k-1,1} & r_{k-1,2} & r_{k-1,3}
\end{bmatrix}
$$

Мысал. Бастапқы мысалды қарастырайық.

$$
B = \begin{bmatrix}
241 & 23 & 86 & 74\\
43 & 26 & 59 & 147\\
237 & 177 & 108 & 169\\
44 & 53 & 196 & 170
\end{bmatrix}
$$

(4.6) формуланы қолданып $B_0 = 4\,044\,838\,474$, $B_1 = 723\,139\,475$, $B_3 = 3\,987\,827\,881$, $B_4 = 741\,721\,258$ аламыз.

Келесі кезекте (4.7) формула бойынша $R_0 = 1\,031\,026\,512$, $R_1 = 723\,139\,475, R_3 = 3\,987\,827\,881$, $R_4 = 741\,721\,258$ есептеп тауып аламыз. Соңында (4.8) формуланы қолдану арқылы келесі нәтижеге қол жеткіземіз:

$$
R = \begin{bmatrix}
61 & 116 & 55 & 80\\
250 & 22 & 111 & 176\\
212 & 27 & 15 & 194\\
242 & 231 & 169 & 232
\end{bmatrix}
$$

Тәжірибелік нәтижелер. Есептеу жылдамдығын анықтау бойынша жұмыс нәтижесінде Mixer2 түрлендіруін есептеудің үш алгоритмі жасалды. Алгоритмдер Delphi6 бағдарламалау тілінде жеке бағдарламалар ретінде жүзеге асырылды. Mixer2 түрлендіруі $m_i(x)$ және $m_i^{-1}(x)$ бастапқы параметрлеріне қатысты симметриялы болғандықтан, Mixer2 есептеуін орындаудың жылдамдығын бағалау үшін тек шифрлау үдерісі қарастырылды. Есептеу

уақытын өлшеу келесідей параметрлі компьютерде жүргізілді: Intel(R) Core i-7 8700 жиілігі 3,20 ГГц және 32 Гбайт жедел жады. Тәжірибелік нәтижелер кесте 4.1 - де келтірілген.

Кесте 4.1 – Үш тәсілдің жылдамдығын салыстыру

	1-есептеу тәсілі	2-есептеу тәсілі	3-есептеу тәсілі
Блок ұзындығы 128 бит	0,00039 сек.	0,0000219 сек.	0,0000031 сек.
Блок ұзындығы 192 бит	0,00051 сек.	0,0000312 сек.	0,0000063 сек.
Блок ұзындығы 256 бит	0,00066 сек.	0,0000375 сек.	0,0000078 сек.

Тәжірибенің негізгі мақсаты – Mixer2 түрлендіруінің есептеу жылдамдығын анықтау болды. Кесте 4.1 - ге сәйкес, шифрлау блогының ұзындығына байланысты келесі жағдайларды байқауға болады. Егер бағдарламаны орындау уақыты бойынша салыстыратын болсақ, онда шифр блогының ұзындығының барлық жағдайында (128 бит, 192 бит және 256 бит) олардың жылдамдықтары өзара пропорционал болады. Яғни, есептеудің 2-ші тәсілі 1-ші тәсілге қарағанда 16 есе, ал 3-ші тәсіл 2-ші тәсілге қарағанда 7 есе жылдам. Нәтижесінде 1-есептеу әдісі шифрлау блогының ұзындығына қарамастан 3-шіден 112 есе баяу. Бұл дерек, есептеу уақытының көп бөлігі әрбір элементі жеке байт ретінде сақталатын массивтер мен жолдарды пайдалануға кететіндігімен түсіндіріледі. Массивтер мен жолдардың элементтерін пайдалану есептеу жылдамдығын едәуір баяулататыны белгілі. Осыған байланысты қарапайым логикалық операцияларды қолданатын және үлкен сандармен жұмыс жасауға негізделген 3-тәсіл, есептеу уақытты бойынша ең оңтайлысы. Егер бағдарламалық іске асырудың уақытын шифрлау блогының ұзындығына байланысты салыстыратын болсақ, онда ол пропорционалды өседі, яғни шамадан үлкен ауытқулар табылған жоқ.

ҚОРЫТЫНДЫ

Диссертациялық жұмыста теориялық және тәжірибелік зерттеулер жүргізу нәтижесінде криптографияда кеңінен қолданылатын SP-желісіне негізделген симметриялы блоктық шифрлау алгоритмі құрылды және кілттерді қолданудағы ерекшелігіне қарай ПЕПСЖ негізделген екінші нұсқасы ұсынылды. Әзірленген алгоритмде қолданылған түрлендірулер сипатталып, бағдарламалық жасақтамасы іске асырылды.

Криптографияда барлық блоктық шифрлау алгоритмдері болжамды берік алгоритмдер тобына жататындықтан, осы түрдегі алгоритмдерді криптоталдау әдістеріне берік екендігі көрсетілуі қажет. Сондықтан, жұмыста құрылған алгоритмді жан-жақты зерттеу нәтижелері келтірілген. Ол ең бірінші кезекте, шифрлау алгоритмі бойынша алынған шифрмәтіндерге статистикалық зерттеу жұмыстарынан басталады және алгоритмнің лавиндік әсері тексерілді. Осы бағыттағы зерттеу нәтижелері жақсы нәтиже көрсетті және олардың сандық сипаттамалары жұмыста келтірілген. Сонымен қатар, раундтық кілттерді құру алгоритмі бойынша алынған кілттер тізбегі де статистикалық бағаланып, кілттің 0 мен 1-ді қабылдау ықтималдығының 0,5-тен ауытқу аралығы көрсетілді.

Диссертациялық жұмыста, алгоритмнің криптографиялық беріктігін зерттеу алдымен әрбір түрлендіруге криптоталдаулар жүргізуден басталады. Содан кейін, алынған нәтижелерге байланысты толық алгоритмге, яғни барлық раундтық түрлендірулерге талдау жасалады. Қазіргі кездегі ең көп тараған әдістердің бірі сызықтық және дифференциалдық криптоталдауға негізделген шабуылдар. Бұл шабуылдарға қарсы әзірленген алгоритмге зерттеу жұмыстары жүргізілген. Сонымен бірге, бумеранг және алгебралық шабуылдарға жататын XL және XSL шабуылдарының нәтижелері келтірілген. Көптеген алгоритмдердің дифференциалдық және сызықтық криптоталдауға беріктілігіне олардың S-блоктары жауап беретіндігі белгілі. Бұл өз кезегінде S-блоктардың қасиеттері туралы үлкен зерттеулерге себеп болды. Алгоритмде қолданылған S-блоктарға да зерттеу жұмыстары жүрзілді. Дифференциалдық және сызықтық криптоталдау бойынша құрылған кестенің өлшемі 256x256 болғандықтан, жұмыста осы кестедегі мәндердің ең үлкендері және ең кішілері берілген. Сондай-ақ, S-блок бойынша алгебралық шабуылдарға қажетті теңдеулер жүйесі алынды және өз кезегінде оларды шабуылдар жүргізуге пайдаланылды.

Сонымен бірге, ПЕПСЖ негізделген шифрлау алгоритміне жекелей зерттеу жүргізілді. Зерттеу нәтижелері осы жүйені шифрлауда пайдалану алгоритмнің криптобериктілігін арттыра түсетіндігін көрсетті. Жұмыс негіздері құпия, яғни ол кілттің бір бөлігі болған жағдайда, шифрға дифференциалдық, сызықтық және т.б. криптоталдаулар жүргізу тиімсіз екендігі анықталды. Шифрлау алгоритмі аталған криптоталдауларға және осы талдауларды негізге ала отырып жасалатын басқа да шабуылдарға берік болатындығы көрсетілді.

Құрылған шифрлау алгоритмі бірінші бөлімде келтірілген ақпаратты шифрлау алгоритмдеріне қойылатын жалпы талаптарды қанағаттандарады. Алгоритмді құру және зерттеу барысында сол талаптар орындалуы керектілігі

ескерілді. Яғни, алгоритм криптоберіктілік деңгейін қамтамасыз етеді; мәтінге енгізілген аз ғана өзгеріс, сол кілтті қолданған жағдайда, шифрмәтін айтарлықтай өзгеруіне әкеледі; таңдалған мәтінге негізделген шабуылдарға қарсы тұрады; әр түрлі платформаларда жүзеге асыруға икемді; құрамында микропроцессорларда қолдануға тиімді қарапайым операциялар пайдаланылды; кілт ретінде қажетті ұзындықтағы кез-келген кездейсоқ бит тізбегін қабылдай алады; криптоталдауды жеңілдететін әлсіз кілттері жоқ; қауіпсіздіктің әртүрлі деңгейлеріне сай оңай жетілдіруге болады және оның ең төменгі және ең жоғары талаптарына сай. Қойылатын талаптардың бірі алгоритм бағдарламалықпен қатар, аппараттық іске асыруға да тиімді болуы керек. Бұл салада, ПЕПСЖ аппараттық іске асыруда отандық ғалымдар Тынымбаев С., Айтқожаева Е.Ж. және т.б. еңбектері бар.

Негізгі талаптардың қатарына шифрлау жылдамдығы да жатады. Жұмыста оларды жүзеге асырудың ең жылдам тәсілдерін анықтау үшін шифрлау алгоритмінде қолданылған Mixer2 түрлендірулерін есептеудің үш әдісі қарастырылған. Галуа өрісінде көбейту, бекітілген параметрлерге қатысты теңдеулерді қолдану және үлкен сандарды модуль бойынша көбейтуді биттік жылжыту операциялары арқы іске асыру пайдаланылды. Мәліметтерді енгізу нұсқалары қарастырылды, олардың орындалу уақытын анықтауға арналған тесттер жасалды және тексерілді. Нәтижесінде Mixer2 түрлендіруін жүзеге асырудың ең жақсы тәсілі анықталды. Mixer1 және Mixer2 түрлендірулерінде біреуі баған бойынша, екіншісі жол бойынша араластыруды қамтамасыз етеді және әрбір бағанда немесе жолда түрлендірулер бір-біріне тәуелсіз орындалатындықтан оларды есептеуді параллель жүзеге асыруға болады. Оған қазіргі компьютерлердің мүмкіндігі бар.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

1. Kamol Lek, Naruemol Rajapakse. Cryptography: Protocols, Design, and Applications. – Nova Science Publishers, 2012. – 242p.
2. Keith Martin. Everyday Cryptography: Fundamental Principles and Applications. – Oxford University Press, 2012. – 560p.
3. Гатченко Н.А., Исаев А.С., Яковлев А.Д. «Криптографическая защита информации». – СПб: НИУ ИТМО, 2012. – 142с.
4. Camel Tanougast. Progress in Data Encryption Research. – Nova Science Publishers Inc, – 2013. – 158p.
5. Ященко В.В. Введение в криптографию.– 4-е изд., доп. М.: МЦНМО, 2012. – 348 с.
6. Douglas R. Stinson, Maura B. Paterson. Cryptography: Theory and Practice. – Boca Raton - CRC Press, Taylor & Francis Group, 2019. – 580p.
7. Токарева Н.Н. Симметричная криптография. – НГУ, Новосибирск, 2012. – 234 с.
8. Концепция кибербезопасности («Киберщит Казахстана»): утв. постановлением Правительства Республики Казахстан от 30 июня 2017 года № 407.
9. R. G. Bijashev, S. E. Nyssanbayeva. Algorithm for Creation a Digital Signature with Error Detection and Correction // Cybernetics and Systems Analysis. – 2012. – Vol. 48, № 4. – pp. 489-497.
10. Biyashev R., Nyssanbayeva S., Kapalova N. The Key Exchange Algorithm on Basis of Modular Arithmetic // Proceedings of International Conference on Electrical, Control and Automation Engineering (ECAE2013), Hong Kong**.** – Lancaster, U.S.A.:DEStech Publications, 2013. – P.16.
11. Kapalova N., Haumen A. The model of encryption algorithm based on non-positional polynomial notations and constructed on an SP-network // Open Engineering – 2018. – Vol. 8, Issue 1. – P. 140-146.
12. Kapalova N., Dyusenbayev D. Security analysis of an encryption scheme based on nonpositional polynomial notations // Open Engineering – 2016. –№ 6. – P. 250-258.
13. Амербаев В.М., Бияшев Р.Г., Нысанбаева С.Е. Применение непозиционных систем счисления при криптографической защите // Известия Национальной академии наук Республики Казахстан. Серия физико-математическая. – Алматы: Гылым, 2005. – № 3. – С. 84-89.
14. Бияшев Р.Г., Нысанбаева С.Е. Алгоритм формирования электронной цифровой подписи с возможностью обнаружения и исправления ошибки // Кибернетика и системный анализ. – 2012. – Т. 48, № 4. – С. 14-23.
15. Biyashev R., Nyssanbayeva S., Kapalova N., Khakimov R. Modular models of the cryptographic protection of information // International Conference on Computer Networks and Information Security (CNIS2015). – Changsha, China. 2015. – P.393-398 (Thomson Reuters).

16. Biyashev R.G., Kalimoldayev M.N., Nyssanbayeva S.E., Kapalova N.A., Dyusenbayev D.S., Algazy K.T. Development and analysis of the encryption algorithm in nonpositional polynomial notations // Eurasian Journal of Mathematical and Computer Applications. – 2018. – № 6(2). – С. 19-33.

17. Бияшев Р.Г., Нысанбаева С. Е., Капалова Н.А. Секретные ключи для непозиционных криптосистем. Разработка, исследование и применение. – LAB LAMBERT Academic Publishing, 2014. – С. 126.

18. Капалова Н.А., Хомпыш А. Позициялық емес санау жүйесін қолданып, Эль-Гамаль шифрлау алгоритмінің модификациясын құру // Қ.И. Сәтбаев атындағы Қазақ Ұлттық техникалық зерттеу университетінің, Хабаршысы. – Алматы, 2017. – №4 (122). – 506-510 б.

19. Biyashev R., Nyssanbayeva S., Kapalova N., Haumen A. Modied symmetric block encryption-decryption algorithm based on modular arithmetic // Proceedings of the International Conference on Wireless Communications, Network Security and Signal Processing (WCNSSP2016). – Chiang Mai, Thailand. – 2016. –P. 263-265.

20. Biyashev R. G., Nyssanbayeva S. E. Algorithm for Creation a Digital Signature with Error Detection and Correction // Cybernetics and Systems Analysis. – 2012. – Vol.48. – № 4. – P. 489-497.

21. Петров В.А., Пискарев А.С., Шеин А.В. Информационная безопасность. Защита информации от несанкционированного доступа в автоматизированных системах. – М.: МИФИ, 1995.

22. Скляров Д. Искусство защиты и взлома информации. – СПб.: БХВ-Петербург, 2004 – 288 с.

23. Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф. Защита информации в компьютерных системах и сетях. – М.: Радио и связь, 1999.

24. Алферов А.П., Зубов А.Ю., Кузьмин А.С., Черемушкин А.В. Основы криптографии: Учебное пособие. – М.: Гелиос АРВ, 2002. – 480 с.

25. К. Шеннон. Работы по теории информации и кибернетике. – М: ИЛ, 1963. – С. 333–369.

26. Зубов А.Ю. Криптографические методы защиты информации. Совершенные шифры. – М.: Гелиос АРВ, 2005.

27. Хоффман Л.Д. Современные методы защиты информации / Под ред. В.А. Герасименко. – М.: Сов. радио, 1980. – 264 с.

28. Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. – М.: Издательство ТРИУМФ, 2002. – 816 с.

29. Баричев С.Г., Гончаров В.В., Серов Р.Е. Основы современной криптографии. – М.: Горячая линия-Телеком, 2001.

30. Фомичев В.М. Симметричные криптосхемы. Краткий обзор основ криптологии для шифрсистем с открытым ключом. – М.: МИФИ, 1995.

31. Асамбаев А.Ж. Криптография негіздері. Оқу құралы. – Павлодар, 2012. – 173 бет.

32. Ростовцев А. Алгебраические основы криптографии. – СПб: Мир и Семья, 2000.

33. Нечаев В.И. Элементы криптографии. Основы теории защиты информации. – М.: Высшая школа, 1999.

34. Ахметов Б.А., Корченко А.Г., Сиденко В.П., Дрейс Ю.А., АлимсеитоваЖ.К. Қолданбалы криптология: шифрлау әдістері. – Алматы: Қ. И. Сәтбаев атындағы ҚазҰТЗУ, 2016. – 500 б.

35. Иванов М.А. Криптографические методы защиты информации в компьютерных системах и сетях. – М.: Кудиц-Образ, 2001. – 368 с.

36. Панасенко С.П. Алгоритмы шифрования. Специальный справочник. – СПб.: БХВ-Петербург, 2009 – 576 с.

37. Молдовян А. А. Криптография: скоростные шифры. – СПб.: БХВ-Петербург, 2002. – 496 с.

38. Столлингс В. Криптография и защита сетей: принципы и практика. – М.: «Вильяме», 2002. – 672 с.

39. Варфоломеев А.А., Варфоломеев А.А., Жуков А.Е. Блочные криптосистемы. Основные свойства и методы анализа стойкости. – М.: МИФИ, 1998. – 247 с.

40. Гундарь К.Ю., Гундарь А.Ю., Янишевский Д.А., Защита информации в компьютерных системах. – Киев: Корншчук, 2000. –154 с.

41. Диффи У., Хеллман М.Э. Защищенность и имитостойкость: Введение в критпографию. – ТИИЭР, 1976. – Т.67, № 3. – С. 71-109.

42. Тилборг ван Х.К.А. Основы криптологии. Профессиональное руководство и итерактивный учебник. – М.: Мир, 2006. – 471 с.

43. Домарев В.В. Защита информации и безопасность компьютерных систем. – Киев: Diasoft, 1999. – 480 с.

44. Смарт Н. Криптография. – М.: Техносфера, 2005. – 528 с.

45. Фергюссон Н., Шнайер Б. Практическая криптография. – М.: Вильямс, 2005. – 424 с.

46. Диффи У., Хеллмэн М.Э. Защищенность и имитостойкость: введение в криптографию. // ТИИЭР. – 1979 г. – № 3. – Т.67. – С. 71-109.

47. Бабенко Л.К., Ищукова Е.А. Современные алгоритмы блочного шифрования и методы их анализа. – М.: Гелиос АРВ, 2006. – 376 с.

48. Закон Республики Казахстан «О государственных секретах»: утв. 15 марта 1999 г. № 349-I.

49. Закон Республики Казахстан «Об электронном документе и электронной цифровой подписи». / Утв. 7 января 2003 г. № 370-II.

50. Постановление Правительства Республики Казахстан. Правила электронного документооборота государственных органов Республики Казахстан / Утв. 17 апреля 2004 г. № 430.

51. СТ РК 1073-2007. Средства криптографической защиты информации / Общие технические требования. / Утв. 2009. 01.01. – Астана: 2009.

52. Концепции информационной безопасности Республики Казахстан/ Утв. 10 октября 2006 года N 199.

53. Киберқауіпсіздік тұжырымдамасы («Қазақстанның киберқалқаны») / Қазақстан Республикасы Үкіметінің 2017 жылғы 30 маусымдағы № 407 қаулысымен бекітілген.

54. Акушский И.Я., Юдицкий Д.И. Машинная арифметика в остаточных классах. – М.: Советское радио, 1968.– 439 с.

55. Амербаев В.М., Бияшев Р.Г. Интерполяция и коды, исправляющие ошибки // Теория кодирования и информационное моделирование. – Алма-Ата, 1973. – С. 51-64.

56. Бияшев Р.Г. Разработка и исследование методов сквозного повышения достоверности в системах обмена данными распределенных АСУ: дисс. д.т.н: 05.13.06: защищена 09.10. 1985: утв. 28.03.1986. – М., 1985. – 328 с.

57. Nursulu Kapalova, Ardabek Khompysh, Müslüm Arici, Kunbolat Algazy. A block encryption algorithm based on exponentiation transform // Cogent Engineering. – 2020. – № 7 (1788292). – P. 1-12.

58. R.G. Biyashev, A. Smolarz, K. T. Algazy, A.Khompysh. Encryption algorithm "Qamal NPNS" based on a nonpositional polynomial notation // Вестник КазНУ им. аль-Фараби. – Алматы, 2020. – № 1. – С. 198-208.

59. Algazy K., Biyashev R., Kapalova N., Babenko L., Ishchukova E., Nyssanbayeva S. Investigation of the different implementations for the new cipher Qamal. // Proceedings of the 12th International Conference on Security of Information and Networks. – 2019. – P. 1-8.

60. Алгазы К.Т., Бабенко Л.К., Бияшев Р.Г., Ищукова Е.А., Капалова Н.А., Нысанбаева С.Е. Исследование дифференциальных свойств нового Алгоритма шифрования Qamal // Матер. межд. науч.-практ. конф. «Актуальные проблемы информационной безопасности в Казахстане». – Алматы, 2020. – С. 97-105.

61. K.T. Algazy, L.K. Babenko, R.G. Biyashev, E.A. Ishchukova, N.A. Kapalova, S.E. Nysynbaeva, Andrzej Smolarz. Differential Cryptanalysis of New Qamal Encryption Algorithm // International journal of electronics and telecommunications, № 4, 2020, P. 647-653.

62. Зензин О.С., Иванов М.А. Стандарт криптографической защиты - AES. Конечные поля. – М.: КУДИЦ-ОБРАЗ, 2002. – 176 с.

63. Кнут Д.Э. Искусство программирования. – Т.2. – М.: Вильямс, 2007. – 832 с.

64. Иванов М.А., Михайлов Д.М., Чугунков И.В. и др. Стохастические методы и средства защиты информации в компьютерных системах и сетях. – М.: Кудиц-Пресс, 2009. – 512 с.

65. Бияшев Р.Г., Капалова Н.А., Алғазы К.Т., Дюсенбаев Д.С., Хомпыш А. Криптоанализ генератора псевдослучайных последовательностей и ее модификация // Вестник КазНИТУ. – 2019. – №3. – С. 179-185

66. Ключарёв П.Г. О статистическом тестировании блочных шифров // Математика и математическое моделирование. – 2018. – №5. – С. 35-56.

67. Будько М.Б., Будько М.Ю., Гирик А.В., Грозов В.А. Методы генерации и тестирования случайных последовательностей. – СПб: Университет ИТМО, 2019. – 70 с.

68. Капалова Н.А., Хомпыш А., Алгазы К.Т. ЕМ түрлендіру әдісі негізінде жасалған блокты шифрлеу алгоритміне жүргізілген бағалау тесттері // Матер. IV межд. науч.-практ. конф. «Информатика и прикладная математика». – Алматы, 2019. – Ч.II. – С. 580-587.

69. R.G. Biyashev, N.A. Kapalova, D.S. Duysenbayev, K.T. Algazy, Waldemar Wojcik, Andrzej Smolarz Development and Analysis of Symmetric Encryption Algorithm Qamal Based on a Substitution-permutation Network // International journal of electronics and telecommunications, № 1, 2021, P. 127-132

70. Капалова Н.А., Хомпыш А., Алгазы К.Т. Исследование разработанного алгоритма на основе преобразования ЕМ по критерию «лавинного эффекта» // Вестник КазАТК. – Алматы, 2020. – №3. – С.284-292.

71. Vergili I., Yucel M. D. Avalanche and Bit Independence Properties for the Ensembles of Randomly Cho-sen S-Boxes // Turk J Elec Engin. – 2001 – № 2. P. 137–145.

72. Бияшев Р.Г., Алгазы К.Т., Хомпыш А. Исследование разработанных алгоритмов по критерию «лавинного эффекта» // Матер. межд. науч.-практ. конф. «Актуальные проблемы информационной безопасности в Казахстане». – Алматы, 2020. – С. 107-119.

73. Biham E., Shamir A. Differential Cryptanalysis of the Full 16-round DES // Crypto'92. – Springer-Velgar, 1998. – P. 487.

74. Biham E., Shamir A. Differential Cryptanalysis of DES-like Cryptosystems Extended Abstract // Crypto'92. – Springer-Velgar, 1998. – P.21.

75. Ishchukova E.A., Tolomanenko E.A., Babenko L.K. Differential analysis of 3 round Kuznyechik // Proceedings of the 10th international conference on Security of information and networks. – ACM, NY, USA, 2017. – P. 131-137.

76. Дюсенбаев Д.С., Алғазы К.Т., Сақан Қ.С., Хомпыш А. «MODNPSS14» шифрлау алгоритміне криптографиялық талдау // Хабаршы КазККА. – Алматы, 2019. – № 3. –235-243 б.

77. Mitsuru Matsui Linear Cryptanalysis Method for DES Cipher, / T. Helleseth (Ed.): Advances in Cryptology // EUROCRYPT '93, LNCS. – Springer-Verlag, Berlin Heidelberg, 1994. – P. 386-397.

78. Бияшев Р.Г., Алгазы К.Т., Дюсенбаев Д.С., Сакан К.С. Результаты линейного криптоанализа шифра Qamal // Вестник АУЭС. – Алматы, 2020. – № 2. – С. 96-105.

79. Rustem Biyashev, Dilmuhanbet Dyusenbayev, Kunbolat Algazy, Nursulu Kapalova, Algebraic cryptanalysis of block ciphers // International Conference on Wireless Communication, Network and Multimedia Engineering. Advances in Computer Science Research. – Atlantis press, 2019. – Vol. 89. – P. 129-132

80. Капалова Н., Дюсенбаев Д., Сақан Қ., Алғазы К. «AL01» шифрлау алгоритміне криптографиялық талдау // Хабаршы КазҰТЗУ. – Алматы, 2019. – № 5. – 92-98 б.

81. Дюсенбаев Д.С., Алгазы К.Т., Сакан К.С. Исследование алгоритмов шифровании «Al01» и «Qamal» на основе алгебраического криптоанализа // Вестник КазНИТУ. – Алматы, 2020. – № 5. – С. 620-629.

82. N. Courtois, Goubin L., W. Meier, J.D. Tacier, Solving Underdefined Systems of Multivariate Quadratic Equations // Public Key Cryptography 2002, LNCS. – NY, Springer. – 2002. – Vol. 2274. – P. 211-227.

83. Courtois N., Klimov A., Patarin J., Shamir A. Efficient algorithms for solving overdefined systems of multivariate polynomial equations // Advances in Cryptology – EUROCRYPT, 2000, LNCS. – NY, Springer, 2000. – Vol. 1807. – P. 398–413.

84. Courtois N., Pieprzyk J., Cryptanalysis of block ciphers with overdefined systems of equations // Asiacrypt, 2002, LNCS. – NY, Springer, 2002. – Vol. 2501. – P. 267-287.

85. Nicolas Courtois and Josef Pieprzyk. Cryptanalysis of Block Ciphers with Overdefined Systems of Equations // Cryptology ePrint Archive, Report 2002/044. – 2002.

86. Nicolas Courtois, Algebraic Attacks over GF(2^k) // Applications to HFE Challenge 2 and Sflash-v2. LNCS. – Springer-Verlag, 2004. – Vol. 2947. – P. 201-217.

87. Nicolas Courtois and Jacques Patarin. About the XL algorithm over GF(2) // In M. Joye, editor, Progress in Cryptology - CT-RSA 2003. – Springer-Verlag, 2003. – P. 140-156.

88. Л.К. Бабенко, Е.А. Маро. Анализ стойкости блочных алгоритмов шифрования к алгебраическим атакам // Известия ЮФУ. Технические науки. – 2011. – № 12. – С. 110-119.

89. D. Wagner, The boomerang attack in Fast Software Encryption // FSE'99, LNCS. – Springer-Verlag, 1999. – Vol. 1636 – P. 156–170.

90. J. Chen, A. Miyaji Differential Cryptanalysis and Boomerang Cryptanalysis of LBlock // International Conference on Availability, Reliability and Security CD-ARES 2013: Security Engineering and Intelligence Informatics. – Regensburg, Germany, 2013. – Vol. 8128. – P. 1-15.

91. Дюсенбаев Д., Сақан Қ., Криптографическая атака на алгоритм «Qamal» методом бумеранга // Матер. межд. науч.-практ. конф. «Актуальные проблемы информационной безопасности в Казахстане». – Алматы, 2020. – С. 123-129.

92. Дюсенбаев Д.С., Алғазы К.Т., Позициялы емес полиномды санау жүйесіне негізделген шифрлау алгоритміне дифференциалдық криптоталдау // Труды V межд. науч.-практ. конф. «Информатизация общества». – Астана: Евразийский Национальный Университет им. Л.Н. Гумилева, 2016. – С. 386-387.

93. Сакан К.С., Алгазы К.Т., Дюсенбаев Д.С. О некоторых способах улучшения производительности вычисления блока mixer2 алгоритма шифрования «Qamal» // Вестник КазНИТУ. – Алматы, 2020. – № 4. – С. 492-499.

94. М.Вельшенбах. Криптография на Си и С++ в действии. – М.: Триумф, 2004. – 443 с.

ҚОСЫМША А

Жарияланымдар тізімі

1 Капалова Н.А., Хаумен А., Дюсенбаев Д.С., Алгазы К.Т. Линейные преобразования в современных симметричных блочных алгоритмах шифрования // Материалы III международной научно-практтической конференции «Информатика и прикладная математика», - Алматы, 2018. – Ч.2, – С. 213-220.

2 Қапалова Н.А., Алгазы К.Т., Хомпыш А. Модуль бойынша дәрежеге шығару негізінде ақпаратты криптографиялық қорғау алгоритмінің модификациясы // Хабаршы ҚазККА. – Алматы, 2019. – № 4. – 247-253 б.

3 Rustem Biyashev, Dilmuhanbet Dyusenbayev, Kunbolat Algazy, Nursulu Kapalova, Algebraic cryptanalysis of block ciphers // International Conference on Wireless Communication, Network and Multimedia Engineering. Advances in Computer Science Research. - Atlantis press, 2019. – Vol. 89. - P. 129-132.

4 Бияшев Р.Г., Капалова Н.А., Алғазы К.Т., Дюсенбаев Д.С., Хомпыш А., Криптоанализ генератора псевдослучайных последовательностей и ее модификация // Вестник КазНИТУ. – 2019.- №3. - С. 179-185

5 Капалова Н.А., Хомпыш А., Алгазы К.Т. ЕМ түрлендіру әдісі негізінде жасалған блокты шифрлеу алгоритміне жүргізілген бағалау тесттері // Матер. IV междунар. научно-практ. конф. «Информатика и прикладная математика». – Алматы, 2019. – Ч.II. – С. 580-587.

6 Бияшев Р.Г., Алгазы К.Т., Дюсенбаев Д.С., Ержанов Е.Б. Результаты проверки «лавинного эффекта» алгоритма «AL01» // Матер. IV междунар. науч.-практ. конф. «Информатика и прикладная математика». – Алматы, 2019. – Ч.II. – С. 602-607.

7 Algazy K., Biyashev R., Kapalova N., Babenko L., Ishchukova E., Nyssanbayeva S. Investigation of the different implementations for the new cipher Qamal. // Proceedings of the 12th International Conference on Security of Information and Networks. – 2019. – P. 1-8. (в базе Scopus)

8 Капалова Н., Дюсенбаев Д., Сақан Қ., Алғазы К. «AL01» шифрлау алгоритміне криптографиялық талдау // Хабаршы КазҰТЗУ. – Алматы, 2019. – № 5. – 92-98 б.

9 Дюсенбаев Д.С., Алғазы К.Т., Сақан Қ.С., Хомпыш А. «MODNPSS14» шифрлау алгоритміне криптографиялық талдау // Хабаршы КазККА. – Алматы, 2019. – № 3. –235-243 б.

10 Алгазы К.Т., Бабенко Л.К., Бияшев Р.Г., Ищукова Е.А., Капалова Н.А., Нысанбаева С.Е. Исследование дифференциальных свойств нового Алгоритма шифрования Qamal // Матер. междунар. науч.-практ. конф. «Актуальные проблемы информационной безопасности в Казахстане». – Алматы, 2020. – С. 97-105.

11 Бияшев Р.Г., Алгазы К.Т., Хомпыш А. Исследование разработанных алгоритмов по критерию «лавинного эффекта» // Матер. междунар. науч.-практ.

конф. «Актуальные проблемы информационной безопасности в Казахстане». – Алматы, 2020. – С. 107-119.

12 R.G. Biyashev, A. Smolarz, K. T. Algazy, A.Khompysh. Encryption algorithm "Qamal NPNS" based on a nonpositional polynomial notation // Вестник КазНУ им. аль-Фараби. – Алматы, 2020. – № 1. – С. 198-208.

13 Бияшев Р.Г., Алгазы К.Т., Дюсенбаев Д.С., Сакан К.С. Результаты линейного криптоанализа шифра Qamal // Вестник АУЭС. – Алматы, 2020. – № 2. – С. 96-105.

14 Сакан К.С., Алгазы К.Т., Дюсенбаев Д.С., О некоторых способах улучшения производительности вычисления блока mixer2 алгоритма шифрования «Qamal» // Вестник КазНИТУ. – Алматы, 2020. – № 4. – С. 492-499.

15 Nursulu Kapalova, Ardabek Khompysh, Müslüm Arici, Kunbolat Algazy. A block encryption algorithm based on exponentiation transform // Cogent Engineering. – 2020. – № 7 (1788292). – P. 1-12 // https://doi.org/10.1080/23311916.2020.1788292 (квартиль Q2, процентиль - 62).

16 Дюсенбаев Д.С., Алгазы К.Т., Сакан К.С. Исследование алгоритмов шифровании «Al01» и «Qamal» на основе алгебраического криптоанализа // Вестник КазНИТУ. – Алматы, 2020. – № 5. – С. 620-629.

17 Сакан К.С., Алгазы К.Т. Криптографиялық хеш алгоритмдер жасаудың әртүрлі жолдарын қарастыру // Матер. междунар. науч.-практ. конф. "Информатика и прикладная математика". – Алматы, 2020. – С. 374-378.

18 Алгазы К.Т., Сакан К.С. Принципы построения блочных шифров и требования к ним // Матер. V междунар. науч.-практ. конф. "Информатика и прикладная математика". – Алматы, 2020. – С. 378-384.

19 Капалова Н.А., Хомпыш А., Алгазы К.Т. Исследование разработанного алгоритма на основе преобразования ЕМ по критерию «лавинного эффекта» // Вестник КазАТК. – Алматы, 2020. - №3. - С.284-292.

20 K.T. Algazy, L.K. Babenko, R.G. Biyashev, E.A. Ishchukova, N.A. Kapalova, S.E. Nysynbaeva, Andrzej Smolarz Differential Cryptanalysis of New Qamal Encryption Algorithm // International journal of electronics and telecommunications, № 4, 2020, P. 647-653 (процентиль - 27).

21 R.G. Biyashev, N.A. Kapalova, D.S. Duysenbayev, K.T. Algazy, Waldemar Wojcik, Andrzej Smolarz Development and Analysis of Symmetric Encryption Algorithm Qamal Based on a Substitution-permutation Network // International journal of electronics and telecommunications, № 1, 2021, P. 127-132 (процентиль - 27).

22 А. к. 5200. Файлдарда шифрлауға арналған бағдарлама «Qamal v 1.0.1» / Бияшев Р.Г., Қапалова Н. А., Алғазы К.Т., Дюсенбаева Д.С., Сақан Қ.С.; жариял. 06.09.2019. – 1 б.

ҚОСЫМША Ә

Лицензия және авторлық куәліктер

Ақпаратты криптографиялық қорғау құралдарын (оның ішінде басқа да түрлерін) әзірлеуге және енгізуге мемлекеттік лицензия

17012125

ГОСУДАРСТВЕННАЯ ЛИЦЕНЗИЯ

03.07.2017 года **549**

Выдана **Республиканское государственное предприятие на праве хозяйственного ведения "Институт информационных и вычислительных технологий" Комитета науки Министерства образования и науки Республики Казахстан**

050010, Республика Казахстан, г.Алматы, УЛИЦА ПУШКИНА, дом № 125., БИН: 040740002672

(полное наименование, местонахождение, бизнес-идентификационный номер юридического лица (в том числе иностранного юридического лица), бизнес -идентификационный номер филиала или представительства иностранного юридического лица – в случае отсутствия бизнес-идентификационного номера у юридического лица/полностью фамилия, имя, отчество (в случае наличия), индивидуальный идентификационный номер физического лица)

на занятие **На осуществление деятельности по разработке и реализации (в том числе иной передаче) средств криптографической защиты информации**

(наименование лицензируемого вида деятельности в соответствии с Законом Республики Казахстан «О разрешениях и уведомлениях»)

Особые условия

(в соответствии со статьей 36 Закона Республики Казахстан «О разрешениях и уведомлениях»)

Примечание **Неотчуждаемая, класс 1**

(отчуждаемость, класс разрешения)

Лицензиар **Комитет национальной безопасности Республики Казахстан**

(полное наименование лицензиара)

Руководитель (уполномоченное лицо) МАСИМОВ КАРИМ КАЖИМКАНОВИЧ

(фамилия, имя, отчество (в случае наличия)

Дата первичной выдачи

Срок действия лицензии

Место выдачи **г.Астана**

ПРИЛОЖЕНИЕ К ГОСУДАРСТВЕННОЙ ЛИЦЕНЗИИ

Номер лицензии 549

Дата выдачи лицензии 03.07.2017 год

Подвид(ы) лицензируемого вида деятельности:

- Реализация (в том числе иная передача) средств криптографической защиты информации
- Разработка средств криптографической защиты информации

(наименование подвида лицензируемого вида деятельности в соответствии с Законом Республики Казахстан «О разрешениях и уведомлениях»)

Лицензиат **Республиканское государственное предприятие на праве хозяйственного ведения "Институт информационных и вычислительных технологий" Комитета науки Министерства образования и науки Республики Казахстан**

050010, Республика Казахстан, г.Алматы, УЛИЦА ПУШКИНА, дом № 125., БИН: 040740002672

(полное наименование, местонахождение, бизнес-идентификационный номер юридического лица (в том числе иностранного юридического лица), бизнес-идентификационный номер филиала или представительства иностранного юридического лица – в случае отсутствия бизнес-идентификационного номера у юридического лица/полностью фамилия, имя, отчество (в случае наличия), индивидуальный идентификационный номер физического лица)

Производственная база

(местонахождение)

Особые условия действия лицензии

(в соответствии со статьей 36 Закона Республики Казахстан «О разрешениях и уведомлениях»)

Лицензиар **Комитет национальной безопасности Республики Казахстан**

(полное наименование органа, выдавшего приложение к лицензии)

Руководитель (уполномоченное лицо) МАСИМОВ КАРИМ КАЖИМКАНОВИЧ

(фамилия, имя, отчество (в случае наличия)

Номер приложения 001

Срок действия

Дата выдачи приложения 03.07.2017

Место выдачи г.Астана

Осы құжат «Электронды құжат және электрондық цифрлық қолтаңба туралы» Қазақстан Республикасының 2003 жылғы 7 қаңтардағы Заңы 7 бабының 1 тармағына сәйкес қағаз тасығыштағы құжатқа теңестірілген. Данный документ согласно пункту 1 статья 7 ЗРК от 7 января 2003 года "Об электронном документе и электронной цифровой подписи" равнозначен документу на бумажном носителе.

Авторлық куәлік № 5200 –
«Qamal v1.0.1» Файлдарды шифрлау бағдарламасы

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ

РЕСПУБЛИКА КАЗАХСТАН

АВТОРЛЫҚ ҚҰҚЫҚПЕН ҚОРҒАЛАТЫН ОБЪЕКТІЛЕРГЕ ҚҰҚЫҚТАРДЫҢ МЕМЛЕКЕТТІК ТІЗІЛІМГЕ МӘЛІМЕТТЕРДІ ЕНГІЗУ ТУРАЛЫ

КУӘЛІК

2019 жылғы «6» қыркүйек № 5200

Автордың (лардың) жөні, аты, әкесінің аты (егер ол жеке басын куәландыратын құжатта көрсетілсе):
САКАН ҚАЙРАТ САҚАНҰЛЫ, БИЯШЕВ РУСТЕМ ГАКАШЕВИЧ, КАПАЛОВА НУРСУЛУ АЛЛАЖАРОВНА, ЛЮСЕНБАЕВ ДИЛМУХАНБЕТ САМУРАТОВИЧ, АЛҒАЗЫ ҚҰНБОЛАТ ТІЛЕУХАНҰЛЫ

Авторлық құқық объектісі: ЭЕМ-ге арналған бағдарлама

Объектінің атауы: Программа шифрования файлов «Qamal v1.0.1»

Объектіні жасаған күні: 05.09.2019

Құжат түпнұсқалығын http://www.kazpatent.kz/ru сайтының "Авторлық құқық" бөлімінде тексеруге болады https://copyright.kazpatent.kz

Подлинность документа возможно проверить на сайте kazpatent.kz в разделе «Авторское право» https://copyright.kazpatent.kz

Подписано ЭЦП

Оспанов Е.К.

ҚОСЫМША Б

Енгізу актісі

УТВЕРЖДАЮ
Зам. генерального директора
Института информационных
и вычислительных технологий
PhD Мамырбаев О.Ж.
_____
«21» 01 2021

АКТ
о внедрении результатов диссертационной работы
Алғазы Күнболат Тілеуханұлы

Экспертная комиссия «Института информационных и вычислительных технологий» (ИИВТ) Комитета науки МОН РК в составе:

председатель: заместитель генерального директора по связям с общественностью Айнакулов С.Ж;

члены: д.т.н., ассоциированный профессор, ГНС ЛИБ ИИВТ Нысанбаева С.Е.;
к.т.н., ВНС ЛИБ ИИВТ Капалова Н.А.;
PhD, ученый секретарь ИИВТ Усатова О.А.

составили настоящий акт о том, что результаты диссертационной работы «Разработка и исследование алгоритмов шифрования на базе различных подходов» НС ЛИБ ИИВТ Алғазы К.Т. были получены при выполнении проектов РГП на ПХВ «ИИВТ» КН МОН РК, № гос. регистрации - 0118РК01064), источник финансирования Комитет науки МОН РК:

– программно-целевого финансировании (ПЦФ) КН МОН РК «Разработка программных и программно-аппаратных средств для криптографической защиты информации при ее передаче и хранении в инфокоммуникационных системах и сетях общего назначения» на 2018-2020 годы.

Полученные результаты включены в отчёты проектов ПЦФ за 2018 - 2020 годы.

Краткое содержание внедренных результатов:

1. разработан алгоритм симметричного блочного шифрования;
2. разработан алгоритм генерации раундовых ключей;
3. для исследования надежности разработанного алгоритма симметричного блочного шифрования применялись методы линейного, алгебраического и дифференциального криптоанализа и «атака» методом бумеранга;
4. осуществлена программная реализация разработанного алгоритма симметричного блочного шифрования.

Материалы, указанные в настоящем акте, были рассмотрены на Ученом Совете института (протокол № 1 от 21 января 2021 год).

Председатель комиссии _____ Айнакулов С.Ж.

Члены комиссии _____ Нысанбаева С.Е.

_____ Капалова Н.А.

_____ Усатова О.А.

ҚОСЫМША В

Криптоталдауда қолданылған формулалар

Формула В.1 – Mixer2 түрлендіруінің шығыс мәндерінің сызықты өрнектелуі, мұндағы «+» - биттік қосуды білдіреді

$z_{127}=x_{96}+x_{98}+x_{100}+x_{106}+x_{110}+x_{112}+x_{114}+x_{115}+x_{116}+x_{118}+x_{119}+x_{126}+x_{125}+x_{124}$

$z_{126}=x_{97}+x_{99}+x_{105}+x_{109}+x_{111}+x_{113}+x_{114}+x_{115}+x_{117}+x_{118}+x_{125}+x_{127}+x_{124}+x_{123}$

$z_{125}=x_{96}+x_{98}+x_{104}+x_{108}+x_{110}+x_{112}+x_{113}+x_{114}+x_{116}+x_{117}+x_{127}+x_{124}+x_{126}+x_{123}+x_{122}$

$z_{124}=x_{97}+x_{103}+x_{107}+x_{109}+x_{111}+x_{112}+x_{113}+x_{115}+x_{116}+x_{126}+x_{123}+x_{125}+x_{122}+x_{121}+x_{127}$

$z_{123}=x_{96}+x_{102}+x_{106}+x_{108}+x_{110}+x_{111}+x_{112}+x_{114}+x_{115}+x_{125}+x_{127}+x_{122}+x_{124}+x_{121}+x_{120}+x_{126}$

$z_{122}=x_{101}+x_{105}+x_{107}+x_{109}+x_{110}+x_{111}+x_{113}+x_{114}+x_{127}+x_{124}+x_{126}+x_{121}+x_{123}+x_{120}+x_{119}+x_{125}$

$z_{121}=x_{100}+x_{104}+x_{106}+x_{108}+x_{109}+x_{110}+x_{112}+x_{113}+x_{126}+x_{123}+x_{125}+x_{120}+x_{122}+x_{119}+x_{127}+$
$+x_{118}+x_{124}$

$z_{120}=x_{99}+x_{103}+x_{105}+x_{107}+x_{108}+x_{109}+x_{111}+x_{112}+x_{125}+x_{122}+x_{124}+x_{119}+x_{121}+x_{118}+x_{126}+$
$+x_{117}+x_{123}$

$z_{119}=x_{98}+x_{102}+x_{104}+x_{106}+x_{107}+x_{108}+x_{110}+x_{111}+x_{124}+x_{121}+x_{123}+x_{118}+x_{120}+x_{117}+x_{125}+x_{116}+$
$+x_{122}$

$z_{118}=x_{97}+x_{101}+x_{103}+x_{105}+x_{106}+x_{107}+x_{109}+x_{110}+x_{123}+x_{120}+x_{122}+x_{117}+x_{119}+x_{116}+x_{124}+$
$+x_{115}+x_{121}$

$z_{117}=x_{96}+x_{100}+x_{102}+x_{104}+x_{105}+x_{106}+x_{108}+x_{109}+x_{122}+x_{119}+x_{121}+x_{116}+x_{118}+x_{115}+x_{123}+$
$+x_{114}+x_{120}$

$z_{116}=x_{99}+x_{101}+x_{103}+x_{104}+x_{105}+x_{107}+x_{108}+x_{121}+x_{118}+x_{120}+x_{115}+x_{117}+x_{114}+x_{122}+x_{113}+$
$+x_{119}+x_{127}$

$z_{115}=x_{98}+x_{100}+x_{102}+x_{103}+x_{104}+x_{106}+x_{107}+x_{120}+x_{117}+x_{119}+x_{114}+x_{116}+x_{113}+x_{121}+x_{112}+$
$+x_{118}+x_{126}$

$z_{114}=x_{97}+x_{99}+x_{101}+x_{102}+x_{103}+x_{105}+x_{106}+x_{119}+x_{116}+x_{118}+x_{113}+x_{115}+x_{112}+x_{120}+x_{111}+x_{117}+$
$+x_{125+}x_{127}$

$z_{113}=x_{96}+x_{98}+x_{100}+x_{101}+x_{102}+x_{104}+x_{105}+x_{118}+x_{115}+x_{117}+x_{112}+x_{114}+x_{111}+x_{119}+x_{110}+x_{116}+$
$+x_{124}+x_{126}$

$z_{112}=x_{97}+x_{99}+x_{100}+x_{101}+x_{103}+x_{104}+x_{117}+x_{114}+x_{116}+x_{111}+x_{113}+x_{110}+x_{118}+x_{109}+x_{115}+x_{123}+$
$+x_{125}+x_{127}$

$z_{111}=x_{96}+x_{98}+x_{99}+x_{100}+x_{102}+x_{103}+x_{116}+x_{113}+x_{115}+x_{110}+x_{112}+x_{109}+x_{117}+x_{108}+x_{114}+x_{122}+$
$+x_{124}+x_{126}$

$z_{110}=x_{97}+x_{98}+x_{99}+x_{101}+x_{102}+x_{115}+x_{112}+x_{114}+x_{109}+x_{111}+x_{127}+x_{108}+x_{116}+x_{107}+x_{113}+x_{121}+$
$+x_{123}+x_{125}$

$z_{109}=x_{96}+x_{97}+x_{98}+x_{100}+x_{101}+x_{114}+x_{111}+x_{113}+x_{108}+x_{110}+x_{126}+x_{107}+x_{115}+x_{127}+x_{106}+x_{112}+$
$+x_{120}+x_{122}+x_{124}$

$z_{108}=x_{96}+x_{97}+x_{99}+x_{100}+x_{113}+x_{110}+x_{112}+x_{107}+x_{109}+x_{125}+x_{106}+x_{114}+x_{126}+x_{127}+x_{105}+x_{111}+$
$+x_{119}+x_{121}+x_{123}$

$z_{107}=x_{96}+x_{98}+x_{99}+x_{112}+x_{109}+x_{111}+x_{106}+x_{108}+x_{124}+x_{127}+x_{105}+x_{113}+x_{125}+x_{126}+x_{104}+x_{110}+$
$+x_{118}+x_{120}+x_{122}$

$z_{106}=x_{97}+x_{98}+x_{111}+x_{108}+x_{110}+x_{105}+x_{107}+x_{123}+x_{126}+x_{104}+x_{112}+x_{124}+x_{125}+x_{103}+x_{109}+x_{117}+$
$+x_{119}+x_{121}+x_{127}$

$z_{105}=x_{96}+x_{97}+x_{110}+x_{107}+x_{109}+x_{104}+x_{106}+x_{122}+x_{125}+x_{103}+x_{111}+x_{123}+x_{124}+x_{102}+x_{108}+x_{116}+$

$+x_{118}+x_{120}+x_{127}+x_{126}$

$z_{104}=x_{96}+x_{109}+x_{106}+x_{108}+x_{103}+x_{105}+x_{121}+x_{124}+x_{102}+x_{110}+x_{122}+x_{123}+x_{101}+x_{107}+x_{115}+x_{117}+x_{119}+x_{127}+x_{126}+x_{125}$

$z_{103}=x_{96}+x_{98}+x_{108}+x_{105}+x_{107}+x_{112}+x_{102}+x_{104}+x_{110}+x_{120}+x_{123}++x_{101}+x_{109}+x_{115}+x_{119}++x_{121}+x_{122}$

$z_{102}=x_{97}++x_{107}+x_{104}+x_{106}+x_{111}+x_{101}+x_{103}+x_{109}+x_{119}+x_{122}+x_{100}+x_{108}+x_{114}+x_{118}+x_{120}++x_{121}$

$z_{101}=x_{96}+x_{106}+x_{103}+x_{105}+x_{110}+x_{100}+x_{102}+x_{108}+x_{118}+x_{121}+x_{99}+x_{107}+x_{113}+x_{117}+x_{119}+x_{120}$

$z_{100}=x_{96}+x_{105}+x_{124}+x_{100}+x_{102}+x_{104}+x_{110}+x_{114}+x_{109}+x_{99}+x_{101}+x_{107}+x_{115}+x_{117}+x_{120}+x_{126}++x_{125}$

$z_{99}=x_{96}+x_{104}+x_{110}+x_{123}+x_{99}+x_{101}+x_{103}+x_{109}+x_{113}+x_{115}+x_{108}+x_{112}+x_{118}+x_{126}$

$z_{98}=x_{103}+x_{109}+x_{122}+x_{98}+x_{100}+x_{102}+x_{108}+x_{112}+x_{114}+x_{107}+x_{111}+x_{117}+x_{127}+x_{125}$

$z_{97}=x_{96}+x_{98}+x_{100}+x_{102}+x_{108}+x_{112}+x_{114}+x_{118}+x_{121}+x_{97}+x_{99}+x_{101}+x_{107}+x_{111}+x_{113}+x_{115}++x_{119}+x_{125}$

$z_{96}=x_{97}+x_{99}+x_{101}+x_{107}+x_{111}+x_{113}+x_{115}+x_{116}+x_{117}+x_{119}+x_{120}+x_{127}+x_{126}+x_{125}$

$z_{95}=x_{64}+x_{65}+x_{67}+x_{70}+x_{74}+x_{75}+x_{77}+x_{79}+x_{80}+x_{81}+x_{83}+x_{86}+x_{90}+x_{88}+x_{94}$

$z_{94}=x_{64}+x_{66}+x_{69}+x_{73}+x_{74}+x_{76}+x_{78}+x_{79}+x_{80}+x_{82}+x_{85}+x_{89}+x_{87}+x_{93}$

$z_{93}=x_{65}+x_{68}+x_{72}+x_{73}+x_{75}+x_{77}+x_{78}+x_{79}+x_{81}+x_{84}+x_{88}+x_{95}+x_{86}+x_{92}$

$z_{92}=x_{64}+x_{67}+x_{71}+x_{72}+x_{74}+x_{76}+x_{77}+x_{78}+x_{80}+x_{83}+x_{87}+x_{94}+x_{85}+x_{91}+x_{95}$

$z_{91}=x_{66}+x_{70}+x_{71}+x_{73}+x_{75}+x_{76}+x_{77}+x_{79}+x_{82}+x_{86}+x_{93}+x_{84}+x_{90}+x_{94}$

$z_{90}=x_{65}+x_{69}+x_{70}+x_{72}+x_{74}+x_{75}+x_{76}+x_{78}+x_{81}+x_{85}+x_{95}+x_{92}+x_{83}+x_{89}+x_{93}$

$z_{89}=x_{64}+x_{68}+x_{69}+x_{71}+x_{73}+x_{74}+x_{75}+x_{77}+x_{80}+x_{84}+x_{94}+x_{91}+x_{82}+x_{88}+x_{92}+x_{95}$

$z_{88}=x_{67}+x_{68}+x_{70}+x_{72}+x_{73}+x_{74}+x_{76}+x_{79}+x_{83}+x_{93}+x_{90}+x_{95}+x_{81}+x_{87}+x_{91}+x_{94}$

$z_{87}=x_{66}+x_{67}+x_{69}+x_{71}+x_{72}+x_{73}+x_{75}+x_{78}+x_{82}+x_{92}+x_{89}+x_{94}+x_{80}+x_{86}+x_{90}+x_{93}$

$z_{86}=x_{65}+x_{66}+x_{68}+x_{70}+x_{71}+x_{72}+x_{74}+x_{77}+x_{81}+x_{91}+x_{88}+x_{93}+x_{79}+x_{85}+x_{89}+x_{92}+x_{95}$

$z_{85}=x_{64}+x_{65}+x_{67}+x_{69}+x_{70}+x_{71}+x_{73}+x_{76}+x_{80}+x_{90}+x_{87}+x_{92}+x_{78}+x_{84}+x_{88}+x_{91}+x_{94}$

$z_{84}=x_{64}+x_{66}+x_{68}+x_{69}+x_{70}+x_{72}+x_{75}+x_{79}+x_{89}+x_{86}+x_{91}+x_{95}+x_{77}+x_{83}+x_{87}+x_{90}+x_{93}$

$z_{83}=x_{65}+x_{67}+x_{68}+x_{69}+x_{71}+x_{74}+x_{78}+x_{88}+x_{85}+x_{90}+x_{94}+x_{76}+x_{82}+x_{86}+x_{89}+x_{92}+x_{95}$

$z_{82}=x_{64}+x_{66}+x_{67}+x_{68}+x_{70}+x_{73}+x_{77}+x_{87}+x_{84}+x_{89}+x_{93}+x_{95}+x_{75}+x_{81}+x_{85}+x_{88}+x_{91}+x_{94}$

$z_{81}=x_{65}+x_{66}+x_{67}+x_{69}+x_{72}+x_{76}+x_{86}+x_{95}+x_{83}+x_{88}+x_{92}+x_{94}+x_{74}+x_{80}+x_{84}+x_{87}+x_{90}+x_{93}$

$z_{80}=x_{64}+x_{65}+x_{66}+x_{68}+x_{71}+x_{75}+x_{85}+x_{94}+x_{82}+x_{87}+x_{91}+x_{93}+x_{73}+x_{79}+x_{83}+x_{86}+x_{89}+x_{92}$

$z_{79}=x_{64}+x_{65}+x_{67}+x_{70}+x_{74}+x_{84}+x_{93}+x_{81}+x_{86}+x_{90}+x_{92}+x_{72}+x_{78}+x_{82}+x_{85}+x_{88}+x_{91}$

$z_{78}=x_{64}+x_{66}+x_{69}+x_{73}+x_{83}+x_{92}+x_{80}+x_{85}+x_{89}+x_{91}+x_{71}+x_{77}+x_{81}+x_{84}+x_{87}+x_{90}+x_{95}$

$z_{77}=x_{65}+x_{68}+x_{72}+x_{82}+x_{91}+x_{79}+x_{84}+x_{88}+x_{90}+x_{95}+x_{70}+x_{76}+x_{80}+x_{83}+x_{86}+x_{89}+x_{94}$

$z_{76}=x_{64}+x_{67}+x_{71}+x_{81}+x_{90}+x_{78}+x_{83}+x_{87}+x_{89}+x_{94}+x_{69}+x_{75}+x_{79}+x_{82}+x_{85}+x_{88}+x_{95}+x_{93}$

$z_{75}=x_{66}+x_{70}+x_{80}+x_{89}+x_{77}+x_{82}+x_{86}+x_{88}+x_{93}+x_{68}+x_{74}+x_{78}+x_{81}+x_{84}+x_{87}+x_{94}+x_{92}$

$z_{74}=x_{65}+x_{69}+x_{79}+x_{88}+x_{76}+x_{81}+x_{85}+x_{87}+x_{92}+x_{67}+x_{73}+x_{77}+x_{80}+x_{83}+x_{86}+x_{93}+x_{91}$

$z_{73}=x_{64}+x_{68}+x_{78}+x_{87}+x_{75}+x_{80}+x_{84}+x_{86}+x_{91}+x_{66}+x_{72}+x_{76}+x_{79}+x_{82}+x_{85}+x_{92}+x_{90}$

$z_{72}=x_{67}+x_{77}+x_{86}+x_{74}+x_{79}+x_{83}+x_{85}+x_{90}+x_{65}+x_{71}+x_{75}+x_{78}+x_{81}+x_{84}+x_{91}+x_{89}$

$z_{71}=x_{65}+x_{66}+x_{75}+x_{95}+x_{76}+x_{79}+x_{81}+x_{85}+x_{94}++x_{67}+x_{73}+x_{78}+x_{82}+x_{84}+x_{86}+x_{89}$

$z_{70}=x_{64}+x_{65}++x_{74}+x_{94}+x_{75}+x_{78}+x_{80}+x_{84}+x_{93}+x_{66}+x_{72}+x_{77}+x_{81}+x_{83}+x_{85}+x_{88}$

$z_{69}=x_{64}+x_{73}+x_{93}+x_{74}+x_{77}+x_{79}+x_{83}+x_{92}+x_{65}+x_{71}+x_{76}+x_{80}+x_{82}+x_{84}+x_{87}$

$z_{68}=x_{72}+x_{88}+x_{74}+x_{90}+x_{94}+x_{92}+x_{67}+x_{73}+x_{77}+x_{65}+x_{76}+x_{78}+x_{80}+x_{82}+x_{91}$

$z_{67}=x_{65}+x_{71}+x_{74}+x_{87}+x_{94}+x_{70}+x_{73}+x_{80}+x_{86}+x_{89}+x_{93}+x_{91}+x_{66}+x_{67}+x_{72}+x_{76}+x_{83}+x_{88}$

$z_{66}=x_{64}+x_{70}+x_{73}+x_{86}+x_{93}+x_{69}+x_{72}+x_{79}+x_{85}+x_{88}+x_{92}+x_{90}+x_{65}+x_{66}+x_{71}+x_{75}+x_{82}+x_{87}+x_{95}$

$z_{65}=x_{67}+x_{69}+x_{72}+x_{77}+x_{79}+x_{83}+x_{85}+x_{88}+x_{92}+x_{90}+x_{68}+x_{71}+x_{75}+x_{78}+x_{80}+x_{84}+x_{87}+x_{91}+$

$+x_{89}+x_{95}$

$z_{64}=x_{64}+x_{65}+x_{66}+x_{68}+x_{71}+x_{75}+x_{76}+x_{78}+x_{80}+x_{81}+x_{82}+x_{84}+x_{87}+x_{91}+x_{89}+x_{95}$

$z_{63}=x_{32}+x_{33}+x_{35}+x_{36}+x_{38}+x_{43}+x_{44}+x_{47}+x_{48}+x_{51}+x_{53}+x_{54}+x_{61}+x_{59}+x_{60}$

$z_{62}=x_{32}+x_{34}+x_{35}+x_{37}+x_{42}+x_{43}+x_{46}+x_{47}+x_{50}+x_{52}+x_{53}+x_{60}+x_{63}+x_{58}+x_{59}$

$z_{61}=x_{33}+x_{34}+x_{36}+x_{41}+x_{42}+x_{45}+x_{46}+x_{49}+x_{51}+x_{52}+x_{59}+x_{62}+x_{63}+x_{57}+x_{58}$

$z_{60}=x_{32}+x_{33}+x_{35}+x_{40}+x_{41}+x_{44}+x_{45}+x_{48}+x_{50}+x_{51}+x_{58}+x_{61}+x_{63}+x_{62}+x_{56}+x_{57}$

$z_{59}=x_{32}+x_{34}+x_{39}+x_{40}+x_{43}+x_{44}+x_{47}+x_{49}+x_{50}+x_{57}+x_{60}+x_{62}+x_{61}+x_{55}+x_{56}+x_{63}$

$z_{58}=x_{33}+x_{38}+x_{39}+x_{42}+x_{43}+x_{46}+x_{48}+x_{49}+x_{56}+x_{59}+x_{61}+x_{60}+x_{54}+x_{55}+x_{62}+x_{63}$

$z_{57}=x_{32}+x_{37}+x_{38}+x_{41}+x_{42}+x_{45}+x_{47}+x_{48}+x_{63}+x_{55}+x_{58}+x_{60}+x_{59}+x_{53}+x_{54}+x_{61}+x_{62}$

$z_{56}=x_{36}+x_{37}+x_{40}+x_{41}+x_{44}+x_{46}+x_{47}+x_{62}+x_{54}+x_{57}+x_{59}+x_{58}+x_{52}+x_{53}+x_{60}+x_{61}$

$z_{55}=x_{35}+x_{36}+x_{39}+x_{40}+x_{43}+x_{45}+x_{46}+x_{61}+x_{53}+x_{56}+x_{58}+x_{57}+x_{51}+x_{52}+x_{59}+x_{60}+x_{63}$

$z_{54}=x_{34}+x_{35}+x_{38}+x_{39}+x_{42}+x_{44}+x_{45}+x_{60}+x_{52}+x_{55}+x_{57}+x_{56}+x_{50}+x_{51}+x_{58}+x_{59}+x_{62}$

$z_{53}=x_{33}+x_{34}+x_{37}+x_{38}+x_{41}+x_{43}+x_{44}+x_{59}+x_{63}+x_{51}+x_{54}+x_{56}+x_{55}+x_{49}+x_{50}+x_{57}+x_{58}+x_{61}$

$z_{52}=x_{32}+x_{33}+x_{36}+x_{37}+x_{40}+x_{42}+x_{43}+x_{58}+x_{62}+x_{50}+x_{53}+x_{55}+x_{54}+x_{63}+x_{48}+x_{49}+x_{56}+x_{57}+x_{60}$

$z_{51}=x_{32}+x_{35}+x_{36}+x_{39}+x_{41}+x_{42}+x_{57}+x_{61}+x_{49}+x_{52}+x_{54}+x_{53}+x_{62}+x_{47}+x_{48}+x_{55}+x_{56}+x_{59}+x_{63}$

$z_{50}=x_{34}+x_{35}+x_{38}+x_{40}+x_{41}+x_{56}+x_{60}+x_{48}+x_{51}+x_{53}+x_{52}+x_{61}+x_{46}+x_{47}+x_{54}+x_{55}+x_{58}+x_{62}$

$z_{49}=x_{33}+x_{34}+x_{37}+x_{39}+x_{40}+x_{55}+x_{59}+x_{47}+x_{50}+x_{52}+x_{51}+x_{60}+x_{45}+x_{46}+x_{53}+x_{54}+x_{57}+x_{61}$

$z_{48}=x_{32}+x_{33}+x_{36}+x_{38}+x_{39}+x_{54}+x_{58}+x_{46}+x_{49}+x_{51}+x_{50}+x_{59}+x_{44}+x_{45}+x_{52}+x_{53}+x_{56}+x_{60}$

$z_{47}=x_{32}+x_{35}+x_{37}+x_{38}+x_{53}+x_{57}+x_{45}+x_{48}+x_{50}+x_{63}+x_{49}+x_{58}+x_{43}+x_{44}+x_{51}+x_{52}+x_{55}+x_{59}$

$z_{46}=x_{34}+x_{36}+x_{37}+x_{52}+x_{56}+x_{44}+x_{47}+x_{49}+x_{62}+x_{48}+x_{57}+x_{42}+x_{43}+x_{50}+x_{51}+x_{54}+x_{58}$

$z_{45}=x_{33}+x_{35}+x_{36}+x_{51}+x_{55}+x_{43}+x_{46}+x_{48}+x_{61}+x_{47}+x_{56}+x_{63}+x_{41}+x_{42}+x_{49}+x_{50}+x_{53}+x_{57}$

$z_{44}=x_{32}+x_{34}+x_{35}+x_{50}+x_{54}+x_{42}+x_{45}+x_{47}+x_{60}+x_{46}+x_{55}+x_{62}+x_{40}+x_{41}+x_{48}+x_{49}+x_{52}+x_{56}$

$z_{43}=x_{33}+x_{34}+x_{49}+x_{53}+x_{41}+x_{44}+x_{46}+x_{59}+x_{45}+x_{54}+x_{61}+x_{39}+x_{40}+x_{47}+x_{48}+x_{51}+x_{55}+x_{63}$

$z_{42}=x_{32}+x_{33}+x_{48}+x_{52}+x_{40}+x_{43}+x_{45}+x_{58}+x_{44}+x_{53}+x_{60}+x_{38}+x_{39}+x_{46}+x_{47}+x_{50}+x_{54}+x_{62}+x_{63}$

$z_{41}=x_{32}+x_{47}+x_{51}+x_{39}+x_{42}+x_{44}+x_{57}+x_{43}+x_{52}+x_{59}+x_{37}+x_{38}+x_{45}+x_{46}+x_{49}+x_{53}+x_{63}+x_{61}+x_{62}$

$z_{40}=x_{46}+x_{50}+x_{38}+x_{41}+x_{43}+x_{56}+x_{42}+x_{51}+x_{58}+x_{36}+x_{37}+x_{44}+x_{45}+x_{48}+x_{52}+x_{62}+x_{60}+x_{61}$

$z_{39}=x_{32}+x_{33}+x_{45}+x_{49}+x_{53}+x_{37}+x_{40}+x_{42}+x_{48}+x_{55}++x_{38}+x_{41}+x_{50}+x_{54}+x_{57}$

$z_{38}=x_{32}++x_{44}+x_{48}+x_{52}+x_{36}+x_{39}+x_{41}+x_{47}+x_{54}+x_{37}+x_{40}+x_{49}+x_{53}+x_{56}$

$z_{37}=x_{43}+x_{47}+x_{51}+x_{35}+x_{38}+x_{40}+x_{46}+x_{53}+x_{36}+x_{39}+x_{48}+x_{52}+x_{55}$

$z_{36}=x_{32}+x_{43}+x_{53}+x_{59}+x_{42}+x_{46}+x_{50}+x_{33}+x_{34}+x_{36}+x_{37}+x_{39}+x_{44}+x_{45}+x_{48}+x_{52}+x_{60}+x_{61}$

$z_{35}=x_{42}+x_{48}+x_{52}+x_{58}+x_{41}+x_{54}+x_{45}+x_{49}+x_{53}+x_{63}+x_{61}$

$z_{34}=x_{41}+x_{47}+x_{51}+x_{57}+x_{40}+x_{53}+x_{44}+x_{48}+x_{52}+x_{62}+x_{60}$

$z_{33}=x_{32}+x_{35}+x_{38}+x_{40}+x_{46}+x_{50}+x_{53}+x_{56}+x_{63}+x_{33}+x_{36}+x_{39}+x_{44}+x_{48}+x_{52}+x_{54}+x_{60}$

$z_{32}=x_{33}+x_{34}+x_{36}+x_{37}+x_{39}+x_{44}+x_{45}+x_{48}+x_{49}+x_{52}+x_{54}+x_{55}+x_{62}+x_{60}+x_{61}$

$z_{31}=x_{0}+x_{3}+x_{9}+x_{12}+x_{14}+x_{15}+x_{16}+x_{19}+x_{20}+x_{21}+x_{22}+x_{31}+x_{30}+x_{24}+x_{27}$

$z_{30}=x_{2}+x_{8}+x_{11}+x_{13}+x_{14}+x_{15}+x_{18}+x_{19}+x_{20}+x_{21}+x_{31}+x_{30}+x_{29}+x_{23}+x_{26}$

$z_{29}=x_{1}+x_{7}+x_{10}+x_{12}+x_{13}+x_{14}+x_{17}+x_{18}+x_{19}+x_{20}+x_{30}+x_{29}+x_{28}+x_{22}+x_{25}$

$z_{28}=x_{0}+x_{6}+x_{9}+x_{11}+x_{12}+x_{13}+x_{16}+x_{17}+x_{18}+x_{19}+x_{29}+x_{31}+x_{28}+x_{27}+x_{21}+x_{24}$

$z_{27}=x_{5}+x_{8}+x_{10}+x_{11}+x_{12}+x_{15}+x_{16}+x_{17}+x_{18}+x_{28}+x_{31}+x_{30}+x_{27}+x_{26}+x_{20}+x_{23}$

$z_{26}=x_{4}+x_{7}+x_{9}+x_{10}+x_{11}+x_{14}+x_{15}+x_{16}+x_{17}+x_{27}+x_{30}+x_{29}+x_{26}+x_{25}+x_{19}+x_{22}$

$z_{25}=x_{3}+x_{6}+x_{8}+x_{9}+x_{10}+x_{13}+x_{14}+x_{15}+x_{16}+x_{26}+x_{29}+x_{28}+x_{25}+x_{31}+x_{24}+x_{18}+x_{21}$

$z_{24}=x_{2}+x_{5}+x_{7}+x_{8}+x_{9}+x_{12}+x_{13}+x_{14}+x_{15}+x_{25}+x_{28}+x_{27}+x_{24}+x_{30}+x_{23}+x_{17}+x_{20}+x_{31}$

$z_{23}=x_1+x_4+x_6+x_7+x_8+x_{11}+x_{12}+x_{13}+x_{14}+x_{24}+x_{27}+x_{26}+x_{23}+x_{29}+x_{22}+x_{16}+x_{19}+x_{30}+x_{31}$

$z_{22}=x_0+x_3+x_5+x_6+x_7+x_{10}+x_{11}+x_{12}+x_{13}+x_{23}+x_{26}+x_{25}+x_{22}+x_{28}+x_{21}+x_{15}+x_{18}+x_{29}+x_{30}+x_{31}$

$z_{21}=x_2+x_4+x_5+x_6+x_9+x_{10}+x_{11}+x_{12}+x_{22}+x_{25}+x_{24}+x_{21}+x_{27}+x_{20}+x_{14}+x_{17}+x_{28}+x_{29}+x_{30}$

$z_{20}=x_1+x_3+x_4+x_5+x_8+x_9+x_{10}+x_{11}+x_{21}+x_{24}+x_{23}+x_{20}+x_{26}+x_{19}+x_{31}+x_{13}+x_{16}+x_{27}+x_{28}+x_{29}$

$z_{19}=x_0+x_2+x_3+x_4+x_7+x_8+x_9+x_{10}+x_{20}+x_{23}+x_{22}+x_{19}+x_{25}+x_{18}+x_{30}+x_{12}+x_{15}+x_{26}+x_{27}+x_{28}$

$z_{18}=x_1+x_2+x_3+x_6+x_7+x_8+x_9+x_{19}+x_{22}+x_{21}+x_{18}+x_{24}+x_{17}+x_{29}+x_{11}+x_{14}+x_{25}+x_{26}+x_{27}$

$z_{17}=x_0+x_1+x_2+x_5+x_6+x_7+x_8+x_{18}+x_{21}+x_{20}+x_{17}+x_{23}+x_{16}+x_{28}+x_{10}+x_{13}+x_{24}+x_{25}+x_{26}+x_{31}$

$z_{16}=x_0+x_1+x_4+x_5+x_6+x_7+x_{17}+x_{20}+x_{19}+x_{16}+x_{22}+x_{15}+x_{27}+x_9+x_{12}+x_{23}+x_{24}+x_{25}+x_{30}+x_{31}$

$z_{15}=x_0+x_3+x_4+x_5+x_6+x_{16}+x_{19}+x_{18}+x_{15}+x_{21}+x_{14}+x_{26}+x_8+x_{11}+x_{22}+x_{23}+x_{24}+x_{29}+x_{30}$

$z_{14}=x_2+x_3+x_4+x_5+x_{15}+x_{18}+x_{17}+x_{14}+x_{20}+x_{13}+x_{25}+x_7+x_{10}+x_{21}+x_{22}+x_{23}+x_{28}+x_{29}$

$z_{13}=x_1+x_2+x_3+x_4+x_{14}+x_{17}+x_{16}+x_{13}+x_{19}+x_{12}+x_{24}+x_{31}+x_6+x_9+x_{20}+x_{21}+x_{22}+x_{27}+x_{28}$

$z_{12}=x_0+x_1+x_2+x_3+x_{13}+x_{16}+x_{15}+x_{12}+x_{18}+x_{31}+x_{11}+x_{23}+x_{30}+x_5+x_8+x_{19}+x_{20}+x_{21}+x_{26}+x_{27}$

$z_{11}=x_0+x_1+x_2+x_{12}+x_{15}+x_{14}+x_{11}+x_{17}+x_{30}+x_{10}+x_{22}+x_{29}+x_4+x_7+x_{18}+x_{19}+x_{20}+x_{25}+x_{26}+x_{31}$

$z_{10}=x_0+x_1+x_{11}+x_{14}+x_{13}+x_{10}+x_{16}+x_{29}+x_9+x_{21}+x_{28}+x_3+x_6+x_{17}+x_{18}+x_{19}+x_{24}+x_{25}+x_{30}$

$z_9=x_0+x_{10}+x_{13}+x_{12}+x_9+x_{15}+x_{28}+x_8+x_{20}+x_{27}+x_2+x_5+x_{16}+x_{17}+x_{18}+x_{23}+x_{24}+x_{29}$

$z_8=x_9+x_{12}+x_{11}+x_8+x_{14}+x_{27}+x_7+x_{19}+x_{26}+x_1+x_4+x_{15}+x_{16}+x_{17}+x_{22}+x_{23}+x_{28}$

$z_7=x_8+x_{11}+x_{10}+x_9+x_7+x_{13}+x_{20}+x_{26}++x_6+x_{12}+x_{18}+x_{19}+x_{24}+x_{25}+x_{30}$

$z_6=+x_7+x_{10}+x_{31}+x_9+x_8+x_6+x_{12}+x_{19}+x_{25}+x_5+x_{11}+x_{17}+x_{18}+x_{23}+x_{24}+x_{29}$

$z_5=x_6+x_9+x_{30}+x_8+x_7+x_5+x_{11}+x_{18}+x_{24}+x_4+x_{10}+x_{16}+x_{17}+x_{22}+x_{23}+x_{28}$

$z_4=x_0+x_5+x_8+x_{14}+x_{29}+x_7+x_6+x_{12}+x_{19}+x_{24}+x_{30}+x_4+x_{10}+x_{17}+x_{20}+x_{23}$

$z_3=x_0+x_4+x_7+x_{13}+x_{20}+x_{28}+x_{31}+x_6+x_{12}+x_{15}+x_{27}+x_{30}+x_5+x_{11}+x_{14}+x_{18}+x_{21}+x_{23}+x_{24}+x_{29}$

$z_2=x_3+x_6+x_{12}+x_{19}+x_{27}+x_{30}+x_5+x_{11}+x_{14}+x_{26}+x_{29}+x_4+x_{10}+x_{13}+x_{17}+x_{20}+x_{22}+x_{23}+x_{31}+x_{28}$

$z_1=x_0+x_2+x_5+x_{11}+x_{14}+x_{18}+x_{24}+x_{26}+x_{29}+x_4+x_{10}+x_{13}+x_{15}+x_{20}+x_{31}+x_{25}+x_{28}$

$z_0=x_0+x_1+x_4+x_{10}+x_{13}+x_{15}+x_{16}+x_{17}+x_{20}+x_{21}+x_{22}+x_{23}+x_{31}+x_{25}+x_{28}$

Формула В.2 – Шифрлау алгоритмінде қолданылған S-блоктан XL шабуылы үшін алынған тәуелсіз квадраттық теңдеулер

$0 = 1 \oplus x_0 \oplus x_3 \oplus x_4 \oplus y_0 \oplus y_1 \oplus y_2 \oplus y_4 \oplus y_5 \oplus x_0x_2 \oplus x_0x_3 \oplus x_1x_2 \oplus x_1x_5 \oplus$
$\oplus x_2x_3 \oplus x_2x_4 \oplus x_2x_7 \oplus x_3x_4 \oplus x_4x_5 \oplus x_5x_6 \oplus x_6x_7 \oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_6 \oplus$
$\oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_6 \oplus y_1y_7 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_7 \oplus y_3y_6 \oplus y_4y_6 \oplus y_4y_7 \oplus y_5y_7 \oplus$
$\oplus y_6y_7 \oplus x_0y_0 \oplus x_0y_2 \oplus x_0y_4 \oplus x_0y_5 \oplus x_1y_2 \oplus x_1y_4 \oplus x_1y_5 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus$
$\oplus x_2y_6 \oplus x_3y_0$

$0 = x_6 \oplus y_1 \oplus y_2 \oplus y_4 \oplus y_5 \oplus x_0x_3 \oplus x_0x_5 \oplus x_0x_7 \oplus x_1x_2 \oplus x_1x_3 \oplus x_1x_4 \oplus x_2x_5 \oplus$
$\oplus x_2x_6 \oplus x_3x_6 \oplus x_3x_7 \oplus x_4x_5 \oplus x_4x_6 \oplus x_4x_7 \oplus x_5x_6 \oplus x_6x_7 \oplus y_0y_3 \oplus y_1y_3 \oplus$
$\oplus y_1y_5 \oplus y_1y_6 \oplus y_2y_3 \oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_5 \oplus y_3y_6 \oplus y_4y_7 \oplus y_5y_7 \oplus x_0y_5 \oplus x_0y_6 \oplus$
$\oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_3 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_3 \oplus x_2y_6 \oplus x_2y_7 \oplus x_3y_1 \oplus x_3y_2$

$0 = 1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus y_4 \oplus y_5 \oplus x_0x_2 \oplus x_0x_4 \oplus x_0x_7 \oplus x_1x_2 \oplus x_1x_3 \oplus$
$\oplus x_1x_4 \oplus x_1x_5 \oplus x_1x_6 \oplus x_2x_3 \oplus x_2x_5 \oplus x_3x_5 \oplus x_3x_6 \oplus x_4x_5 \oplus x_5x_6 \oplus x_5x_7 \oplus y_0y_1 \oplus$
$\oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_4 \oplus y_1y_5 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_5 \oplus y_2y_6 \oplus y_2y_7 \oplus$
$\oplus y_3y_4 \oplus y_3y_6 \oplus y_3y_7 \oplus \oplus y_4y_6 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_3 \oplus x_0y_6 \oplus x_1y_2 \oplus x_1y_3 \oplus$
$\oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_5 \oplus x_2y_7 \oplus x_3y_3$

$$0 = x_0 \oplus x_1 \oplus x_3 \oplus x_4 \oplus y_0 \oplus y_2 \oplus y_3 \oplus y_7 \oplus x_0x_3 \oplus x_0x_4 \oplus x_0x_6 \oplus x_1x_2 \oplus x_1x_4 \oplus$$
$$\oplus x_1x_6 \oplus x_3x_5 \oplus x_4x_5 \oplus x_5x_6 \oplus x_5x_7 \oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_4 \oplus y_0y_7 \oplus y_1y_3 \oplus$$
$$\oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_7 \oplus y_2y_3 \oplus y_2y_6 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_7 \oplus y_4y_6 \oplus y_5y_6 \oplus y_5y_7 \oplus$$
$$\oplus y_6y_7 \oplus x_0y_2 \oplus x_0y_3 \oplus x_0y_4 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_3 \oplus x_1y_6 \oplus x_2y_7 \oplus x_3y_1 \oplus x_3y_4$$

$$0 = 1 \oplus x_1 \oplus x_3 \oplus x_5 \oplus y_0 \oplus y_7 \oplus x_0x_1 \oplus x_0x_2 \oplus x_0x_7 \oplus x_1x_3 \oplus x_1x_4 \oplus x_1x_5 \oplus$$
$$\oplus x_2x_3 \oplus x_2x_5 \oplus x_3x_6 \oplus x_3x_7 \oplus x_5x_6 \oplus x_6x_7 \oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_4 \oplus y_0y_5 \oplus$$
$$\oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_7 \oplus y_2y_7 \oplus y_3y_4 \oplus y_3y_7 \oplus y_4y_5 \oplus y_5y_7 \oplus y_6y_7 \oplus x_0y_0 \oplus$$
$$\oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_5 \oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_3 \oplus x_1y_4 \oplus x_2y_0 \oplus x_2y_1 \oplus$$
$$\oplus x_2y_2 \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_7 \oplus x_3y_1 \oplus \oplus x_3y_5$$

$$0 = x_0 \oplus x_1 \oplus x_3 \oplus x_4 \oplus y_1 \oplus y_2 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7 \oplus x_0x_1 \oplus x_0x_4 \oplus x_0x_6 \oplus$$
$$\oplus x_0x_7 \oplus x_1x_3 \oplus x_1x_5 \oplus x_1x_7 \oplus x_2x_4 \oplus x_2x_5 \oplus x_3x_5 \oplus x_4x_5 \oplus x_4x_7 \oplus x_5x_6 \oplus x_6x_7 \oplus$$
$$\oplus y_0y_2 \oplus y_0y_4 \oplus y_0y_7 \oplus \oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_6 \oplus y_1y_7 \oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_5 \oplus y_4y_5 \oplus$$
$$\oplus y_4y_7 \oplus x_0y_1 \oplus x_0y_3 \oplus x_0y_4 \oplus x_0y_5 \oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_5 \oplus x_1y_7 \oplus x_2y_5 \oplus x_3y_6$$

$$0 = 1 \oplus x_3 \oplus x_7 \oplus y_3 \oplus y_4 \oplus y_7 \oplus x_0x_1 \oplus x_0x_2 \oplus x_0x_4 \oplus x_0x_6 \oplus x_0x_7 \oplus x_1x_2 \oplus$$
$$\oplus x_1x_5 \oplus x_1x_6 \oplus x_2x_5 \oplus x_2x_6 \oplus x_3x_4 \oplus x_3x_6 \oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_1 \oplus y_0y_3 \oplus y_0y_5 \oplus$$
$$\oplus y_0y_6 \oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_7 \oplus y_2y_4 \oplus y_2y_5 \oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_4 \oplus$$
$$\oplus y_3y_5 \oplus y_3y_6 \oplus y_3y_7 \oplus y_4y_5 \oplus y_4y_6 \oplus y_4y_7 \oplus y_6y_7 \oplus x_0y_5 \oplus x_0y_6 \oplus x_1y_1 \oplus x_1y_3 \oplus$$
$$\oplus x_1y_5 \oplus x_2y_0 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_7 \oplus x_3y_1 \oplus x_3y_7$$

$$0 = 1 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_5 \oplus x_7 \oplus y_5 \oplus y_6 \oplus y_7 \oplus x_0x_2 \oplus x_0x_4 \oplus x_0x_6 \oplus x_0x_7 \oplus$$
$$\oplus x_1x_3 \oplus x_1x_4 \oplus x_1x_5 \oplus x_1x_6 \oplus x_2x_4 \oplus x_3x_7 \oplus x_4x_7 \oplus x_5x_6 \oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_1 \oplus$$
$$\oplus y_0y_4 \oplus y_0y_5 \oplus y_1y_3 \oplus y_1y_4 \oplus y_1y_7 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_5 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_7 \oplus$$
$$\oplus y_5y_6 \oplus y_5y_7 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_6 \oplus x_1y_1 \oplus x_1y_4 \oplus x_1y_5 \oplus x_1y_6 \oplus x_2y_0 \oplus x_2y_1 \oplus$$
$$\oplus x_2y_2 \oplus x_2y_3 \oplus x_2y_5 \oplus x_2y_6 \oplus x_4y_0$$

$$0 = x_3 \oplus x_5 \oplus x_6 \oplus x_7 \oplus y_3 \oplus y_7 \oplus x_0x_4 \oplus x_0x_5 \oplus x_0x_6 \oplus x_0x_7 \oplus x_1x_3 \oplus x_1x_5 \oplus$$
$$\oplus x_1x_7 \oplus \oplus x_2x_5 \oplus x_2x_6 \oplus x_2x_7 \oplus x_3x_4 \oplus x_3x_5 \oplus x_4x_5 \oplus x_4x_6 \oplus y_0y_2 \oplus y_0y_3 \oplus$$
$$\oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_5 \oplus y_2y_3 \oplus y_2y_5 \oplus y_2y_6 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_6 \oplus y_3y_7 \oplus y_4y_6 \oplus$$
$$\oplus y_4y_7 \oplus y_5y_6 \oplus x_0y_1 \oplus x_0y_5 \oplus x_0y_7 \oplus x_1y_1 \oplus x_1y_3 \oplus x_1y_5 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus$$
$$\oplus x_2y_6 \oplus x_2y_7 \oplus x_4y_1$$

$$0 = 1 \oplus x_0 \oplus x_3 \oplus x_4 \oplus x_6 \oplus x_7 \oplus y_0 \oplus y_1 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus x_0x_4 \oplus x_0x_7 \oplus$$
$$\oplus x_1x_4 \oplus x_1x_5 \oplus x_1x_7 \oplus x_2x_3 \oplus x_2x_4 \oplus x_2x_5 \oplus x_2x_6 \oplus x_3x_4 \oplus x_4x_7 \oplus y_0y_1 \oplus y_0y_3 \oplus$$
$$\oplus y_0y_5 \oplus y_0y_6 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_7 \oplus y_2y_5 \oplus y_3y_5 \oplus x_0y_0 \oplus x_0y_2 \oplus x_0y_6 \oplus x_1y_0 \oplus$$
$$\oplus x_1y_1 \oplus x_1y_3 \oplus x_1y_5 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_3 \oplus x_2y_7 \oplus x_4y_2$$

$$0 = 1 \oplus x_0 \oplus x_2 \oplus x_3 \oplus x_5 \oplus y_0 \oplus y_2 \oplus y_5 \oplus y_6 \oplus x_0x_1 \oplus x_0x_3 \oplus x_0x_4 \oplus x_0x_7 \oplus$$
$$\oplus x_1x_2 \oplus x_1x_6 \oplus x_2x_6 \oplus x_2x_7 \oplus x_3x_5 \oplus x_3x_7 \oplus x_4x_7 \oplus x_5x_6 \oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_2 \oplus$$
$$\oplus y_0y_3 \oplus y_0y_4 \oplus y_0y_5 \oplus y_1y_2 \oplus y_1y_4 \oplus y_1y_7 \oplus y_2y_4 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_7 \oplus y_4y_5 \oplus$$
$$\oplus y_4y_7 \oplus y_5y_6 \oplus y_5y_7 \oplus y_6y_7 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_3 \oplus x_0y_4 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_3 \oplus$$
$$\oplus x_1y_4 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_3 \oplus \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_6 \oplus x_4y_3$$

$$0 = x_0 \oplus x_2 \oplus x_4 \oplus x_5 \oplus x_7 \oplus y_0 \oplus y_1 \oplus y_2 \oplus y_4 \oplus y_5 \oplus x_0x_1 \oplus x_0x_2 \oplus x_0x_4 \oplus x_0x_5 \oplus$$
$$\oplus x_0x_6 \oplus x_1x_4 \oplus x_1x_7 \oplus x_2x_3 \oplus x_2x_6 \oplus x_2x_7 \oplus x_3x_4 \oplus x_3x_5 \oplus x_3x_6 \oplus x_3x_7 \oplus x_4x_5 \oplus$$
$$\oplus x_4x_6 \oplus x_5x_6 \oplus x_6x_7 \oplus y_0y_1 \oplus y_0y_4 \oplus y_1y_2 \oplus y_1y_3 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_6 \oplus y_2y_7 \oplus$$
$$\oplus y_3y_4 \oplus y_3y_6 \oplus y_4y_5 \oplus y_5y_7 \oplus x_0y_1 \oplus x_0y_6 \oplus x_1y_2 \oplus x_1y_3 \oplus x_1y_4 \oplus x_2y_0 \oplus x_2y_2 \oplus$$
$$\oplus x_2y_4 \oplus x_2y_5 \oplus x_3y_1 \oplus x_4y_4$$

$$0 = x_0 \oplus x_1 \oplus x_3 \oplus y_0 \oplus x_0x_1 \oplus x_0x_2 \oplus x_0x_3 \oplus x_0x_4 \oplus x_0x_5 \oplus x_0x_6 \oplus x_1x_3 \oplus x_1x_4 \oplus$$
$$\oplus x_1x_7 \oplus x_2x_7 \oplus x_3x_6 \oplus x_4x_6 \oplus x_5x_6 \oplus x_6x_7 \oplus y_0y_4 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_4 \oplus y_2y_3 \oplus$$
$$\oplus y_2y_6 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_6 \oplus y_4y_5 \oplus y_4y_7 \oplus y_5y_6 \oplus y_5y_7 \oplus x_0y_0 \oplus x_0y_4 \oplus x_0y_7 \oplus$$
$$\oplus x_1y_2 \oplus x_1y_3 \oplus x_1y_4 \oplus x_1y_7 \oplus x_2y_2 \oplus x_2y_5 \oplus x_2y_6 \oplus x_4y_5$$

$$0 = x_1 \oplus x_3 \oplus x_4 \oplus x_7 \oplus y_0 \oplus y_2 \oplus y_3 \oplus y_5 \oplus x_0x_1 \oplus x_0x_2 \oplus x_0x_4 \oplus x_0x_7 \oplus x_1x_5 \oplus$$
$$\oplus x_1x_6 \oplus x_2x_3 \oplus x_2x_4 \oplus x_2x_6 \oplus x_3x_4 \oplus x_3x_5 \oplus x_3x_7 \oplus x_4x_5 \oplus x_4x_7 \oplus x_5x_7 \oplus y_0y_2 \oplus$$
$$\oplus y_0y_6 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_4 \oplus y_2y_3 \oplus y_2y_5 \oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_6 \oplus y_3y_7 \oplus$$
$$\oplus y_4y_7 \oplus y_5y_6 \oplus y_6y_7 \oplus x_0y_0 \oplus x_0y_2 \oplus x_0y_3 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_4 \oplus x_1y_5 \oplus$$
$$\oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_3 \oplus x_2y_5 \oplus x_2y_7 \oplus x_4y_6$$

$$0 = 1 \oplus x_2 \oplus x_4 \oplus x_5 \oplus y_0 \oplus y_1 \oplus y_3 \oplus y_7 \oplus x_0x_4 \oplus x_0x_5 \oplus x_1x_2 \oplus x_1x_6 \oplus x_2x_4 \oplus$$
$$\oplus x_2x_6 \oplus x_2x_7 \oplus x_3x_4 \oplus x_3x_5 \oplus x_3x_7 \oplus x_4x_5 \oplus x_4x_6 \oplus x_4x_7 \oplus x_5x_6 \oplus x_5x_7 \oplus y_0y_1 \oplus$$
$$\oplus y_0y_2 \oplus y_0y_4 \oplus y_0y_5 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_5 \oplus y_1y_6 \oplus y_1y_7 \oplus y_2y_3 \oplus y_2y_6 \oplus y_3y_4 \oplus$$
$$\oplus y_3y_5 \oplus y_3y_6 \oplus y_3y_7 \oplus y_4y_5 \oplus y_4y_7 \oplus y_6y_7 \oplus x_0y_2 \oplus x_0y_4 \oplus x_0y_5 \oplus x_1y_1 \oplus x_1y_2 \oplus$$
$$\oplus x_1y_3 \oplus x_1y_4 \oplus x_1y_6 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_5 \oplus x_4y_7$$

$$0 = x_0 \oplus x_1 \oplus x_2 \oplus x_4 \oplus x_6 \oplus y_2 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7 \oplus x_0x_2 \oplus x_0x_3 \oplus x_0x_5 \oplus$$
$$\oplus x_1x_3 \oplus x_1x_4 \oplus x_1x_6 \oplus x_1x_7 \oplus x_2x_3 \oplus x_2x_4 \oplus x_2x_5 \oplus x_2x_7 \oplus x_4x_6 \oplus x_4x_7 \oplus x_5x_6 \oplus$$
$$\oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_2 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_6 \oplus y_2y_4 \oplus y_2y_5 \oplus$$
$$\oplus y_2y_6 \oplus y_3y_6 \oplus y_4y_5 \oplus y_4y_7 \oplus y_5y_6 \oplus y_5y_7 \oplus y_6y_7 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_5 \oplus$$
$$\oplus x_0y_6 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_3 \oplus x_1y_4 \oplus x_2y_2 \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_6 \oplus x_5y_0$$

$$0 = x_0 \oplus x_1 \oplus x_4 \oplus x_6 \oplus y_0 \oplus y_2 \oplus y_6 \oplus x_0x_2 \oplus x_0x_5 \oplus x_0x_6 \oplus x_1x_2 \oplus x_1x_4 \oplus x_1x_6 \oplus$$
$$\oplus x_1x_7 \oplus x_2x_3 \oplus x_2x_6 \oplus x_2x_7 \oplus x_3x_4 \oplus x_3x_6 \oplus x_4x_6 \oplus x_5x_6 \oplus x_5x_7 \oplus y_0y_2 \oplus y_0y_3 \oplus$$
$$\oplus y_0y_4 \oplus y_0y_6 \oplus y_1y_5 \oplus y_1y_6 \oplus y_2y_3 \oplus y_2y_5 \oplus y_2y_6 \oplus y_3y_4 \oplus y_3y_6 \oplus y_3y_7 \oplus y_4y_5 \oplus$$
$$\oplus y_4y_7 \oplus y_6y_7 \oplus x_0y_1 \oplus x_0y_3 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_2 \oplus x_1y_4 \oplus x_2y_0 \oplus x_2y_3 \oplus x_2y_4 \oplus$$
$$\oplus x_2y_5 \oplus x_3y_1 \oplus x_5y_1$$

$$0 = x_1 \oplus x_2 \oplus x_5 \oplus x_6 \oplus y_0 \oplus y_5 \oplus y_7 \oplus x_0x_5 \oplus x_0x_6 \oplus x_1x_3 \oplus x_1x_4 \oplus x_1x_5 \oplus x_1x_6 \oplus$$
$$\oplus x_1x_7 \oplus x_2x_3 \oplus x_2x_4 \oplus x_3x_4 \oplus x_3x_5 \oplus x_3x_6 \oplus x_3x_7 \oplus x_4x_6 \oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_7 \oplus$$
$$\oplus y_1y_3 \oplus y_1y_4 \oplus y_1y_7 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_5 \oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_4 \oplus y_4y_5 \oplus y_4y_7 \oplus$$
$$\oplus y_6y_7 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus x_0y_4 \oplus x_0y_5 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_2 \oplus x_1y_3 \oplus$$
$$\oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus x_5y_2$$

$$0 = x_1 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_7 \oplus y_0 \oplus y_1 \oplus y_7 \oplus x_0x_1 \oplus x_0x_3 \oplus x_0x_7 \oplus x_1x_2 \oplus x_1x_3 \oplus$$
$$\oplus x_1x_4 \oplus x_1x_5 \oplus x_1x_6 \oplus x_1x_7 \oplus x_2x_4 \oplus x_2x_5 \oplus x_4x_7 \oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_5 \oplus y_1y_2 \oplus$$
$$\oplus y_1y_4 \oplus y_1y_5 \oplus y_2y_3 \oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_5 \oplus y_4y_5 \oplus y_4y_6 \oplus y_5y_6 \oplus x_0y_2 \oplus x_0y_7 \oplus$$
$$\oplus x_1y_2 \oplus x_1y_6 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_4 \oplus x_2y_6 \oplus x_2y_7 \oplus x_3y_1 \oplus x_5y_3$$

$$0 = 1 \oplus x_4 \oplus x_7 \oplus y_1 \oplus y_3 \oplus y_6 \oplus x_0x_2 \oplus x_0x_3 \oplus x_0x_5 \oplus x_0x_6 \oplus x_1x_2 \oplus x_1x_3 \oplus x_1x_4 \oplus$$
$$\oplus x_1x_6 \oplus x_1x_7 \oplus x_2x_7 \oplus x_3x_5 \oplus x_3x_6 \oplus x_3x_7 \oplus x_4x_5 \oplus x_4x_6 \oplus x_5x_6 \oplus x_5x_7 \oplus y_0y_1 \oplus$$
$$\oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_4 \oplus y_0y_6 \oplus y_1y_6 \oplus y_2y_3 \oplus y_2y_5 \oplus y_3y_5 \oplus y_4y_6 \oplus y_4y_7 \oplus y_5y_6 \oplus$$
$$\oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_5 \oplus x_0y_6 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_4 \oplus x_1y_6 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_3 \oplus$$
$$\oplus x_2y_4 \oplus x_2y_6 \oplus x_2y_7 \oplus x_5y_4$$

$$0 = x_0 \oplus x_1 \oplus x_2 \oplus x_4 \oplus x_6 \oplus x_7 \oplus y_0 \oplus y_3 \oplus y_6 \oplus x_0x_3 \oplus x_0x_4 \oplus x_0x_5 \oplus x_1x_2 \oplus$$
$$\oplus x_1x_3 \oplus x_1x_6 \oplus x_2x_3 \oplus x_2x_4 \oplus x_2x_7 \oplus x_4x_6 \oplus x_4x_7 \oplus x_5x_6 \oplus x_5x_7 \oplus y_0y_1 \oplus y_0y_4 \oplus$$
$$\oplus y_0y_6 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_6 \oplus y_1y_7 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_5 \oplus y_2y_7 \oplus$$
$$\oplus y_3y_6 \oplus y_3y_7 \oplus y_4y_5 \oplus y_4y_7 \oplus y_6y_7 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_4 \oplus x_0y_5 \oplus x_1y_2 \oplus x_1y_4 \oplus$$
$$\oplus x_1y_5 \oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_1 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_6 \oplus x_3y_1 \oplus x_5y_5$$

$$0 = x_2 \oplus x_5 \oplus x_6 \oplus y_0 \oplus y_4 \oplus y_6 \oplus y_7 \oplus x_0x_1 \oplus x_0x_4 \oplus x_0x_5 \oplus x_0x_7 \oplus x_1x_4 \oplus x_1x_5 \oplus$$
$$\oplus x_1x_7 \oplus x_2x_3 \oplus x_2x_5 \oplus x_2x_7 \oplus x_4x_6 \oplus x_5x_6 \oplus x_6x_7 \oplus y_0y_4 \oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_5 \oplus$$
$$\oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_7 \oplus y_3y_5 \oplus y_3y_7 \oplus y_4y_6 \oplus y_5y_6 \oplus y_5y_7 \oplus y_6y_7 \oplus x_0y_3 \oplus x_1y_0 \oplus$$
$$\oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_3 \oplus x_1y_5 \oplus x_1y_6 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_3 \oplus x_2y_5 \oplus x_3y_1 \oplus x_5y_6$$

$$0 = 1 \oplus x_0 \oplus x_1 \oplus x_2 \oplus x_4 \oplus x_7 \oplus y_1 \oplus y_2 \oplus y_3 \oplus y_6 \oplus x_0x_1 \oplus x_0x_2 \oplus x_0x_4 \oplus x_0x_5 \oplus$$
$$\oplus x_0x_6 \oplus x_1x_4 \oplus x_1x_5 \oplus x_1x_6 \oplus x_2x_3 \oplus x_2x_6 \oplus x_3x_5 \oplus x_4x_6 \oplus x_4x_7 \oplus x_5x_6 \oplus x_5x_7 \oplus$$
$$\oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_5 \oplus y_1y_3 \oplus y_1y_4 \oplus y_1y_5 \oplus y_2y_5 \oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_5 \oplus$$
$$\oplus y_4y_6 \oplus y_4y_7 \oplus y_5y_6 \oplus y_6y_7 \oplus x_0y_2 \oplus x_0y_4 \oplus x_0y_5 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_1 \oplus x_1y_3 \oplus$$
$$\oplus x_2y_0 \oplus x_2y_4 \oplus x_2y_5 \oplus x_5y_7$$

$$0 = 1 \oplus x_0 \oplus x_3 \oplus x_7 \oplus y_3 \oplus y_4 \oplus y_6 \oplus y_7 \oplus x_0x_1 \oplus x_0x_3 \oplus x_1x_2 \oplus x_1x_4 \oplus x_2x_3 \oplus$$
$$\oplus x_2x_4 \oplus x_2x_5 \oplus x_3x_5 \oplus x_3x_6 \oplus x_4x_5 \oplus x_4x_7 \oplus x_5x_6 \oplus y_0y_1 \oplus y_0y_4 \oplus y_0y_7 \oplus y_1y_2 \oplus$$
$$\oplus y_1y_4 \oplus y_1y_5 \oplus y_2y_3 \oplus y_3y_7 \oplus y_4y_6 \oplus y_4y_7 \oplus y_5y_7 \oplus y_6y_7 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus$$
$$\oplus x_0y_4 \oplus x_1y_0 \oplus x_1y_3 \oplus x_1y_6 \oplus x_2y_0 \oplus x_2y_2 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_6 \oplus x_6y_0$$

$$0 = x_3 \oplus x_4 \oplus y_1 \oplus y_2 \oplus y_3 \oplus y_4 \oplus x_0x_2 \oplus x_0x_4 \oplus x_0x_5 \oplus x_0x_7 \oplus x_1x_7 \oplus x_2x_3 \oplus x_3x_4 \oplus$$
$$\oplus x_3x_5 \oplus x_3x_6 \oplus x_4x_6 \oplus x_4x_7 \oplus x_5x_6 \oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_4 \oplus$$
$$\oplus y_1y_6 \oplus y_2y_4 \oplus y_2y_5 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_7 \oplus y_4y_6 \oplus y_4y_7 \oplus y_5y_7 \oplus x_0y_1 \oplus x_0y_2 \oplus$$
$$\oplus x_0y_4 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_3 \oplus x_1y_4 \oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_2 \oplus x_2y_3 \oplus$$
$$\oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_7 \oplus x_3y_1 \oplus x_6y_1$$

$$0 = 1 \oplus x_3 \oplus x_5 \oplus y_1 \oplus y_4 \oplus y_6 \oplus y_7 \oplus x_0x_2 \oplus x_0x_3 \oplus x_0x_5 \oplus x_0x_6 \oplus x_0x_7 \oplus x_1x_2 \oplus$$
$$\oplus x_1x_4 \oplus x_1x_5 \oplus x_2x_3 \oplus x_2x_7 \oplus x_3x_7 \oplus x_4x_6 \oplus x_6x_7 \oplus y_0y_1 \oplus y_0y_3 \oplus y_0y_5 \oplus y_0y_6 \oplus$$
$$\oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_5 \oplus y_2y_3 \oplus y_3y_5 \oplus y_3y_6 \oplus y_3y_7 \oplus y_4y_6 \oplus x_0y_3 \oplus x_0y_4 \oplus x_0y_5 \oplus$$
$$\oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_3 \oplus x_1y_4 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_6 \oplus x_6y_2$$

$$0 = 1 \oplus x_2 \oplus x_4 \oplus x_5 \oplus x_6 \oplus y_0 \oplus y_4 \oplus y_5 \oplus x_0x_3 \oplus x_0x_6 \oplus x_1x_2 \oplus x_1x_3 \oplus x_2x_4 \oplus$$
$$\oplus x_2x_5 \oplus x_2x_6 \oplus x_3x_4 \oplus x_3x_6 \oplus x_4x_7 \oplus x_6x_7 \oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_6 \oplus y_0y_7 \oplus$$
$$\oplus y_1y_2 \oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_6 \oplus y_1y_7 \oplus y_2y_3 \oplus y_2y_6 \oplus y_3y_4 \oplus y_3y_7 \oplus y_4y_5 \oplus y_5y_7 \oplus$$
$$\oplus x_0y_2 \oplus x_0y_5 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_0 \oplus x_2y_3 \oplus x_2y_5 \oplus x_2y_7 \oplus x_3y_1 \oplus x_6y_3$$

$$0 = 1 \oplus x_0 \oplus x_2 \oplus x_3 \oplus x_5 \oplus x_6 \oplus x_7 \oplus y_1 \oplus y_4 \oplus y_7 \oplus x_0x_6 \oplus x_1x_2 \oplus x_2x_3 \oplus x_2x_4 \oplus$$
$$\oplus x_2x_5 \oplus x_3x_5 \oplus x_3x_6 \oplus x_3x_7 \oplus x_4x_5 \oplus x_4x_7 \oplus x_6x_7 \oplus y_0y_1 \oplus y_0y_3 \oplus y_0y_4 \oplus y_0y_6 \oplus$$

$$\oplus\, y_1y_2 \oplus y_2y_3 \oplus y_2y_7 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_7 \oplus y_4y_7 \oplus y_5y_7 \oplus x_0y_0 \oplus x_0y_3 \oplus x_0y_5 \oplus$$
$$\oplus\, x_1y_2 \oplus x_1y_4 \oplus x_1y_5 \oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_7 \oplus x_3y_1 \oplus x_6y_4$$

$$0 = x_0 \oplus x_2 \oplus x_3 \oplus x_4 \oplus y_0 \oplus y_4 \oplus y_6 \oplus y_7 \oplus x_0x_4 \oplus x_0x_5 \oplus x_0x_7 \oplus x_1x_2 \oplus x_1x_3 \oplus$$
$$\oplus\, x_1x_5 \oplus x_1x_7 \oplus x_2x_3 \oplus x_2x_4 \oplus x_2x_6 \oplus x_2x_7 \oplus x_3x_4 \oplus x_3x_6 \oplus x_3x_7 \oplus x_4x_6 \oplus x_4x_7 \oplus$$
$$\oplus\, x_6x_7 \oplus y_0y_1 \oplus y_0y_3 \oplus y_0y_6 \oplus y_0y_7 \oplus y_1y_4 \oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_6 \oplus y_3y_7 \oplus y_4y_5 \oplus$$
$$\oplus\, y_5y_6 \oplus x_0y_1 \oplus x_0y_4 \oplus x_0y_5 \oplus x_0y_7 \oplus x_1y_5 \oplus x_1y_6 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_7 \oplus$$
$$\oplus\, x_3y_1 \oplus x_6y_5$$

$$0 = x_4 \oplus x_6 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7 \oplus x_0x_1 \oplus x_0x_2 \oplus x_0x_4 \oplus x_0x_5 \oplus x_0x_6 \oplus x_1x_4 \oplus$$
$$\oplus\, x_1x_5 \oplus x_1x_6 \oplus x_2x_3 \oplus x_2x_6 \oplus x_3x_5 \oplus x_4x_6 \oplus x_4x_7 \oplus x_5x_6 \oplus x_5x_7 \oplus y_0y_4 \oplus y_0y_7 \oplus$$
$$\oplus\, y_1y_2 \oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_7 \oplus y_2y_5 \oplus y_2y_6 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_7 \oplus y_4y_5 \oplus y_4y_6 \oplus$$
$$\oplus\, y_4y_7 \oplus y_5y_6 \oplus y_5y_7 \oplus x_0y_1 \oplus x_0y_3 \oplus x_0y_4 \oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_4 \oplus$$
$$\oplus\, x_1y_6 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_6 \oplus x_3y_1 \oplus x_6y_6$$

$$0 = x_0 \oplus x_1 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7 \oplus x_0x_2 \oplus x_0x_3 \oplus x_0x_4 \oplus x_0x_6 \oplus$$
$$\oplus\, x_1x_3 \oplus x_1x_4 \oplus x_1x_7 \oplus x_2x_4 \oplus x_2x_6 \oplus x_3x_4 \oplus x_3x_6 \oplus x_4x_7 \oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_3 \oplus$$
$$\oplus\, y_0y_4 \oplus y_0y_5 \oplus y_1y_2 \oplus y_1y_4 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_5 \oplus y_2y_6 \oplus y_3y_7 \oplus y_4y_7 \oplus y_5y_7 \oplus$$
$$\oplus\, x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus x_0y_4 \oplus x_0y_5 \oplus x_1y_2 \oplus x_1y_4 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_3 \oplus$$
$$\oplus\, x_2y_4 \oplus x_2y_5 \oplus x_2y_6 \oplus x_6y_7$$

$$0 = 1 \oplus x_0 \oplus x_1 \oplus x_5 \oplus x_6 \oplus x_7 \oplus y_6 \oplus x_0x_1 \oplus x_0x_2 \oplus x_1x_4 \oplus x_1x_6 \oplus x_1x_7 \oplus x_2x_5 \oplus$$
$$\oplus\, x_3x_5 \oplus x_3x_7 \oplus x_4x_5 \oplus x_5x_7 \oplus y_0y_2 \oplus y_0y_3 \oplus y_0y_4 \oplus y_0y_6 \oplus y_0y_7 \oplus y_1y_3 \oplus y_1y_4 \oplus$$
$$\oplus\, y_1y_6 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_5 \oplus y_2y_7 \oplus y_4y_6 \oplus x_0y_1 \oplus x_0y_5 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_1 \oplus$$
$$\oplus\, x_1y_3 \oplus x_1y_4 \oplus x_1y_6 \oplus \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_6 \oplus x_3y_1 \oplus$$
$$\oplus x_7y_0$$

$$0 = 1 \oplus x_2 \oplus\ x_4 \oplus x_7 \oplus y_6 \oplus x_0x_1 \oplus x_0x_2 \oplus x_0x_5\ \oplus x_0x_7 \oplus x_1x_2 \oplus x_1x_6 \oplus x_1x_7 \oplus$$
$$\oplus\, x_2x_3 \oplus x_3x_4 \oplus x_3x_6 \oplus x_3x_7 \oplus x_4x_5 \oplus x_4x_6 \oplus x_4x_7 \oplus x_5x_6 \oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_3 \oplus$$
$$\oplus\, y_0y_4 \oplus y_0y_5 \oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_6 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_7 \oplus y_3y_5 \oplus y_3y_6 \oplus y_3y_7 \oplus$$
$$\oplus\, y_4y_5 \oplus y_4y_7 \oplus y_6y_7 \oplus x_0y_2 \oplus x_0y_5 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_2 \oplus x_1y_4 \oplus x_1y_6 \oplus$$
$$\oplus\, x_1y_7 \oplus x_2y_0 \oplus x_2y_2 \oplus x_2y_6 \oplus \oplus x_3y_1 \oplus x_7y_1$$

$$0 = 1 \oplus x_0 \oplus x_6 \oplus y_0 \oplus y_3 \oplus y_4 \oplus y_6 \oplus x_0x_1 \oplus x_0x_4 \oplus x_0x_5 \oplus x_0x_6 \oplus x_0x_7 \oplus x_1x_2 \oplus$$
$$\oplus\, x_1x_4 \oplus x_1x_6 \oplus x_2x_4 \oplus x_3x_4 \oplus x_3x_5 \oplus x_4x_6 \oplus x_5x_6 \oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_1 \oplus y_0y_2 \oplus$$
$$\oplus\, y_0y_5 \oplus y_0y_6 \oplus y_1y_4 \oplus y_1y_5 \oplus y_2y_4 \oplus y_2y_5 \oplus y_2y_7 \oplus y_3y_6 \oplus y_3y_7 \oplus y_4y_5 \oplus y_4y_6 \oplus$$
$$\oplus\, y_5y_6 \oplus y_6y_7 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_5 \oplus x_0y_7 \oplus x_1y_4 \oplus x_1y_5 \oplus x_1y_6 \oplus x_2y_2 \oplus x_2y_7 \oplus$$
$$\oplus\, x_7y_2$$

$$0 = x_0 \oplus x_2 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus y_0 \oplus y_1 \oplus y_2 \oplus y_4 \oplus x_0x_3 \oplus x_0x_4 \oplus x_0x_7 \oplus x_1x_3 \oplus$$
$$\oplus\, x_1x_4 \oplus x_1x_6 \oplus x_2x_4 \oplus x_2x_5 \oplus x_2x_7 \oplus x_3x_4 \oplus x_3x_5 \oplus x_3x_6 \oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_1 \oplus$$
$$\oplus\, y_0y_3 \oplus y_0y_5 \oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_6 \oplus y_2y_6 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_7 \oplus$$
$$\oplus\, y_4y_7 \oplus y_5y_7 \oplus y_6y_7 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_4 \oplus x_1y_5 \oplus$$
$$\oplus\, x_1y_6 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_3 \oplus x_2y_5 \oplus x_2y_6 \oplus x_7y_3$$

$0 = x_0 \oplus x_1 \oplus x_3 \oplus x_5 \oplus y_0 \oplus y_3 \oplus y_4 \oplus y_6 \oplus y_7 \oplus x_0x_2 \oplus x_0x_5 \oplus x_0x_6 \oplus x_1x_2 \oplus$
$\oplus x_1x_4 \oplus x_1x_6 \oplus x_1x_7 \oplus x_2x_3 \oplus x_2x_6 \oplus x_2x_7 \oplus x_3x_4 \oplus x_3x_6 \oplus x_4x_6 \oplus x_5x_6 \oplus x_5x_7 \oplus$
$\oplus y_0y_1 \oplus y_0y_2 \oplus y_0y_4 \oplus y_0y_5 \oplus y_0y_6 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_5 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_5 \oplus$
$\oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_4 \oplus y_3y_5 \oplus y_3y_7 \oplus y_4y_6 \oplus y_4y_7 \oplus y_5y_7 \oplus y_6y_7 \oplus x_0y_0 \oplus x_0y_2 \oplus$
$\oplus x_0y_3 \oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_3 \oplus x_1y_5 \oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_3 \oplus x_2y_5 \oplus x_2y_7 \oplus$
$\oplus x_7y_4$

$0 = 1 \oplus x_1 \oplus x_2 \oplus x_5 \oplus x_6 \oplus y_1 \oplus y_2 \oplus y_4 \oplus y_7 \oplus x_0x_1 \oplus x_1x_2 \oplus x_1x_3 \oplus x_1x_4 \oplus$
$\oplus x_1x_5 \oplus x_1x_6 \oplus x_1x_7 \oplus x_2x_4 \oplus x_2x_5 \oplus x_2x_6 \oplus x_2x_7 \oplus x_4x_5 \oplus x_5x_6 \oplus x_5x_7 \oplus x_6x_7 \oplus$
$\oplus y_0y_4 \oplus y_0y_6 \oplus y_1y_2 \oplus y_1y_3 \oplus y_1y_5 \oplus y_1y_6 \oplus y_1y_7 \oplus y_2y_3 \oplus y_2y_5 \oplus y_3y_5 \oplus y_3y_6 \oplus$
$\oplus y_3y_7 \oplus y_4y_5 \oplus y_5y_6 \oplus y_6y_7 \oplus x_0y_1 \oplus x_0y_3 \oplus x_0y_4 \oplus x_1y_0 \oplus x_1y_2 \oplus x_1y_3 \oplus x_1y_4 \oplus$
$\oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_2 \oplus x_2y_3 \oplus x_2y_5 \oplus x_7y_5$

$0 = 1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus y_1 \oplus x_0x_3 \oplus x_1x_3 \oplus x_1x_5 \oplus x_1x_6 \oplus x_1x_7 \oplus x_2x_3 \oplus x_2x_6 \oplus$
$\oplus x_2x_7 \oplus x_3x_5 \oplus x_3x_6 \oplus x_4x_7 \oplus x_5x_7 \oplus x_6x_7 \oplus y_0y_3 \oplus y_0y_4 \oplus y_0y_7 \oplus y_1y_2 \oplus y_1y_3 \oplus$
$\oplus y_1y_4 \oplus y_1y_5 \oplus y_1y_6 \oplus y_2y_7 \oplus y_3y_4 \oplus y_4y_5 \oplus y_5y_6 \oplus x_0y_1 \oplus x_0y_3 \oplus x_0y_7 \oplus x_1y_2 \oplus$
$\oplus x_1y_3 \oplus x_1y_5 \oplus x_2y_1 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_5 \oplus x_7y_6$

$0 = x_2 \oplus x_3 \oplus x_5 \oplus x_6 \oplus y_0 \oplus y_1 \oplus y_2 \oplus y_3 \oplus y_4 \oplus x_0x_1 \oplus x_0x_3 \oplus x_0x_5 \oplus x_0x_6 \oplus$
$\oplus x_0x_7 \oplus x_1x_2 \oplus x_2x_3 \oplus x_2x_5 \oplus x_3x_4 \oplus x_3x_5 \oplus x_3x_6 \oplus x_3x_7 \oplus x_4x_6 \oplus x_4x_7 \oplus y_0y_7 \oplus$
$\oplus y_1y_3 \oplus y_1y_4 \oplus y_1y_7 \oplus y_2y_3 \oplus y_2y_4 \oplus y_2y_5 \oplus y_2y_6 \oplus y_2y_7 \oplus y_3y_4 \oplus y_4y_5 \oplus y_4y_7 \oplus$
$\oplus y_6y_7 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus x_0y_4 \oplus x_0y_6 \oplus x_1y_0 \oplus x_1y_2 \oplus x_1y_3 \oplus x_1y_4 \oplus x_1y_7 \oplus$
$\oplus x_2y_0 \oplus x_2y_4 \oplus x_2y_6 \oplus x_2y_7 \oplus x_3y_1 \oplus x_7y_7$

Формула В.3 – Шифрлау алгоритмінде қолданылған S-блоктан XSL шабуылы үшін алынған тәуелсіз квадраттық теңдеулер

$0 = 1 \oplus x_0 \oplus x_1 \oplus x_3 \oplus y_2 \oplus y_7 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_7 \oplus x_1y_1 \oplus x_1y_2 \oplus x_2y_3 \oplus x_2y_4 \oplus$
$\oplus x_2y_5 \oplus x_2y_6 \oplus x_2y_7 \oplus x_3y_2 \oplus x_3y_3 \oplus x_3y_4 \oplus x_3y_5 \oplus x_3y_6 \oplus x_3y_7 \oplus x_4y_1 \oplus x_4y_3 \oplus$
$\oplus x_4y_5 \oplus x_5y_0$

$0 = 1 \oplus x_0 \oplus x_2 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus y_1 \oplus y_3 \oplus y_5 \oplus y_6 \oplus y_7 \oplus x_0y_0 \oplus x_0y_2 \oplus$
$\oplus x_0y_5 \oplus x_1y_0 \oplus x_1y_2 \oplus x_1y_4 \oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_6 \oplus x_2y_7 \oplus$
$\oplus x_3y_2 \oplus x_3y_3 \oplus x_3y_4 \oplus x_4y_2 \oplus x_4y_3 \oplus x_4y_4 \oplus x_4y_6 \oplus x_5y_2$

$0 = x_3 \oplus x_4 \oplus x_7 \oplus y_4 \oplus y_6 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_4 \oplus x_0y_5 \oplus x_0y_7 \oplus x_1y_0 \oplus$
$\oplus x_1y_1 \oplus x_1y_3 \oplus x_1y_5 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_7 \oplus x_3y_0 \oplus x_3y_2 \oplus$
$\oplus x_3y_6 \oplus x_3y_7 \oplus x_4y_3 \oplus x_4y_4 \oplus x_4y_5 \oplus x_4y_7 \oplus x_5y_3$

$0 = x_0 \oplus x_1 \oplus x_3 \oplus x_4 \oplus x_5 \oplus y_2 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_7 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus$
$\oplus x_0y_5 \oplus x_0y_6 \oplus x_1y_1 \oplus x_1y_3 \oplus x_1y_5 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_5 \oplus x_2y_6 \oplus$
$\oplus x_3y_1 \oplus x_3y_2 \oplus x_3y_4 \oplus x_3y_5 \oplus x_3y_6 \oplus x_4y_0 \oplus x_4y_4 \oplus x_4y_5 \oplus x_4y_6 \oplus x_5y_4$

$0 = x_0 \oplus x_1 \oplus x_6 \oplus y_0 \oplus y_1 \oplus y_3 \oplus y_4 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus x_0y_5 \oplus x_1y_2 \oplus x_1y_5 \oplus$
$\oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_5 \oplus x_3y_0 \oplus x_3y_2 \oplus x_3y_4 \oplus x_4y_1 \oplus$
$\oplus x_4y_2 \oplus x_4y_4 \oplus x_4y_5 \oplus x_4y_7 \oplus x_5y_1 \oplus x_5y_5$

$$0 = 1 \oplus x_1 \oplus x_4 \oplus x_5 \oplus x_7 \oplus y_1 \oplus y_6 \oplus y_7 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus x_0y_5 \oplus x_0y_6 \oplus$$
$$\oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_4 \oplus x_1y_5 \oplus x_1y_6 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_3 \oplus x_2y_6 \oplus x_3y_1 \oplus x_3y_3 \oplus$$
$$\oplus x_3y_4 \oplus x_3y_5 \oplus x_4y_1 \oplus x_4y_2 \oplus x_4y_4 \oplus x_4y_5 \oplus x_5y_6$$

$$0 = x_0 \oplus x_1 \oplus x_2 \oplus x_5 \oplus x_6 \oplus y_2 \oplus y_5 \oplus y_6 \oplus y_7 \oplus x_0y_0 \oplus x_0y_4 \oplus x_0y_6 \oplus x_0y_7 \oplus$$
$$\oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_4 \oplus x_2y_4 \oplus x_2y_7 \oplus x_3y_4 \oplus x_3y_5 \oplus x_3y_6 \oplus x_4y_0 \oplus x_4y_2 \oplus$$
$$\oplus x_4y_6 \oplus x_4y_7 \oplus x_5y_1 \oplus x_5y_7$$

$$0 = x_0 \oplus x_3 \oplus x_4 \oplus x_7 \oplus y_2 \oplus y_4 \oplus x_0y_1 \oplus x_0y_3 \oplus x_0y_5 \oplus x_0y_7 \oplus x_1y_1 \oplus x_1y_3 \oplus x_1y_5 \oplus$$
$$\oplus x_1y_6 \oplus x_2y_1 \oplus x_2y_4 \oplus x_2y_7 \oplus x_3y_5 \oplus x_4y_0 \oplus x_4y_5 \oplus x_4y_7 \oplus x_5y_1 \oplus x_6y_0$$

$$0 = x_1 \oplus x_3 \oplus x_5 \oplus x_7 \oplus y_4 \oplus y_6 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus x_0y_5 \oplus x_1y_3 \oplus x_1y_4 \oplus$$
$$\oplus x_1y_5 \oplus x_2y_1 \oplus x_2y_4 \oplus x_2y_5 \oplus x_3y_3 \oplus x_3y_5 \oplus x_3y_7 \oplus x_4y_0 \oplus x_4y_2 \oplus x_4y_5 \oplus x_4y_6 \oplus$$
$$\oplus x_5y_1 \oplus x_6y_1$$

$$0 = x_1 \oplus x_3 \oplus x_7 \oplus y_2 \oplus y_3 \oplus y_5 \oplus y_6 \oplus y_7 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_1 \oplus$$
$$\oplus x_1y_7 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_4 \oplus x_2y_7 \oplus x_3y_2 \oplus x_3y_3 \oplus x_3y_6 \oplus x_4y_0 \oplus x_4y_1 \oplus x_4y_2 \oplus$$
$$\oplus x_4y_3 \oplus x_4y_4 \oplus x_4y_5 \oplus x_4y_7 \oplus x_6y_2$$

$$0 = x_0 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_7 \oplus y_0 \oplus y_1 \oplus y_4 \oplus y_5 \oplus y_7 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus$$
$$\oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_2 \oplus x_2y_4 \oplus x_2y_6 \oplus x_2y_7 \oplus x_3y_0 \oplus x_3y_2 \oplus x_3y_5 \oplus x_3y_6 \oplus x_4y_0 \oplus$$
$$\oplus x_4y_1 \oplus x_4y_2 \oplus x_4y_3 \oplus x_4y_4 \oplus x_4y_5 \oplus x_4y_6 \oplus x_6y_3$$

$$0 = x_0 \oplus x_1 \oplus x_2 \oplus x_4 \oplus x_5 \oplus y_0 \oplus y_1 \oplus y_4 \oplus y_5 \oplus x_0y_1 \oplus x_0y_4 \oplus x_0y_5 \oplus x_0y_6 \oplus$$
$$\oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_4 \oplus x_1y_5 \oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_2 \oplus x_2y_3 \oplus x_2y_4 \oplus$$
$$\oplus x_2y_6 \oplus x_3y_1 \oplus x_3y_3 \oplus x_3y_6 \oplus x_3y_7 \oplus x_4y_1 \oplus x_4y_2 \oplus x_4y_3 \oplus x_4y_4 \oplus x_4y_5 \oplus x_4y_6 \oplus$$
$$\oplus x_4y_7 \oplus x_6y_4$$

$$0 = 1 \oplus x_2 \oplus x_3 \oplus x_5 \oplus x_6 \oplus y_1 \oplus y_4 \oplus y_5 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_4 \oplus x_0y_5 \oplus$$
$$\oplus x_0y_7 \oplus x_1y_1 \oplus x_1y_3 \oplus x_1y_4 \oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_7 \oplus x_3y_3 \oplus$$
$$\oplus x_3y_4 \oplus x_3y_7 \oplus x_4y_1 \oplus x_4y_2 \oplus x_4y_3 \oplus x_4y_5 \oplus x_6y_5$$

$$0 = x_2 \oplus x_3 \oplus x_5 \oplus y_1 \oplus y_2 \oplus y_3 \oplus y_4 \oplus y_7 \oplus x_0y_3 \oplus x_0y_7 \oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_4 \oplus$$
$$\oplus x_1y_6 \oplus x_2y_1 \oplus x_2y_3 \oplus x_2y_5 \oplus x_2y_6 \oplus x_2y_7 \oplus x_3y_0 \oplus x_3y_3 \oplus x_3y_4 \oplus x_3y_5 \oplus x_4y_1 \oplus$$
$$\oplus x_4y_4 \oplus x_4y_5 \oplus x_4y_6 \oplus x_4y_7 \oplus x_5y_1 \oplus x_6y_6$$

$$0 = 1 \oplus x_0 \oplus x_2 \oplus x_3 \oplus x_5 \oplus x_6 \oplus y_0 \oplus y_4 \oplus y_6 \oplus y_7 \oplus x_0y_0 \oplus x_0y_4 \oplus x_0y_5 \oplus x_0y_6 \oplus$$
$$\oplus x_1y_1 \oplus x_1y_2 \oplus x_1y_3 \oplus x_1y_4 \oplus x_1y_6 \oplus x_2y_1 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_6 \oplus x_2y_7 \oplus x_3y_1 \oplus$$
$$\oplus x_3y_4 \oplus x_3y_7 \oplus x_4y_4 \oplus x_4y_7 \oplus x_6y_7$$

$$0 = x_0 \oplus x_1 \oplus x_7 \oplus y_0 \oplus y_2 \oplus y_4 \oplus y_6 \oplus x_0y_3 \oplus x_0y_5 \oplus x_0y_6 \oplus x_0y_7 \oplus x_1y_2 \oplus x_1y_6 \oplus$$
$$\oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_4 \oplus x_3y_0 \oplus x_3y_1 \oplus x_3y_3 \oplus x_3y_5 \oplus x_3y_7 \oplus x_4y_0 \oplus$$
$$\oplus x_4y_1 \oplus x_4y_2 \oplus x_4y_3 \oplus x_4y_6 \oplus x_7y_0$$

$$0 = 1 \oplus x_0 \oplus x_1 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus y_0 \oplus y_1 \oplus y_2 \oplus y_6 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_5 \oplus$$
$$\oplus x_0y_7 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_4 \oplus x_2y_1 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_6 \oplus x_3y_0 \oplus x_3y_2 \oplus x_3y_4 \oplus$$
$$\oplus x_3y_6 \oplus x_4y_0 \oplus x_4y_1 \oplus x_4y_3 \oplus x_4y_4 \oplus x_4y_5 \oplus x_7y_1$$

$$0 = 1 \oplus x_0 \oplus x_1 \oplus x_4 \oplus x_5 \oplus x_7 \oplus y_2 \oplus y_5 \oplus y_7 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus \oplus x_0y_5 \oplus x_1y_1 \oplus x_1y_5 \oplus x_1y_7 \oplus x_2y_3 \oplus x_2y_4 \oplus x_2y_6 \oplus x_2y_7 \oplus x_3y_5 \oplus x_3y_7 \oplus x_4y_7 \oplus \oplus x_5y_1 \oplus x_7y_2$$

$$0 = x_0 \oplus x_1 \oplus x_3 \oplus x_5 \oplus x_6 \oplus y_2 \oplus y_3 \oplus y_4 \oplus y_5 \oplus x_0y_0 \oplus x_0y_4 \oplus x_0y_7 \oplus x_1y_1 \oplus \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_7 \oplus x_3y_2 \oplus x_3y_3 \oplus x_3y_5 \oplus x_3y_7 \oplus x_4y_1 \oplus x_4y_3 \oplus x_4y_4 \oplus x_4y_6 \oplus \oplus x_7y_3$$

$$0 = x_1 \oplus x_2 \oplus x_4 \oplus x_6 \oplus x_7 \oplus y_5 \oplus y_7 \oplus x_0y_0 \oplus x_0y_3 \oplus x_0y_4 \oplus x_0y_5 \oplus x_1y_1 \oplus \oplus x_1y_2 \oplus x_1y_5 \oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_3 \oplus x_2y_4 \oplus x_3y_2 \oplus x_3y_3 \oplus x_3y_5 \oplus \oplus x_3y_7 \oplus x_4y_1 \oplus x_4y_3 \oplus x_4y_4 \oplus x_5y_1 \oplus x_7y_4$$

$$0 = x_2 \oplus x_3 \oplus x_4 \oplus x_6 \oplus y_1 \oplus y_2 \oplus y_3 \oplus y_5 \oplus y_6 \oplus x_0y_3 \oplus x_0y_5 \oplus x_1y_1 \oplus x_1y_2 \oplus \oplus x_1y_3 \oplus x_1y_7 \oplus x_2y_0 \oplus x_2y_1 \oplus x_2y_2 \oplus x_2y_4 \oplus x_2y_7 \oplus x_3y_0 \oplus x_3y_1 \oplus x_3y_6 \oplus x_4y_0 \oplus \oplus x_4y_3 \oplus x_4y_4 \oplus x_4y_7 \oplus x_5y_1 \oplus x_7y_5$$

$$0 = x_5 \oplus y_0 \oplus y_4 \oplus x_0y_0 \oplus x_0y_1 \oplus x_0y_2 \oplus x_0y_3 \oplus x_0y_4 \oplus x_0y_7 \oplus x_1y_1 \oplus x_1y_3 \oplus x_1y_4 \oplus \oplus x_1y_6 \oplus x_1y_7 \oplus x_2y_1 \oplus x_2y_5 \oplus x_2y_6 \oplus x_2y_7 \oplus x_3y_0 \oplus x_3y_1 \oplus x_3y_3 \oplus x_3y_5 \oplus x_3y_6 \oplus \oplus x_4y_3 \oplus x_4y_5 \oplus x_4y_6 \oplus x_7y_6$$

$$0 = x_0 \oplus x_2 \oplus x_5 \oplus y_4 \oplus y_6 \oplus x_0y_0 \oplus x_0y_2 \oplus x_0y_3 \oplus x_0y_6 \oplus x_1y_0 \oplus x_1y_1 \oplus x_1y_2 \oplus \oplus x_1y_3 \oplus x_1y_5 \oplus x_2y_0 \oplus x_2y_4 \oplus x_2y_5 \oplus x_2y_7 \oplus x_3y_2 \oplus x_3y_4 \oplus x_3y_6 \oplus x_3y_7 \oplus x_4y_0 \oplus \oplus x_4y_2 \oplus x_4y_4 \oplus x_4y_5 \oplus x_4y_6 \oplus x_7y_7$$