Дәрістік қысқаша конспектісі

Дәріс №1. Тақырыбы: Компьютерлік жүйеде ақпаратты қорғау проблемалары.

Жоспар:

1. Қауіпсіздік түсінігі
2. Деректердің бүтіндігі, қол жетерлігі, құпиялылығы
   Қауіп қатердің жіктелуі
3. Қауіпсіздікті қамтамасыз ету үшін жүйелі ықпал ету
4. Аутентификация, авторизация, аудит
5. Аутентификация технологиясы
6. Сертификат негізіндегі аутентификация
7. Kerberos жүйесі
8. Сервер қорына енуге рұқсат алу
9. Қорға енуге рұқсат алу
10. Есептер мен жаттығулар

Мақсаты: Компьютерлік жүйеде ақпаратты қорғау проблемалары туралы мәлімет беру

Кілттік сөздер: құпиялылық (confidentiality*), е*ну мүмкіндігі (availability), бүтіндігі (integrity), аутентификация

Ақпараттық қауіпсіздік жүйесін қарастырғанда әдетте екі мәселелер тобына бөледі: компьютер қаупсіздігі және желілік қауіпсіздік. Компьютердің қауіпсіздігіне деректерді қорғаудың барлық мәселелері жатады, яғни компьютерде сақталатын, өңделетін автономды жүйе ретінде қарастырылатын деректер мәселелері. Бұл мәселелер деректер қорымен,компьютердің енгізілген аппаратты құрылғыларымен, операциялық жүйе құралдарымен шешіледЖелілік қауіпсіздік түсінігі желілік құрылғылар араындағы ақпарат алмасу кезіндегі қорғау және рұқсатсыз енуден қорғау мәселелері болшып табылады. Бірақ қазіргі уақытта компьютер мен желілік қауіпсіздікті бір-бірінен айырмашылығын анықтау өте қиын мәселе, олар бір –бірімен өте тығыз байланыста жатады. Тек желілік қауіпсіздік өзінің мамандығын қажет етеді. Жеке дара жағдайда жұмыс істеп тұрған компьютерге ішкі қол салудан сақтау үшін әртүрлі тиімді әдістерді қолдануға болады: мысалы клавиатураны құлыпқа жабу немесе қатты дискіні шығарып алып сейфке салып тастау. Желіде жұмыс істеп тұрған компьютер қоғамнан бөлініп қала алмайды ол басқа компьютерлермен кез-келген уақытта байланыста болуы керек. Сондықтан желіде қауіпсіздік қорғау өте күрделі жағдай болып есептеледі. Басқа қолданушының желіде жұмыс істеп тұрған компьютерді қолдануы логикалық тұрғыдан болуы шарт. Осындай жағдайда қауіпсіздікті қорғауды қамтамасыз ету бір ойға әкеліп соқтырады – яғни әрбір қолданушы үшін өзінің ақпаратты қолданатын құқығы желідегі әрбір компьютер үшін желілік байланыстарын реттеп отыру қажет. Б9л м2селелерден басқа желілер өзінің

табиғатында тағы да бір қауіп түріне кездеседі. желімен берліген хабарды ұстап қалу анализдеу «жалған» трафиктер құру. Желі қауіпсіздігін қорғаудың негізгі бөлшегі осы қылмыстың алдын алуға жұмсалады. Желілік қауіпсіздігінің мәселелері коорпаративті желілерден бөлінген каналдарға өту барысында көптеп байқалады. (Интернет, frame relay). Жалпы желіні қамтамсыз етушілер қазірше берілгендерді өзінің магистралы бойынша тасымалдау кезінде қолданушы деректерін қорғауды сирек қолданады. Бұл мәселені көбінесе құпиялылығын, бүтіндігін, ену мүмкіндігін қолданушының өзінің қамқорына қалдырады. Құпиялылықтың негізгі түсінігі деректердің бүтіндігі және неу мүмкіндігі. Қауіпсіз ақпараттық жүйе – ол біріншіден рұқсат етілмеген енуден қорғайды, екіншіден ақпаратты өзінің қолданушыларына барлық уақытта деректерге енгізе алады, үшіншіден ақпаратты сенімді сақтайды және деректердің өзгермеуін қамтамасыз ететін жүйе. Осылайша, анықтама бойынша қауіпсіз жүйе құпиялық, ену мүмкіндігімен, бүтіндік қасиетімен анықталады.

құпиялылық (confidentiality) — бұл құпия деретерге ену тек қана рұқсат берілген қолданушыларға берілетіндігінің кепілі. (Бұл қолданушылар авторластырылған деп аталады).

Ену мүмкіндігі (availability) — авторластырылған қолданушылар барлық уақытта деректерге ену құқығы бар болуының кепілі.

Бүтіндігі (integrity) — деректерді түрлі жолдармен өзгертуге авторластырылмағандар үшін рұқсат етілмеуді қамтамасыз ететін дұрыс мәнді деректерді сақтаудың кепілі

Қауіпсідікті қорғау жүйесі жүйенің арналымына, қолданылатын деретердің мінедемесіне, қауіп мүмкіндігінің типіне байланысты өзгеруі мүмкін. Бүтін және ену мүмкіндігі бар жүйені көз алдымызға елестету өте қиын, бірақ құпиялық қасиеті бар болуы міндетті емес. Мысалы: егер Интернетке Web-сервер сіз ақпарат жариялайсыз және сіздің мақсатыңыз оны өте үлкен көлемде адамдарға ену мүмкіндігін беру. Онда сізден осы жағдайда құпилылықты талап етпейді. Бірақ-та бүтіндік және ену мүмкіндігі көкейкесті мәселе болып қала береді. Шындығында сіз егер деретердің бүтіндігін қамтамасыз ететін арнайы шаралар қолданбасаңыз, онда қаскүнем сіздің серверіңіздегі деректерді өзгертуі мүмкін және сіздің ұжымыңызға шығын әкеледі. Қаскүнем мысалы Web-серверде орналасқан деректерге мынадай өзгерістер енгізуі мүкін. Мысалы: прайс –парақ тағы сіздің бәсекелес қабілеттілін төмендететін ақпараттар орналастыруы мүмкін. Ұжым біраз шығын шығара отырып интернетте серверді қамтамасыз ету барысында мынадай жетістіктерге қолжеткізуі мүмкін: тұтынушылар санының, сату санының ұлғайюына, т.с.с. Бірақта қаскүнем атака жасауы мүмкін нәтижесінде сервердегі арнайы арналған деректер ену мүмкіндігі болмай қалады. Мұндай қаскүнем әрекеттің мысалы ретінде кері адрестері дұрыс емес IP-пакеттермен «атқылау», сәйкесінше протокол жұмысын тайм-аут шақырады немесе басқа да сұранымдарға жауап бере алмайтын жағдайға жетеді. Құпиялылық, бүтіндік, ену мүмкіндігі түсінігі тек қана ақпаратқа байланысты анықталмайды сонымен бірге есептеу желісінің басқа да қорларына, мысалы

ішкі құрылғыларына немесе қосымшаларға. Жүйенің қауіпсіздігінің бұзылуына әкеліп соқтыратын көптеген «заңсыз» қолданылу мүмкіндігі бар жүйелік қорлар бар. Мысалы шексіз ену құқығы бар баспа құрылғысына қаскүнемге баспадағы құжаттың көшірмесін алуға, баптау параметрлерін өзгертуге мүмкіндік береді. Бұл жұмыстың кезектілігіне әсерін тигізіп немесе құрылғы істен шығып қалуы да мүмкін.

Құпиялылық қасиеті қолданушыға тек қана анықталған қолданушар ғана қолдана алатындай және құрылғымен тек қана анықталған операцияларды орындай алатындай интерпретациялауға болады.

Ену мүмкіндігі қасиеті барлық уақытта қолдануға дайын екендігін көрсетеді.

Бүтіндік қасиеті осы құрылғының баптау параметрлері өзгермеуі қасиетімен анықталады. Құрылғы бірнеше қызметтер жасайды: мәтінді теру, факс жіберу, Интернетке ену, электронды пошта және т.с.с. Бұл жғдайды заңсыз қолданғанда ұжымға айтарлықтай шығын әкелуі мүмкін және жүйенің қауіпсіздігінің бұзылуына әкеліп соқтырады. Ақпараттың қ ұпилылығына, бүтіндігіне , ену мүмкіндігіне бағытталға кез -келген іс-әрекет және басқа да қорларды заңсы қолдану ***қауіп*** болып саналады. Таратылған қауіп ***шабуыл*** деп аталады. Риск — бұл ойдағыдай өткізілген шабул нәтижесінде ақпарат қоры иесінің шығынға ұщырау мүмкіндігінің өлшемі. Қауіпсіздік жүйесі осал жері көп болғанда риск өлшемі өте үлкен. Қауі классификациясында әмбебаб қауіп болмайды, мүмкін, өйткені адамның творчестволық мүмкіндігінде шек жоқ, күн сайын желіге заңсыз ену әдісі қолданылып жатады, жаңа вирустар пайда болып жатады, бағдарламалық және аппараттық желілік тауарларда жаңа ақаулар табылып жатады. Бұған жауап ретінде көптеген қауіп көздеріне шек қоятын қауіпсздік құралдары өңделіп шығады, одан кейін ол шабуылдың жаңа объектісі болып қалады. Сонда да болса біз бірнеше талдау жасап көрелік.Біріншіден қауіп қасақана жасалған және байқамай жасаған болып екіге бөлеміз. Байқамай жасаған қауіп деңгейі төмен немесе жауапкершілігі жоқ қызметкердің қате жасалған іс әрекетінен туындайды. Бұдан басқа осы типті қауіп жүйедегі бағдарламалық және аппараттық құрылғылардың сенімді емес жұмысынан штуындайды. Мысалы дискінің істен шығуына байланысты ұжымға керек ақпараттарды өз уақытындам жібере алиай, қолданушыларды ену құқығынан айырады. Сондықтан қауіпсіздік жағдайы сенімділік жағдайымен тығыз жанасып жатады. Бағдарламалық-аппараттық құралы жұмысының сенімділігінен шығатын қауіпсіздік қауіптерін, оларды барлық уақытта аппаратура деңгейінде резервтеуді қолдану, толық жетілдіріп отыру жолымен шешіледі (RAID-массивтер, көппроцессорлы компьютерлер, үзіліссіз ток көзі, кластерлі архитектуралар) немесе деректерді массивтеу деңгейінде (файлды тираждау, резервті көшірмелер). әдейі жасалған қауіп қатер пассивті деректерді оқумен шектелуі мүмкін немесе өзіне активті іс-әрекеттер қосатын мониторингті жүйе. Мысалы ақпараттың бүтіндігі мен ену мүмндігін, құрылғылар мен қосымшалардың істен шығуына әкеліп соқтырады. Қасақана асалған қауіп хакерлардың тәжірибесінің нәтижесінде

пайда болады және ол нақ ұжымның жұмысына шығын әкелетіні сөзсіз. Есептеу желісінде қасақана жасалған қауіпті келесі типтерге бөлуге болады:

Заңды қолданушы ретінде компьютерлердің біріне заңсыз ену;

Вирус – бағдарламалары көмегімен жүйені бұзу әрекеттері;

Заңды қолданушының заңсыз әрекеттері;

Ішкі желіні «тыңшы» тыңдау.

Заңсыз ену операциялық жүйенің құжатталмаған мүмкіндіктерін қолдана отырып, қауіпсіздік жүйесінің осал жері арқылы таралуы әбден мүмкін. Бұл мүмкіндіктер қаскүнемге желіге енуді бақылайтын стандартты процедураны «айналып» өтуге мүмкіндік туғызады. Өзге қолданушының паролын көріп алу арқылы енуі де әбден мүмкін. Сондықтан да барлық қолданушылар өз паролдарын құпия ұстаған жөн. Паролды көрудің тағы бір амалы ол өзге компьютерге **«троянского конь»** енгізу арқылы. Бұл қаскүнемнің берілген іс-әрекетін орындайтын, компьютер иесіне бағынбайтын резидентті бағдарлама. Бұл бағдарлама компьютер иесі жүйеге енген уақытта енгізілген паролды жаттап жібереді. «троянский конь» бағдарламасы барлық уақытта пайдалы утилиттер мен ойындармен бірге маскіленеді.

Аутентификация, авторизация, аудит

Аутентификация

Аутентификация (authentication) қажет емес адамдарды желіге енгізбей және заңды қолданушыларға желге енуге рұқсат береді. «аутентификация» термині латын тілінен аударғанда «шынайылықты орнату» дегенді білдірді. Аутентификацияны идентификациядан айырып білу қажет. Қолданушы идентификаторлары файлдың, прцесстің, басқа да объектілер сияқты жүйеде қолданылады, бірақ олары қауіпсіздікті қамтамасыз етумен байланысты. Идентификация қолданушымен жүйеге хабар өзінің идентификатоын тіркейді, осы уақытта аутентификация – бұл енгізілген идентификатор тек қана өзінікі екендігі, және ол сол адам екендігін дәлелдетін процедура. Аутентификация процедурасында екі жақ қатысады: бір жақ өзінің бірнеше дәлелден тұратын аутентті екендігін дәлелдейді, ал екінші жақ — аутентификатор — осы дәлелдерді тексеріп шешім шығарады. Аутенттіліктік дәлелі ретінде неше түрлі әдістер қолданылады: Ауденттілікті мумкін болғанша екі жаққада құпия ұстап сақтауға болады: парольдық сөз немесе фактні (болған оқиғаның уақыты,адамның аты т.б.) сонымен қатар ауденттілік керемет қасиеттерімен ерекшеленеді (физикалық кілтпн) мысалы электрондық магниттік карта ; аудендификация ұқсастығын көрсетеді, өзінің мінездемесін пайдалана отырып;қол таңба немесе көз қарашығының суреті ,аудентификатордың мәліметтер базасына алдын ала енгізілген болуы керек.

Ақпараттық аутентификация

Алынған баяндамадағы мәліметтер негізінде , жүйе арқылы алынған мәліметтердің компьютерлік жүйедегі нақтылығын аутентмфикация арқылы түсінуге болады.Егер мәліметтерді шифрлау соңғы мақсаты ретінде бұл мәліметтер мен санкциясыз танысуға қорғаныс қоюмен қамтамасыз ету,онда аутентификацияның соңғы мақсаты мәліметтер мен алмасу кезіндегі оған қатысушылардың жалған мәлімет алмауын қамтамасыз ету.аутентификацияның концепциясы кең мағынасында мәліметтердің жалғандығын анықтауға қатысушылардың арасындағы сенімділіктің болуымен және болмауы.Компьютерлік жүйеде аутентификациялық мәліметтердің екі түрі бар:мәліметтер мен бағдарламада сақталынатын аутентификация- мәліметтер модификацияға ұшырамағанын дәлелдейтін факті,берілген мәліметтермодификацияланбауы тиіс.Ауденфикациялық баяндама-алынған баяндаманың жалған болып орналастырылуы, сондай-ақ осы баяндаманың авторлығы жайлы жауаптың шешімі және қабылдау фактісінің орналастырылуы.

Бекіту сұрақтары:

1. Қауіпсіздік түсінгі дегеніміз не?
2. Деректердің бүтіндігі, қол жетерлігі, құпиялылығы дегеніміз не?
3. Қауіп қатердің жіктелуі не үшін қажет?
4. Қауіпсіздікті қамтамасыз ету үшін жүйелі ықпал ету
5. Аутентификация, авторизация, аудит міндеттері.
6. Аутентификация технологиясы дегеніміз не?
7. Kerberos жүйесі
8. Сервер қорына енуге рұқсат алу жолдары.
9. Қорға енуге рұқсат алу

Пайдаланылатын әдебиет: [1], 7-14 бет

Дәріс №2. Тақырыбы: Ақпаратты қорғаудың құқықт ық негізі
Жоспар:

1. Ақпараттар категориялары.
2. Ақпараттық ресурстардың қауіпсіздік жағдайы
3. Толассыз мәліметтерді шифрлау

Мақсаты: Ақпараттық ресурстардың қауіпсіздік жағдайларымен таныстыру

Кілттік сөздер: ПЭВМ, *гаммалар,* шифр, PRG

1. Ақпараттар категориялары.

Ақпараттарды қорғау ауданында ең негізгі заны 26 маусымда 1998 жылы еңгізілген. Ол № 233-1 “Қазақстан Республикасының халық қауыпсыздығы туралы” заны, 22 бабы. “ақпараттық қауыпсыздықты қамтамасыз ету” Қазақстан Республикасында ақпараттық ресурстарды қолдану және қалыптастыру кезінде пайда болған қатынасты жинақтау, сақтау, тарату және қолданушыларға құжаттық ақпараттарды қамтамасыз етеді, сонымен қатар ақпараттық технологияларды құру және қолдану кезінде, ақпаратты қорғау кезінде қамтамасыз етеді.

Мемлекттік органдарда, ұйымдарда, тәуелсіздік формасына қарамастан, азаматтар мен қызметкер адамдарға занға байланысты мынадай шараларды міндетті түрде қолдамауға қажет:

1. Қазақстанның ақпараттық тәуелділігіне;
2. Басқа мемлекеттер тарапынан ақпараттық жаулап алу;
3. Президент, Парламент, Правительство және Қазақстан Республиаксының халықтық қаупсыздық күшінің ақпараттық изоляциясын қаматамасыз етуде.

Халықтын қауыпсыздығына күмән келтіретін әрекеттерді қолдануға тиым салынады, мысалы:

1. мемлекеттік құпия жұмыстармен айналысқан қызметшілер бес жыл бойы Қазақстан Республика шек арасынаң тұрақты тұрғын жерін ауыстыруға болмайды.
2. Қазақстан Республика территориясында халықтың қауыпсыздығына күмән келтіретін жазба ақпататтарды, радио- және теледидарда шетелдік массалық ақпараттарды таратуға болмайды.
3. Мемлекеттін мүдесіне байланысты жұмыс бабындағы немесе басқада бір ақпараттарды жаюға болмайды
4. шетел адамдарға тура немесе жанама түрінде 20 шақты процеңттік акция пакеттерін қолдануға, білуге, пайдалануға болмайды.

Осы заңнаң :

• барлық білім және іс әрекет аймағындағы ақпараттар ашық және көпшілік қолды болады, егер заң шығарушылар белгілі бір тәртіппен шектелмеген болса.

• «конфиденциялдық ақпарат» категориясы барлық қорғаныш ақпараттар/құпия/ түрлерін біріктіреді. Бір ерекшелігі, ақпарат мемлекеттік құпия болып саналады: ол конфиденциялдық ақпаратқа жатпайды, белгілі бір шектелген енісі бар құрама болімді ақпарат.

Ақпараттар категорияларын бөлінеді:

- Мемлекеттік құпия;
- Конфиденциялдық ақпараттар;
- азаматтар туралы персоналдық мәліметтер.

2. Ақпараттық ресурстардың қауіпсіздік жағдайы.

1. **ақпараттық ресурстардын қауыпы.**
2. **ақпараттық ресурстарға шабуыл**

ақпараттық ресурстардын қауыпы— ақпараттық жүйеге әсер ететін потенциальді ықтималды оқиға. Компьютерлік жүйенің күнділіктілігі— бұл қауп түндірудін мүмкіндігін тудыратын, оның үйлесімсіз сипаттамасы.

Ақпараттық қауіпсіздіктін қауып түрлері.

Үш негізгі түрі болады:

1 ақпараттық ресурстардың ашылуы;
2 олардың бүтіндігін бұзу;
3 қамтамасыз етуден бас тарту.

Ақпараттық ресурстардың ашылуы дегеніміз, ол ақпараттың және білімнің барлығына қол жетерлігі.

Олардың бүтіндігін бұзу деп әртүрлі әди өзгерткен әрекеттерді айтады.

Қамтамасыз етуден бас тарту бір немесе бірнеше ақпараттық жүйенің ресурстарының блокировкасы кезінде пайда болады.

Қауіптің классификациясын қарастырайық.

Ұрлап алу мүмкін:

- ПЭВМ және желілер жабдықталатын аппараттық құрылғылар (блоктар, узелдар және дайын бөлшектер);
- Программалық қамтамсыз ету және ақпараттық жетектер;
- Сақталған ақпараттардың баспадан басып шыққан көшірмесін.

Ұрлап алып кету мүмкін:

- Қолданушылардың жұмыс орнынаң;
- Транспортировка кезінде;
- Сақталып тұрған жерден.

Қауіп жетектердін потенциальдық субъектілері болып:

- конкурентар;
- бірге әскерде болғандар;
- посетителдер;
- қаскүнемдер;
- жинауштар;
- жөндеушілер;
- жабдықтаушылар;
- күзет;
- құрылысшылар;
- терезе жүғыштар;
- қызметкерлер.

Қызметкерлерге келесі өкілдер инстанциялары жатады;

- санитарлі-эпидемиологиялық станциялар;
- телефондық компаниялар;
- өрт сөндіруді қадағалаушылар;
- ведомстводан тыс күзет;
- жалға берушілер;

- қаржы органдар;
- салық органдар.

ақпараттық ресурстарға шабуыл— бұл бұзушылардың қолданылатын әрекеті. Яғни шабуыл — бұл қаіпт тудыру.

3. Толассыз мәліметтерді шифрлау

Шифрлер құрылысы.

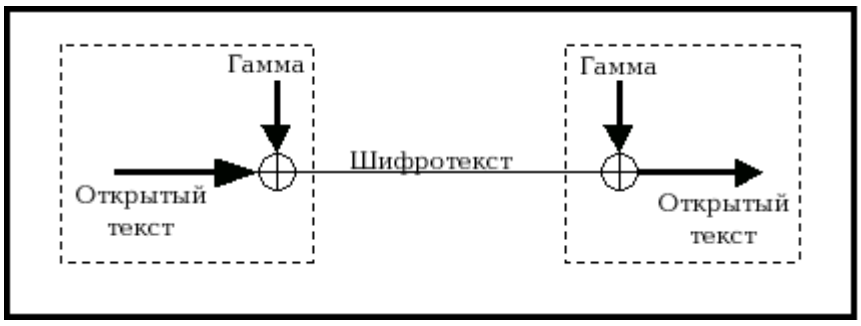
Толассыз мәліметтерді шифрлау бірнеше кілт кезегінің қосындысы негізінде (*гаммалар*), ашық мәтіндік хабарымен жүзеге асырылады. Шифрленген теңдеуі мынандай болады:

$$c_i = m_i \square\square k_i \text{ для } i=1,2,3...$$

мұнда $c_{i\ шифрмәтіннін\ белгісі,}$ m_i –ашық мәтіннің белгісі, , k_i –кілт кезегінің белгісі. Шешіп алу түрі мынадай болады:

$$m_i = c_i \square k_i \text{ для } i=1,2,3...$$

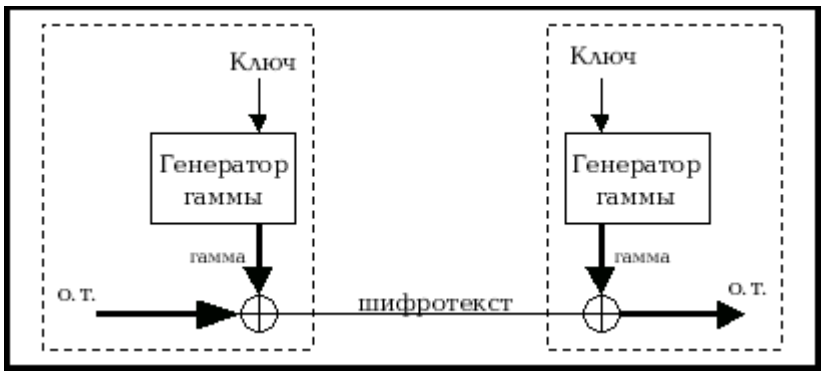
Белгі түрінде бөлек биттер, сонымен қатар символдарда (байттар) болуы мүмкін. Сондықтан толласыз шифрлер арқылы дауысттарды, видео және т.б. дыбыстарды шифрлауға болады.



Шығыс гамма хабар үшін кілттік ағын болып саналады. Толласыз шифрлер келесі түрде жетіледі:

- *синхрондық*;
- *самосинхрондалған (асинхрондық)*.

Синхрондық толласыз шифрлер –Шифрленген мәтінге байланысты емескілттік ағын (шығыс гамма). Мұндай жағдайда біздің иллюстрация былай өзгереді:



Құпия кілт негізінде шифр гамманы істеп шығарады. Гамманы істеп шығаратын блогты гамма генераторы деп немесе кездесоқ генераторы *(гаммалар)* - PRG*(Pseudo Random Generator)*деп атайды.

Синхрондық толласыз шифрлер келесі мүмкіншіліктері бар:

- *Синхронизация бойынша талабы*. Синхрондық толласыз шифрлеуді қолданған кезде қабылдаушы мен жіберуші синхрондалған болу керек, яғни кілт ағымның манызы бірдей өндеу керекЕгер синхронизация бұзылса, мысалы жіберілген кезде бір белгі жоғалса, онда шифрдан шешіп алу процессі нәтеже бермейді.
- *Қатесіздіктін көбеюінің жоқтылығы*. Шифрмәтіннің белгісінің өзгеруі шешіп алу кезінде қате болып саналмайды.
- *Шабуыл қасиетінің белсенділігі*. Шифрмәтінге белгілі бір немесе бірнеше символдарды енгізу немесе жоюы синхронизацияның бұзылуына әкеліп соғады, сондықтанмұны алдын ала лу үшін қосымша механизмдер қажет.

Осындай шифрлерге бір шабылуды қарастырайық..

$O_1O_2O_3$... – Ашық мәтіннің белгісі.

$K_1K_2K_3$... – кілт кезегінің белгісі.

$C_1C_2C_3$... –Келесі түрде алынған шифрмәтіннің белгісі:

$$\begin{array}{l} O_1\ O_2\ O_3\ O_4\ \ldots \\ \underline{K_1\ K_2\ K_3\ K_4\ \ldots} \\ C_1\ C_2\ C_3\ C_4\ \ldots \end{array} \quad (\oplus)$$

Қайта шифрлеу кезінде осы кілтте бір белгі енгізілсе O':

$$\begin{array}{l} O_1\ O'\ O_2\ O_3\ \ldots \\ \underline{K_1\ K_2\ K_3\ K_4\ \ldots} \\ C_1\ C'_2\ C'_3\ C'_4\ \ldots \end{array} \quad (\oplus)$$

Жау криптоаналитигі екі кезектіде жаулап алды $C_1C_2C_3C_4$ и $C_1C'_2C'_3C'_4$. уравнения құрастырайық: $K_2 = C'_2 \oplus O'$; $O_2 = C_2 \oplus K_2$; $K_3 = C'_3 \oplus O_2 \oplus O_3 = C_3 \oplus K_3$ және т.б., және бір белгінің O' мәнін тандап ол осы белгіден соң хабарды оқи алады Зерттеу кезінде онда кілт кезегінің фрагменті болады, сондықтан ол хабарды толығымен оқи алады . Сондықтан мынадай тұжырым жасауға болады, бір кілтті екірет қолдануға болмайды.

Хотя описание атаки носит гипотетический характер, тем не менее она очень и очень реальна, Ведь в качестве вставленного знака с таким же успехом может выступать и последовательность знаков. В этом случае подбираться будет последний знак вставленной последовательности. Представим себе ситуацию шифрования 2 файлов одним ключом, причем файлы имеют одинаковый заголовок и, скажем, середину (довольно реальная ситуация!!!). Проведя описанную атаку, криптоаналитик получит либо полностью исходные файлы, либо их фрагменты, что может иметь непоправимые последствия.

Егерде 2 түрлі мәтінді бір кілтпен шифрлесе, жау шифрмәтіннің белгісінің суммасын есептейді $C_i^1 \oplus C_i^2 = O_i^1 \oplus K_i \oplus K_i \oplus O_i^2 = O_i^1 \oplus O_i^2$, мұнда C_i^1- i-ый бірінші шифрмәтіннің белгісі, C_i^2 - i-ый екінші шифрмәтіннің белгісі, O_i^1 и O_i^2 – ашық мәтіннің белгісі. Сондықтан жау хабарды да шешіп алалады.

Самосинхрондалған толласыз шифрлер – кілт потогінін белгісі шифрмәтіннің белгісі ретінде жазып алынған саннмен анықталады.

Схемамен ол былай көрінеді:

Өз бетінше синхрондалған толласыз шифрлердін мүмкіншіліктері:

- Өз бетінше синхрондау. Өз бетінше синхрондау шифрмәтінде кейбір белгілер жойылғанда немесе енгізлгенде пайда болады.
- Қателердін санаулы көбеюы.
- *Шабуыл қасиетінің белсенділігі*.

- *Ашық мәтінді статистикасың себу*. Ашық мәтіннін статистикалық манызы шифрмәтінде сақталмайды, өйткені ашық мәтіннің әр белгісі келесі шифрмәтінге әсер етеді.

Бақылау сұрақтары:

1. Ақпараттар категорияларын атап беріңіз.
2. Мемлекттік органдарда, ұйымдарда, тәуелсіздік формасына қарамастан, азаматтар мен қызметкер адамдарға заңға байланысты қандай шараларды міндетті түрде қолдамауға қажет:
3. Халықтын қауыпсыздығына күмән келтіретін әрекеттерді қолдануға тиым салынады, соған мысалы келтіріңіз.
4. Симмертиялық алгоритмдер дегеніміз не?
5. Симметриялық алгоритмнің бөлімдерін атап беріңіз?
6. Олардың атқаратың қызметтері?.
7. Толласыз мәліметтерді шифрлеу дегеніміз не?.
8. криптография алгоритмі дегеніміз не?
9. шифрлау алгоритмі не үшін қажет?
10. шифрлау алгоритммен қандай әрекеттер жасауға болады?

Пайдаланылатын әдебиет: [2], 179 бет.

Дәріс №3. Тақырыбы: Ақпаратты өңдеудің автоматтық жүйесінің (АӨАЖ) қауыпсыздігінің негізгі қауыпы. Ақпаратты қоғаудың криптографиялық принциптері

2.1 АӨАЖ (АСОИ) қауіпсіздігінің негізгі қауіптері

АӨАЖ қауіпсіздігінің негізгі үш түрлі қаупі бар:

1. Ақпараттық құпиялығының бұзылу қаупі;
2. Ақпараттық бүтінділігінің бұзылу қаупі;
3. Жүйенің жұмыс жасауының бұзылу қаупі.

Компьютерлік бұзылулардың неғұрлым кең тараған және әр алуан түрі болып санккционерленбеген рұқсаты табылады. Онда парольдерді бұзу, «Маскарат», мүмкіншілікті заңсыз қолдану сияқты бұзуларды атауға болады.

Парольдерді ұстау арнайы жасалған программалармен жүзеге асады. Жүйеге заңсыз пайдаланушының кіруі кезінде ұстаушы программа экранда қолданушының паролін және атын енгізуді иммитациялайды, ол тікелей ұстаушы программаны иесіне беріледі, бұдан кейін экранда қате жөнінде хабар шығады және басқару операциялық жүйеге қайтарылады.

Пайдаланушы парольді енгізу кезіндегі жіберілген қате деп шамалайды. Заңды пайдаланушының атын және паролін алған ұстаушы программа иесі, оны өзінің мақсаттарында қолдана алады.

«Маскарад» - бұл мүмкіншіліктері бар бір пайдаланушының басқа пайдаланушының атынан қандайда бір әрекетті орындау. «Маскараттың» мақсаты болып қандай да бір әрекетті басқа пайдаланушыға жазу немесе басқа

пайдаланушының мүмкіншіліктерін және қасиеттерін өзіне меншіктеу. «Маскарат» әсіресе банктік жүйедегі электрондық төлеуде қауіпті.

2.2 Мүмкіндікті заңсыз қолдану.

Әрбір пайдаланушы өзінің мүмкіндік жиынын алады: әдеттегі пайдаланушылар – минимальды, администраторлар – максималды.

Компьютерлік желілерде болатын қауіптерге ерекше тоқталу қажет, олардың компонентреі кеңістікте бөлінген және олардың арасындағы байланыс физикалық тұрғыдан байланыс сызығы көмегімен және программалық тұрғыдан хабарлау механизмі көмегімен жүзеге асырылады.

Қазкүнем желіге ену кезінде басып енудің белсенді және белсенді емес әдістерін қолдана алады.

Белсенді емес басып енуде (ақпаратты ұрлау) ол ақпараттың ағымына тимей ақпараттың өтуін ғана бақылайды, бірақ ол тағайындау пунктін, айырбастау жиілігін және т.б. анықтай алады, яғни берілген каналда график анализін орындайды (хабар ағымын).

Белсенді басып енуде бұзушы ақпаратты алмастыруға (жою, ұстау, хабардың берілу ретін өзгерту) ұмтылады.

Желілердің сипаты, олардың шабуылды жоюдың, ақпараттың бұзылу әрекетін, байланыс каналы бойынша программаның жүзеге асқанын көрсету болып келеді.

Қауіптердің кең тараған түрлері:

1. «Троянский конь» - құжатта жазылған әрекеттермен бірге, жүйенің қауіпсіздігінің бұзылуына әкелетін әрекеттерді орындайтын программа; әдетте бұл программа қандай да бір пайдалы функцияларды орындайды деп қабылданатын, жекелей олар маскеленетін, көбіне қандай да бір пайдалы утилиттермен.
2. Компьютерлік вирус – басқа программаларды бүлдіретін, есептеу процесінің модификациялауына және өзінен көбеюіне қабілетті программа.

Алғашында мұндай программалар, басқа компьютерлер желісінде ресурстарды, бөлінген есептеулерді орындау мүмкіншілігін алу негізінде , іздеу үшін жазылған. Бірақ ол зиян келтіретін программаға оңай айналды. Неғұрлым белгілісі – Червь Меррис, Си тіліндегі программа.

Көрсетілген зиян келтіретін программалардан қорғану үшін, келесі шараларды қолдану қажет:

- Орындалатын файлдарға енудің санкционерленбегенін болғызбау;
- Алынатын программалық құралдардың тестіленуі;
- Жүйелік облыстардың және орындалатын файлдардың бүтіндігін бақылау;
- Программаның орындалуының тұйық ортасын құру.

АӨАЖ қорғау жүйесі – бұл зиянның пайда болуын минимумға әкелу мақсатында АӨАЖ қауіптеріне қарсы әрекеттерге негізделген заңдылық, моральдық-этикалық нормалар, административтік-ұйымдастырушылық шаралар, физикалық және техникалық-программа құралдардың жиыны.

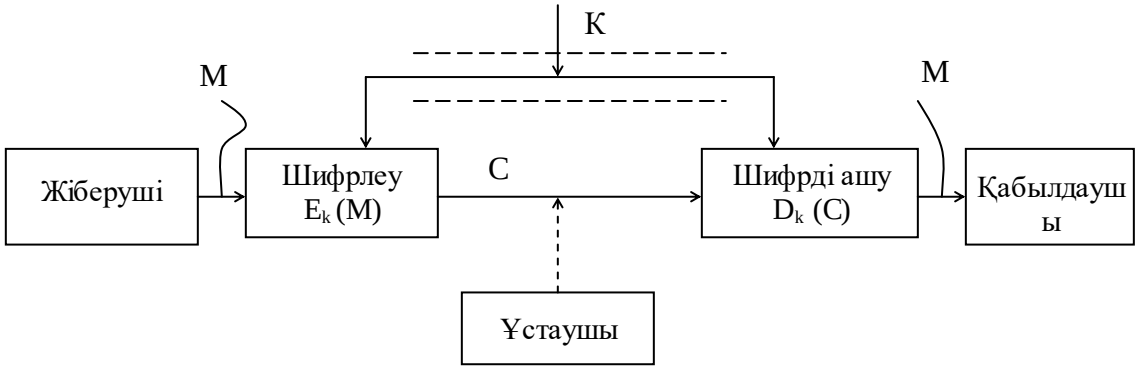
Қорғаудың соңғы құралдары негізінен ақпаратты қорғаудың криптографикалық әдістерімен жүзеге асады.

2.3 Ақпаратты қорғаудың криптографиялық принциптері.

Криптография –бұл мәліметтерді алмастыру әдістерінің жиынын көрсетеді, қарсы жақ үшін бұл мәліметтерді пайдасыз ету үшін бағытталған.(Криптография - cryptos - құпия, белгісіз logos – хабар байланыс)

Криптоанализ – (қорғаныс жүйесін бұзу) бұл кілтке ену рұқсатынсыз шифрленген хабардың бастапқы мәтінін ашу жөніндегі ғылым.

Криптожүйенің жалпыланған схемасын салайық.



Сур.1.1 Криптожүйенің жалпы схемасы

Жіберуші қорғалмаған канал бойынша заңды қабылдаушыға берілуі қажет бастапқы М хабарының ашық текстін генерациялайды. Каналды ұстап алушы берілетін хабарды ашу және ұстап алу мақсатында бақылайды. Ұстап алушы М хабарының мазмұнын біліп алмау үшін, жіберуші оны $E_к$ алмастыру көмегімен шифрлейді, және қабылдаушыға жіберілетін шифртекстті $C=E_к(M)$ алады.

Қабылдаушы С шифрді кері алмастыру көмегімен ашады $E_к^{-1}=D_k(C)$ және бастапқы хабарды алады.

$D_k(C)=E^{-1}_k(E_k(M))=M$

$E_к$ алмастыру криптоалгоритмі деп аталатын криптографиялық алмастыру түрлерінен таңдалады.

Жеке қолданылатын алмастыру таңдалатын параметр криптографиялық кілт К деп аталады.

Криптографиялық жүйе – бұл $E_к: \overline{M} \to \overline{C}$ түрге келтіретін ашық текстi $\overline{M}$ хабарды кеңістіктен $\overline{C}$ шифрленген текстті кеңістікке көшірілетін бір параметрлік (E_k) түрі.

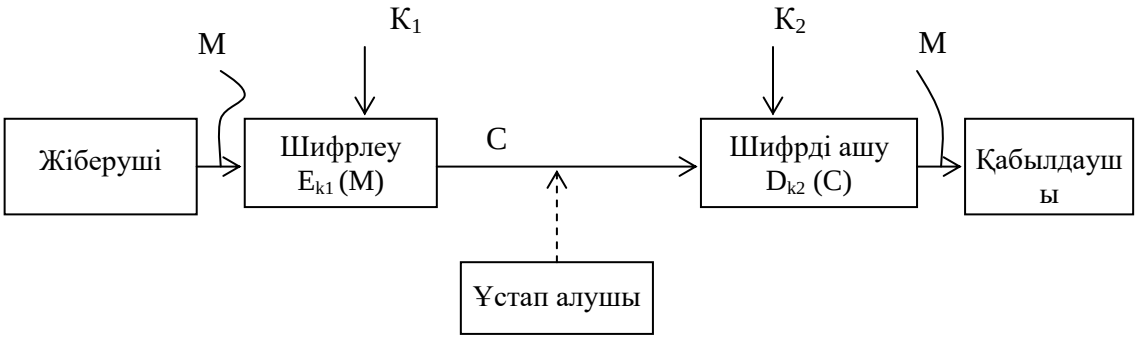
К(кілт) параметрі кілттер кеңістігі деп аталатын ақырлы К жиынынан таңдап алынады.

Криптожүйенің класын екіге бөлеміз:

1) симметриялық (біркілтті) криптожүйе;
2) асимметриялы (екікілтті) криптожүйе (ашық кілтпен).

1-ші кластың жүйелерінде шифрлеу және шифрді ашу кезінде бір ғана құпия кіл қолданылады.

Асимметриялық жүйеде шифрлеу және шифрді ашу үшін К1 және К2 әр түрлі екі кілтті қолданады.



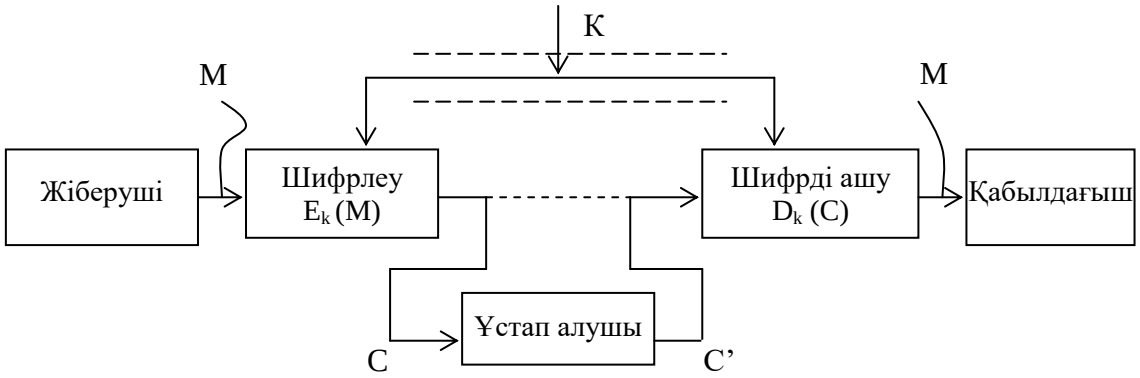
Сур.2.2 Асимметриялық криптожүйенің ашық кілтпен берілген жалпы схемасы

Асимметриялық криптожүйеде қорғалмаған канал бойынша тек қана ашық кілтті, ал құпия кілтті оның генерациясының орнында сақтайды.

Симметриялы криптожүйенің құрия кілтін жіберушіге және қабыладушыға беру керек.

Ұстап алушының кез-келген шифрленген мәтінді ашу С ашылған текстіалу үшін М немесе өзіңнің мәтініңді қайта шифрлеу М’ ал осыған сәйкес шифрленген мәтін алу үшін С’, және де жалған кілтсіз ұмтылыс криптоанаитикалық шабуыл деп аталады.

Егер криптоаналитикалық шабуыл, қойылған міндетті орындай алмаса және криптоаналитик жалған кілтсіз С дан М және М’ тан С’ шығара алмаса, онда мұндай криптожүйе криптотұрақты деп аталады.



Сур2.3 Криптожүйедегі ақпараттар ағымының активті ұстап алу хабары

Бақылау сұрақтары:

1. «Маскарад» деген не?
2. «Троянский конь» деген не?
3. Криптография мәліметтерді қорғаудың қандай мәселелерін шеше алады?
4. Шифрдың беріктілігі немен анықталуы керек?

Пайдаланылатын әдебиет: [2] – 12- 36 б.

Дәріс №4. Тақырыбы: Криптография негіздері мен түсініктері. Криптографияның дамуының қысқаша тарихы

Жоспар:
1. Шифрлау алгоритм негіздері
2. Цифрлық қол қою.
3. Криптографиялық хэш- функциялар
4. Кездесоқ сандардың криптографиялық генераторлары.
5.Шифрланған алгоритм.

Мақсаты: Криптография және криптоанализ мәселелерімен таныстыру
Кілттік сөздер: Криптография, Криптоанализ

Криптография екі бөліктен тұрады: криптография және криптоанализ.
Криптография – бұл заңсыз қолданушылардан қорғау үшін ақпараттарды өндейтін ғылым.
Криптоанализ – бұл шрифтарды шешу және тәжірибеде оны қолдану әдістері туралы ғылым.

Криптографиалық терминдарда шығыс жолдауы ашық мәтінмен аталады. Шифрлеу дегеніміз шығыс мәтінің мазмұнын басқалардан жасыру үшін өзгертуді айтады. Шифромәтін деп шифрленген хабарды айтады. Шифрмәтіннен ашық мәтінді шығару процесін шифрден алып шығу /дешифровка/ дейміз. Көбінесе шифрлау және шифрдан алып тастау процесінде кілт қолданылады, осы кілтті білген адам ғана шифрден мәтінді алып шығарады.

1. **Шифрлау алгоритм негіздері**

Криптоалгоритмге қарасты бірнеше классификация схемалары пайдаланады.сондықтан қолданылатын алгоритм бірнеше схемалардан «өтеді».
Барлық криптоалгоритм дардың классификациялар схемасының негіздері мыналар:

- **Құпиятізім. /Тайнопись**/Жіберуші және қабылдаушы хабарламаны өздері ғана өңдей алады. Басқа адамдарға шифрлау алгоритмі белгісіз.
- **Криптография кілтімен**. Барлық сыртқы адамдарға алгоритм белгілі, бірақ ол «кілтке» байланысты,ол кілт тек жіберушімен қабылдаушыда ғана бар.
- **Симметриялық криптоалгоритмдар**.Хабарламаны шифрлау және шифрдан алып шығару үшінкілт қолданады.
- **Асимметриялық криптоалгоритмдар.** Хабарламаны шифрлау үшін «ашық» кілтті қолданады, оны барлығына белгілі, ал шифрдан алып шығу үшін «жабық» кілтті қолданады, ол тек қабылдаушыда ған болады.

Кез келген криптоалгоритм кілтімен біз құпиятізімге айналдыруымызға болады. Ол үшін бағдарламадағы қолданып жүрген кодты өзгерту керек. Қайта ашу мүмкін емес.

Алгоритмдар деректерге байланысты бөлінеді:

- **Орын алмастыру/Перестановочные/.** Ақпараттар жиыны /байттар,биттар, және т.б./ өзгермейды, бірақ олардың қатар тәртібі өзгереді, сондықтан басқаларға ақпаратты қолдану мүмкін емес.
- **ауыстыруды тексеру /Подстановочные/.** Ақпараттар жиыны өздері өзгереді криптоалгоритм заңы боынша. Көбінесе бастапқы алгоритмдер осы топқа жатады.

  Жайдақтау /Важно/: Кез келген қайта құру криптографиялар ақпараттың көлемін кеңейтпейді, тек оның түсінігін өзгертеді. Сондықтан, егер шифрлау бағдарламасы шығыс файлдың көлемін кеңейтсе, онда оның негізінде оптимальды емес, дерекілік криптоалгоритм жатуы мүмкін.

Криптоалгоритмдар ақпарат жиынының көлеміне байланысты бөлінеді:

- **Үздіксіз шифрлер /Потоковые шифры/.** Кодтау өлшем бірлігі бір бит болады.Кодтау шешімі алғаш үздіксіз кіріске байланысты емес.Бұл схема ақпараттарды жіберу жүйесінде қолданады, яғни кейбір жағдайда ақпарттарды жіберу кезінде ықтиярлы уақытта басталуы және аяқталуы мүмкін не болмаса әлдеқалай тоқтап қалуы мүмкін. Көп қолданатын үздіксіз шифрлер – скремблерлар.
- **Блоктық шифрлер.** Кодтау өлшем бірлігі бірнеше байттар тізбегі /қазіргі кезде 4-32/ жатады. Кодтау шешімі осы блоктың барлық негізгі байттар түрінең тұрады.Бұл схема файлдарды кодтау және ақпараттар пакетің жіберу кезінде қолданады.

Шифрлеу/ Шифрдең шешу әдісті **шифр** (cipher) деп атайды. Кейбір шифрлеу алгоритмдер шифрлау әдісі құпия болғаның негіздейді. Бірақ қазіргі кезде оның тәжірибелік мағынасы жоқ. Барлық қазіргі уақытта алгоритмдерде шифрлеу және шифрден шешу үшін арнайы кілт қолданады. Сонымен, кілт қолданатын алгоритмдер екі топқа бөлінеді: симметриялық/алгоритмдар құпия кілтімен/ және асимметриялық. /алгоритмдер ашық кілтімен/. Айырмашылығы, симметриялық алгоритм шифрлау және шифрды шешу үшін бір кілтті қолданады, ал асиметриялық алгоритм әр түрлі кілттерді қолданады және шифрді шешу кілті мен шифрлеу кілті сәйкес келмейді.

2. Цифрлық қол қою.

Кейбір асимметриялық алгоритмдерді цифрлық қол қою генерациясында қолданады. **Цифрлық қол қою** деп кейбір құпия кілтті қолданатың генерлирленген мәліметтер тізімін айтады. Цифрлық қол қоюды хабар шынымен жіберуші адамнаң келгенің анықтау үшін пайдаланады.

Құжатта цифрлық қол қоюбылай құрылады: құжаттан дайджест (message digest) генерациялайды және оған қоса ақпарат қосылады, кім құжаттқа қол қойғаны, штамп уақыты, және т.б Шыққан қатар шифрленеді құпия кілтпен. Осы пайда болған шифрленіп алынған бит қол/подпись/ болып саналады.

3. Криптографиялық хэш- функциялар

Криптографиялық хэш- функцияларды көбінесе дайджест хабарлау генерациясында цифрлық қол құру кезінде қолданылады.

Хэш-функциялар хабарларды жазып алынған көлемді бірнеше хэш-мәндерге (hash value) бірдей қылып бөліп тастайды.

Криптографиялық хэш- функцияларды көбінесе мәндерін 128 еңдігімен және битпен өндіріледі.

4. Кездесоқ сандардың криптографиялық генераторлары.

Кездесоқ сандардың криптографиялық генераторлары кездесоқ сандардан құрайды, және олар криптографиялық қосымшаларда қолданады., мысалы-кілттерді генерациялау кезінде.

Жалған кездесоқтық сандардың криптографиялық генераторы көбінесе кездесоқ ақпаратты құрушы үлкен пул қолданады

Сонымен кездесоқ сандардың криптографиялық дәекті генераторың іске асыру аса қиын емес.

5.Шифрланған алгоритм.

Шешу жұмысы қиын болу үшін жақсы криптографиялық жүйелерді құрады. Оны іске асыру үшін көп күш жұмсау қажет емес, тек ұқыптылық пен базалық білім болу керек. Теориялық, барлық шифрланған алгоритм кілтпен қолдануы ашылады барлық кілт мәндерін тандап шығу әдісімен. Егер кілт добалда күш әдісімен тандалса компьютердің керек қуаты экспоненциалды өседі, кілттін ұзындығыда өседі. Кілт ұзыңдығы 32 бит болса ,онда 2" (109) қадам. Мұндай есеп кез келген дилентант шығарады және үйдегі компьютерде шығаруға болады. Жүйлер 40-биттық кілтімен 2^{40} қадам қажет . Жүйлер 56-биттық кілтімен кішкене қиыңырақ болады, ол үшін арнайы қымбат аппарат қажет 64 битті кілттерді қазіргі уақытта үлкен мемлекеттерде шеше алады.

Ақпараттану қазіргі таңда өмірдің негізгі мүмкіншілігі болып табылады. Жаңа ақпараттық технологиялар қазіргі кезде барлық жерде қолданылуда. Компьютер арқылы ғарыш кемелері мен самолеттерді басқарады, атом электростанцияларының жұмысын қадағалайды және де банк жүйелеріне қызмет етеді. Компьютер көптеген ақпараттарды өтеудің автоматизациялық жүйесінің негізі болып табылады және де ақпараттарды сақтау, өңдеу, оны қолданушыға таныстыру осы жаңа ақпараттық технологиялармен жүзеге асады.

Компьютерлік жүйе ақпараттарының қауіпсіздігі жайлы негізгі түсініктемелерді талдаймыз.

АӨАЖ (АСОИ) қауіпсіздігі дегенде біз оның джақсы бір процесске әдейі немесе кездейсоқ шатысуынан және де ұрлау әрекеттерінен, өзгерту немесе бұзу компоненттерінен қорғауын айтамыз.

Ақпаратқа «рұқсат» алу дегенде біз – ақпаратпен танысу, оны өңдеу, көшіру, модификациялау немесе жою деп түсінеміз.

Ақпаратқа рұқсат алу екіге бөлінеді. Ол санкционерленген және санкционерленбеген.

Ақпаратқа санкционерленген рұқсат алу – бұл ақпаратқа рұқсат алу, яғни рұқсаттың шектелген ережелер жиынтығын бұзбауы.

Ақпаратқа санкционерленбеген рұқсат алу – бұл керісінше рұқсаттың шектелген ережелер жиынтығының бұзылуы. Компьютерлік бұзылулардың неғұлым кең тараған және әр алуан түрі осы санкционерленбеген рұқсат болып табылады.

Конфеденциальды мәліметтер – бұл қауіпсіздіктің қандай дәрежеде керектілігін және берілген мәліметтер жайлы мәртебе. Субъект – бұл ақпараттардың объекттен субъектіге немесе жүйе құрылымының өзгеруіне себеп болатын жүйенің активті компоненті.

Объект – бұл ақпаратты сақтайтын, қабылдайтын немесе өткізетін жүйенің пассивті компоненті.

Объектіге рұқсат алу, яғни ақпарат құрылымына рұқсат алу дегенді білдіреді.

Ақпараттың біртұтастылығы – егер жүйедегі берілген мәліметтер құжаттағы берілген мәліметтермен симантикалық қатынасынан айырмашалағы болмаса және әдейі немес кездейсоқ бұзылудан қамтамасыздандырады.

АӨАЖ қауіпсіздігінің қаупі дегенде біз АӨАЖ мүмкін болған барлық зияндардан қауіпсіздігін түсінеміз. Қауіпсіздік зияны АӨАЖ ақпараттың қауіпсіздік жағдайының құрамы мен өңделуін пайымдайды.

Қауіпсіздің зияны АӨАЖ негізінен тығыз байланысқан.

АӨАЖ негізі – бұл жүйенің кейбір қасиеттерін сәтсіздікке алып келеді.

Компьютер жүйесіне шабуыл – бұл зиян келтірушілердің арқасында пайда болады. Шабуыл – бұл қаіпсіздік зиянының жүзеге асуы. Ақпаратты өңдеу жүйесінің қорғаныс мақсаты зиянына қарсылық тудырады.

Қорғалған немесе қауіпсіз жүйе- бұл қауіпсіздің зиянына қарсы тұра алатын тиімді қорғаныс жүйесі.

Қауіпсіздік политикасы – бұл нормалар, ережелер және практикалық кепілдемелер жиынтығы.

Бақылау сұрақтары:

1. Компьютерлік жүйелердің ақпараттық қауіпсіздігінің негізгі түсініктерін ата.
2. АСӨЖ қауіпсіздігінің негізгі қауіп типтерін ата.
3. Компьютерлік бұзулардың ең көп тараған түрі?
4. Шифрлау алгоритм негіздерің атап беріңіз.
5. Цифрлық қол қою дегеніміз не,оның атқаратың қызметі.
6. Криптографиялық хэш- функцияларнеше бөліктен тұрады?
7. Кездесоқ сандардың криптографиялық генераторлары атқаратың қызметі.
8. Шифрланған алгоритм дегеніміз не?, ол не үшін қажет?.

Пайдаланылатын әдебиет: [3]–12-15 б.

Дәріс №5. **Тақырыбы:** Құпия кілтті криптография. Классикалық симметриялы криптожүйелер

Жоспар:

1. Орын алмастыру шифрлер.
2. алмастыру шифрлері: моноалфавиттық, полиалфавиттық - Вижинер шифрі.
3. DES шифрлері.

Мақсаты: Орын алмастыру шифрлер ережелерімен таныстыру және жұмыста қолдану аясымен таныстыру

Классикалық шифрлер деп электрондық ақпараттық жүйенің алдында қолданған шифрлерді айтады. Құпия кілтті криптографияны(симметриялық криптоалгоритмдер) шифрлаудың классикалық әдісіне жатқызады.

1. Орын алмастыру шифрлер.

Ашық мәтіннің символдарын берілген әдіспен шифлеу кезінде кейбір ережелер қолданады.

1 мысал. Ашық мәтін: "ШИФРОВАНИЕ ПЕРЕСТАНОВКОИ". Кілт (орын алмастыру ережесі): тоб 8 әріптен тұрады, реттік нөмірін 1.2.....8 орын алмастыру ретіне 3-8-1-5-2-7-6-4 келтіреміз.

1	Ш	И	Т
2	И	Е	А
3	Ф	_	Н
4	Р	П	О
5	О	Е	В
6	В	Р	К
7	А	Е	О
8	Н	С	Й

Шифрмәтін : "ФНШОИАВР_СИЕЕЕРПННТВАОКО".
Өздігімен.Шифрмәтінді шығарып оқыныз
: «РПИКТАГЛООИМТРЫ» орын алмастыру әдісімен. Кілті: 2-4-3-1-5 (ашық мәтін – КРИПТОАЛГОИТМ).

2. алмастыру шифрлері

Моноалфавиттық. Орын алмастырудың екітүрін қарастырайық (алмастыру): моноалфавиттық, полиалфавиттық. Енді әрі қарай барлық мысалдарды орыс алфавит әріптерін кодтауда қолданамыз.

1 Кесте

Әріптер

А	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	П	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь

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

Моноалфавиттық алмастыру кезінде ашық мәтіннің әр бір алфавит әріпі шифрмәтіннің әр бір алфавит әрібіне сәйкес келеді.

2 мысал. Ашық мәтін: "ШИФРОВАНИЕ_ЗАМЕНОЙ".

2 кестеде қойылуы келтірілген.

2 Кесте

Негізгі мәтіннің алфавит А Б В Г Д ...

Шифрмәтіннің алфавиті _ Я Ю Э Ь

Шифрмәтін: "ИШМРТЮ_УШЫАЩ_ФЫУТЧ".

Полиалфавиттік шифр. Вижинер шифрі. Формуламен берілген шифрді:

$$yi=xi+ki(\mathrm{mod}\ n),$$ Вижинер шифрі дейміз.

Мүнда yi – алфавит символы, xi – ашық мәтіннің символы (алфавит әрібінің нөмірі), ki- і-ая кілттін әрібі, оның ішінде сөз немесе фраза қолданылады, n – қолданылатын алфавит ұзындығы.

3 мысал. Ашық мәтін: "ЗАМЕНА".

Кілті: "КЛЮЧ"

3 Кесте

З А М Е Н А

К Л Ю Ч К Л

y1=8+11(mod 33)=19 -> Т

y2=1+12(mod 33)=13 -> М

y3=13+31(mod 33)=11-> К

y4=6+24(mod 33)=30 -> Э

y5=14+11(mod 33)=25 -> Ш

y6=1+12(mod 33)=13 -> М.

Шифрмәтін: "Т М К Э Ш М".

Өздігінше. Шифрмәтінді шешіп алыңыз «Ю Ы Ц Ь А Р Ю». Кілті: «ОКНО» (Ашық мәтін – ПРИНТЕР)

Ақпараттың қорғану амалдарының (құралдары) көпшілігі шифрлау – кері шифрлау (шифрды ашу) процедураларына және криптографиялық шифрларды қолдануына негізделеді. ГОСТ 28147-89 стандартына сәйкес **шифр** деп криптографиялық түрлендіру алгоритмі мен кілтпен берілген қайтымды түрлендіру жиынтығы көптеген ашық мәліметтерді көптеген шифрланған мәліметтер ретінде түсінеміз.

Кілт – барлық мүмкін варианттардан бұл алгоритм үшін тек бір вариантты таңдауды қамтамасыз ететін, мәліметтер криптографиялық түрлендіру алгоритмнің кейбір параметрлерінің нақты құпия жағдайы.

Криптоберіктілік шифрдың негізгі сипаттамасы болып табылады. Ол криптоанализ әдістерімен ашатын беріктігін анықтайды. Әдетте бұл сипаттама кері шифрлау үшін керек уақыт аралықты анықтайды.

Ақпараттың криптографиялық қорғау үшін пайдаланған шифрларға бірнеше талаптар қойылады:

- жеткілікті криптоберіктілік (деректер жабудың беріктігі);
- шифрлау және кері шифрлау процедурларының жеңілдігі;
- шифрлаудың есебіне мол ақпараттың шамалығы;
- шифрлаудың кішкентай қателеріне сезімсіздігі және т.б.

Сол немесе басқа шамада бұл талаптарға жауап беретіндер:

- орын ауыстыру шифрлары;
- алфавит ауыстыру шифрлары;
- гаммалау шифрлары;
- шифрланатын деректердің аналитикалық түрлендіруіне негізделген шифрлар.

Орын ауыстыру арқылы шифрлау – шифрланатын мәтіннің символдары бұл мәтіннің кейбір блок ішінде нақты ереже бойынша қойылады. Блоктың жеткілікті ұзындығында ауыстыру орындалады және күрделі қайталанбайтын ауыстыру тәртібінде шифрдың тұрақты практикалық қолданулар үшін жетуге тиімді болады.

Алфавит ауыстыру арқылы шифрлау – мәтіннің шифрланатын символдары алдын ала келісіп алған алмастырылған сұлбамен сәйкес сол немесе басқа алфавит символдарымен алмастырылады.

Гаммалау арқылы шифрлау – шифрланатын мәтіннің символдары гамма шифры деп аталатын кейбір кездейсоқ бір ізділік символдарымен қосылады. Шифрлау тұрақтылығы негізінен шифр гамманың қайталанбайтын бөлігінің ұзындығымен (периодымен) анықталады. ЭВМ көмегімен шексіз гамма шифрын жасауға болатындықтан, ол автоматталған жүйелерде ақпаратты шифрлайтын негізгі әдістерінің біреуі болып табылады.

Аналитикалық түрлендіру бойынша шифрлау шифрланатын мәтін қайсібір аналитикалық ереже (формула) бойынша түрлендіріледі.

Мысалы, векторды матрицаға көбейту ережесін қолдануға болады. Көбейтілетін матрица шифрлау кілті болып табылады (сондықтан оның көлемі мен мазмұны құпия күйінше сақталу керек), ал көбейтілетін вектордың символдары шифрланатын мәтіннің символдары бір ізділі атқарады. Мысалдың басқа түріне ашық кілтті криптожүйелерді құратын бірбағытты функция кіреді.

Шифрлау және кері шифрлау процесі криптожүйелер шеңберінде орындалады. Симметриялық криптожүйенің маңызды сипаттамасы – хабарларды шифрлау немесе кері шифрлау негізінде сол бір құпия кілтті қолдану болып табылады.

Алфавит деп аталатын шектік символдар жиынына кіретін ашық мәтін де, шифрмәтін де әріптерден тұрады. Алфавиттер мысалдары барлық бас әріптердің шектік жиыны, барлық бас және кіші әріптері және цифрлар және т.с.с. шектік жиыны болып табылады.

Жалпы түрде $\sum$ қайсібір алфавитті былай көрсетуге болады:

$\sum=\{a_0, a_1, a_2, ..., a_{m-1}\}$.

Негізгі ереже бойынша $\sum$ алфавиттегі әріптерді біріктіргенде, жаңа алфавитті құруға болады:

- (a_0a_0, a_0,a_1, ..., $a_{m-1}a_{m-1}$) m^2 биграммалары бар $\sum^2$ алфавиті;

- ($a_0a_0a_0$, $a_0a_0a_1$, ..., $a_{m-1}a_{m-1}a_{m-1}$) m^3 триграммалары бар $\sum^3$ алфавиті.

Жалпы жағдайда, n әріптері бойынша біріктірсек, m^n n–граммалары бар $\sum^n$ алфавиті шығады.

Мысалы, ағылшын алфавиті $\sum$ = {ABCDEFGH...WXYZ} m=26 әріптер көлемі бар тіркестіру операциясы арқылы 26^2 = 676 (AA, AB, ..., XZ, ZZ) биграммалары бар алфавитті,

26^3 = 17576 (AAA, AAB, ..., ZZX, ZZZ) триграммалары бар алфавитті және тағы басқаларды жасауға мүмкіндік береді.

Криптографиялық түрлендіруді орындау кезінде алфавит әріптерін бүтін сандарға 0, 1, 2, 3, ... ауыстыруға пайдалы. Бұл керекті алгебралық манипуляциялардың орындалуын жеңілдетеді. Мысалы, қазақ алфавиті $\sum_{қаз.}$ = {А°БВГ±ДЕ...Þß} және бүтін жиындар $\bar{Z}_{42}$ = {0, 1, 2, 3, …, 41};
ағылшын алфавиті $\sum_{англ.}$ = {ABCDEF...YZ} және бүтін жиындар $\bar{Z}_{26}$ = {0, 1, 2, 3, …, 25} аралығында бірмәнді өзара сәйкестікті орнатуға болады.

Кейінірек m "әрпі" бар (сан түрінде) әдетте алфавитті пайдаланады $\bar{Z}_m$ = {0, 1, 2, 3, …, m-1}.

Дәстүрлі алфавиттің сандарымен алмастыру негізгі концепциялар мен криптограммалық түрлендіру әдістерін нақты ұйымдастыруға мүмкіндік береді. Көбінесе алфавиттің табиғи тілі қолданылады.

$\bar{Z}_m$ алфавитінен алынған n әріптері бар мәтінді n-грамма сияқты қарастыруға болады:

$$\bar{\bar{x}} = (x_0, x_1, x_2, \ldots, x_{n-1}),$$

мұнда $x_i \in \bar{Z}_m$, $0 \le i < n$, кейбір бүтін n=1, 2, 3, … .

$\bar{Z}_{m,n}$ арқылы $\bar{Z}_m$ жиынының әріптерінен құралған n-грамм жиынын белгілейміз.

E криптографикалық түрлендіруі

$$E = \{E^{(n)} : 1 \le n < \infty\},$$

$$E^{(n)} : \bar{Z}_{m,n} \to \bar{Z}_{m,n}$$

түрлендірулердің жиынтығын көрсетеді.

$E^{(n)}$ түрлендіруі қалай $\bar{\bar{x}} \in \bar{Z}_m$ ашық мәтіннің әрбір n-граммасына $\bar{\bar{y}}$ шифрмәтіннің n-граммасына алмастырылатынын анықтайды, яғни

$$\bar{\bar{y}} = E^{(n)}(\bar{\bar{x}}), \text{ және } \bar{\bar{x}}, \bar{\bar{y}} \in \bar{Z}_{m,n};$$

бұл кезде міндетті түрде бірмәндік $E^{(n)}$ түрлендіруінің $\bar{Z}_{m,n}$ жиынына бірдей талабы болып табылады.

Криптографиялық жүйе кілт деп аталатын К параметрімен белгіленген $\bar{\mathbf{E}} = \{E_k : K \in \bar{\bar{\mathbf{K}}}\}$ криптографиялық түрлендірудің жиыны сияқты түсіндірілуі мүмкін.

Кілттің мәнінің жиыны $\bar{\bar{\mathbf{K}}}$ кілттік кеңістігін құрайды.

3.2 Орын ауыстыру шифрлары

Шифрланатын мәтіннің символдарын орын ауыстырумен шифрлаған кезде бұл мәтіннің блогының шегінде анықталған ереже бойынша орындары ауыстырылады. Орын ауыстыру шифрлары ең қарапайым, сондай-ақ ең ежелгі шифрлар болып табылады.

Шифрлайтын кестелер

Шифрлайтын кестелерде кілт ретінде мыналар қолданылады:

- кестенің өлшемі;

- орын ауыстыруды беретін сөз немесе сөздер тіркесі;

- кестенің құрылымының ерекшеліктері.

Орын ауыстыру кестелік шифрларының ең үнемдісі кестенің өлшемі қызмет ететін жай орын ауыстыру болып табылады. Мысалы, КОМПЬЮТЕРЛІК ЖҮЙЕЛЕРДІ ҚОРҒАУ хабар кестеге баған бойынша кезектесіп жазылады. Кестенің 4 қатардан және 7 бағаннан тұратын толтыру нәтижесі 1-суретте көрсетілген.

Шифрмәтінді қалыптастыру үшін хабар мәтінін баған бойынша кестені толтырудан кейін қатар бойынша кестенің құрамын есептейді. Егер шифрмәтінді жеті әріп бойынша тобымен жазып отырса мынадай шифрланған хабар алынады:

КЬРЖЛІҒ ОЮЛҮЕҚА МТІЙРОУ ПЕКЕДР.

Шифрды ашу кезінде іс-әрекеттер кері ретпен орындалады.

К	Ь	Р	Ж	Л	І	±
О	Þ	Л	Ү	Е	Қ	А
М	Т	І	Й	Р	О	У
П	Е	К	Е	Д	Р	.

3.1-сурет. Кестенің 4 қатардан және 7 бағаннан тұратын толтырылуы

Кілт бойынша орын ауыстыру әдісі. Алдыңғы тәсілден бұл тәсіл кестенің бағандары кілттік сөз, сөздер тіркесі немесе кестенің қатарына теру ұзындығының саны бойынша орын ауыстырылады.

Мысалы, кілт ретінде ТЕХНИКА сөзін қолданайық, ал хабардың мәтінін алдыңғы мысалдан алайық. 3.2-суретте хабардың мәтінімен кілттік сөзбен толтырылған екі кесте көрсетілген, бұл кезде сол жақ кесте орнын ауыстыруға дейінгі толтыруға, ал оң жақ кесте – орнын ауыстырудан кейінгі толтыруға сәйкес.

Кілт ⟶

Т	Е	Х	Н	И	К	А
6	2	7	5	3	4	1
К	Ь	Р	Ж	Л	І	Ғ
О	Ю	Л	Ү	Е	Қ	А
М	Т	І	Й	Р	О	У
П	Е	К	Е	Д	Р	.

а) Орын ауыстыруға дейін

А	Е	И	К	Н	Т	Х
1	2	3	4	5	6	7
Ғ	Ь	Л	І	Ж	К	Р
А	Ю	Е	Қ	Ү	О	Л
У	Т	Р	О	Й	М	І
.	Е	Д	Р	Е	П	К

б) Орын ауыстырудан кейін

3.2-сурет. Хабардың мәтіні мен кілттік сөзбен толтырылған

Сол жақ кестенің жоғарғы қатарында кілт, ал кілттің әріптерінің астындағы нөмірлер алфавитте кілттің әріптерінің ретімен сәйкес анықталған. Егер кілтте бірдей әріптер кездессе, олар солдан оңға қарай нөмірленетін еді. Оң жақ кестенің бағандары кілттің әріптерінің реттелген нөмірімен сәйкес орындары ауыстырылған.

Оң жақ кестенің құрамындағы қатар бойынша және жеті әріп бойынша шифрмәтіннің тобының жазбасын есептеу кезінде шифрланған хабарды аламыз: ҒЬЛІЖКР АЮЕҚҮОЛ УТРОЙМІ .ЕДРЕПК

Қосымша жасыруды қамтамасыз ету үшін шифрланудан өткен хабарды қайта шифрлауға болады. Шифрлаудың мұндай тәсілі **екі рет орын ауыстыру** деп аталады. Бұл әдісте орын ауыстыру кестелері жеке баған үшін және жеке қатар үшін анықталады. Кестеге алдымен хабардың мәтіні жазылады, ал содан кейін кезекпен бағандар, сосын қатарлар ауыстырылады. Шифрды ашу кезінде ауыстырулар кері ретте жүргізіледі.

3.3-суретте екі рет орын ауыстыру әдісін іске асыр мысалы көрсетілген.

	2	3	1
3	А	Қ	П

	1	2	3
3	П	А	Қ

	1	2	3
1	А	А	Р

1	А	Р	А
5	Т	Т	Ы
2	Қ	О	Р
4	Ғ	А	У

Бастапқы кесте

1	А	А	Р
5	Ы	Т	Т
2	Р	Қ	О
4	У	Ғ	А

Бағандардың орнын ауыстыру

2	Р	Қ	О
3	П	А	Қ
4	У	Ғ	А
5	Ы	Т	Т

Қатарлардың орнын ауыстыру

3.3-сурет. Екі рет орын ауыстыру әдісінің мысалы

Бастапқы кестенің бағандарының нөмірлері мен қатарларының нөмірлерінің тізбегі қосарлы алмастыру шифрының кілтіне қызмет етеді. (Біздің мысалымызда 231 және 31524 тізбектері сәйкес).

Егер шифрмәтінді оң жақ кестеден 5 әріп бойынша блок қатарымен оқыса, онда келесі шығады: ААРРҚ ОПАҚУ ҒАЫТТ

Сиқырлы квадраттар

Сиқырлы квадрат деп әрбір бағаны, әрбір қатары және әрбір диагональдарының қосындысы бірдей сан беретін, оның клеткаларына бірден басталатын натурал сандардың тізбегі жазылған квадраттық кестені атайды.

Шифрланатын мәтін сиқырлы квадратқа оның клеткаларының нөмірленуіне сәйкес жазылады. Егер содан кейін қатар бойынша осындай кестенің құрамын жазып алса, онда бастапқы хабардың әріптерін орнын ауыстыру арқасында жинақталған шифрмәтін алынады.

3.4-суретте АҚПАРАТТЫ ҚОРҒАУ мәтінін сиқырлы квадраттың көмегімен шифрлау мысалы көрсетілген. Қатар бойынша оң жақ кестенің құрамын оқу кезінде алған шифрмәтіннің жұмбақты түрі бар: .ПҚҒ РҚОТ ЫАТР АУАА

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

.	П	Қ	Ғ
Р	Қ	О	Т
Ы	А	Т	Р
А	У	А	А

3.4-сурет. АҚПАРАТТЫ ҚОРҒАУ хабарымен толтырылған 4х4 сиқырлы квадраттың мысалы

3.3 Қарапайым ауыстыру шифрлары

Шифрланатын мәтіннің символдарын ауыстырумен шифрлаған кезде ауыстырудың алдын ала қойылған ережесіне сәйкес сол немесе басқа алфавиттің символдарымен ауыстырылады. ²арапайым ауыстыру шифрында бастапқы мәтіннің әрбір символы мәтіннің соңына дейін сол алфавиттің бірдей символдарымен ауыстырылады. ²арапайым ауыстыру шифрын көбінесе біралфавиттік ауыстыру шифрлары деп атайды.

Полибий квадраты

Полибий квадраты қарапайым ауыстырудың алғашқы шифрларының бірі болып есептеледі. Полибий шифрлау мақсатында грек алфавитінің әріптерімен кездейсоқ ретпен толтырылған, өлшемі 5х5 болып келетін квадраттық кестені жасапты.

Бұл полибий квадратта шифрлау кезінде ашық мәтіннің кезекті әрпін тауып, сол бағанда одан төмен орналасқан әріпті шифрмәтінге жазған. Егер мәтіннің әрпі кестенің төменгі қатарында болса, онда шифрмәтін үшін сол бағаннан ең жоғарғы әрпін алады.

Цезарь шифрлау жүйесі

Цезарь шифры қарапайым ауыстыру (біралфавиттік ауыстыруы) шифрының меншікті жағдайы болып табылады. Бастапқы мәтінді шифрлау кезінде әрбір әріп келесі ереже бойынша сол алфавиттің әрпіне ауыстырылады. Ауыстырылған әріп бастапқы әріптен К әріпке алфавит бойынша ығысқан жолмен анықталады. Алфавиттің соңына жеткен кезде оның басына циклдық өту орындалған. Цезарь К=3 ығысуы кезінде ауыстыру шифрын қолданған. Осындай ауыстыру шифрының құрамында ашық мәтіннің және шифрмәтіннің жұп әріптері сәйкес келетін ауыстыруы кестесімен беруге болады. К=3 үшін мүмкін болатын ауыстырудың жиынтығы 3.1-кестеде көрсетілген.

3.1 кесте

Біралфавиттік ауыстыру

A→D	J→M	S→V
B→E	K→N	T→W
C→F	L→O	U→X
D→G	M→P	V→Y
E→H	N→Q	W→Z
F→I	O→R	X→A
G→J	P→S	Y→B
H→K	Q→T	Z→C
I→L	R→U	

Мысалы, Цезарьдің жолдауы VENI VIDI VICI (қазақшаға аударғанда “Келді, Көрді, Жеңді” дегенді білдіреді), Митридаттың ұлы понтийлік патша Фарнакты жеңгеннен кейін өзінің досы Аминтийге жіберілген, шифрды ашқан кезде мына түрде болған:

YHQL YLGL YLFL

Біралфавиттік ауыстыруы жүйесіне қарсы криптоаналитикалық шабуыл, символдардың пайда болу жиілігін есептеумен басталады: шифрмәтінде әрбір

әріптің пайда болуы санмен анықталады. Содан әріптердің алынған бөлу жиіліктері шифрмәтінде бастапқы хабарламаның, мысалға ағылшынша, алфавиттегі әріптердің бөлу жиілігімен салыстырылады. Шифрмәтінде жоғарғы жиілікті пайда болған әріп ағылшын және т.б. тілдердегі жоғарғы жиілікпен пайда болған әріпке ауыстырылады. Шифрлау жүйесінің табысты ашылу мүмкіндігі шифрмәтіннің ұзындығының көбеһімен жоғарылайды.

Кілттік сөзі бар Цезарь жүйесі

Кілттік сөзі бар Цезарь жүйесі ауыстырудың біралфавиттік жүйесі болып табылады. Бұл жүйенің ерекшелігі - ауыстыру алфавитіндегі символдардың ығысқан және өзгертілген реті үшін кілттік сөз қолданылуы.

Кілттік сөз ретінде К санын, 0≤K<25 және сөз немесе қысқа сөздер тіркестігі таңдап алынады. Кілттік сөздің барлық әріптері әртүрлі болғаны жақсы. Мысалы, кілттік сөз ретінде DIPLOMAT сөзін және К=5 саны таңдалсын.

Кілттік сөз сандық коды таңдалған К санымен сәйкес келетін әріптен басталатын алфавиттің әріптерінің астына жазылады:

0	1	2	3	4	5				10						15					20						25
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
					D	I	P	L	O	M	A	T															

Ауыстыруы алфавитінің қалған әріптері алфавиттік ретпен кілттік сөзден кейін жазылады:

					5																					
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
V	W	X	Y	Z	D	I	P	L	O	M	A	T	B	C	E	F	G	H	J	K	N	Q	R	S	U	

Енді бізде кез келген хабарламаның әрбір әрпі үшін ауыстыруы бар.

Бастапқы хабарлама SEND MORE MONEY

былай шифрланады HZBY TCGZ TCBZS

Кілттік сөздің барлық әріптері әр түрлі болуы тиісті деген талаптың міндетті емес екенін ескеру керек. Кілттік сөзді (немесе сөздер тіркестігін) жай ғана бірдей әріптерді қайталамай жазу керек. Мысалы, кілттік сөйлем АУЫЛДЫҢ ЖАНЫ ТЕРЕҢ САЙ және К=3 саны ауыстырудан келесі кестесі туындайды:

	0			3										
1	А	Ә	Б	В	Г	Ғ	Д	Е	Ж	З	И	Й	К	Қ
2	Э	Ю	Я	А	У	Ы	Л	Д	Ң	Ж	Н	Т	Е	Р
1	Л	М	Н	Ң	О	Ө	П	Р	С	Т	У	Ұ	Ү	Ф
2	С	Й	Ә	Б	В	Г	Ғ	З	И	К	Қ	М	О	Ө
1	Х	Һ	Ц	Ч	Ш	Щ	Ь	Ы	І	Ъ	Э	Ю	Я	
2	П	Ұ	Ү	Ф	Х	Һ	Ц	Ч	Ш	Щ	Ь	І	Ъ	

Кілттік сөзі бар Цезарь жүйесінің жетістігі мүмкін болатын кілттік сөздердің саны тәжірибе түрінде өшірілмейтін болып табылады. Бұл жүйенің кемшілігі пайда болатын әріптердің жиіліктерін талдау негізіндегі шифрмәтіннің бұзылу мүмкіндігі болып табылады.

Трисемустың шифрлайтын кестесі

Осындай ауыстыруы шифрын алу үшін әдетте алфавиттің әріптері мен кілттік сөз (немесе сөздер тіркестігі) жазбасына арналған кесте қолданылған. Кестеге алдымен кілттік сөзі жазылып, қайталанатын әріптері алынып тасталады. Содан бұл кесте алфавиттің кілтке кірмей қалған әріптермен реттелген түрде толықтырылады. Кілттік сөз немесе сөздер тіркестігі жадыда оңай сақталатындықтан, мұндай жағдай шифрлау немесе шифрды ашу процестерін жеңілдеткен.

Бұл шифрлау тәсілін мысалда анықтайық. Қазақ алфавиті үшін шифрлайтын кестенің өлшемі 6х7 болады. Кілт ретінде АЛГОРИТМ сөзін алайық. Осындай кілтпен шифрлайтын кесте 3.6-суретте көрсетілген.

А	Л	Г	О	Р	И	Т
М	Ә	Б	В	Ғ	Д	Е
Ж	З	Й	К	Қ	Н	Ң
Ө	П	С	У	Ұ	Ү	Ф
Х	Һ	Ц	Ч	Ш	Щ	Ъ
Ы	І	Ь	Э	Ю	Я	

3.6-сурет. АЛГОРИТМ кілттік сөзімен шифрлайтын кесте

Шифрлау кезінде Полибий квадратындағы сияқты осы кестеден ашық мәтіннің кезекті әрпін тауып, одан төменгі бағанда орналасқан әріпті шифрмәтінге жазады. Егер бастапқы мәтіннің әрпі кестенің төменгі қатарында болса, онда шифрмәтін үшін сол бағандағы ең жоғарғы әрпін алады.

МЫСАЛЫ, ОСЫ КЕСТЕНІҢ КӨМЕГІМЕН АҚПАРАТТЫ ҚОРҒАУ ХАБАРДЫ ШИФРЛАҒАН КЕЗДЕ МҰҺМҒМЕЕАТҰВҒҚМЧ ШИФРМӘТІНДІ АЛАМЫЗ.

Мұндай кестелік шифрларда шифрлау бір әріп бойынша орындалатындықтан моногрaммды деп аталады. Трисемус шифрлайтын кестелердің екі әріптері бойынша шифрлауға болатынын байқаған. Мұндай шифрлар ***биграммалық*** деп аталады.

Плейфердің биграммалық шифры

Плейфер жүйесінің шифрлайтын кестесінің құрылымы Трисемустың шифрлайтын кестесінің құрылымына ұқсас болады. Сондықтан Плейфер жүйесінде шифрлау және шифрды ашу процедураларын түсіну үшін өткен тараудан (3.6-суретті қара) Трисемустың шифрлайтын кестесін қолданамыз.

Шифрлау процедурасы келесі қадамдардан тұрады:

1. Бастапқы хабарламаның ашықмәтіні әріптер жұбына (биграммаларға) бөлінеді. Мәтінде әріптердің саны жұп болу керек және онда құрамында екі бірдей әріп, биграммалар, болмау керек. Егер бұл талаптар орындалмаса, онда мәтін мәні жоқ орфографиялық кестелердің арқасында түрлендіріледі.

2. Ашық мәтіннің биграммалар тізбегі шифрлайтын кестенің көмегімен келесі ережелер бойынша түрленеді:

а) егер ашық мәтіннің биграммасының екі әрпі де бір қатарға немесе бағанға (мысалы, 6 суреттегі кестедей М және П әріптері сияқты) түспесе, онда берілген әріптердің жұбымен анықталатын тікбұрыштың бұрышындағы әріптер ізделінеді. (Біздің мысалда бұл МП°′ әріптері. МП әріптер жұбы °′ жұбына бейнеленеді. Шифрмәтіндегі биграммаларда әріптердің тізбегі ашық мәтіннің биграммасындағы әріптер тізбегінің қатынасы бойынша айнадай орналасу керек);

б) егер ашық мәтіннің биграммасының екі әріптері де кестенің бір бағанында орналасса, онда шифрмәтіннің әріптері болып оның астында жатқан әріптер есептелінеді. (Мысалы, КО биграммасы УВ шифрмәтіннің биграммасын береді). Егер ашық мәтіннің әрпі төменгі қатарда орналасса, онда шифрмәтін үшін сол бағанның жоғарғы қатарындағы сәйкес келетін әріп алынады;

в) егер ашық мәтіннің биграммасының екі әрпі де кестенің бір қатарына орналасса, онда шифрмәтіннің әріптері болып олардың оң жағында жатқан әріптер есептелінеді. (Мысалы, ЬЮ биграммасы °ß шифрмәтіннің биграммасын береді). Егер ашық мәтіннің әрпі соңғы оң жақ бағанда орналасса, онда шифр үшін сол қатардағы сол жақ бағаннан сәйкес келетін әріпті алады.

КОМПЬЮТЕРЛЕР сөзін шифрлайық. Бұл мәтіннің биграммаларға бөлуі мынаны береді: КО М П ЬЮ ТЕ РЛ ЕР.

Ашық мәтіннің берілген биграммалар тізбегі келесі шифр мәтіннің биграммалар тізбегіне шифрлайтын кестенің (6 суретті қара) көмегімен түрленеді: УВ ӘӨ ЭЯ ЕҢ ИГ ҒТ.

Шифрды ашу кезінде әрекеттердің кері реті қолданылады.

3.4 Күрделі ауыстыру шифрлары

Күрделі ауыстырудың шифрларын көпалфавитті деп атайды. r-алфавитті ауыстыру кезде негізгі хабардың x_0 символы B_0 алфавитіндегі y_0 символымен, x_1 символы B_1 алфавитіндегі y_1 символымен ауыстырылады, тағы да сол сияқты, x_{r-1} символы B_{r-1} алфавитіндегі y_{r-1} символымен, x_r символы B_0 алфавитіндегі y_r символымен ауыстырылады.

r=4 жағдай үшін көпалфавитті ауыстырудың жалпы сұлбасы 3.7-суретте келтірілген.

Енгізу символы	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
Ауыстыру алфавиті	B_0	B_1	B_2	B_3	B_0	B_1	B_2	B_3	B_0	B_1

3.7-сурет. r=4 жағдай үшін көпалфавитті ауыстырудың жалпы сұлбасы

Көпалфавитті ауыстырудың тиімді қолдануы негізгі тілдің шынайы қалқалау статистикасын қамтамасыз етуінде. Өйткені нақты символ А негізгі алфавитімен B_i бірнеше әртүрлі алфавиттік шифрлық символдарына түрлендіріледі. Қолданылатын қорғаныс жасаудың дәрежесі B_i алфавитті тізбегіндегі r периодының ұзындығына теориялық пропорционалды болады.

Гронсфельд шифры

Гронсфельд шифры деп аталатын бұл күрделі ауыстыру шифры Цезарь шифрының сандық модификациясын білдіреді. Ол үшін негізгі хабардың әріптерінің астына сандық кілттің цифрларын жазады. Егер кілт хабардан қысқа (аз) болса, онда кілттің цифрларын циклді түрде қайталайды.

Мысалы, кілт ретінде 2718 санын қолдана отырып ҰЛЫ ЖІБЕК ЖОЛЫ хабары үшін келесі шифрмәтін аламыз:

Хабар	Ұ	Л	Ы	Ж	І	Б	Е	К	Ж	О	Л	Ы
Кілт	2	7	1	8	2	7	1	8	2	7	1	8
Шифрмәтін	Ф	Р	І	Н	Э	З	Ж	П	И	Ұ	М	Б

Хабардың бірінші Ұ әрпін шифрлау үшін 2 кілттің бірінші цифрін қолдана отырып Ұ-дан басталатын қатар бойынша екінші әріпті, яғни алфавиттегі $Ұ\text{-}Ү_1\text{-}Ф_2$ есептеу керек. Шифрмәтіннің Ф деген бірінші әрпі алынады.

Ары қарай шифрлау осылайша жалғаса береді.

Вижинер шифрлау жүйесі

Вижинер жүйесі Цезарь шифрлау жүйесіне ұқсайды**.**

Бұл көпалфавитті ауыстыру шифрын шифрлау кестесімен жазуға болады. Бұл шифрлау кестесі Вижинер кестесі деп аталады. 4-кестеде қазақ тіліне арналған Вижинер кестесі көрсетілген.

Вижинер кестесі шифрлау және шифрды ашу үшін қолданылады. Кестенің екі кірісі бар:

- негізгі ашық мәтіннің әрпін анықтайтын жоғарғы қатардың сызылған символдары;
- кілттің шеткі сол бағанасы.

Шифрлау процесі кезінде кестенің жоғарғы қатарында негізгі мәтіннің кезекті әрпін және сол бағанада кезекті кілттің мәнін табады. Осы екі әрпін байланыстыратын сызықтардың қиылысқан жерінде шифрмәтіннің әрпі алынады. Вижинер кестесі көмегімен алынған шифрмәтіннің мысалын қарастырайық. РЕСПУБЛИКА кілттік сөз таңдап алынсын. КОМПЬЮТЕРЛІК ЖҮЙЕЛЕРДІ ҚОРҒАУ деген хабарды шифрлау керек.

Негізгі хабарды қатарға көшіреміз және оның астына қайта қайталанатын кілттік сөзді жазамыз. Үшінші қатарға Вижинер кестесінен анықталған шифрмәтін әріптерін көшіреміз.

Хабар	КОМПЬЮТЕРЛІК	ЖҮЙЕЛЕРДІ	ҚОРҒАУ
Кілт	РЕСПУБЛИКАРЕ	СПУБЛИКАР	ЕСПУБЛ
Шифрмәтін	ЩҰЬЯПАЬҢЩЛНӨ	ҢҒЫЗХҢШДН	ПЯАРБЭ

3.4-кесте

Қазақ әліпбиіне арналған Вижинер кестесі

а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я
ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а
б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә
в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б

г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в
ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г
д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ
е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д
ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е
з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж
и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з
й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и
к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й
қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к
л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ
м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л
н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м
ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н
о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң
ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о
п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө
р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п
с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р
т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с
у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т
ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у
ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ
ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү
х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф
һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х
ц	ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ
ч	ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц
ш	щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч
щ	ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш
ъ	ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ
ы	і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ
і	ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы
ь	э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і
э	ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь
ю	я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э
я	а	ә	б	в	г	ғ	д	е	ж	з	и	й	к	қ	л	м	н	ң	о	ө	п	р	с	т	у	ұ	ү	ф	х	һ	ц	ч	ш	щ	ъ	ы	і	ь	э	ю

Уитстонның “қос квадрат” шифры

1894 жылы ×арльз Уитстон ағылшыны “қос квадрат” деп аталатын биграммалармен шифрланатын жаңа әдіс тапты. “Қос квадрат” шифры Плейфейр шифрындағы сияқты биграммалармен шифрлау жүргізілетін екі кестені бірден қолданады.

Осы шифрмен шифрлауға мысал келтірейік. Қазақ алфавитінің кездейсоқ орналасқан символдары бар екі кесте берілсін (3.8-сурет).

Шифрлау алдында негізгі хабарды биграммаларға бөледі. Әрбір биграмма бөлек шифрланады. Биграмманың бірінші әрпі сол жақтағы кестеден, ал екінші әрпі – оң жақ кестеден алынады. Содан кейін қарсы төбелерінде жатқан биграмма әріптері бар тіктөртбұрышты ойдағыдай құрастырады.

Бұл тіктөртбұрыштың басқа екі төбесі шифрмәтіннің биграмма әріптерін береді.

1	2	3	4	5		1	2	3	4	5
Р	А	М	,	Һ	1	Л	.	Н	Ы	П
Ү	Ш	Ж	П	Қ	2	Ә	М	У	Б	Ұ
З	Й	.	У	Н	3	С	К	З	:	Ф
І	С	Э	Ұ	Ы	4	Ч	Ш	А	Һ	О
Ң	Л	Ө	Х	Ч	5	Х	,	Ү	И	Ң
В	Ь	К	Я	:	6	Ъ	Щ	Д	Ц	Э
Ә	Ф	О	И	Б	7	Ғ	Й	І	Т	Ж
Щ	_	Д	Ю	Е	8	_	Р	Ь	Ө	Я
Т	Г	Ц	Ъ	Ғ	9	Қ	Ю	Е	Г	В

3.8-сурет. “Қос квадрат” шифрына арналған қазақ алфавиттің кездейсоқ орналасқан символдары бар екі кесте

БҮ негізгі мәтіннің биграммасы шифрлансын дейік. Б әрпі сол кестенің 5-ші бағаны мен 7-ші қатарында орналасқан. Ү әрпі оң кестенің 3-ші бағаны мен 5-ші қатарында орналасқан. Бұл тіктөртбұрыштың 5 және 3 қатарлары, сонымен бірге сол кестенің 7 және оң кестенің 5 бағаналары бойынша пайда болғанын білдіреді. Сондықтан, шифрмәтіннің биграммасына оң кестесіндегі 3-ші баған мен 7 қатарда орналасқан І әрпі және сол кестедегі 5 баған мен 5 қатарда орналасқан Ч әрпі кіреді. ІЧ шифрмәтін биграммасы алынады.

Егер хабар биграмманың екі әріпі де бір қатарда жатса, онда шифрмәтін әріптері де осы қатардан алынады. Хабар биграмманың екінші әрпіне сәйкес келетін шифрмәтін биграмманың бірінші әрпі сол кестедегі қатардан алынады.

Бастапқы хабар	БҮ	ГІ	Н_	ЖА	ҢБ	ЫР	ЛЫ	_Қ	ҮН
Шифрмәтін	ІЧ	ЕФ	СЕ	УЭ	ИУ	ШЕ	ИА	РЙ	УР

Бірреттік шифрлау жүйесі

Бірреттік шифрлау жүйесінің ерекшелігі кілттік тізбегінің бір рет қолданылуы болып табылады. Бұл шифрлау жүйесі $X=(x_0, x_1, \ldots, x_{n-1})$ бастапқы ашық мәтінді мынадай $Y_i= (X_i + K_i) \bmod m$, $0 \le i \le n-1$ Цезарь ауыстыруын қолдана отырып $Y=(y_0, y_1, \ldots, y_{n-1})$ шифрмәтініне түрлендіреді. Бұл жерде K_i – кездейсоқ кілттік тізбектің і-ші ӯлементі

Бірреттік жүйенің $\bar{\kappa}$ кілттік кеңістігі $\bar{Z}_m$-ден дискреттік кездейсоқ шамаларының жиынтығын және m^n мәндерінен тұратынын көрсетеді.

Шифрды ашу процедурасы мына теңдігімен жазылады

$Y_i = (X_i - K_i) \bmod m$.

Бірреттік жүйе 1917 жылы американдықтар Дж.Моборн және Г.Вернаммен ойлап тапқан. Бұл алмастыру жүйесін іске асыру үшін бірреттік блокнот пайдаланылады. Бұл блокнот жыртылмалы парақтардан құралған. Парақтың әрқайсында k_i кездейсоқ сандармен (кілттермен) кесте басылған. Блокноттың екі ýкземпляр орындалады: біреуі – жіберушімен, ал екіншісі – алушымен пайдаланылады. Хабардың x_i әрбір символы үшін кестеден k_i өз кілті тек бір рет қолданылады. Кесте қолданылып болған соң, ол блокноттан өшіріліп және жойылу тиіс. Жаңа хабарды шифрлау үшін жаңа беттен бастайды.

Бірреттік блокноттың кейбір вариантарында кілттік тізбектің жай басқаруына сүйенеді, бірақ бұл шифрдың беріктігінің азаþына алып келеді. Мысалы, хабарды жіберушімен алушыға белгілі кілт кітаптағы көрсетілген жерлермен анықталады. Кілттік тізбек сол кітаптың көрсетілген жерінен басталады және Вижинер жүйесіндегі сияқты қолданылады. Кейде мұндай шифрды жүгірікті кілті бар шифр деп атайды. Кілттік тізбекті басқару шифрдың мұндай вариантында жеңілірек, өйткені ұзын кілттік тізбек компактты пішінде көрсетілген. Бірақ басқа жағынан бұл кілттер кездейсоқ болмайды. Сондықтан криптоаналитикте табиғи негізгі тілдің әріптер жиілігі туралы ақпаратты пайдалану мүмкіндігі пайда болады.

Вернам әдісімен шифрлау

Вернам шифрлау жүйесі m=2 модулінің мәні ретінде Вижинер шифрлау жүйесінің жеке жағдайы болып табылады. Негізгі мәтін символдардың екілік көрсетімін қолданады.

Негізгі мәтіннің алты көмекші символдармен кеңейтілген (пробел, возврат каретки және т.с.с.) ағылшын алфавитіндегі {A,B,C,D...Z} әрбір символы алдымен Бодо телеграф кодасы бар 5-биттік блокқа (b_0, b_1,... b_4) кодаланған.

K_0, K_1, K_2... екілік кілттердің кездейсоқ тізбегі алдын-ала қағаз таспаға жазылады.

Алдын-ала *x* екілік символдар тізбегіне түрленген негізгі мәтінді шифрлау екілік кілттер тізбегі мен *x* символдары модуль 2 бойынша қосылады.

Шифрды ашу шифрмәтінінің сол *к* кілттер тізбегі мен *у* символдары модуль 2 бойынша қосылуынан анықталады:

$y \oplus к = x \oplus к \oplus к = x$

Шифрлау және шифрды ашуда пайдаланған кілттер тізбегі бір-бірінің орнын толтырады (модуль 2 бойынша қосқанда) және қорытындыда негізгі мәтіннің *x* символдары қалыпқа келтіріледі.

Бақылау сұрақтары:

1. Орын алмастыру шифрлер дегеніміз не?.
2. алмастыру шифрлері: моноалфавиттық дегеніміз не?
3. полиалфавиттық - Вижинер шифрі атқаратың қызметі.
4. DES шифрлері не үшін қажет?
5. орын алмастыру және алмасу шифлердің айырмашылығы недең?

6. Көпалфавитті алмастыруды қолданудың мақсаты?
7. Шифрлаудың қарапайым әдістерін ата.
8. Биграммалық шифрларды ата.

Пайдаланылатын әдебиет: [1] – 4-18 б.

Дәріс №6. Тақырыбы: Симметриялық алгоритімдер. Қазіргі симметриялы криптожүйелер

1. Қазіргі кездегі симметриялық шифрлеу алгоритмі.
2. Алгоритмдер түрлері.

Мақсаты: Симметриялық және асимметриялық шифрлеу алгоритммен таныстыру

DES (Data Encryption Standard). өте тиімді және қолайлы симметриялық шифрлеу алгаритмі. АҚШ Мемлекеттік Стандарттау және технология Институтында өнделген (NIST – National of Standards and Technology) 1977 жылы.

DES негізгі мүмкіншіліктері:

1. DES 64 битпен деректер блогін шифрлейді.
2. DES шифрлау кілті 64 бит көлемі бар, бірақ кілттін мағыналы бөлігі 56 биттен тұрады, әр сегізінші бит жұптылықты бақылау үшін қолданылады.
3. Шифрлеу алдында және алғашқы құру кезінде кілт – нәтижесінде кілттін көлемі 64 битке тең болады.
4. Алғашқы орын алмасу өндіріледі.
5. 16 раундынегізгі криптоөндеу жүзеге асырылады.
6. Осылардың соңында финалды орын алмасады, алғашқыға инверсальды.

DES келесі жұмыс режимі бар:

- электронды кодтық кітап ECB (Electronic Code Book);
- CBC шифр блогінің ұстасуы (Cipher Block Chaining)
- CFB шифромәтін бойынша қайта байланысы (Cipher Feed Back)
- OFB шығу бойынша қайта байланысы (Output Feed Back).

Алгоритмдер келесі минимальды талаптары бар:

- алгоритм мәліметтерді симметриялық шифрлау блогын орындау қажет;
- алгоритм минимум келесі шифрленген болк көлемін және шифрлау кілтінің ұзындығын сақтау қажет : 128/128, 128/192 және 128/256 бит.

Алгоритмдер түрлері.

Конкурсқа 15 алгоритм шифрлеуі жіберілді.Нәтежесінде 5 алгоритм озды, оның кемшіліктері болған жоқ.

Алгоритмдер:

MARS	IBM Corporation
RC6	RSA Laboratories
RIJNDAEL	Joan Daemen, Vincent Rijmen, Бельгия *
SERPENT	Ross Anderson Eli Biham, Lars
TWOFISH	Bruce Schneier.

Негізгі талаптарынаң басқа олар келесі критерияларға сай:

- жоғарға криптотабанды;
- әлсіз және эквиваленті кілттері жоқ ;
- түсінікті құрылымы бар;
- оперативті және энергобайланыссыз жадтарға көп талаптанбайды.
- Барлық платформаларда жоғарғы тез әрекеті бар – 8 ден бастап 64 битқа дейін;

Енді конкурстан өтпеген 10 алгоритмнің критерияларын қарастырайық .

Алгоритмдер

DEAL Шифрлау жылдамдығы өте төмен.

FROG алғашқы өндеу кілті және шифрлеу жылдамдығы төмен, алгоритм құрылысы қиын.

HPC барлық платформаларда шифрлеу аяндау Эквивалентті кілттердін саны көп

LOKI 97 Сызықтық және дифференциалдық криптоанализ әдістері жоғарғы табанды емес.

MAGENTA Тандап алған ашық мәтінді жаулап алуы мүмкін.

CAST – 256 Шифрлеу жылдамдығының үлкенсіздігіне байланысты энергобайланыс жадына жоғарға талабы.

CRYPTON Шифрлеу жылдамдығы төмен, бірақ RIJNDAEL алгоритміне ұқсайды.

TWOFISH

DFC барлық платформаларда шифрлеу жылдамдығы жоғарғы емес, 64- биттан басқа.

E2 оперативті және энергобайланыссыз жадтарға көп талаптанады.

SAFER+ негізгі сиппаттамасы бойынша SERPENT ұқсайды.

К.Шеннон пікірі бойыша шифрларларда екі жалпы ұстанымды пайдалану қажет: шашырау және араластыру.

Шышырау кезінде ашық мәтіндегі бір таңбаның шифрмәтіндегі көп таңбаға ықпалының таралуы болады. Ол ашық мәтіннің статистикалық қасиеттерін жасыруға мүмкіндік береді. Араластыру кезінде шифрлаушы түрлендірулер пайдаланылады. Олар ашық және шифрланған мәтіндердің

статистикалық қасиеттерінің өзара байланысын қалпына келтіруді қиындатады. Дегенмен шифр ашуды қиындатып қана қоймай (егер пайдаланушыға құпия кілт белгілі болған жағдайда) шифрлау мен оны қайта ашудың оңай болғанын да қамтамасыз етуі керек.

Шашырау мен араластыруды іске асыруға арналған тәсілдердің біріне құрама шифрды пайдалану тәсілі жатады. Бұл тәсілде пайдаланылатын шифр қарапайым шифрлер тізбегінен тұрады. Мұндағы әрбір шифр не шашырау, не араластыру арқылы нәтижеге өз үлесін қосады. Құрама шифрлерде қарапайым шифр есебінде, әдетте ауыстыру және қарапайым орын ауыстыру жиі пайдаланылады. Орын ауыстыру кезінде ашық мәтін символдарын тек араластырады және араластырудың нақты түрін құпия кілт анықтайды. Ауыстыру кезінде ашық мәтіннің әрбір символы, сол алфавиттегі басқа символмен ауыстырылады, ал алмастырып қоюдың нақты түрін, бұл жерде де құпия кілтпен анықтайды. Сонымен бірге мынаны еске ала кету керек: қазіргі кездегі блоктық шифрда ашық мәтін мен шифрмәтіннің блоктары әдетте ұзындықтары 64 бит болатын екілік тізбектерден тұрады. Әрбір блок 2^{64} мән қабылдауы мүмкін. Сондықтан ауыстыру құрамында $2^{64}=10^{19}$ символдары бар өте үлкен алфавитте орындалады.

5.1. DES стандарты

DES (Data Encryption Standart) стандартын 1977 жылы АҚШ-тың ұлттық стандарттар бюросы жарияланған [Романец]. DES алгоритмінің негізгі жағымды жақтары:

- ұзындығы 56 бит болатын бір ғана кілт пайдаланылады;
- DES стандартына сәйкес болатын программалардың бір дестесінің көмегімен хабарларды шифрланған соң, осы стандартқа сәйкес кез келген шифрды ашу программалар дестесін пайдалануға болады;
- алгоритмнің қарапайымдылығы өңдеудің жоғары шапшаңдығын қамтамасыз етеді;
- алгоритмнің жеткілікті түрдегі криптоберіктілігі.

DES алгоритмі орын ауыстыру мен ауыстырулар қисындасуын пайдаланады. DES 64-биттік кілт көмегімен 64-биттік деректер блогын шифрлауға мүмкіндік береді. Кілттегі 8 бит - жұптылықты бақылауға арналған тексеру биттері болып табылады. Шифрды ашу - шифрлауға кері операция болып табылады.

DES алгоритмін сипаттағанда мынадай шартты белгілер пайдаланылды:

- L және R - сол (left) және оң (right) биттер тізбектері;
- LR - L және R тізбектерінің конкатенациясы. LR биттер тізбегінің ұзындығы L және R ұзындықтарының қосындысына тең. LR биттер тізбегінде L тізбегінің битерінен соң R тізбегінің биттері жүреді;
- $\oplus$ - модуль 2 бойынша битті битке қосу операциясы.

Бастапқы мәтіні бар файлдан кезекті 64-биттік (8-байттық) T блогы оқылады. Бұл T блогы IP бастапқы *орын ауыстыру матрицасы* көмегімен түрлендіріледі (2-кесте).

Содан сон 16 қадамнан тұратын шифрлаудың итеративтік процесі орындалады.

$$T_i = L_i\ R_i\ ,$$

мұндағы T_i - і-ші итерациясының нәтижесі; $L_i = t_1 t_2 \ldots t_{32}$ (бастапқы 32 бит); $R_i = t_{33} t_{34} \ldots t_{64}$ (соңғы 32 бит).

Сонда і-ші итерациясының нәтижесі келесі формула арқылы жазылады:

$$L_i = R_{i-1}, \quad i=1, 2, \ldots, 16;$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i), \quad i=1, 2, \ldots, 16$$

f-функциясы шифрлау функциясы деп аталады. Оның аргументтері болып итерацияның алдыңғы қадамында алынған R_{i-1}-тізбегі мен 64-биттік К-шифрын түрлендіруден пайда болған 48-биттік K_i-кілті табылады. (f-шифлау функциясы мен K_i-кілт алгоритмі төменде келтірілген).

Итерацияның соңғы қадамында R_{16} және L_{16} тізбегі алынады (орын ауыстырусыз). Олар 64-биттік $R_{16}L_{16}$ тізбегіне конкатенцияланады.

Шифрлау аяқталған соң, кері ауыстыру IP^{-1} матрицасы (3-кесте) көмегімен биттер алғашқы орнына келтіріледі.

IP^{-1} матрицасы мен IP матрицасының бірінші қатарындағы элементтердің бір-біріне өзара қатынасы 4-кестеде келтірілген.

Кері шифрлау былайша жүргізіледі: кері шифрланатын деректер алдымен IP^{-1} матрицасы бойынша орын ауыстырылады, содан соң $R_{16}L_{16}$ биттер тізбегіне шифрлау процесінде болатын амалдардың кері түрі қолданылады.

Кері шифрлау процесі келесі формула түрінде жазылады:

$$R_{i-1} = L_i, \quad i=1, 2, \ldots, 16;$$

$$L_{i-1} = R_i \oplus f(L_i, K_i), \quad i=1, 2, \ldots, 16.$$

Сонымен, орын ауыстырылған $R_{16}L_{16}$ кірме блогы бар кері шифрлау процессі үшін 1-ші итерацияда - K_{16} кілті, 2-де - K_{15} кілті т.с.с. пайдаланылады. 16-шы итерацияда K_1-кілті пайдаланылады. Итерацияның ең соңғы қадамында L_0 және R_0 тізбектері алынады. Олар 64 биттік L_0R_0 - тізбегіне конкатенцияланады. Содан соң осы тізбектегі 64 бит IP матрицасына сәйкес орын ауыстырылады. Бұл түрлендіру нәтижесінде бастапқы биттер тізбегін алынады (кері шифрланғ 64 биттік мән).

2-кесте

IP бастапқы орын ауыстыру матрицасы

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

3-кесте

IP^{-1} кері ауыстыру матрицасы

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

63	55	47	39	31	23	15	7

33	1	41	9	49	17	57	25

4-кесте

Матрица элементтерінің байланысы

IP^{-1} матрица элементі	IP матрица элементі
40	01
8	02
48	03
16	04
56	05
…	…

$f(R_{i-1},K_i)$-функциясының мәндерін есептеу үшін мыналар пайдаланылады:

- E функциясы (32 биттен 48 кеңейуі)
- S_1, S_2, …, S_8 функциясы (6 биттік санның 4 биттік санына алмастыру);
- P функциясы (32 биттік тізбектегі биттердің орын ауыстыруы).

Осы функциялардың анықтамасын келтірейік. f - шифрлау функциясының аргументтері болып R_{i-1}(32 бит) және K_i (48 бит) табылады. E (R_{i-1}) функциясының нәтижесі - 48 биттік сан. 32 биттен 48 битке дейін кеңейтуды орындайтын (32 биттік блокты қабылдап, 48 биттік блокты тудырады). E кеңейту функциясы 5-кесте бойынша анықталады.

5-кесте

E кеңейту функциясы

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

5-кестеге сәйкес $E(R_{i-1})$ - алғашқы үш биті ол - 32, 1 және 2 биттер, ал соңғы - 31, 32, 1. Алынатын нәтиже (оны $E(R_{i-1})$ деп белгілейік) модуль 2 бойынша K_i кілтінің ағымдағы мәнімен қосылады (XOR операциясы) содан сон 6 биттік B_1, B_2, …, B_8 блоктарына бөлінеді:

$$E(R_{i-1}) \oplus K_i = B_1\, B_2 \ldots B_8$$

Ары қарай бұл блоктардың әрқайсысы, 4 биттік мәні бар S_1, S_2, …, S_8 - функция-матрицаларының элементтерінің нөмірі ретінде пайдаланылады (6-кесте).

6-кесте

S_1 ,S_2, …, S_8 түрлендіру Функциялары

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

S_j матрицасында элементті таңдау ерекше түрде өтеді. S_j-матрицасының кірісіне 6-биттік $B_j=b_1\,b_2\,b_3\,b_4\,b_5\,b_6$ блок кірсін дейік. Сонда 2-біттік b_1b_6 саны матрица жолының нөмірін, ал төртбіттік $b_2b_3b_4b_5$ саны - баған нөмірін көрсетеді. Мысалы, егер S_1 матрицасының кірісіне $B_1=b_1b_2b_3b_4b_5b_6=100110$ алтыбиттік блок түссе, онда $b_1b_6=10_{(2)}=2_{(10)}$ екібиттік сан S_1 матрицасының нөмірі 2 жолын, ал $b_2b_3b_4b_5=0011_{(2)}=3_{(10)}$ 4 биттік сан S_1 матрицасының нөмірі 3 бағанын көрсетеді. Бұл S_1 матрицасындағы $B_1=100110$ блогы нөмірі 2 жол мен нөмірі 3 баған қиылысқан жеріндегі элементті, яғни $8_{(10)}=1000_{(2)}$ таңдап

алады. Алты биттік B_1, B_2, …, B_8 - блоктар жиынтығы S_1 ,S_2, …, S_8 әр матрицасындағы 4 биттік элементті таңдап алуды қамтамасыз етеді.

Нәтижесінде $S_1(B_1)S_2(B_2)…S_8(B_8)$, яғни 32 биттік блок аламыз (S_j матрицасы құрамында 4 биттік элементтер бар). Бұл 32-биттік блок P биттердің орын алмастыру функциясы бойынша түрлендіріледі (7-кесте).

Сонымен, шифрлау функциясы: $f(R_{i-1}, K_i) = P(S_1(B_1), …, S_8(B_8))$ болады.

Әр итерация сайын K_i кілтінің (ұзындығы 48 бит) жаңа мәні пайдаланылады. K_i кілтінің жаңа мәні бастапқы К кілтінен есептеледі (15-сурет). К кілті есебінде 64 биттік блокты алуға болады. Оның 8 биті (8, 16, 24, 32, 40, 48, 56, 64 орындарында орналасқан) жұптық бақылауы болады. Бақылау биттерін алып тастап, кілті жұмысқа пайдалану үшін кілтті алдын ала дайындайтын G функциясы пайдаланылады (8-кесте).

7-кесте

Биттерді орын ауыстыру P-функциясы

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

8-кесте

Кілтті бастапқы дайындау G функциясы (1-ші орын ауыстырып іріктеме)

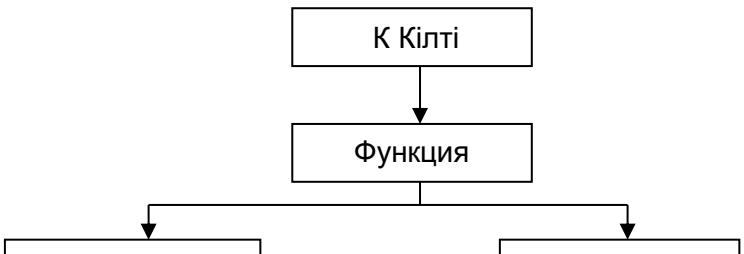
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

8-кесте екі бөлікке бөлінген. G(K) турлендіру нәтижесі әр қайсысы 28 биттік екі C_0 және D_0 бөлігіне бөлінген. G-матрицасының бірінші 4 жолы C_0 тізбегінің биттері қалай таңдалады - соны анықтайды (C_0 бірінші биті шифр кілтінің 57 биті, содан кейінгі 49 бит т.б.с, ал соңғы биттері 44 және 36 кілт биттері болады).

G матрицасының келесі 4 жолы D_0 тізбегінің биттері қалай таңдалатының анықтайды (яғни D_0 тізбегі шифр кілтінің 63, 55, 47, …, 12, 4 биттерінен тұрады).

8-кестеде келтірілгендей C_0 және D_0 тізбегін генерациялау үшін шифр кілтінің 8, 16, 24, 32, 40, 48, 56, 64 биттері пайдаланылмайды. Бұл биттер шифрлауға әсер етпейді және оларды басқа мақсаттарға пайдалануға болады (мысалы жұптылықты бақылау үшін). Осылайша шифр кілті іс жүзінде 56 биттік болып табылады.

C_0 және D_0 анықталған соң, рекурсивті түрде C_i және D_i анықталады i=1, 2, …, 16. Ол үшін, 9-кестеде көрсетілгендей, итерация қадамының нөміріне байланысты бір немесе 2 битке циклды түрде солға ығысу операциясы қолданылады.



5.1 сурет. K_i кілтін есептеу алгоритмінің сұлбасы

Итерацияның әр қадамында анықталатын K_i кілті - 56 биттік тізбегі және онын орны ауыстыруынан іріктеп алынған нақты биттер нәтижесі. Басқаша айтқанда K_i кілті мынаған тең, $K_i=H(C_iD_i)$, мұндағы H-Функциясы кілтті өңдеуді аяқтайтын матрица арқылы анықталады (10-кесте).

9-кесте

Кілтті есептеуге арналған S_i ығысу кестесі

Итерация нөмірі	S_i солға ығысу саны (бит)	Итерация нөмірі	S_i солға ығысу саны (бит)
1	1	9	1
2	1	10	2
3	2	11	2
4	2	12	2
5	2	13	2
6	2	14	2
7	2	15	2
8	2	16	1

10-кесте

Кілтті өңдеуді аяқтайтын Н-Функциясы
(2-ші орын ауыстырып іріктеу)

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

10-кестеде көрсетілгендей, K_i кілтінің 1-ші биті болып C_iD_i тізбегінің 14-ші биті, екінші - 17-ші бит, K_i кілтінің 47 биті болып C_iD_i -дің 29 биті, ал 48-шісі болып - 32 биті табылады.

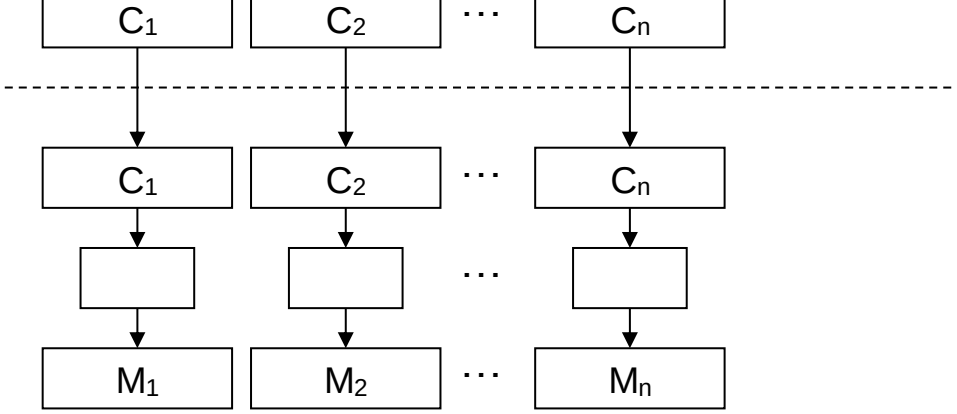
5.2 DES алгоритм жұмысының негізгі режимдері

DES алгоритмі деректерді шифрлеуге де аутентификация жүргізуге де жарайды. Әртүрлі криптографиялық есептерді шығаруда DES алгоритмін пайдалану үшін 4 жұмыс режимі жасалған:

1. ECB (Electronic Code Book) электрондық кодалық кітап;
2. CBC (Cipher Block Chaining) шифр блоктарын тіркемі;
3. CFB (Cipher Feed Back) шифрмәтін бойынша кері байланыс;
4. OFB (Output Feed Back) шығыс бойынша байланыс.

“Электрондық кодалы кітап” режимі

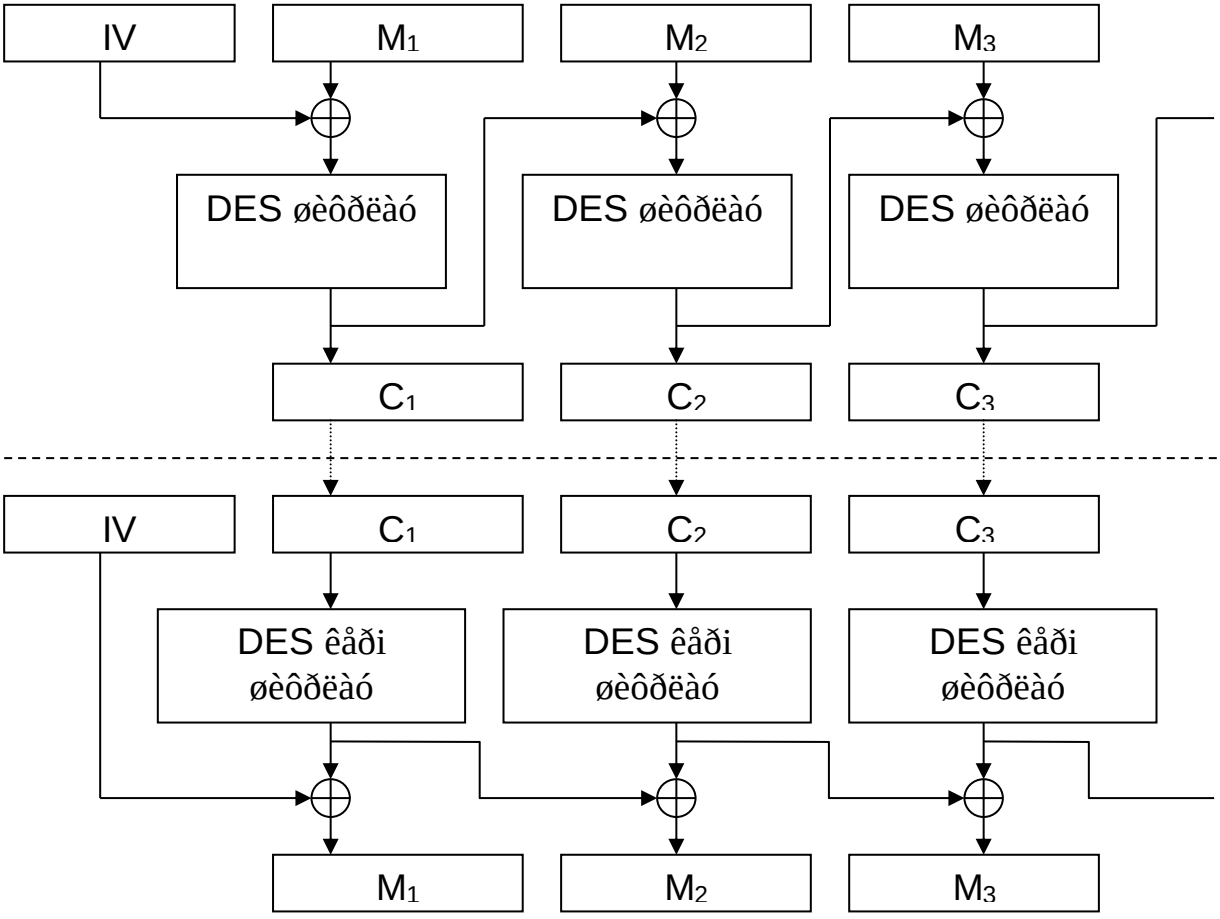
ұзын файлды әрқайсысы 8 байттан 64 биттік бөліктерге (блоктарға) бөледі. Әр блокты бір-біріне тәуелсіз түрде шифрлаудың бір ғана кілтін пайдалана отырып шиФрлайды (3.5 сурет). Негізгі жетістігі - іске асыру қарапайымдылығы, кемшілігі – криптоталдау кезінде тұрақтылығы төмен. Блок ұзындығы 64 битпен шектелген шифрлаудың бекітілген сипатына байланысты “сөздікпен” криптоталдау жасау мүмкіндігі бар. Осындай өлшемдегі блок хабарламада қайталанып тұруы мүмкін. Бұл хабарламадағы ашық мәтіннің ұқсас блоктары шифр-мәтіннің ұқсас блоктарымен беріледі деген сөз. Ол криптоаналитикке хабарлама мазмұны жөнінен мәлімет беруі мүмкін.



5.2 сурет. Электрондық кодалық кітап режиміндегі DES алгоритмінің сұлбасы.

“Шифр блоктарының тіркемі” режимі

Бұл режимде бастапқы M файлы 64 биттік блоктарға бөлінеді: $M=M_1M_2…M_n$. M_1 блогы модуль 2 бойынша 64 биттік бастапқы IV-векторымен қосылады. Бастапқы вектор күнде өзгеріп отырады және құпия сақталынады (3.6 сурет). Алынған қосынды DES кілтін пайдалану арқылы шифрланады. Ол кілт ақпарат жіберуші және қабылдап алушыға ғана белгілі болады. Алынған 64 биттік C_1 шифры модуль 2 бойынша мәтіннің 2-ші блогымен қосылады, нәтиже шифрланады және 64 биттік C_2 - шифры алынады ж.т.б.с.с. Процедура мәтіннің барлық блоктары өңделіп болғанша қайталана береді.



5.3 сурет. Шифр блоктарын тіркемі режиміндегі DES алгоритмінің сұлбасы.

Осылайша, барлық i -1…n үшін (n-блоктар саны) C_i шифрлау нәтижесі анықталады: $C_i=DES(M_i \oplus C_{i-1})$, мұндағы $C_0=IV$ – шифрдың бастапқы мәні, ол бастапқы векторға тең (инициализация векторы).

Әрине шифрмәтіннің соңғы 64-биттік блогы құпия кілттің, бастапқы вектордың және оның ұзындығына байланысты емес ашық мәтіннің әр биттің функциясы болып табылады. ШиФрмәтіннің бұл блогы хабарлаудың аутентификация кодасы(ХАК) деп аталады (КАС код аутентификации сообщения).

ХАК кодасын ақпаратты қабылдап алатын құпия кілтті және бастапқы векторды білетін адам еркін тексере алады. Ол ақпаратты жіберуші адам жасаған процедураны қайталайды. Бөтен адам, әрине КАС генерациясын жасай алмайды.

Бұл режимнің жақсы жағына ол хабарларды беру кезінде қателердің жинақталуына жол бермеуін жатқызуға болады.

M_i блогы C_{i-1} және C_i–нің функциясы болып табылады. Сондықтан хабарларды жібергендегі қателер тек бастапқы мәтіннің 2 блогындағы ғана ақпаратты жоғалту қауіпін туғызады.

Мәліметтерд Ресей стандарт арқылы шарт белгілеу

Бұл стандарт криптографиялық талаптарға сай жасалған және қорғалатын ақпарат құпиялығының дәрежесіне ешқандай шектеу қоймайды. Деректерді шифрлау алгоритмі - 256 биттік кілті бар 64 биттік блокты алгоритм.

Мынадай шартты белгілер пайдаланылады:

- L және R-биттер тізбектері;
- LR - L және R тізбектерінің конкатенциясы, мұнда R тізбегінің биттері L тізбегінің биттерінен соң жүріп отырады;
- $\oplus$ - модулі 2 бойынша әр биттердің өзара қосу операциясы;
- [+] - 2^{32} модулі бойынша екі 32 разрядты екілік сандарды қосу операциясы;
- [+]’ - $(2^{32}-1)$ модулі бойынша екі 32 разрядттық екілік сандарды қосу операциясы.

Екі бүтін $\boldsymbol{a}$ және $\boldsymbol{b}$ сандары $0 \le \boldsymbol{a}, \boldsymbol{b} \le 2^{32}-1$, яғни $a=(a_{32}a_{31} \dots a_2 a_1)$ және $b=(b_{32}b_{31} \dots b_2b_1)$ екілік түрде берілсін дейік:

$$a=a_{32}2^{31}+a_{31}2^{30}+ \dots +a_2 2^1 +a_1,$$

$$b=b_{32}2^{31}+b_{31}2^{30}+ \dots +b_2 2^1 +b_1.$$

бұл екілік сандар 2^{32} модулі бойынша ([+] операциясы) келесі ережеге сәйкес қосылады:

$a[+]b=a+b$, егер $a+b<2^{32}$ болса,

$a[+]b=a+b-2^{32}$, егер $a+b \ge 2^{32}$ болса,

ал $2^{32}-1$ модулі бойынша (операция [+]’) келесі ережеге сәйкес қосылады:

$a[+]’b=a+b$, егер $a+b<2^{32}-1$,

$a[+]’b=a+b-(2^{32}-1)$, егер $a+b \ge 2^{32}-1$

Алгоритмде төрт жұмыс режімі қарастырылған:

- қарапайым ауыстыру режімінде деректерді шифрлау;

- гаммалау режімінде деректерді шифрлау;
- кері байланысы бар гаммалау режімінде деректерді шифрлау;
- имитовставканы жасау.

Қарапайым ауыстыру режімі

Қарапайым ауыстыру режімінде деректерді шифрлау алгоритмін іске асыру үшін жалпы криптожүйе блоктарының бір бөлігі ғана пайдаланылады (16-сурет). Сұлбадағы шартты белгілер:

- N_1, N_2 - 32 разрядтық жинағыштар;
- $СМ_1$ - 2^{32} модулі бойынша 32 разрядтық қосындылағыш ([+]);
- $СМ_2$ - 2 модулі бойынша 32 разрядтық қосындылағыш ($\oplus$);
- R - циклдік ығысудың 32 разрядттық регистрі;
- КЗУ - 256 биттік кілттік жадтайтын құрылғысы ол X_0, X_1, X_2, … X_7 32 разрядттық 8-жинағыштан тұрады;
- S - (S_1, S_2, S_3, …, S_7, S_8) 8 ауыстыру торабынан (узел) тұратын (S - ауыстыру блогы) ауыстыру блогы.

Ашық деректерді қарапайым ауыстыру режімінде шифрлау**.** Шифрлауға жататын ашық деректерді T_o деген 64-разрядтық блоктарға бөледі. T_o блоктарды шифрлау процедурасы 32 циклдан тұрады (i=1…32). Кілттік жаттайтын құрылғысына К кілтінің 256 битін, сегіз 32 разрядттық K_i кілтшілері түрінде енгізеді:

$$К=К_7 К_6 К_5 К_4 К_3 К_2 К_1 К_0$$

блогының биттер тізбегін:

$T_o = (a_1(0), a_2(0), …, a_{31}(0), a_{32}(0), b_1(0), b_2(0), …, b_{31}(0), b_{32}(0))$

32-биттік екі тізбекке бөледі: b(0) a(0), мұндағы b(0) сол немесе үлкен биттер, a(0) оң немесе кіші биттер.

бұл тізбектерді a(0) -> N_1 және b(0) -> N_2 жинақтағыштарына шифрлаудың бірінші циклының басында енгізеді.

64 разрядтты блогы бар ашық деректерді шифрлау процедурасының j цикл нөміріне байланысты келесі теңдеу арқылы жазуға болады.

$$\begin{cases} a(j) = f\,(a(j-1)[+]\,K_{j-1(mod8)}) \oplus b(j-1) \\ b(j) = a(j-1) \end{cases} \quad j=1, …, 24 \text{ болғанда}$$

$$\begin{cases} a(j) = f\,(a(j-1)[+]\,K_{32-j}) \oplus b(j-1) \\ b(j) = a(j-1) \end{cases} \quad j=25, …, 31 \text{ болғанда}$$

$$\begin{cases} a(32) = a(31) \\ b(32) = f\,(a(31)[+]\,K_0) \oplus b(31) \end{cases} \quad j=32 \text{ болғанда}$$

Мұнда:

- a(j)=(a_{32}(j), a_{31}(j), …, a_1(j) j циклы шифрлауынан соңғы N_1 жинақтағышы;
- b(j)=b_{32}(j), b_{31}(j), …, b_{31}(j) j-ші циклы шифрлауынан соңғы N_2 жинақтағышы, j=1, …, 32;
- $T_ш$ - шифрланған деректер блогы (64 разряд) N_1 және N_2 жинақтағыштарынан келесі тәртіппен шығарылады, алдымен N_1 жинақтағышының 1, …, 32 разрядтарынан, сосын N_2 жинақтағышының 1, …, 32 разрядтарынан, яғни кіші разрядтардан бастап

$T_ш$ = (a_1(32), a_2(32), …, a_{32}(32), b_1(32), b_2(32), …, b_{32}(32)).

f - шифрлау функциясы.

f-функциясының аргументі болып 2^{32} модулі бойынша алынған a(j) санымен K_j санының қосындысы табылады.

K_j – X_j КЖҚ жинақтағышынан оқылған ішкілті (subkey). бұндағы әрбір сан 32 битке тең.

f-функциясында алынатын 32 разрядтық қосындыға екі операция жасалады (a(j)[+]K_j).

бірінші операция *ауыстыру* деп аталады да, S ауыстыру блогы арқылы орындалады. S ауыстыру блогы сегіз ауыстыру торабынан тұрады (S ауыстыру блогы) S_1, S_2, …, S_8 оның әрбіреуінің жады көлемі 64 бит. $СМ_1$-дан S ауыстыру блогына түсетін 32 разрядтық векторды сегіз тізбектелген 4 разрядтық векторларға бөледі. Олардың әрқайсысы өздеріне сәйкес ауыстыру торабы бар төрт разрядтық векторға түрленеді. Әр ауыстыру торабының диапазоны 0000…1111 болып келетін оналты төрт разрядты екілік сандардың орын ауыстыру-кестесі түрінде көрсетуге болады. Кіру векторы кестедегі қатар (жол) адресін көрсетеді. Ал бұл жолдағы сан шығу векторы болып табылады. Содан соң төрт разрядттық шығу векторлары бірінен соң бірі 32 разрядты векторға бірігеді. Ауыстыру түйіндері (ауыстыру-кестесі) есебінде, ҮЕМ-дер торабы үшін жалпы болып табылып өте сирек өзгеретін кілттік үлементтері жүреді. бұл ауыстыру түйіндері өте құпия сақталады.

Екінші операция S ауыстыру блогының шығуынан алынған 32-разрядтық векторды, циклдік солға (11 разрядқа) ығыстыру. Циклдік ығыстыру R-ығыстыру регистрі арқылы орындалады. Одан соң f шифрлау функциясының нәтижесімен N_2-жинақтағышының 32 разрядтық бастапқы b(j) толтырылуы $СМ_2$ қосындылауышта әр разрядты өзара 2 модулі бойынша қосуды жүргізеді.

Қосындыда N_2 жинақтағышының 32 разрядтық бастапқы b(j) толтырылуы болады. Содан соң $СМ_2$ шығуына алынған нәтиже N_1 жинақтағышына жазылады, ал N_1-нің ескі мәні N_2 жинақтағышына көшіріліп жазылады (b(j) = a(j-1)).

32 цикл өткенше шифрлауда КЖҚ–дағы кілтшелерді таңдау мына тәртіпте жүреді:

K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7

K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_7, K_6, K_5, K_4, K_3, K_2, K_1, K_0

32 циклда $СМ_2$ қосындылауыштағы нәтиже N_2 жинақтағышына еңгізіледі де, ал N_1 жинақтағышында бұрынғы толтырылым сақталады. Шифрлаудың 32 цикл нәтижесінде алынған N_1 мен N_2 жинақтағыштардың толтырулар ($Т_о$ ашық деректер блогына сәйкес келетің) шифрланған деректер блогы $Т_ш$ болып табылады.

Қарапайым ауыстыру режимінде шифрды ашу. Қарапайым ауыстыру режимінде - шифрды ашудың алгоритмі 16 суретегі сұлба бойынша орындалады. КЖҚ-ға шифрлау кезінде пайдаланған кілттің 256 биті енгізіледі. Шифрды ашуға дайындалған деректерді әр біреуінде 64 биттер бар $Т_ш$ - деген блоктарға бөлінеді.

Кез келген $Т_ш$ - блогын

$$Т_ш = (a_1(32), a_2(32), \ldots, a_{32}(32), b_1(32), b_2(32), \ldots, b_{32}(32))$$

$a(j) \to N_1$, $b(j) \to N_2$ болатындай етіп енгізеді.

Шифрды ашу шифрлау алгоритміндей жасалады. Тек оның айырмашылығы $X_0, X_1, \ldots, X_7$ жинақтағыштарынын толтыруда КЖҚ-дан шифрды ашу циклы бойынша келесі тәртіппен оқылады:

$$К_0, К_1, К_2, К_3, К_4, К_5, К_6, К_7, К_7, К_6, К_5, К_4, К_3, К_2, К_1, К_0,$$
$$К_7, К_6, К_5, К_4, К_3, К_2, К_1, К_0, К_7, К_6, К_5, К_4, К_3, К_2, К_1, К_0.$$

Шифрды ашу тендеуінің түрі мынадай болады:

$$\begin{cases} a(32-j) = f\,(a(32-j+1)[+]\,K_{j-1}) \oplus b(32-j+1) \\ b(32-j) = a(32-j+1) \end{cases} \quad j=1,\ldots, 8 \text{ болғанда}$$

$$\begin{cases} a(32-j) = f\,(a(32-j+1)[+]\,K_{32-j(\mathrm{mod}8)}) \oplus b(32-j+1) \\ b(32-j) = a(32-j+1) \end{cases} \quad j=9, \ldots, 31 \text{ болғанда}$$

$$\begin{cases} a(0) = a(1) \\ b(0) = f(a(1)[+]\,K_0) \oplus b(1) \end{cases} \quad j=32 \text{ болғанда}$$

N_1 және N_2 жинақтағыштары 32 цикл жұмыс бойынша ашық деректер бар блок пайда болады:

$$Т_о = (a_1(0), a_2(0), \ldots, a_{32}(0), b_1(0), b_2(0), \ldots, b_{32}(0)),$$

ол шифрланған $Т_ш$ деректер блогына сәйкес келеді.

Егер 64 биті бар $Т_о$ блокты қарапайым ауыстыру режімінде шифрлау алгоритмін А арқылы белгілесек, онда

$$A(Т_о) = A(a(0), b(0)) = (a(32), b(32)) = Т_ш.$$

Қарапайым ауыстыру режімін деректерді шифрлау үшін шектелген жағдайда ғана қолдану, мысалы кілт жасағанда, яғни кілтті байланыс арналары бойынша беру немесе оны ҮЕМ жадында сақтауда, оның қорғанысын қамтамасыз етуде ғана қолдану тиімді болады.

Бақылау сұрақтар:

1. Симмертиялық алгоритмдер дегеніміз не?
2. Симметриялық алгоритмнің бөлімдерін атап беріңіз?
3. Олардың атқаратың қызметтері?.
4. Толласыз мәліметтерді шифрлеу дегеніміз не?.
5. Шифрді ауыстырып қою дегеніміз не? Ол не үшін қажет?
6. DES алгоритмінде қандай шартты белгілер қолданылады?
7. Алгоритмнің негізгі артықшылықтары қандай?
8. Шифрлау функциясы қалай орындалады?

Пайдаланылатын әдебиет: [1]-20-25 б. [2] – 82- 102 б.

Дәріс №7. Тақырыбы: Кілттерді ашық түрде тарату жүйесі

Жоспар:

1. **Кілттерді ашық түрде тарату жүйесі**
2. **Шифрлеу**

Мақсаты: Кілттерді ашық түрде тарату жүйесімен таныстыру
Кілттік сөздер: ассиметрия, RSA, шифр, дешифр

1. Кілттерді ашық түрде тарату жүйесі

Кілттерді ашық түрде тарату жүйесің (асимметриялық шифрлер) ашық кілт барлық адамдарға қол жеткілікті болу үшін қоланады. Бұл кез келген адамға шифрді шешіп алу мүмкіндігін береді. Бырақ бұл хабарды тек керек адамғана шешіп алалады (шешіп алатың кілтті білетін ғана адам). Шифрлеу кілтті ашық кілт еп атайды, шешіп алу кілтті – жабық немесе құпия кілт деп атайды.

RSA – ашық кілттін криптографиялық жүйесі. Криптожүйе RSA 1977 жылы шығарылған және оны ойлап шығарушылардың атымен аталған Ronald Rivest, Adi Shamir и Leonard Adleman.
RSA алгоритмі келесі әдіспен жұмыс істейді: Екі қарапайым үлкен сандар алынып, p және q, олардың көбейтіндісі есептеледі n = p*q; n – модуль деп атайды. Сан қарапайым болады, егер ол 1-ге немесе өз өзіне бөлінсе. Содан кейін e саны таңдалады, мына шартты қанағаттайтын 1< e < (p - 1)*(q - 1) және ортақ бөлгіш саны жоқ (p - 1)*(q - 1) (өзара жай сан), 1-ден басқа. Содан кейін d саны есептеледі, осылайша (e*d - 1) мынаған (p - 1)*(q – 1) бөлінеді.

- (n; e) – ашық (public) кілт
- (n; d). – жабық (private) кілт.

Егер де көбейткіштерге ажыратудың тиімді әдісі болғанда, онда p и q көбейткіштерден n ажыратып құпия кілтті алуға болады. Сондықтан, RSA криптожүйенің дәйектілігі тәжірибелік шешілмес есепке негізделеді.

2. Шифрлеу

Хабар жіберу үшін Жіберуші С шифрланған мәтінің құрады да, М хабарың n: $C = M^e \pmod n$ модуль дәрежесіне шығарады, мұндағы е және n – Алушының ашық (public) кілті. Содан кейін Жіберуші С (шифрланған мәтінді)Алушыға жібереді. Алған мәтінді шешіп алу үшін Алушы С мәтінің d дәрежесі бойынша n: $M = C^d \pmod n$ модуліне ауыстырады; мұндағы е және d Алушының М есептеу үшін қажет.Алушы d мәнің қойған кезде ол оның алынған хабардың шешуіне әкеледі.

Мысалы: p=7; q=17 екі жай сан аламыз /тәжірибеде ол сандар бір неше рет үлкен болуы мүмкін/. Бұл жағдайда n = p*q 119 тең. Енді е таңдаймыз, мұндағы e=5. Келесі орындайтың қадамымыз d санының табу, мұндағы $d*e=1 \bmod [(p-1)(q-1)]$. d=77 (Эвклидтың кенейтілген алгоритмі қолданған). d – құпия кілт, ал е және n ашу кілтін мінездейді. Егер бізге берілген мәтінді шифрлеу M=19 түрінде берілсе,ол $C = M^e \bmod n$. Онда C=66 шифрленген мәтін аламыз. Бұл «мәтінді» керекті адресаттқа жіберуге болады. Алушы алынған хабарды $M= C^d \bmod n$ және C=66 дисшифрлаторларың қолданады, сонда қорытындысы M=19

RSA криптожүйесі көптеген салалармен өнімдерде қолданылады. RSA BSAFE шифрлеу технологиясымен әлемнің 500 миллион қолданушылары қолданып келеді. Өткені көп жағдайда RSA алгоритмі қолдана отырып оның көп таратылған криптожүйелерінің ашық кілтін Internet жүйесінде қолданған ыңғайлы.

RSA –тың бұзу әдістері бар. Ең тиімді жолы: жеке кілтті (private) табу, қажетті ашық (public) кілтке сәйкес келетін. Бұл шабуылшыға хабарды оқуға және қолды қолдан жасауға мүмкіндік береді. Мұндай шабуылды ортақ модульды n – p және q негізгі көбейткішін табу арқылы жүзеге асырады. p, q және e (ортақ көрсеткіш) негізінде, шабуылшы d көрсеткішті онай таба алады. Негізгі қиыншылық - n қажетті көбейткішті іздеу (факторинг); RSA қауыпсіздігі көбейткіштердің ажыратуына байланысты.

RSA алгоритмінде кілттің көлемі n модуль көлеміне байланысты. p және q екі санының туындысы модуль болатын, ұзындығы шамалы бірдей болуы керек. Мұндай жағдайда көбейтіндісін табу қиындыққа соқпайды.

RSA криптожүйесі көптеген өнімдерде, әр түрлі платформаларда, көптеген салаларда қолданылады. Сонымен қатар оны Microsoft, Apple, Sun және Novell операциялық жүйелерде қолданады.

RSAкриптожүйе – көптеген стандарт бөлімдеріне жатады. Қазіргі уақытта көптеген өнделіп жатқан стандарттар не болмаса RSA алгоритмін немесе RSA криптожүйесін ұсынады.

Бақылау сұрақтары:

1. Кілттерді ашық түрде тарату жүйесі
2. RSA криптожүйесі дегеніміз не?
3. RSA криптожүйесі қай жылы шығарылған?
4. Шифрлеу дегеніміз не?

5. RSA криптожүйесі стандарт бөлімдеріне жатама?

Пайдаланылатын әдебиет: [1], 13-15 бет.

Дәріс №8. Тақырыбы: Дискреттік экспоненттер негізінде Диффи-Хеллман жүйесі

Жоспар: Диффи-Хеллман жүйесі

Мақсаты: Дискреттік экспоненттер негізінде Диффи-Хеллман жүйесімен таныстыру

Кілттік сөздер: mod, код, Диффи, Эйлер,

1976 жылы Диффи және Хеллман статья шығарды, онда екі кілттік крипография ашылғаны туралы жазылды Онда тәжірибелі табанды құпиялық жүйе құрылымының мүмкіншілігі бар, сондықтан құпия кілтті жіберу қажет емес.

Диффи-Хеллман жүйесін қолдануына мысал.

А абонентпен В абонент бір біріне шифрланған ақпарат ашық кілтімен жіберді Әр абонент екі үлкен қарапайым сан таңдайды, олардың көбейтіндісін анықтайды және кездесоқ бір сан тандайды

АБОНЕНТ А	**АБОНЕНТ В**
1. p және q екі қарапайым сан тандау.	
p=7	p=11
q=13	q=23
2. Көбейтіндісін есептейді. r=p*q	
r=7*13=91	r=11*23=253
3. Эйлера функциясының есептейді j(r)=r-p-q+1. Эйлер функциясы – арифметикалық функция j(r), оның мәніоң сандарға тең, r аса басым түспейтін және r өз ара қарапайым .	
j(r)=91-7-13+1=72	j(r)=253-11-23+1=220
4. s кездесоқ санды тандау, j(r) өз ара қарапайым 0<s< j(r) интервалдан	
s=5	s=31
5. t құпия кілтті есептеу s*t mod j(r) = 1 ара қатынасынаң	
5*t mod j(r)=1	31*t mod j(r)=1
t=29	t=71
6. Ашық кілтті жариялау	
s=5, r=91	s=31, r=253

А абоненті В абонентке хабарды жібергісі келді деп ойлайық. Біріншіден орын алмасу әдісімен әр символды сонмен ауыстырайық.

Мәтін: Д Е П О З И Т

Коды: 5 6 16 15 8 9 19

А абоненті В абонентіне формула бойынша ашық кілтті қолданып хабарды шифрлайды.

$C=M^s \bmod r$

$C(1)=5^{31} \bmod 253 = 126$

$C(2)=6^{31} \bmod 253 = 39$

$C(3)=16^{31} \bmod 253 = 192$

$C(4)=15^{31} \bmod 253 = 37$

$C(5)=8^{31} \bmod 253 = 239$

$C(6)=9^{31} \bmod 253 = 163$

$C(7)=19^{31} \bmod 253 = 217$

Шыққан сандар: 126 39 192 37 239 163 217, ДЕПОЗИТ мәтіннің шифрі болады. В абоненті шифрограмманы қабылдап өз құпия кілтін қолданады, және хабарды формула бойынша шешеді: $M=C^t \bmod r$

$M(1)=126^{71} \bmod 253 = 5$

$M(2)=39^{71} \bmod 253 = 6$

$M(3)=192^{71} \bmod 253 = 16$

$M(4)=37^{71} \bmod 253 = 15$

$M(5)=239^{71} \bmod 253 = 8$

$M(6)=163^{71} \bmod 253 = 9$

$M(7)=217^{71} \bmod 253 = 19$

Шифрден шешіп алу нәтежесінде сандар шығады 5 6 16 15 8 9 19, ДЕПОЗИТ мәтініне сәйкес келеді кодтық кесте бойынша.

Осы жүйенің басқа ашық кілтімен криптожүйелерден қарағанда бір мүмкіншілігі, ол құпия кілттін жабық каналдардаң беріп жібермеуі. Хабарды басқа абонент беріп жібермегеніне көз жеткізгісі келсе , онда аутенфикация жасау қажет, яғни құжаттың жіберілген авторлық дәлелдігі. Мұнда электрондық қол қою қолданылады.

Бұл әдістін мағынасы, ол хабар тек ашық кілтпен ғана шифрленіп жіберілмейді онда хабар жіберуші абоненттін өз құпия кілті болады.

Мысал келтірейік. Абонент В негізгі мәтін жібергісі келді дейік.

Д И С К

Рассмотрим пример.

5 9 18 11

Ең бірінші абонент А ашық кілтін пайдаланып хабарды шифрлейді.

$C(1) = 5^5 \bmod 91 = 31$

$C(2) = 9^5 \bmod 91 = 81$

$C(3) = 18^5 \bmod 91 = 44$

$C(4) = 11^5 \bmod 91 = 72$

Содан соң абонент В осы хабарды қайта өз құпия кілтімен мына формуланы қолданып шифрлейді: $N=C^t \bmod r$

$N(1)=31^{71} \bmod 253=108$

$N(2)=81^{71} \bmod 253=202$

$N(3)=44^{71} \bmod 253=198$

$N(4)=72^{71} \bmod 253=105$

Шифрограмма 108 202 198 105 абонент А жіберіледі.

Абонент А, Құпия хабарды алған соң ең бірінші абонент В ашық кілтін қолданып шешеді.

$C=N^s \bmod r$

$C(1) = 108^{31} \bmod 253 = 31$

$C(2) = 202^{31} \bmod 253 =81$

$C(3) = 198^{31} \bmod 253 =44$

$C(4) = 105^{31} \bmod 253 =72$

Содан соң абонент А өз ашық кілтін қолданып негізгі мәтінді шығарады.

$M(1)=31^{29}$ MOD 91 = 5

$M(2)=81^{29}$ MOD 91 =9

$M(3)=44^{29}$ MOD 91 =18

$M(4)=72^{29}$ MOD 91 =11

Электрондық қол қоюды қолданған кезде ешкім абонент А-ға абонент В-дан хабар жібере алмайды, өйткені абонент В-ға тек белгілі құпия кілтті ғана қолдану керек.

Бекіту сұрақтары:

1. Кілттерді ашық түрде тарату жүйесі дегеніміз не?**.**
2. Шифрлеу қажеттілігі.
3. RSA криптожүйе дегеніміз не?
4. RSAкриптожүйесін бұзу жолдары.

Пайдаланылатын әдебиет: [2], 35-166 бет.

Дәріс №9. Тақырыбы: Ақпараттың нақтылығы

Жоспар:

1. Ақпараттың нақтылығы
2. Жіберушінің іс әрекеті
3. Қабылдаушының әрекеті
4. Парольдау.
5. Көшірме алу.
6. Архивтау.
7. Шифрлау.
8. Ішкі аудит.
9.

Мақсаты: Ақпараттың нақтылығымен таныстыру

Кілттік сөздер: Пароль, кілттік фраза, Конфиденциал

1. Ақпараттың нақтылығы

Цифрлік қол қою (digital signature) – ол хабардағы ақпараттың толықтығын тексеру және жіберушінің нақтылығын тексеру әдісі. Ассиметриялық шрифтерді қолданып, өз қолың және жүп кілттерін қолданып жүзеге асырылады.

Жіберуші хабарға қол қойып, кілтпен шифрлеп хатпен бірге жібереді. Қабылдаушы хабарды алып, ашық кілтпен оны шешеді. Сонымен қатар қабылдаушы алған хабардан цифрлық қолды есептеп, шифрден шешіп алғандағымен салыстырады. Егер нәтежесінде екеуі бірдей болса, онда қол нақты дегенді білдіреді. Әйтпегенде, не хабардағы ақпарат өзгертілген, не қолы өтірік қойылған.

Мысалы, Алиса Бобқа хабар жібермекші дейік. Сонымен қатар, Боб хабарды біреу оқып қоймағаны және де шындығында Алисадан екендігіне сенімді болу керек. Ол үшін Алиса цифрлық қол S құрастырады, M санды d дәрежеге келтіріп және n модуліне көбейтіп өзінің жеке кілтін құрады.:
$S = M^d (\mathrm{mod}\ n)$, мұнда d және n – Алисаныңжеке кілті. Ол Бобқа M және S жібереді.
Боб қолды тексеру үшін S –ты e дәрежеге келтіреді және n модуліне көбейтеді:
$M = S^e (\mathrm{mod}\ n)$, мұнда e және n – Алисаның ашық (public) кілті.
Сондықтан, шифрлеу және хабардың авторлық нақтылығын тексеру үшін құпия (private) кілттердің жіберуі қажет емес: корреспонденттін екеуіде тек ашық (public) кілтті немесе жеке өз (private) кілтін қолданады.

Ақпаратты жіберген кезде бірге немесе бөлек болу керек:
1. Конфиденциалдық (privacy) – жаугер жіберілетін хабардың мазмұның білмеу керек.
2. Нақтылығы (authenticity), ол екі мағынаны білдіреді
- тұтастық (integrity) – хабар кездесоқ немесе әди өзгерістерден қорғалған болу керек;
- жіберушінің идентификациясы (авторлықты тексеру) – қабылдаушы хабар кімнен жіберілгенін тексеру мүмкіндігі болу керек.

Шифрлеу конфиденциялдықты қамтамасыз ете алады, ал кейбір жүйелерде тұтастықты.

Хабардың тұтастылығы бақлау функциялар арқылы тексеріледі (check function) Хабардан – үзындығы үлкен емес белгілі бір сан. Осы бақылау функциясы хабардың кішкентай бір өзгерістерінде де жоғарғы ықтималымен өзгерілу керек (ақпараттың жою, енгізу, орын алмасу немесе қайта ретке келтіру кезінде). Бақылау функциясын әр түрлі атайды және тандайды:
- хабардың нақты коды (Message Authentical Code, MAC);
- бақылау суммасы;
- хеш-функциясы (hash);
- **ГОСТ 28147-89-қа имитоенгізгіш;**

Мысалы:

Жіберушінің ашық және құпия кілттері мыналар:
e=3, d=7, n=33

Қабылдаушының ашық және құпия кілттері мыналар:
e=5, d=17, n=21

2. ЖІБЕРУШІНІҢ ІС ӘРЕКЕТІ

Хабарды жіберу үшін Жіберуші негізгі мәтінді келесі формуламен шифрлейді $C=M^e \bmod n$, мұнда e және n – Қабылдаушының жүп ашық кілттері. Негізгі мәтінді алайық ЗВЕЗДА, кодтық кесте арқылы әріптерді санға аударайық 8 3 6 8 5 1

Қабылдаушының ашық кілті арқылы шифрлейміз:
$C(1)=8^5 \bmod 21=8$ - З
$C(2)=3^5 \bmod 21=12$ - Л
$C(3)=6^5 \bmod 21=6$ - У
$C(4)=8^5 \bmod 21=8$ - З
$C(5)=5^5 \bmod 21=17$ - Р
$C(6)=1^5 \bmod 21=1$ – А

Енді Жіберуші хабарға қолын қояды, цифрлық қолды S-деп формуламен құрады. $S=M^d \bmod$, мұнда d және n –Жіберушінің жүп құпия кілттері
$S(1)=8^7 \bmod 33 = 2$ - Б
$S(2)=3^7 \bmod 33 = 9$ - И
$S(3)=6^7 \bmod 33 = 30$ - Э
$S(4)=8^7 \bmod 33 = 2$ – Б
$S(5)=5^7 \bmod 33 =14$ - Н
$S(6)=1^7 \bmod 33 = 1$ - А

Жіберуші Қабылдаушыға қол қойылған шифрмәтінді жібереді:
ШИФРМӘТІН: 8 12 6 8 17 1
Цифрлық қолы: 2 9 30 2 14 1

3. Қабылдаушының әрекеті

Ең біріншіден Қабылдаушы хабарды формуламен шешеді $M=C^d \bmod n$, мұнда d және n – Жіберушінің құпия кілті
$M(1)=8^{17} \bmod 21 = 8$ - З
$M(2)=12^{17} \bmod 21 = 3$ – В
$M(3)=6^{17} \bmod 21 = 6$ – Е
$M(4)=8^{17} \bmod 21 =8$ – З
$M(5)=17^{17} \bmod 21 = 5$ – Д
$M(6)=1^{17} \bmod 21 =1$ – А

Одан негізгі мәтін шығады. Енді хабардың нақтылығын анықтау қажет. Ол үшін цифрлық қолды мына формуламен есептейді:
$M=S^e \bmod n$, мұнда e және n – Жіберушінің ашық жүп кілттері
$M(1) = 2^3 \bmod 33 = 8$ – З
$M(2) = 9^3 \bmod 33 =3$ – В
$M(3) = 30^3 \bmod 33 =6$ – Е

$M(4) = 2^3 \bmod 33 = 8 – З$
$M(5) = 14^3 \bmod 33 = 5 – Д$
$M(6) = 1^3 \bmod 33 = 1- А$

Нәтежесі бірдей болса, онда қолы нақты екен, бұл құжаттын өзгермегендігін және хабар автордікі екендігін білдіреді.

4. Парольдау.

Компьютерлік жүйенің кез келген түрін қорғау ең қарапайым және арзан жолы: пароль қолдану.

Парольдар, жүйеге кіру кілті болып қарастырылады, бірақ оларды басқа мақсаттарда да қолданылады: дискжетекте жазуды блокировкалау, деректерді шифрлау командаларында немесе файлдарды архивтан шығару — қолданушының программалық қамтамасыздандыру немесе заң түріндегі иесі арқылы әрекеттер жасалады.

Парольдар жеті негізгі тобтқа бөлінеді:

1) қолданушы қоятын парольдар;
2) жүйе генерациялайтын парольдар;
3) жүйе генерациялайтын кездесоқ қол жету коды;
4) жарты сөз;
5) кілттік фразалар;
6) интерактивтік кезек түрінде сұрақ — жауаб;
7) қатал парольдар .

Егер сіз біреуді жаулап алғыныз келсе, біріншіден біліп алу керек сол жүйеде осы жеті пароль түрінен қайсысы екенін. Біріншісі көп қолданбалы. Кездесоқ парольдер мен кодтар бірнеше түрден тұруы мүмкін. Жүйелік программалық қамтамасыздандыру толық кездесоқ қатар символдарды қолданыла алады— кездесоқ регистрларға дейін, цифр, пунктуаци жәнеүзындығы; немесе генерацияланған процедураларда шек ара қолдануы мүмкін. Мысалы, әр қолжетерлік коды алғашқы дайындалған шаблонмен сәйкестелген болады (oabc-12345-efghn, мұнда әріптермен сандарберілген позицияда кездесоқ түрде генерацияланады).

Жарты сөз көбінесе қолданушы арқылы құрылады, ал кейбірде — кейбір кездесоқ процесспен. Яғни, егер де қолданушы оңай табылатын пароль құрсада, мысалы «секрет», компьютер оны толықтырып қояды, сонымен ол күрделі парольға айналады «секрет,5гЫ 1».

Кілттік фразалар тиімділігі, ол үзынырақ және оны шешу өте қиынға түседі, бірақ оны жаттап алу оңай. Фразалар мағыналы болу — we were troubled by that немесе мағынасыз болу мүмкін — fished up our nose. Парольдың алтыншы түрі — интерактивтік кезек түрінде сұрақ — жауаб қолданушыға бірнеше сұрақтарға жауап беруды сұрайды: әйелініздің қыз кезіндегі фамилиясы? Сіздің жақсы көретін түсініз? Және т.б. Компьютерде осындай сұрақтардың бірнеше жауабтары сақталып тұрады. Қолданушы жүйеге кірер кезінде, компьютер жауаптарды салыстырып отырады. *Қатал парольдер,* көбінесе сыртқы электрондық немесе механикалық құрылыстарда— церберде қолданылады. Компьютер бірнеше шақыру вариантарды ұсынады, ал қолданушы оған жауап іздеу қажет.

5. Көшірме алу.

Ақпараттық жүйені эксплуатациялау кезінде ең негізі— деректердің толығымен бүтіндігін сақтау. Сондықтан резервтік көшірме алу жүйесі қолданылады. Сонымен қатар бүл жүйе корпоративтік деректер архивін жинап және қамтамасыз ете алады.

Резервтық көшірме алу сыртқы жетектерде ақпаратты сақтау мақсатында қолданылады , авария немесе ақпараттық жүйеде қателік кеткен жағдайда оны қажет етеді.

Желілік резервтық көшірме алу жүйесінің іске асыру үшін клиент-сервер технологиясының мүмкіншілігі:

- резервтық көшірмесін сақтау үшін бір құрылғыны қолдану мүмкіндігі;
- Ортақтарлыған жоспарланған жұмыс және оларды басқару;
- Резервтік көшірмені локалдық сақтау.

6. Архивтау.

Істеп қоиылған ақпараттарды көп мерзімге сақтап қою үшін архивтап тастайды. Көбінесе бұл ақпаратты қолданбайды, бірақ керек кезде әр қашанда алып шығыра алады. Ол процесті архивтан шығару деп атайды.

Барлық резервтық көшірме алу (архивтау) программаларын үш категорияға бөлуге болады.

1. операциялық жүйе құрылыма кіретін Алғашқы деңгейлі жүйе.
2. Қазіргі кезде нарықта ортанғы деңгейлі жүйе қолданылады.мысалы көбіне белгілі ол ARCservelT компаниясы Computer Associates, Backup Exec от Seagate Software және NetWorker компаниясы Legato Systems.
3. Жоғарғы деңгейлі жүйе, ол күрделі гетерогенды ортада қолданылады. Мысалы оған ADSM компаниясы IBM және OpenView OmniBack II от Hewlett-Packard жатады.

7. Шифрлау.

Шифрға ақпаратты жіберу кезінде қолданылатын атауы немесе шартты белгілеу жүйесінің коды жатпайды. Кодтау көбінесе жіберілген ақпараттың сапасы жоғарғы болу үшін қолданылады.

Криптография — Заңсыз қолданушылардан қорғау мақсатында ақпараттарды өзгерту(шифрлау) әдістері туралы ғылым

Стеганография — хабарды жіберу фактісін жасырындылық әдісімен құралдар жиыны.

Шифр — заңсыз қолданушылардаң қорғау мақсатында ақпараттарды өзгерту әдісі, амалы.

Ақпараттарды қорғау бойынша криптографиялық өзгерту екі мақсатты жетілдіру үшін арналған:

- Кілт жоқ адамдардан ақпаратты қол жетпейтіндей қамтамасыз ету;
- Бекітілмеген бұрмалануды табу сенімділігін қолдау.

Басқа ақпараттарды қорғау әдістерінен қарағанда классикалық криптография қорғауды тек мынадай жағдайларда кепіл болады:

- тиімді криптографиялық алгоритмді қолдану;
- кілттін құпиялығын және нақтылығын сақтау.

Криптология — екі тараудан тұратын ғылым: криптографиядан және криптоанализдан.

Криптоанализ — шифрлерді шешу әдістерімен амалдары туралы ғылым (және тәжірибеде оны қолдану).

5.Ішкі аудит.

Бұл ақпаратты қорғау элементі мекемелердің ішінде ақпаратты қорғау политикасын іске асыруына бағытталған. Ол келесі мәселелерді қарастырады.

Компьютерлік вирус — бұл өз көшірмесін құрайтын және оларды бірнеше объектілерге, локальдық ресурстарға, компьютерлік жүйелерге, желілерге және т.б. енгізе алатын программа. Сонымен қатар көшірмелері өз қабілеттерін жоғалтпай көбие береді.

Компьютерлік вирустар көп болғандықтан олар *классификацияны* қажет етеді. Вирустарды келесі нашындары бойынша классификациялауға болады:

- Қоршаған ортада;
- Қоршаған ортаны жұғу әдісі;
- Деструктивтік мүмкіншілігімен;
- Вирус алгоритмнің ерешелігімен.

Қоршаған орта бойынша вирустарды файлдық, жүклеулік, және макровирустар деп бөлуге болады.Файлдық вирустар орындалып жатқан файлдарға енеді (*.COM, *.EXE, *.SYS, *.BAT, *.DLL). Жүктелік — дисктың жүктелетін секторына (Boot-сектор) немесе жүйелік жүктелу винчестер секторында (Master Boot Record). Макровирустар жұмыс істеп жатқан кезде макрос деп аталатын жүйеге енеді (например, Word, Excel). Кейбір жағдайда бірігіп енуі мүмкін, мұндай вирустарда күрделі алгоритм і болады және оларды іздеп табу қиындыққа соғады.

Жұғу әдісі бойынша вирустар резиденттық және резиденттық емес болып бөлінеді. Резиденттық вирустар компьютердыңоперативтық жадында өзінің резиденттық бөлімін қалдырып кетеді, сосын әрі қарай теренірек енеді. Резиденттық вирустар жадта компьютер өшіп немесе қайта қосылғанға дейін белсенді болады.Ал резиденттық емес вирустар жадқа жұқпайды және шектелген уақытта ғана белсенді болады.

Деструктивті мүмкіншілігі бойынша вирустарды келесі түрлерге бөлуге болады:

- Зиянсыз, компьютердың жұмысына әсер етпейді, бірақ өзінің таралуына байланысты дисктін бос орының азайтады;
- қауыпсыз, дисктін бос орының азайтады сонымен қатар графикалық, дыбыстық және басқа эффектілеріне әсер етеді;
- қауыпты — жұмыс істеп жатқанда қателіктерге әкеп соғатын вирустар;
- өте қауыпты, программалардың жоюлуына әкеп соғатын; деректерді жою; компьютерге қажет ететін ақпараттарды жою, және т.б.

Алгоритм ерекшелігіне байланысты вирустарды келесі тобтарға бөлуге болады:

- компаньон-вирустар (companion) —COM кенейткіші бар;
- «құрт»-вирустар (worm) — компаньон-вирустар варианті;

- желілік (құрттар)
- «паразиттік»
- «студенттық»
- «стелс»-вирустар (көрінбейтін вирустар, stealth)
- «полиморфик»-вирустар (өз бетінше шифрлейтін немесе призрак-вирустар, polymorphic)
- «макровирустар»

Бақылау сұрақтар:

1. Цифрлық қол қою дегеніміз не?
2. ақпаратты жіберу кезінде не сақталу қажет?
3. ақпараттың нақтылығы дегеніміз не?
4. Жіберушінің іс әрекеті қандай?
5. Қабылданушының іс әрекеті қандай?
6. Компьютерлік жүйенің қорғау ең арзан әдісі не?
7. Ақпараттық жүйені эксплуатациялау кезінде қай есеп ең негізгі болып саналады? Ол немен қамтамасыз етеді?
8. Архивтау программасын қандай категорияларға болуге болады?
9. Стенография, Криптография, Шифр терминдеріне анықтама беріңіз.
10. компьютерлік вирус дегеніміз не?

Пайдаланылатын әдебиет: [2], 35-166 бет.

Дәріс №10. Тақырыбы: Аппаратты шифрлау

Жоспар:

1. Аппараттық шифрлау.
2. аппараттық шифраторлар құрылымы.
3. Шифропроцессорлар.
4. программдық интерфейстын жұмыс принциптері
5. Кілттік схемалар.
6. Электрондық үй.
7. техникалық іске асыру нұсқалары.
8. Техникалық сиппаттамалар.

Мақсаты: Аппаратты шифрлеуді үйрету. Аппараттық шифратордың құрылымымен таныстыру

Кілттік сөздер: Шифропроцессор, криптография, шифратор

Криптографичлық қорғау тәсілі бойынша көбінесе арнайы аппараттық құралдар түрінде іске асырылған. Бұл құралдар линия байланысына құрылып,

барлық берілетін мұндағы ақпараттарды шифрлейді. Аппараттық шифрлауды бағдарламада қолдануының бірнеше себебі бар.
Біріншіден, аппараттық шифрлеудін жылдамдығы өте үлкен. Криптографиялық алгоритмдер үлкен көлемді күрделі операциялардан тұрады,. Осы операцияларды шешу үшін қазіргі универсальды компьютерлер ынғайлы емес. Арнай құралдар оларды өте тез орындайды.

Екіншіден, аппаратураны сырттан тиіскендерден қорғау өте оңай. Ал компьютерде орындалған бағдарлама қорғансыз. Жау көрінбейтіндей қылып өзгерістер енгізе алады, криптографиялық алгоритмнің табандылығын төмендету үшін, оны ешкім байқамай қалуы мүмкін.Ал аппаратураны арнайы контейнерларға орналастырады, олардың функциялау схемасын өзгертуге мүмкіндік бермейді. Чиптін сыртынан химиялық құрамынан түратын арнайы қосқынды құяды, сонын нәтежесінде осы чиптін қорғау беттін жоямын деген кезде оның ішкі логикалық құрылымның өздігімен жойылады.

Үшіншіден, Шифрлеу аппараты орнатуы өте оңай. Көбінесе компьютерлік құрылғы артық болған кезде шифрлеуді қажет етеді. Телефондар, факсимальды аппараттар және модемдерді кішкене компьютерлерге кірістіргеннен аппараттық шифрлеумен жабдықтандыру өте арзан .
Компьютерлерде де арнайы шифрлеу құрылғысын орнатуы қиындық тудырмайды.

Қазірге нарықта ақпаратты шифрлеу аппаратық құрылғының үш түрін ұсынады — өздік жетерлік шифровальдық модулдер (олар кілттермен барлық жұмыстарды өздігімен орындайды), байланыс каналдарында шифрлеу блогі және ДК орнату үшін шифровальдық кенейткіш тақташасы.Бірінші және екінші типті құрылғыларды көбінесе тар мамандырылған, сондықтан оны сатып алған кезде оның мүмкіншілігін білу керек.

1.Аппараттық шифратордың құрылымы

ДК үшін аппараттық шифратор классикалық варианті –ДК аналық тақшасында PCI слоты енгізілген кенейтілу тақша. АНКАД фирмасы құрастырған КРИПТОН-9 криптографиялық деректерді қорғау құрылғысын (УКЗД) мысал ретінде құрылымымен аппараттын жұмыс принципін қарастырайық. (рис. 5.1).

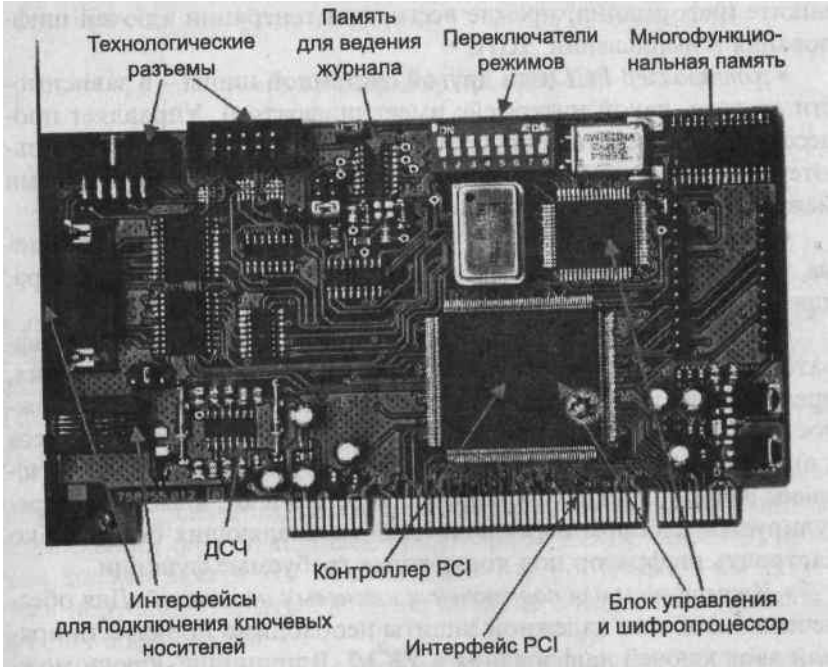


Рис. 5.1. Шифратор КРИПТОН-9

Кәдімгі аппараттық шифратор келесі бөліктерден тұрады [28, 43]:

• *Басқару блогі /Блок управления/. Барлық шифрратордың жұмыс басқаруының негізгі модулі*. Көбінесе микроконтроллер базасында іске асырылады.

• *Шифропроцессор. Арнайы* микросхема немесе логиканы бағдарламалайтын микросхемаға (PLD – Programmable Logic Device) ұқсайды. Шифропроцессор кілт арқылы деректерді шифрлайды, шифропроцессорлар бірнеше болу мүмкін – бір бірін бақылау үшін. сур. 5.1 көрсетіліп тұрған КРИПТОН-9 бір шифропроцессордан тұрады PLD базасында.

- *Кездесоқ сандардың аппараттық датчигі.* Бұл статистикалық кездесоқ және молжаға болмайтын сигнал беретін әрі қарай цифрлық формаға өзгереді құрылғы.
- *PCI контроллері*. Компьютердің аналық тақшасымен шифратордың сәйкестігін басқаратын процесс.
- *Жадтын микросхемалары*. Микроконтроллердін программалық қамтамасыздандыруын сақтауоперация журналына деректер жазу және т.б. мақсаттар үшін энергобайланыссыз жад қажет етеді.
- *Жұмыс режимін ауыстырып қосқыш*. Көбінесе аппараттық шифраторлар шифрлау функцияларының орындалуын шектемейді, қолданушыға көптеген қосымша мүмкіншіліктерді қамтамасыз етеді.
- *Кілттік жетектерді қосуға арналған интерфейстер.* УКЗД-да шифрлеу кілтін тікелей енгізу қажет, сенімді қорғауды қамтамасыз ету үшін.

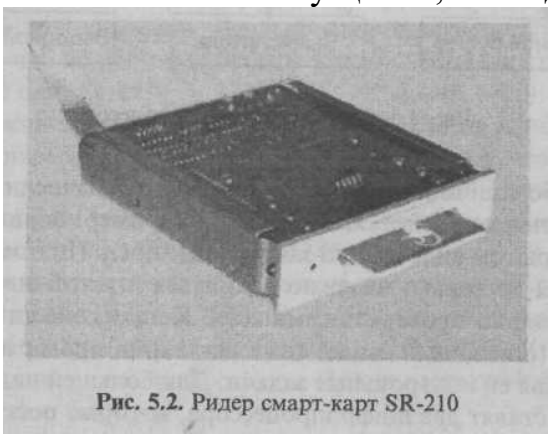


Рис. 5.2. Ридер смарт-карт SR-210

3. Шифропроцессорлар.

Шифропроцессордың құрылымы. Шифропроцессор бірнеше құрылым бірлігінен тұрады (сурет. 5.3) [28, 43]:

- *Есептеуіш* - регистрлер, сумматорлар, блоктар жиыны, бір бірімен деректер жіберу шина арқылы байланысқан. Криптографиялық әрекеттерді максимальді тез арада орындайды.
- *Басқару блогі.* Шын мәнінде бұл аппаратты реализацияланған программа, егер бір себептен программа өзгерсе, онда есептеуіш дұрыс жұмыс істемейді.

Рис. 5.3. Структура шифропроцессора

Сондықтан программа өз өзін тексеріп отыру қажет.

• Е*нгізу/шығару буфері* шифропроцессордың өнімділігін арттыру үшін қажет – деректердің бірінші болгі шифрлеп жатқанда, келесінікі жүктеліп жатады. Шығу кезінде дәл осылай қайталанады. Мұндай деректерді жіберу конвейерлігі шифрлау жылдамдығын көбейтеді.

Жылдамдығының сипаттамасы. Шифрлау жылдамдығы шифратор ең негізгі сипаттамасы болып келеді. Қазірғы кездегі УКЗД ДК орталық процессорын алаңдырмай деректерді шифрлейді.

Бір компьютерде бірнеше УКЗД болу мүмкін, мысалы, криптографикалық маршрутизаторда: желі арқылы жіберілген деректерді шифрлейді,ал екіншісі – қабылданғандарды шифрден шешіп алады.

Деректерді үздіксіз өндеу жылдамдығы - аппаратные шифраторларды бағалайтын ең негізгі параметр. Ол бір секундта мегабайтпен өлшенеді және шифрлау алгоритмнің күрделігіне байланысты. Ең онайы шифратордың жылдамдығын формула арқылы бағалаған:

$V=F*K/n$,

мұнда F – тактілік жиілік;

K – стандартты шифрлау болгінің размері;

n – жиілік саны.

4. Программдық интерфейстын жұмыс принциптері

Аппараттық шифраторларда негізгі екі жұмыс режимі бар:

Алғашқы жүктелу режимі. Компьютер енгізіліп жатқанда ДК-ден BIOS сұраныс жасаған жағдайда, шифратор басқаруды өз қолына алып оның жадында жатқан командаларды орындайды.

Орындалу операцияның режимі. Шифратор алғашқы жүктелу аяқталған соң ДК-ден шифрлау операциясының орындалуын күтеді. Әр шифратор бұл режимде келесілерді орындау қажет:

- Шифропроцессорде жүктелуді жүзеге асыру және шифрлеу кілтін одан түсіру;
- Деректер мен кілттер үшін имитоприставканы өлшеу;
- Сұраныс бойынша кездесоқ сандарды беру.

Программа УКЗД-ға қарағанда әр бір команда бірнеше деңгейден өтеді:

- Қосымша деңгей;
- Қосымша және УКЗД драйверлер арасындағы интерфейсті қамтамасыз ету деңгейі;
- ОЖ ядро деңгейі - УКЗД драйвері;
- Аппараттық деңгей - УКЗД.

Кілттік схемалар.

Аппараттық шифраторлар бірнеше шифрлау кілттер деңгейін қолдану қажет, идеалды варианты- үш өлшемді кілттер иерархиясы:

1. Сеанстық, файлдық және пакеттік кілттер.
2. Көп уақытты қолданушылық немесе желілік кілттер.
3. Негізгі кілттер.

Әр кілттер деңгейлеріне шифропроцессор жадындағы кілттік ұяшық сәйкес келеді..

Үш кілттік схеманың қажеттілігін файлды шифрлау процесіндегі мысалда қарастырайық. Ол бірнеше қадам бойынша орындалады:

1 қадам. ДСЧ шифраторға кездесоқ сан алуға сұраныс шығады, берілген сан кілттік ұя№ 1 жүктеледі (файлдық кілт).

2 қадам. Файлдың мазмұны файлдық кілтте шифрленеді, шифрленген ақпаратты сақтау үшін жаңа файл құрылады.

3 қадам. Көп мерзімді кілт қолданушыдан сұранып кілттік ұя № 2 жүктеледі.

4 қадам. шифрден шешілген көп мерзімді кілтте файлдық кілт шифрленеді. Содан соң шифратордан шығып шифрленген файлдын аталуына жазылады.

5 қадам. Файлды шифрден шешу кезінде қолданушының көп мерзімді кілті арқылы файлдық кілт шифрден шешіледі, сосын керек ақпарат шешіледі.

Электрондық үй.

«электрондық үй» функциясы компьютерді бекітілмеген енуден қорғауды ұйымдастырады, сонымен қатар ОЖ файлдардың толықтығын бақылауды қамтамасыз етеді.

Әр шифратордың жадында электрондық үй режимінде жұмыс істейтін келесі ақпараттар болу мүмкін:

- Қолданушылардың тізімі;
- Әр қайсысына хэш-мағынасы бар бақылаушы файлдар тізімі;
- журнал, компьютерге кіруге тырысқандардың тізімі.

Алғашқы жүктеу режимінде шифратордың электрондық үйі келесілерді орындайды:

• Қолданушыдан ақпараттың аутентификациясынсұрайды;

- Егер де бір бақылаушы файлдың толықтығы бұзылса, онда компьютердің жүктелуі блокировкаланып қалады, ал шифратор арнайы жұмыс режиміне көшеді;
- Шифратор өз журналында кіруді жазып қойып компьютерге басқаруды қайтарады, сосын ОЖ жүктелуін әрі қарай жалғастырады;

Электрондық үйдің кемшілігі – оны компьютерден алып тастауға болады.

техникалық іске асыру нұсқалары.

Аппараттық шифраторлардың бағдарламалықпен салыстырғанда бір кемшілігі – бұл жоғарғы құны. Бірақ қолданушы қымбатырақ алғанмен ақпаратты жоғарғы сапалығымен қорғай алады. Аппараттық шифраторлар көп мүмкіншілігі бар:

- Шифрлаудың аппараттық алгоритм реализациясы алгоритмнің өзін өзгертпеуге кепіл болады;
- аппараттық ДСЧ қолдануы абсолютті кездесоқ сандарға кепіл болады.
- аппараттық шифратор шифропроцессорге тікелей шифрлау кілтін жүктелуге мүмкіндік береді;
- аппараттық шифратор компьютерге шек қою жүйесін құру мүмкіндігін береді;
- аппараттық шифрлау арнайы шифропроцессор арқылы компьютердің орталық процессорын жүктейді.

Техникалық сиппаттамалар.

Аппараттық шифраторлардың техникалық сипаттамалары:

- Шифрлау алгоритмін қолданылуыжәне шифрлеу кілт ұйқастығы;
- Шифрлау жылдамдығы;
- Шифратордың кілттік жүйе деңгейінің мөлшері;
- интерфейс (ISA/PCI/USB және т.б.);
- кілттік жетектерді қолдайтын жиын;
- функционалдық электрондық үйдің бары;
- Әр түрлі ОЖ үшін шифратор драйверлерінің бары;
- Программалық қамтамасыз етудің бары.

Бекіту сұрақтары:

1. Аппараттық шифрлау дегеніміз не?
2. аппараттық шифраторлар құрылымы.
3. Шифропроцессорлар не үшін қажет?.
4. программдық интерфейстын жұмыс принциптері
4. Кілттік схемалардың атқаратын қызметі?
5. Электрондық үй дегеніміз не?

6. техникалық іске асыру нұсқалары.
7. Техникалық сиппаттамалар.

Пайдаланылатын әдебиет: [3], 78 бет.

Дәріс №11. Тақырыбы: Программалық шифрлау

Жоспар:
1. *PGP криптографиялық жүйе*
2. PGP бағдарламмасын құру және қолдану.
3. PGP қалай жұмыс істейді.
4. Кілттер. Цифрлық қол қою
5. Шифрленген хабарды қалай жіберуге болады.
6. Хабарды шешіп алу

Мақсаты: Алгоритмдерді программалық іске асыру. PGP криптографиялық жүйемен жұмыс жасауды үйрету
Кілттік сөздер: PGP, RSA, DES, хэш-функция

PGP криптографиялық жүйе

Pretty Good Privacy (PGP) криптожүйе Phil's Pretty Good Software фирмасы шығарған және MS-DOS, Windows, Unix, VAX/VMS және т.б. ОЖ үшін үлкен құпия дәрежелі криптографиялық жүйе болып табылады. PGP қолданушыға файлдармен, хабарлармен алмасға мүмкіндік береді және құпиялық функцияларын, нақтылығын тексеруге, сақтауға, жұмыс істеуге өте тиімді.

PGP криптожүйесі RSA криптографиялық жүйесін ашық кілтімен тиімді қолдануды өзіне біріктіреді. PGP ашық кілт функциясын тез орындайды. PGP — бұл барлығы үшін ашық кілтті криптографиялық жүйе.

PGP модеммен жұмыс істеу мүмкіншілігі жоқ. Ол үшін бөлек бағдарламманы қамттамасыз ету керек.

Стандартты криптографиялық жүйелерде мысалы, US Federal Data Encryption Standart (DES), бір кілтті шифрлеу кезінде және шешіп алу кезінде қолданады және кілт құпия каналдар арқылы жіберіледі. Бұл жағдай тиімді емес.

Криптографиялық жүйеде ашық кілтімен әр адамда бір бірімен байланысты екі кілті болады: жарияланған ашық кілт және құпия кілт. Осы адамдардың әр қайсысында шифрды шешіп алатын коды болады. Мұндай протокол арнайы байланыс каналдарсыз құпияны қамтамасыз етеді.

Сонымен қатар, хабардың нақтылығы анықталады. Электрондық қол қою арқылы қабылдап алған хабар шынымен жіберілген адамдікі ме екендігін анықтайды.

Осы екі процессті құпиялықты және нақтылықты анықтау үшін біріктіруге болады.

Ашық кілттер «кілттер сертификаттар» түрінде сақталады. Ашық кілттер сертификаттары ашық кілттерден, ал құпия кілттер сертификаттары – құпия

кілттерден тұрады.Әр құпия кілт бөлек парольмен шифрленеді. Кілттер файлы, немесе кілттер каталогі («кольцо с ключами» — «keyring») бір немесе бірнеше осындай сертификаттардан тұрады.
PGP қол қоюды құру үшін «хабарлар дайджестлерін» қолданады. Хабар дайджест — бұл криптографиялық 128-биттік,бірқатарлы, алымды хэш-функция. дайджест екі бірдей дайджестпен хабарды құруға мүмкіндік бермейді. Хабар дайджестсі электрондық қол құру үшін құпия кілтпен шифрленеді.
Қабылдаушының бағдарламалық қамтамасыз ету автоматты түрде құпия кілттер каталогінаң шифрден шешіп алу құпия кілтің іздейді.

Осы екі кілттер каталог түрлері ашық және құпия кілттермен жұмыс істеудін және сақтаудың негізгі әдісі болып табылады.

PGP бағдарламмасын құру және қолдану.
PGP (Pretty Good Privacy)- жоғарғы дәрежелі беріктігімен криптографиялық /шифровальдық/ бағдарламма. Ол қолданушыларға толық конфиденциальды режимінде электрондық түрде ақпараттар алмасуды қамтамасыз етеді.
Бұл бағдарламманың негізгі мүмкіншілігі қолданушылар шифрленген хабарды бір біріне жіберген кезде құпия кілтпен алмасу қажет емес. өйткені бұл бағдарламма жаңа жұмыс принципі бойынша құрылған- публикалық криптография немесе ашық кілтпен алмасу, мұнда қолданушылар «Интернет»арқылы ашық түрде бір біріне кілттерін жібере алады, бірақ басқа адамдар олардың хабарын оқып қоя алмайды.
PGP бағдарламмасы принципінде екі бір біріне байланысқан кілттер қолданылады: ашық және жабық кілттер. Жабық кілтке тек сізге қол жеткірлікті, ал ашық кілтінізді басқаларға қол жетерлік. Бұл бағдарламманың тағы бір мүмкіншілігі , ол тегін, және әрбір қолданушы оныИнтернеттен көшіріп ала алады.
PGP шифрленген хабарды тек қабылдаушы ғана шеше алады. PGP Филипп Циммерман құрушы бағдарламманың кодың ашық түрде жариялаған, оны бірнеше специалисттар зерттеп осы бағдарламмма жайлы кемшіліктер байқаған жоқ.

PGP қалай жұмыс істейді.
Қолданушы PGP арқылы хабарды шифрлеген кезде бағдарламма ең бірінші мәтінді сығыстырады, ол модем арқылы жіберу уақытың қысқарту және шифрлеудін сенімділігін тудыру үшін қолданылады. Содаң сон PGPсессиялық кілтті генерациялайды.
Деректер шифрленген сон сессиялық кілт хабар алушының бәріне белгілі кілті арқылы шифрленеді және шифрленген мәтін мен бірге жіберіледі.
Шифрден шешіп алу қайта кезегімен өтеді. Хабар қабылдаушының PGP бағдарламмасы қабылдаушының жабық кілтін қолданады, сессиялық кілтті шешу үшін, содаң кейін бағдарламма шифрленген мәтінді шеше бастайды.

Кілттер

Кілт – бұл мәтінді шифрлеу үшін криптографиялық алгоритмда қолданылатын сан. Ереже бойынша кілттер – бұл өте көп сандар. Кілт саны битпен өлшенеді. 1024 битпен берілген сан – өте көп. Публикалық криптографияда кілт үлкен болған сайын, оны басқа біреулер шешіп алу ға қиынға түседі. Бірақ егер компьютер өте қуатты болса шіп алуға мүмкін. Сондықтан кілттердің көлемі сәйкес келетін болған қажет.

Кілттер сіздің компьютерінізде қатты дискта сақталып тұрады, екі файл түрінде: біреуі ашық кілт үшін, екіншісі – жабық кілт үшін. Бұл файлдарды «сақина» (keyrings) деп атайды.

Цифрлық қол қою

Публикалық криптографияның тағы бір мүмкіншілігі ол цифрлық қол қою мүмкіндігі. Цифрлық қол қою қолмен қол қою функцияларын орындайды, бірақ цифрлық қол қоюды басқа біреу өтірік қоя алмайды, ал қолмен қол қоюды өтірік біреу ұқсастырып қоя солу мүмкін.

ХЭШ-ФУНКЦИЯСЫ

PGP тағы бір мүмкіншілігі, ол «хэш-функция» қолданылады , ол арқылы егер ақпарат бір битке болса да өзгертілсе, онда «хэш-функция» нәтежесе басқа болып шығады. «хэш-функции» және жабық кілт арқылы «қол қою» құрылады, сосын ол бағдарламма арқылы мәтінмен жіберіледі. Қабылдаушы PGP арқылы негізгі мәтінді және қол қоюды тексере алады.

ПАРОЛЬДЫҚ ФРАЗА

Парольдық фраза – бұл бірнеше сөз жиыны, ол теория жағынаң парольдық сөзден қарағанда, өте тиімді болып саналады. Паральдық фразалар өте үзың және күрделі болу керек, сонымен қатар онда цифрлық белгілермен пунктуация белгілері болу қажет. Парольдық фразаны ұмытып қалмайтындай және үшінші адам оны шешіп алмайтындай болу қажет.

Шифрленген хабарды қалай жіберуге болады.

Сіздін корреспондентініздін ашық кілтін компьютерінізге енгізгеннен соң қабылдаушыға хабарды келесі түрде жіберуге болады:

почталық программада Outlook Express хабар құрастырамыз.

Хабар жіберуге дайын болған соң Outlook Express панеліндегі жасыл конверт бейнеге басамыз немесе tools мәзірінде encrypt using PGP тандаймыз және file мәзірінен send later командасын орындаймыз.

Бұл кезде экран бетінде PGP программасының терезесі пайда болады Recipient selection атауымен, мұнда корреспонденттін ашық кілтін тауып, тышқанмен басып, O'K батырмасын басу қажет.

Осыдан кейін программа автоматты түрде мәтінді шифрлейді және оны outbox бумасына кіргізеді. Одан кейін интернетке кіріп хабарды жіберуге болады.

Хабарды шешіп алу
Бірінші әдіс
Қабылданған хабарды ашып, Outlook Express панеліндегі екінші сол жақтағы белгіні басамыз, немесе PGP мәзірінен decrypt message командасын орындаймыз. Бірнеше секундтан соң хабар шифрдаң шешіледі және терезеде пайда болады.
Екінші әдіс
Тағы бір әдіс бар, PGP қолданып Outlook Express арқылы шифрлеуден қарағанда күрделі. Бұл әдісті PGP-ді Outlook Exress программасын қолданған кезде енгізе алмаған жағдайда.
Outlook Express арқылы хабар құрастырамыз, сосын оны edit - select all командасы арқылы белгілеп Windows буферіне copy командасымен көшіреміз.
Осыдан кейін тышқанды PGP белгісіне әкеліп encrypt clipboard командасын орындаймыз.
PGP сұхбат терезесі пайда болады key selection dialog атаумен.
Осы терезеден корреспонденттін ашық кілтін тандап алып тышқанмен екі рет басу қажет, сол кезде ол астында пайда болады, содан кейін O'K батырмасын басамыз, бұл кезде программа clipboard барлық мазмұнын шифрлейді.
Сосын хабарға кіріп тышқанды хабардын жолына қойып оң жақ батырманы басып paste командасын орындаймыз. Осының нәтежесінде шифрленген clipboard мазмұны алдынғы хабарды ауыстырады, сонымен шифрлеу процессі аяқталады. Енді хабарды жіберуге болады.
Хабарды шифрден шешіп алуда осылайша болады: яғни қабылданған мәтінді белгілеп Windows буфері арқылы clipboard-қа көшіреміз, PGP мәзіріне кіріп decrypt and verify clipboard командасын тандаймыз. PGPпрограммасының терезесі пайда болады, онда пароль енгізу қажет біздін алдымызға шифрден шешілген хабар шығады.
Үшінші әдіс:
Мәтінді бір редакторда құрып, мысалы блокнотта, және оны файл түрінде сақтаймыз. Содан кейін проводниктен файлды белгілесек, PGPатты тағы бір команда пайда болады. Тышқанмен PGP бассақ біз төрт командадан тұратын ашылған мәзірді көреміз:
encrypt
sign
encrypt and sign
wipe
бірінші команданы бассақ корресподенттін ашық кілтін тандау сұхбаты пайда болады, кілтті тандап ОК басамыз, пароль енгізсек файл шифрленеді. Осыдан кейін PGP мәзірінде тағы бір команданы орындау керек: wipe (оригиналды файлды жою).
Осы операциядан кейін файлдын атауы сол қалыпында қалады, бірақ кенейтілу түрі өзгеріледі <*.pgp> . Осыдан кейін бұл файлды хабармен бірге жіберуге болады.

PGP бағдарламасыны кемшілігі: хабарды өзінін корреспондентінің ашық кілтімен шифрлаған кезде, жіберуші оны сосын оқып шыға алмайды.
PGP «настройкасында» опция бар, ол арқылы хабарды шифрлаған сон қайта оқу мумкіндігі бар (архивтан алып оқу).
Ол үшін тышқанмен PGP символын басып PGP preferences командасын орындау , сосын General кіріп Always encrypt to default key командасынын алдына белгі қою. Сонымен бірге PGP keys кіріп, тышқан арқылы өз кілтінді тандап, keys мәзіріне кіріп set as default key командасын орындау қажет. Бұл жерде парольдық фразаны өзгертуге болады:
Тышқанмен өз кілтінді тандап key properties командасынан change passphrase орындау керек және парольдық фразаны өзгерту керек.

Бекіту сұрақтары:

1. PGP криптографиялық жүйе мүмкіншілігі
2. PGP бағдарламмасын құру және қолдану жолдары.
3. PGP қалай жұмыс істейді.
4. Кілттер. Цифрлық қол қою тәсілдері.
5. Шифрленген хабарды қалай жіберуге болады.
6. Хабарды шешіп алу жолдары.

Пайдаланылатын әдебиет: [3], 80 бет.

Дәріс №12. Тақырыбы: Кілттермен операциялар жасау
Жоспар:

1. **Кілттермен операциялар жасау жолдары.**
2. **Виртуальды желілерді ұйымдастыру.**
3. **Криптоқорғанысты логикалық дискілерді жасау.**

Мақсаты: Кілттермен операцияларды жасауды үйрету
Кілттік сөздер: NEXT, PGP Disk volume security, Hot keys

1.Инсталляция.

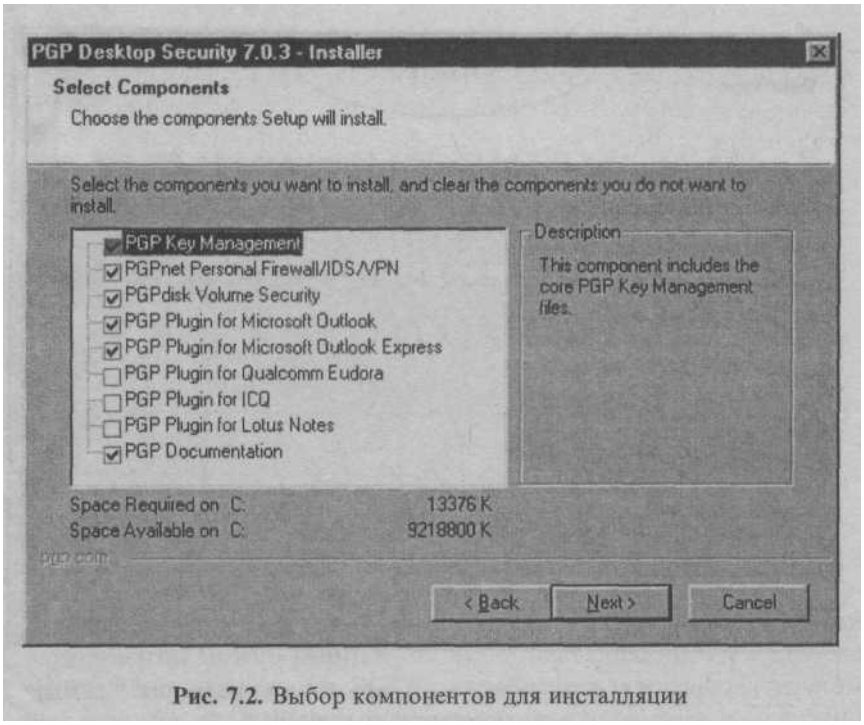


Рис. 7.2. Выбор компонентов для инсталляции

7.0.3. нұсқасынң PGP жүйесінің ертеректе қарастырылған.5.5.3. нұсқасымен салыстырғанда қосымша компьютерге ие.Бұл компоненттерді инсталляция кезеңінде таңдауға болады.PGP DESKTOP SECURITY папкасын Setup.exe файлын жібергенде және содан кейін “NEXT” нүктесін басқанда адымдардын біреуіде қолданушыда генерацияланған қос кілт барлығы туралы сұрағы бар. PGP DESKTOP SECURITY 7.0.3-Installer терезесі пайда болады.7.1-сурет.

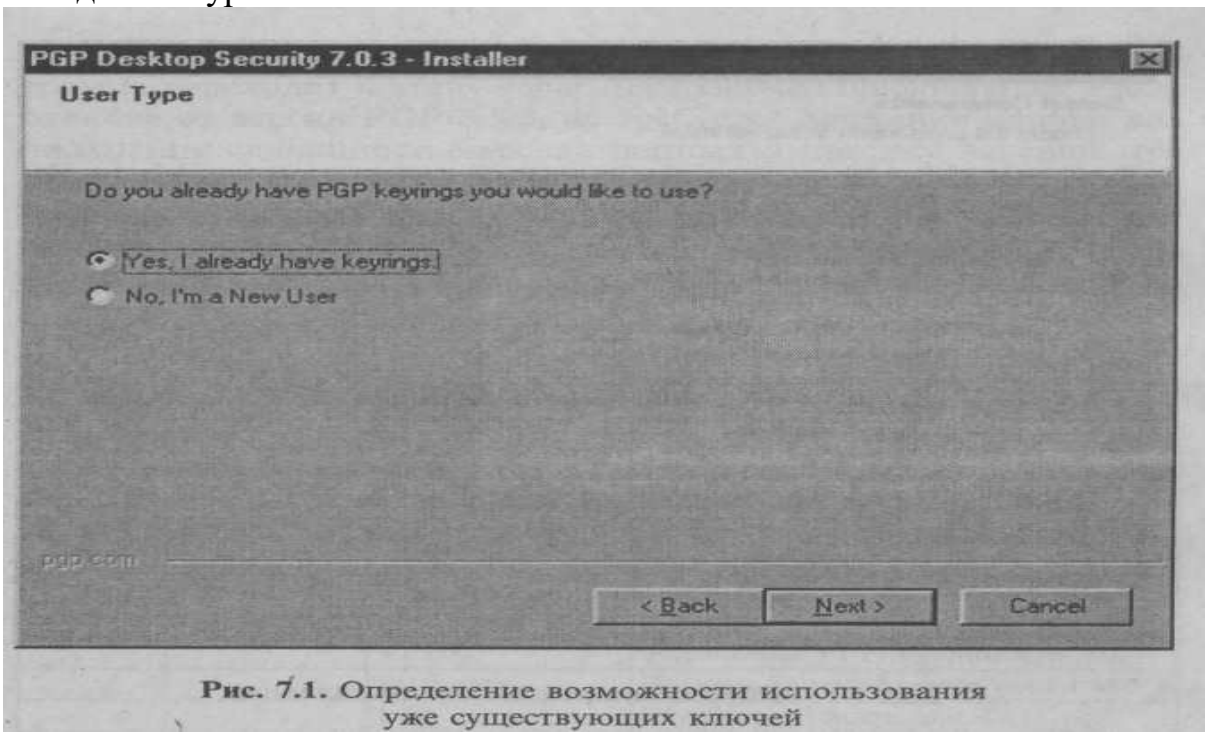


Рис. 7.1. Определение возможности использования уже существующих ключей

Егер кілттер әлі жасалмасы онда тандау батырмасына “NO,I’m a nem user” тышқанмен нүкте қою қажет және әрі қарай “NEXT батырмасын басу керек. PGP DESKTOP SECURITY 7.0.3- Installer бағдарламасы инсталляция папкасы көрсетілген келесі терезеге көшеді.”BROWSE” БАТЫРМАСЫНА

батырмасына баса отырып оған басқа орын таңдауға болады. Одан әрі инсталляция бағдарламасы тиісті жалаушаларды орнатумен инсталляцияланатын компоненттерді таңдауды ұсынады.7.2-сурет

PGP Кілттері менеджерсіз жұмыс істемейтіндіктен PDP KEY MANADEMENT жалаушасы міндетті түрде орнтылуы тиіс.

PGPnet Personal Firewal /IDS /IPN компоненті виртуалды желілерді ұйымдастыру және оларды кілттеу рұхсатсыз қол жеткізулерден қорғау үшін мәліметтер мен трафиктерді филтрациялау бағдарламасын қосады.

PGP PLUDINFOR MICROSOFT OUTLOOK,PGP PLUDINFOR MICROSOFT OUTLOOK EXPRESS,PGP PLUDINFOR QUALCOMM комноненттері үшін жалаушалар.EUDORA PGP PLUDINFOR LOTUS NOTESU PGP PLUDINFOR ICQи тиісті бағдарламалар инсталляцияланған кезде ғана орнатылады. Онда PGP функциялары осы бағдарламадан шақыра алады.

PGP Disk volume security жалаушаларын орнату қатты дисклердің көлемінің бөлігін қолдана отырып кілттенген дискіні инсталляциялау кезінде клавиатурадан терілетін парольді теру арқылы алуға болады.

PGP Dokumentation жалаушасын орнату инсталлциялаудан кейін бағдарламадын сипаттамасын және дәлдеу жөніндегі нұсқауларды көру мүмкіндігін береді.

Компоненттерді тандағаннан және “NEXT батырмасын басқаннан кейін бағдарламалар файлы көшіріледі және виртуальды желілерді құрастыру кезінде қолданылатын адепьарлардан тізімі орнатылатын келесі терезе пайда болады.Жойылған жұмыстар үшін виртуальді желілер технологиясын пайдалана отырып жойылған рұхсатты бақылаушы жалаушасы іске қосылады,ал локальдық желі үшін Компьютердің желілік картасын білдіретін жасауша іске қосылады. Бұдан кейін ағдарлама кілттерді генерациялау кезеңіні көшеді, онда PGP 5.5.3 нұсқасына қарағанда деректкрдің келдейсоқ массивін алу үшін тышқанның қимылы қажет емес және үрдіс компьютерді қайта жүктеу ұсынысымен аяқталады.

Қайта жүктегеннен кейін жұмыс столының тапсырмаларпанелінің оң жағында Щ белгісі пайда болады. Тышқанмен белгіге басу кезінде бағдарламаның негізгіməзірі пайда болады, оның құрамы 7.1 кестеде көрсетілген.

Көптеген операциялар және бұйрықтар алдыңғы нұсқадағы PGP бағдарламасы жұмыстарының сипаттамасы бойынша таныс. Алдында айтылғандардан басқа WINDOWS белсенді терезесі бар жұмыстың түсінікті функциясы қосылған. (негізгі мәзірдің 9 тармағы бар)

5.5.3 PGP нұсқасынан негізгі ерекшелігі виртуальдық желілерді ұйымдастыру мүмкіндігі (VPN) мен шифрланған логикалықдискілерді жасаудан тұрады. С.Д.Е үш копьютерден тұратын локальдық желіде виртуальдық желіні (VPN) С және Д компьютерлерінің арасында жасауға болады. Оларға VPN құрғаннан кейін Е компьютерінің желілік қоршауында С және Д компьютерлерінің жалғану бейнелері жоғалып кетпейді, бірақ С және Д компьютерінің арасында берілетін

Мәліметтерді алу PGP-шифрларды қолдануға байланысты емес .*Е* комрьютері *С* және *Д* компьютерлерінен жекелеуге (ошаулар) /*PX/ SPX* желілік хаттамасын қолдану есебінен қол жеткізіледі , онда *С* және *Д* компьютерлерінде *TCP/ IP* хаттамасы қолданылады . *С* және *Д* компьютерлеріне *Е* компьютері тарапынан қол жеткізу жабық болады және *С* және *Д* компьютерлеріндегі желілік қоршауында *С* және *Д* белгілері болады , бірақ *Е* белгісі болмайды .(7.5-сурет). Майда *Е* компьютерінде қолданылатын хаттамалар тізіміне *TCP/ IP* хаттамасын қосу кезінде *Е* компьютерінің желілік қоршауында *С* және *Д* белгілері пайда болады , яғн и *С* және *Д* компьютерлеріне *Е* компьюеріне қол жеткізе алады .*С* және *Д* компьютерлерінің арасында *VPN* жалғауды ұйымдастыру үшін *PGP* бағдарламасын инсталяциялаудан кейін әр компьютерде қос кілттерді (ашық және жабық) генерациялау қажет , және бұдан басқа ашық қою керек , атап айтқанда *С* компьютеріне ашық кілтті *Д* компьютерінен ашып және *С* компьютерінің жабық кілтін пайдалана отырып осы ашық кілтке *ЭЦП* қою қажет ;осылайша *Д* компьютеріне *С* компьютерінен ашық кілтті пайдалана отырып *ЭЦП* қою қажет Одан әрі

Компьютердің IP-адресі Windows-тың бақылау панелінің желілер конфигурациясы бөлімінде бекітіледі.Бұл жерде конфигурация ілмегінде тышқанмен TCP9PPGPnet VPN Adapter жолы ерекшеленеді және “СВОЙСТВА”батырмасы басылады.Одан әрі Свойства TCP/IP мәзірінде IP-адрес ілмесі таңдалар және “анық IP-адресті көрсету”радиобатырмасында нүкте орнатылады.

Толтырғаннан кейін IP-адреске қол жеткізуге болады және жеке желілерге арналған адресс диапазонынан таңдалады(10.0.0.0-дан 10.255.255.255-ке дейін 172.11116.0.0-172.31.255.255.192.168.0.0-192.168.255.255)Бұл жағдайда 255.255.255.0.Маскасы (7.6сурет) 192.168.0.17. адресі таңдалған.Д компьютеріне ұқсас операциялар өткізіледі.

IP-адрес сол(192.168.0.44). адрес диапазонынан таңдалады.Маскасы өзгермейді.Адрес таңдалғаннан кейін компьютердің желілік қондырғысының конфигурациясын аяқтау үшін екі рет “OK”батырмасы басылады.Өзгертулер күшіне ену үшін компьютерді қайта жүктеу қажет.

7.2 Жалғауды орнату.

Қайта жүктеу үрдісінде желілік пароль енгізіледі және бір мезгілде PGP бағдарламасы кілттерді генерациялауда қолданған фразалық пароль терілетін PGP net Logon Pass p hrase терезесін экранға шығарады. (бұл мысалда ол v(DSS)1024)және “OK” батырмасы басылады.Бұдан кейін C және D компьютерінің арасында VPN-жалғамын

Орнату мүмкін болады. Бұл мақсат үшін бағдарламаның негізгі мәзірінің 5 тармағының көмегімен (7.1 кесте)VPN бұйрығы іске қосылады. (VPN – жалғалым қондырғысы)

Д компьютерінде VPN-жалғалымының орнатылған белгісі болып SA(7.8 сурет) бағанындағы жасыл шеңбердің барлығы болып табылады,ол оның пайда болуы негізгі мәзірдің optionsбөлімінде жасау қажет тағы да екі

әрекетпен белгіленген.VPN ілмегінде және Option PGP терезесіне Fnable VPN Connectionалаңында VPN-жалғалымына рұқсат ететін жалауша орнатуды қажет етеді. Бұдан басқа VPN Authentication (PGP Authentication алаңында) ілмегінде аутентификация кілтін «Select Key» батырмасымен (бұл жағдайда ол v) таңдалады, Д компьютері үшін бұл wкілті болады.

Өзгертілген қондырғыларды «OK» батырмасын басу арқылы сақтағаннан кейін «Connect» батырмасы ашық болады, және VPN-жалғалымды орнату мүмкін болады. Енді алмастыру кезінде пайдаланылатын ашық кілтті орнату қалды, ол VPN-жалғалымының қасиетіне жүгіну арқылы жасалады.(«Properties» батырмасы PGPnet терезесінде 7.8 суретінде) .«Properties» батырмасын басу кезінде HostGateway терезесі пайда болады, Remote Authentication алаңында «PGP-Key» радиобатырмасында нүкте қойылады. Бұдан кейін Selest Key (s) қосымша терезе пайда болады, мұнда

Таңдалған ашық кілтті тышқанмен екі рет басу арқылы Таңдалған ашық кілтті тышқанмен екі рет басу арқылы WN-жалғалым кезінде қолданылатын ашық кілт орнатылады. Бұл кілтRemote Authentication алаңында пайда болды.(7-11 суреттер)

Дербес компьютер үшін бұл v ашық кілт болар. Енді С және D екі компьютері олардың арасында VPN-жалғасында орнату үшін дайын PGPnet терезесінде «connect» батырмасы басылады және жалғалым орнатылады, яғни SA бағанында жасыл шеңбер пайда болады. Орнату туралы мәлімет LOG-файлға жазылады, оның мазмұны LOG ілмегіне жүгінгенде ашылады және ол 7-12 суретте келтірілген.

Алдында белгілі болғандай мәліметтерді VPN-жалғалым бойынша беру кезінде оның мазмұнын шифрлаудың тиісті кілтіне ие болғанда ғана білуі мүмкін. Бұл жағдайда бейнелеу үшін бұл мысалда С және D компьютерлері арасында 1.txt мәтінді файл беріледі, оның құрамында 51 дана бірлік болады. (7-13 сурет)

Файлды желі бойынша беру кеззінде С компьютерін D компьютеріне берілетін мәлімет жолдан ұсталынады және талданады.

Бұл операция ақпарат пакеттерін ұстап алу және талдаудың арнайы бағдарламасының көмегімен жүзеге асады, олар сниффер деп аталады. Бұл үшін сниффер бағдарламасы филтрінің жұмыстары параметріне компьютерлердің IP – адресі көрсетіледі, олар өз арасында ақпарат алысқанда барлық ақпараттық пакеттер файлға жазылады. Біздің мысалда бұл С және Д компьютерлері, олардың IP – адрестері (7.14 суретіне) сәйкес 192.168.0.17 және 192.168.0.44.

7.15 – суретте осындай сниффер бағдарламасы жұмысының нәтижесінің фрагменті берілген, мұнда шифрсіз қарапайым беру кезінде сниффермен ұсталынған ақпараттық пакеттердегі 1 TXT файлының мазмұны көрсетілген (SOURCE ADDRESS). 7.15 суретте С компьютерінен Д компьютерінің IP адресі 192.168.0.44 (Dest Address) бойынша 1txt файлы бар №26 пакет берілген, яғни 51 бірлік (көрсетілген аймақ). Егер VPN жалғалым қолданылса

ұсталынған ақпараттық пакеттер ақпаратты шифрлаған күйі сақтайды, ал оларды шифрді шешу кілтін білмесе оқу мүмкін емес.

7.3. Шифрланған логикалық диск жасау.

Ақпараттарды шифрланған күйі сақтау қатты дискіде шифрланған лоикалық дискіні жасау жолымен мүмкін.

Бұл дискінің мазмұны pgd – сы кеңейтілген файл түрінде сақталады да, (мысалы, үнсіздігі бойынша бұл файл New PGP disk Volume, pgd), ол инсталляция кезінде таңдалған Windows файлдық жүйесінде қалыптасады және жазылады. Бұл үшін PGP Disk негізгі мәзірінің 6 тармағы қолданылады, онда New Disk бұйрықтарының көмегімен шығатын мәзірде жаңа шифрланған диск жасалады. Бұл бұйрықты орындағаннан кейін PGP Disk Greation Wizard терезесі ашылады, мұнда Choose a location алаңына PGP Disk файлына жол және Choose a size алаңына диск көлемі таңдалады. (7.16 сурет).

Тышқанмен “Advanced Option” батырмасына басқан кезінде Options терезесінде шифрланған логикалық дискінің әріпі немесе Windows 2000 операциялық жүйесіндегі папка анықталады, онда шифрланған ақпарат сақталады. Осы терезеде шифрланған ақпарат сақталады. Осы терезеде шифрлаудың қолданылатын алгоритмі және файлдық жүйенің форматы таңдалады. Орнатылған Mount it at start up жалаушасы кезінде шифрланған диск операциялық жүйені қосқан кезде қалыптасады. Тышқанмен «Далее» (Әрі қарай) батырмасын басқан кезде шифрланған дискінің қорғану тәсілі не ашық кілттің көмегімен, не паролдік сөздің көмегімен белгіленетін келесі терезе ашылады. (7.17 сурет)

«Далее» басқаннан кейін келесі адымда не ашық кілт таңдалады, не паролдік сөз еңгізіледі. PGP disk Greation Wizard бағдарламасы қорғалған Логикалық дикіні форматтайды және шифрлайды,бұдан кейін үрдіс “готово”(дайын) батырмасын басументаяқталады.Жасалған диск қолданушылар үшін анықтық қасиетіне ие.Желілік жұмыс кезінде оған өзге қолданушылар тарапынан толық әрі тек қана оқу үшін жол ашылады. Қорғаныс үшін ашық кілт дискісін қолдану кезінде шифрленген дискіге жол диалогтік терезеде кілттер жұбын генерациялау кезінде қолданылған парольдік сөзді тергеннен кейін ашылады.Егер екінші тәсілді таңдасақ , онда қолданушының атын көрсетуге және келесі терезеде оның парольдік сөзін көрсетуге болады.Бұл жағдайда дискіге жол дискіні жасау кезінде енгізілген парольдік сөзді диалогтік терезеде көрсеткен кезде ашылады.Осылайша жол кілттер жұбын генерациялағандарға ғана емес барлық қолданушыларға да ашылады.Қолданылған қорғаныс тәсілін негізгі мәзірдің 6 тармағын таңдау кезінде шығатын мәзірдің Edit disk бұйрығының көмегімен көруге болады.Кеңейтілген prd файлы бар папканы таңдау қажет Choose a PGP disk терезесі пайда болады, оны ерекшелеу және тышқанмен “Открыть”(ашу) батырмасын басу керек.Бұл жағдайда ашылған PGP disk EDIT or <New PGP disk Volume> терезеде шифрланған логикалық диск туралы ақпарат

сақталған <New PGP disk Volume>файл v ашық кілттің көмегімен қорғалғандығы туралы ақпарат бар.
Тышқанмен New PGP disk Volume файлын ашқанда оның парольдік сөздердің көмегімен қорғалғандығын көруге болады.Қорғалған дискінің фацлын жоюды (және шифрланған барлық ақпаратты жоғалтуды)тоқтату оның атын өзгертуиен жіне pgd кеңейтуді басқасына,мысалы, txt өзгерту көмегімен мүмкін болады.Бұдан кейін файл басқс папкаға көшіріледі және өрісін біледі.Компьютерді қайта жүктеу кезінде PGP бағдарламасы бұл файлдың табылмауы мүмкін екенін ескертеді,ал егер пайда болған PGP disk терезесінде тышқанмен жоқ батырмасына басса бұл ескерту бұдан әрі қайталанбайды.

7.1.0. Шифрланған ақпарат қайта рұқсатты болу үшін файлдын кеңейтілуін қайта қалыптастыру және тышқанмен екі рет файлға басу керек. Бұдан әрі Enter Passphrase терезесі пайда болады және паролдік сөзді тергеннен кейін логикалық дискілер тізімінде бұрын жазылған ақпарат бар қосымша қорғалған диск пайда болады. Қорғалған диск файлынын дискета көлемін 1,44 Мбайт аспайтын көлемі болса, онда оны компьютерден бөлек дискетада үлкен көлем жеке қатты дискіде қажет болса одан жоғарыда корсетілгендей қосымша қорғалған логикалық диск қалыптастыра отырып сақтауға болады
Қорғалған дискілермен жұмыс параметрлерін қосымша дәлдеу негізгі мәзірдің (Options) 4 тармағында болады.Оған жүгінген кезде экранға PGP Options терезесі шығады, онда PGP disk ілмегінде Allow forcible unmounting of PGPdisk with open files жолдамасының көмегімен Unmount Options алаңында қорғалған дискіні егер осы дискіде қандай-да бір файл ашылған жағдайда жөндеуге (яғни рұқсаттылардың тізімінен алып тастауға) тиым салуға болады.Осы жерде оған жүгіну жоқ кезінде диск автоматты түрде .минут саны белгіленеді.

Дискіге қайта рұқсат алу үшін негізгі мәзірдің 6 тармағының Mount Disk бұйрығына жүгіну қажет.Бұдан әрі осы дискіге тиісті файлды ашып, парольдік сөзді теру қажет.

7.4 Құрылған желіаралық экранның конфигурациялануы.

Personal Firewall ілмегіндегі PGP Options терезесінде желіге клиент немесе сервер ретінде қосылған компьютерге ақпараттар легін шектейтін желіаралық экранның PGP бағдарламасына құрылған жұмыстар параметрі көрсетіледі.(7.21 сурет)

Порттардың рұқсат етілген нөмірлері (25,139 және т.б.) немесе хаттамалар (ICMP,
IPSRC) бойынша барлық лектерді фильтрациялау есебінен ақпараттарды қорғау қарастырылады.

«Custon» радиобастырмадағы нүктені орнату кезінде желіаралық экран жұмысының әртүрлі нұсқасын таңдауға болады. «Protection Levei» радиобастырмасындағы нүктені орнату арқылы қорғаныстың төменгі , орташа және жоғарғы деңгейін таңдауға немесе одан бас тартуға болады.Жұмыстың,қондырғылардың принциптері және желіаралық

экрандарда бағдарламалаудың және аппараттың параметрлер туралы толық ақпаратты [23] табуға болады.

Атакаларды табу функцияларының белсендігі.

Personal IDC ілмегінде файлдарға оларды көшіру, оқу, модификациялау немесе жою мақсатында санкцияланбаған рұқсатты табу режимі қосылған PGP бағдарламасының жұмыс функциясының белсенділігі қарастырылған(7.22 сурет) мұндай килігу мен атакаларды(шабуылдарды) табу үшін арналған бағдарламаларды IDC деп атайды.Бұл бағдарламалар қосымшалардан(СУБД,Webсерверлер және басқалар) алынған ақпараттарды жинайды және талдйды.өткізілген талдаудын еегізінде шабуыл бұғатталады. Бұл (automaticalu Block attackers жалаушасын орнатумен жасалады), шабуыл жайлы электрондық мәлімет алаңда көрсетілген адрес бойынша жіберіле алады, оған Send attack alert by email to жалаушасын орнатқан кезде жол ашылады. Бұдан басқа шабуыл сәтінде егер play sound when attacked жалаушасы орнатылса, дыбыстық белгі беріледі және монитор экранында Flash PG Ptray icon wen attacked жалаушасы орнатылса иконалардың жарқырауы көрінеді.

7.6. Файл қалдықтарын өшіру PGP бағдарламасында негізгі мәзірдің алтыншы тармағында алынатын Free space Wipe бағдарламасының көмегімен файлдардың қалдығын өшіру функциясы қарастырылған (7.1. кестесі). Бағдарламаны жіберу жіберу тышқанмен PGP tools терезесінің функциялары қатарындағы шеткі оң белгіні басу арқылы жүргізіледі. Белгіні басқаннан кейін ашылған терезеде операцияларды жүргізуге және қайталау санына арналған логикалық диск таңдалады.

2 Г байт логикалық диск көлемі және 200м байт бос кеңістік үшін бір рет өту уақыты шамамен 3-минутты құраиды.

7.7 Ыстық пернелердің белгіленуі

PGP option терезесінде Hot keys ілмесінің көмегімен пернетақтада пернелердің әр түрлі комбинациялары орнатылады, оларды басу негізінде PGP бағдарламасының кең таралған түрі орындалады.

Punge passphare ofches жалаушасы парольді кештеу режимін қосатын пернені тағайындау мүмкіндігінбереді.

En crypt curcent wind жалаушасы windovs-тың белсенді терезесіндегі ақпараттарды шифрлеу режимін қосатың пернені анықтаиды.

Sign current windov жалаушасы windows белсенді процесінің ЭЦП режимін қосу пернесін береді

En crypt Sisn current window жалаушасы windows-тың белсенді терезесіндегі құжаттарды шифрлеу және ЭЦП пернесін қосуды белгілеиді.

Dencrypt verify current window жалаушасы ЭЦП құжаттарын тексеру және шифрін шешетің операциаларды қосу пернесін белгілеуге мүмкіндік береді.

Unmount all PGP discs жалаушасы басқанда қорғалған дискінің жөнделуін жүргізетің пернелер комбинациясы белгілеиді.

PGP бағдарламасы осы крипта жүйені бағдарламалық жүзеге асыру нұсқасымен толық сәикес екендігін атау қажет.

Бекіту сұрақтары:

1. Кілттермен операциялар жасау жолдары**.**
2. Виртуальды желілерді ұйымдастыру кезеңдері.
3. Криптоқорғанысты логикалық дискілерді жасау жолдары.
4. PGP криптографиялық жүйе мүмкіншілігі
5. PGP бағдарламмасын құру және қолдану жолдары.
6. PGP қалай жұмыс істейді.
7. Кілттер. Цифрлық қол қою тәсілдері.
8. Шифрленген хабарды қалай жіберуге болады.
9. Хабарды шешіп алу жолдары.

Пайдаланылатын әдебиет: [2], 196 бет.

Дәріс №13. Тақырыбы: Гамма тәсілі арқылы шарт белгілеу

Гаммалау деп белгілі бір заң бойынша ашық деректер үстіне шифрдың гаммасын беттестіру (қосу) процесін түсінеміз. Шифрдың гаммасы - дегеніміз ол белгілі бір алгоритм бойынша ашық деректерді шифрлауға және шифрланған деректерді ашуға арналып жасалған жалған кездейсоқ тізбек.

Шифрлау процесінің мәні мынада: шифрдың гаммасын генерациялау және алынған гамманы бастапқы мәтінге, қайтадан кері аударуға болатындай етіп, мысалы модуль 2 бойынша қосу операциясын пайдалану арқылы беттестіру.

Мына жағдайды атап өтуіміз керек.

Шифрлау алдында ашық деректерді, ұзындығы бірдей, әдетте 64 биттен, $T_0^{(i)}$ блоктарына бөледі. Шифрдың гаммасы осыған ұқсас, ұзындығы $Г_ш^{(i)}$ блоктарынан тұратын тізбектер түрінде құрылады.

Шифрлау теңдеуі мына түрде болады:

$$T_ш^{(i)} = Г_ш^{(i)} + T_0^{(i)}, \; i=1 \ldots M,$$

мұндағы $T_ш^{(i)}$ - мәтін-шифрдың і-ші блогі,

$Г_ш^{(i)}$ - гамма-шифрдың і- ші блогі

$T_0^{(i)}$ - ашық мәтіннің і- ші блогі

М- ашық мәтін блоктарының саны.

Шифрды ашу процесі шифр гаммасын қайтадан генерациялау және осы гамманы шифрланған деректер үстіне беттестіруден тұрады.

Шифрды ашу теңдеуі мына түрде болады:

$$T_0^{(i)} = Г_ш^{(i)} \oplus T_ш^{(i)}.$$

Осындай әдіспен алынған мәтін-шифр, ашуға қиындық тудырады, өйткені оның кілті айнымалы шама. Шын мәнінде шифр гаммасы әр шифрланған блок үшін кездейсоқ түрде өзгеріп тұруы қажет. Егер гамма периоды барлық шифрланған мәтін ұзындығынан көп болса және шифрды бұзушыға бастапқы мәтіннің ешқандай бөлігі белгілі болмаса, онда мұндай шифрды тек кілттердің барлық вариантарын түгел тікелей қарап шығу (перебор) арқылы ғана ашуға

болады. Бұл жағдайда шифрдың криптограммалық тұрақтылығы (криптостойкость) кілт ұзындығымен анықталады.

Жалғанкездейсоқ сандар тізбегін генерациялау әдістері. Гаммалау әдісімен шифрлағанда кілт есебінде биттердің кездейсоқ қатары пайдаланылады. Бұл қатар екілік түрде берілген (мысалы А=00000, В=00001, С=00010 және т.с.с) ашық мәтінмен қосылады. Бұл қосулу екі модулі бойынша биттерді өзара қосу арқылы жүзеге асырылады. Нәтижесінде шифрланған мәтін пайда болады. Күні бұрын болжауға болмайтын ұзындығы үлкен екілік тізбектерді генерациялау классикалық криптографиядағы маңызды проблемалар қатарына жатады. Бұл проблеманы шешу үшін екілік жалғанкездейсоқты тізбектер генераторлары пайдаланылады.

Жалғанкездейсоқ бүтін сандар тізбегін генерациялайтын белгілі процедуралар ішінде ең жиі қолданылатыны сызықты конгруэнтті генератор. Бұл генератор $У_1$, $У_2$, ..., $У_{i-1}$,$У_i$, ... жалғанкездейсоқ сандар тізбегін келесі қатынасты пайдалана отырып құрастырады: $У_i=(a*E_{i-1}+b) \bmod m$, мұнда $У_i$-тізбектің і-ші (ағымдағы) саны; $У_{i-1}$ тізбектің алдыңғы саны; a, b және m – тұрақтылар; m – модулі; a – еселік (коэффицент), b - өсімше; $У_i$ – тудырушы сан (алғашқы мәні).

Бұл теңдеу таңдап алынған a, b, m – параметрлеріне байланысты және m мәніне жете алатын қайталау периодымен Жалғанкездейсоқ сандарды генерациялайды. m модулінің мәні 2^n тең деп немесе қарапайым санға тең деп алынады. b -өсімі m- санымен өзара қарапайым, ал a- коэффиценті тақ сан болуы керек.

Жалғанкездейсоқ сандар тізбектерін генерациялайтын сызықты рекурентті қатынастарға негізделген тәсіл бар. Рекурентті қатынастар және олардың айырым теңдеуін қарастырайық

$$\sum_{j=0}^{k} h_j a_{i+j} = 0, \;\; a_{i+k} = -\sum_{j=0}^{k-1} h_j a_{i+j},$$

мұндағы $h_0 \neq 0, h_k = 1,$ әрбір h_i GF (q) өрісіне жатады.

Тізбектердің мұндай түрі компьютерде оңай іске асырылады. Сонымен қатар егер барлық h_i және a_i - GF(2)өрісінде тек 0 және 1 деген мағынада болса іске асыру тіпті оңай болады.

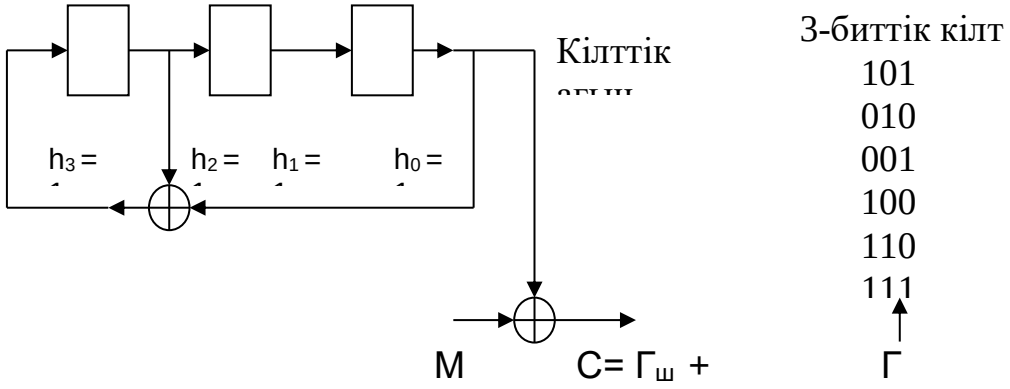
Мысалы ретінде қызықты кері байланысы бар үш разрядтты ығысу регистрін қарастырайық. Ол қарапайым көпмүшеге сәйкес құрылған: $h(x)=x^3+x^2+1$. Мұндағы коэффициенттер мәні $h_3=1$ $h_1=0$ $h_0=1$.

Кілттің мәні 101 болсын. Регистр осы күйден жұмыс істей бастайды. Регистрдің күй-жайы суретте көрсетілген. Регистр өзінің барлық 7 нөл емес күйі арқылы өтіп, қайтадан өзінің 101 қалпына келеді. Бұл – сызықтық кері байланысы бар регистрдің ең ұзын периоды. Мұндай тізбек ығысу регистрінің ең үлкен тізбегі деп аталады. (Maximal Length Shift Register Sequence – MLSRS).

Питерсон мен Уэлдон зерттеуі бойынша кез келген m-бүтін саны үшін периоды (2^m-1)-ге тең m биттік MLSRS – тізбегі болады. Мысалы, егер m=100

болса, онда тізбектің периоды 2^{100}-1 болады да, оны 1Мбит/с жылдамдықпен байланыс жолдарымен жібергенде ол 10^{16} жыл өткенше қайталанбайды.

Біздің мысалымыздағы ығысу регистрінің шығу тізбегі $Г_ш$ (шифр гаммасы) 1010011 тізбегі болып табылады. Ол циклды түрде қайталанып отырады. Бұл тізбекте 4 бірлік пен 3 ноль бар.



4.1 сурет. Кері байланысы бар үшразрядты ығысу

Бақылау сұрақтары:

1. Гамма ұзындығының периоды неге байланысты?
2. Жалғанкездейсоқ сандар тізбегінің генерациялауының криптографиялық беріктілігіне қандай талаптар қойылады?
3. Жылжыту регистрі үшін максималды ұзындық тізбегі қандай тізбек деп аталады?

Пайдаланылатын әдебиет: [1]-18-20., [2] – 73-81 б.

Дәріс №14. Тақырыбы: Internet желісі арқылы шеттелген жабуылдан қорғану мәдістері және тәсілдері

Желіаралық экрандардың қызмет ету ерекшеліктері.

Коммерциялық мақсатта ауқымды желілерді пайдалану және сонымен бірге конфиденциалды сипаттағы мәліметі бар ақпаратты пайдалану осыларды қорғау жүйесін жасауды қажет етеді.

Internet-ке жергілікті немесе корпоративті желіге қосылғанда осы желінің ақпараттық қауіпсіздігін қамтамасыз ету керек.

Internet ауқымды желісі ақпаратты тәуелсіз түрде өзара ауысуға құрылған болатын. Өз идеологиясының ашық болуына байланысты бұзушылар үшін әдеттегі ақпараттың жүйелерге қарағанда көп мүмкіндік береді. Internet арқылы бұзушы мынадай іс-әрекеттер жасай алады:

Кәсіпорынның ішкі желісіне кіріп оның конфиденциалды ақпаратына рұқсатсыз қатынайды;

Кәсіпорын үшін маңызы бар және бағалы ақпаратты көшіру;

Парольдерді, сервер адрестерін, оның мазмұнын алу;

Кәсіпорынның ақпараттық жүйесіне тіркеудегі пайдаланушы атымен кіру.

Желіаралық экрандар (браундмауэр, firewall) ішкі желілерге қауіп тудыратын мәселелерді шешуге мүмкіндік береді. Олар бұзушының ішкі желіге, ақпаратты көшіру, өзгерту, өшіру сияқты немесе осы желідегі компьютерлер жадысының қуатын пайдалану.

Желіаралық экран (ЖЭ) – желаралық қарау жүйесі. Жүйе көмегімен желі екі не одан да көп бөлікке бөлінеді және желі шекарасына өтетін пакеттердің өту шарттарын анықтайды және ережелер жиынтығын іске асырады.

Желаралық экранды пайдаланудың негізгі себебі ретінде, бұл жүйесіз ішкі желідегі ақпараттар Internet қызмет жағынан да, сыртқы желіде орналасқан. Хост-компьютерлердің бір жағынан шабуылға ұшырауын атап өтуге болады.

Internet желісіндегі хабарларды жіберудегі басқаратын хаттамалар жиынтығы (TCP (IP)) біркелкі емес желілік ортадағы коммуникацияларды ұйымдастыру үшін пайдаланылады.

TCP/IP пакеттерінің тақырыпшаларында хакерлер шабуылынан ұшырауы мүмкін мәліметтері болады. Мысалы, хакер жіберуші адресін өзінің «зиянкес» пакеттеріне жазып жіберуі мүмкін, содан соң бұл пакет авторлығы көрсетілген клиенттік болып шығады. Internet-тің кейбір кең тараған қызметтеріндегі «әлсіз» жақтарын атап өтелік:

Электрондық поштаны жіберетін қарапайым хаттама. (SMTP). Internet-тің көлік пошталық қызметін атқарады. Осы хаттамадағы қауіпсіздік мәселесіне қатысты проблеманың біріне: пайдаланушы электрондық пошта хабарының тақырыпшасындағы жіберуші адресін тексере алмайды. Соның нәтижесінде хакер ішкі желіге көптеген пошта хабарларын тастай алады да, оның жұмысы тоқтап қалуы да мүмкін. Көпке мәлім, Internet-тегі электрондық пошта Sendmail программасы өз жұмысында белгілі бір желілік ақпаратты – жіберушінің IP адресін пайдаланады. Sendmail көмегімен жіберілетін хабарларды ұстап алу арқылы, хакер бұл ақпаратты шабуыл үшін пайдалана алады, мысалы: спуоринг (адрестіауыстыру үшін).

Файлдарды жіберу хаттамасы (File Transfer Protocol - FTP)- екілік және мәтіндік файлдарды жіберуді қамтамасыз етеді. Өз сервері үшін белгісіз FTP пайдаланғанда пайдаланушы онда кең түрде таратылатын файлдар сақталатынына сенімді болуы керек.

Желілік есімдер қызметі (Domain Name System - DNS)- үлестірілген дерек қор онда пайдаланушы мен хост-компьютер аттары пакет тақырыптпрында көрсетілген IP- адреске түрлендіріледі және керсінше түрлендіру де жүреді. DNS- проблемаларының біріне бұл дерек қорды автор емес пайдаланушылардан жасыру өте қиын. Соның нәтижесінде DNS-ті хакерлер хост-компьютерлер аттарының мәлімет көзі ретінде пайдаланады.

Қашықтықтағы эмуляция терминалының қызметі (TELNET)- қашықтықтағы жүйелерге қосылуға пайдаланылады. Серверге қосылған

TELNET арқылы хакер оның программасын аттар мен парольдерді жазып алатындай көшіреді.

Дүниежүзілік шырмауық (World Wide Web - WWW) – бұл жүйе пайдаланушыға Internet пен интражелідегі серверлердің мазмұнын қарауға мүмкіндік береді. Web- тораптағы гипермәтін құжаттарында сақталған ақпарат көмегімен қажетті тораптарға қалай қатынау керектігі анықталады. Хакерлер осы мәліметтерді пайдаланып Web- торапты бұзады немесе ондағы маңызды ақпаратқа қатынау мүмкіндігін алады.

Желілік қауіпсіздік саясаты құрамында екі түрлі талап болуы қажет:

желілік сервистерге қатынау саясаты;

желіаралық экранды іске асыру саясаты.

Желілік сервистерге қатынау саясатына сәйкес пайдаланушы үшін шектелген қатынауы бар Internet сервистерінің тізімі анықталады.

Сонымен бірге қатынау әдістеріне де шектеу белгіленеді, мысалы SLIP {Serial Line Internet Protocol) және PPP (Point-to-Point Protocol) хаттамаларын пайдалану. Қатынау әдістеріне шектеу белгілеу, пайдаланушы «шектеу қойыған» Internet сервистеріне басқа жолдармен қатынамауы үшін орнатылады. Мысалы, егер Internet-ке қатынау үшін желілік администратор пайдаланушылар WWW жүйесінде жұмыс істемеуі үшін арнайы шлюз тағайындаса, онда коммутация бойымен PPP- қосылуы арқылы Web-серверге қатынау мүмкіндігі болады.

Желілік сервиске қатынау саясаты әдетте келесі принциптерге негізделеді:

Internet-тен ішкі желіге қатынауға рұқсат жоқ, бірақ ішкі желіден Internet-ке қатынауға болады;

Internet-тен ішкі желіге шектелген қатынауға рұқсат ету. Ол үшін жекеленген «авторлық» жүйелерді қосу, мысалы пошта серверін.

Желіаралық экранды іске асыру саясатына сәйкес ішкі желі ресурстарына қатынайтын ережелер анықталады. Ең алдымен қорғау жүйесі қаншалықты «сенімді» екенін анықтау керек. Басқана айтқанда, ішкі ресурстағы қатынайтын ережелер мына принциптерге негізделуі керек:

анық формада рұқсатсыз нәрселердің бәріне рұқсат жоқ;

анық формада рұқсаты бардың бәріне рұқсат бар.

Бірінші принцип бойынша іске асатын желі аралық экран түрі қорғаудың жақсы түрін бере алады. Бірақ осы принципке сәйкес қатынау ережелері пайдаланушыға көптеген ыңғайсыздық тудырады және оны іске асыру да қымбат.

Екінші принципті іске асырғанда ішкі желі хакерлер шабуылынан, біріншіге қарағанда аздау қорғалады, дегенмен оны пайдалану ыңғайлы да, кететін шығын да аз болады.

Ішкі желілерді желіаралық экранмен қорғау тиімділігі желілік серверлер мен ішкі желіге қатынау саясатына ғана емес сонымен бірге желіаралық экранның негізгі компоненттерін таңдап, оны пайдалануға да байланысты.

Желіаралық экранға деген функционалдық талаптарға мыналар жатады:

желілік деңгейінде іріктеу (фильтрация) талаптары;

қолданба деңгейінде іріктеу талаптары;

іріктеу ережелері мен администрациялауды баптпу талаптары;

желілік аутентификация құралдарына талаптар;

журналдар жүргізу мен есепке алу талаптары.

13.2 Желіаралық экранның негізгі компоненттері

Желіаралық экранның негізгі компоненттерінің көпшілігін келесі үш категорияға жатқызуға болады:

- іріктейтін маршрутизаторлар;
- желілік деңгейдегі шлюздер;
- қолданба деңгейіндегі шлюздер.

Іріктейтін маршрутизаторлар

Іріктейтін маршрутизаторлар-серверде жұмыс істеп тұрған программа. Ол енгізілетін және шығатын пакеттерді іріктейді. Пакеттерді іріктеу пакеттердің TCP және IP- тақырыпшаларындағы мәлімет негізінде болады.

Іріктейтін маршрутизаторлар әдетте IP- пакеттерді пакеттің тақырыпшасындағы мына өрістер (поле) тобының негізінде орындалады:

жіберушінің IP- адресі (пакетті жіберетін жүйе адресі);

алушының IP- адресі (пакетті қабылдап алатын жүйе адресі);

жіберуші – порты (жіберу жүйесіндегі қосу порты);

алушы – порты (алушы жүйесіндегі қосу порты).

Порт- ол программалық ұғым. Оны клиент немесе сервер хабарды жіберу не алу үшін пайдаланады. Порт – ол 16 биттік сан. Қазіргі кезде жіберушінің пакетін TCP/UDP порты бойынша іріктейтін маршрутизаторлар барлығы бірдей іріктейбермейді. Дегенмен көптеген маршрутизатор өндірушілер осындай қызмет түрін көрсете бастады. Кейбір маршрутизаторлар алдымен пакет қай желілік интерфейстен келді соны тексереді де, содан соң бұл мәліметті іріктеудің қосымша критериі есебінде пайдаланады.

Белгілі бір хост-компьютер немесе порттар үшін фильтрация әр түрде іске асырылады. Мысалы сенімсіз не қарсы жақтан болып есептелетін хост-компьютер мен желілердің нақты адресіне келетін қосуларды тоқтатып қою. TCP/UDP порттары фильтрациясын (іріктеуін) IP- адресі фильтрациясына қосу үлкен икемділік береді.

TELNET демоны сияқты серверлер әдетте нақты порттармен байланысқан (мысалы TELNET хаттамасының 23 порты). Егер желіаралық экран TCP немесе UDP қосуларын белгілі бір порттардан ажырата алса, онда қауіпсіздік саясатын іске асыруға болады. Бұл жағдайда кейбір қосулар тек белгілі бір нақты хост-компьютерлермен орындалады.

Адресі 123,4.*.* болатын ішкі желімен қосылатын жағдайдағы қауіпсіздік саясатының іске асуын қарастырайық. TELNET қосуы адрес 123.4.5.6, болатын бір хост-компьютерде болады, ол қолданбалы TELNET шлюзы болуы мүмкін, ал SMTP- қосулары адрестері 123.4.5.7 және 123.4.5.8 болатын екі хост-компьютерде болады. Ол электрондық поштаның екі шлюзы болуы мүмкін. NNTP (Network News Transfer Protocol) бойынша алмасу адресі 129.6.48.254 болатын жаңалықтар сервері бойынша өтеді, және ол 123.4.5.9

адресі бар NNTP-серверінде болады да, ал NTP хаттамасы (желілік уақыт) – барлық хост-компьютерлер үшін өтеді. Басқа барлық серверлер мен пакеттер жабылады. Осыларға сәйкес ережелер жиыны 13.1 кестесінде келтірілген.

13.1 кесте

Фильтрациялар ережелері

Типі	Жіберуші адресі	Алушы адресі	Жіберуші порты	Алушы порты	Іс-әрекет
TCP	*	123.4.5.6	> 1023	23	Рұқсат ету
TCP	-	123.4.5.7	> 1023	25	Рұқсат ету
TCP	*	123.4.5.8	> 1023	25	Рұқсат ету
TCP	129.6.48.254	123.4.5.9	> 1023	119	Рұқсат ету
UDP	*	123.4.*.*	> 1023	123	Рұқсат ету
*	•	*	*	*	Рұқсат жоқ

Бақылау сұрақтары:

1. Internet арқылы бұзушы не істей алады?
2.Internet-тың қандай протоколдары және қызметтері әлсіз болады являются уязвимыми?
3. Әрбір мекеменің желілік қауіпсіздік саясатына не кіреді?

Пайдаланылатын әдебиет: [2] – 187- 217 б.

Дәріс №15. Тақырыбы: Зерттеуден программаларды қорғау. Антивирустық қорғау

Программаларды қорғаудағы ең сапалы әдістердің бірі болып ақпаратты криптографиялық түрлендіру болып табылады. Алайда шифрлау жүйені жөндеуіш басқаруында немесе дизассемблер көмегімен зерттеу бұзушыға криптографиялық қорғау алгоритмін түсінуге және оны қайталауға мүмкіндік береді. Сондықтан шифрлауды программаның кодасын статикалық және динамикалық талдаудан қорғаумен бірге қолдану қажет. Статикалық зерттеу – дизассемблер пайдаланумен, ал динамикалық зерттеу – жөндеуіш қолданумен орындалады. Дизассемблер – түсініксіз машиналық кодасын жеңіл оқылатын Ассемблер тілінде мәтінге түрлендіреді. Жөндеуіш – программаның әрбір кодасы немесе бөлек кескіні орындалған соң компьютерде барлық өтетін процестер туралы хабарлайды. Программалық коданың құпиялығын қамтамасыз ету үшін барлық статикалық және динамикалық құралдардың жұмысының берекесін қашыру (парализовать) немесе жұмысын теріс істеуін мәжбүр ету керек. Жалпы қабылданған «көрнекілік» пен «құрылымдықтан»

өзге механизмдерді қорғауда «изощренностті» қолданған дұрыс, яғни бұл стиль күрделі және шатастыратын орындалу модулін алуға мүмкіндік береді.

Соңғы уақытта бұзуға қарсы қолданылатын құралдар ескірді деген ұғым жиі айтылады, хакерлік құралдар программалық коданы зерттеуден қорғайтын қорғаныстардың кез келгенінен өтуге мүмкіндігі бар.

Шынында да, кез келген қорғаныс құралын ашуға болады, себебі оның коды процессор арқылы қойылады. Бұзушы бағдарламаны жан-жақты виртуалды жүйеде зерттей алады, онда процессор, жады, сыртқы құрылғылар, операциялық орта эмулирленеді. Бұл жағдайда көптеген қарсы тұру тәсілдері тиімсіз болып саналады. Қандай да ортаның эмуляциясы сапалы болғанымен соңғысы болмыстан ерекше болады. Мысалы, аппаратураның мерзімдік сипаттамасының дәл эмуляциясының болмауынан және қорғалған бағдарлама осыдан шығатын барлық ақпаратты тани білетіндей.

Қорғаныс жүйесінің жасаушылардың барлығына ортақ мынадай қателіктерін көрсетуге болады:

- бағдарлама тек қана статистикалық анализ құралдарынан қорғалады да, нәтижесінде ол динамикалық түрде оңай зерттеледі, және керісінше;
- аналогиялық жағдай мынадай орынды иеленеді, ағымдағы ақпаратты қайтаратын функция жұмысының нәтижесі эталондық (күтімдік) мәнге ауыстырылған жағдайда;
- жүйелік қорғаныс ашылғаннан кейін код барлығына бірдей ашық болады және басқа жүйенің ойында сақталады немесе дискіде сол мезетте не жедел түрде басқаруға беруге болады.

Сонымен, бұзылмайтын қорғаныс жүйесі жоқ және болуы да мүмкін емес, бірақ кез келген бұзушыдан сақтайтын қорғаныста қою ешкім және ешқашан мақсат етіп қойған жоқ. Қорғаныста қоюшылардың негізгі мақсаты:

- қарсыластардың қорғаныста бұзуға уақытын алу, осы уақытта контрмер қолдануға уақыт ұту;
- қорғаныста бұзуға кеткен құралды, жаңадан қорғаныс бағдарламасын жазуға;
- бір-бірімен сәйкестендірілген кепілдендіру.

Тағы бір топ DOS-қа қорғаныс керек жоқ, себебі DOS ескірді деген пікір айтып жүр. Бірақ бұл кеңселік программистің логикасы! Шынымен, бір де бір кеңсе DOS-пен жұмыс істемейді. Алайда жауапкершілікті толық тағайындалған компьютерлік жүйедегі программалардың барлығы дерлік соның қарамағында жұмыс істейді. Дәл осылар зерттеуден қорғаныста керек етеді. Кеңселік программаны бұзғаннан тек интеллектуалдық жеке меншік азап шегеді, ол тек үстінен түсетін пайдадан ғана айырылады. Ал жауапкершіліг тағайындалған программаны бұзуды диверсанттар немесе террористер пайдалануы мүмкін, ол апатқа әкелуі мүмкін.

Сонымен, біз толық жауакершілікті қолданыс үшін программа жазамыз. Сәйкесінше, біз ықтимал қарсылас барын ескеруіміз керек, ол біздің кодты зерттей отырып, өзінің мақсатына жету үшін программа алгоритмін өзгертуі керек.

14.1 Автоматикалық және интерактивті дизассемблермен күрес.

Автоматикалық дизассемблер атқарылатын файлдың кодын талқылайды және листингті немесе соған тиісті қорытынды текстi қалыптастырады. Статистикалық кодтың анализі противотрассирлық жүйенің жасалу мүмкіншілігін жоққа шығарады. Дизассемблерленген мәтіннің бағдарламасын қарап шығып қорғау механизмін табуға және айналып кетуге болады. Сондықтан дизассемблерлеуден бағдарламаны қорғайтын жүйеастын реализациялау қажет.

Бағдарламаны статистикалық зерттеуден қорғау үшін бағдарламаның өзінің кодын модификациялау, кодты шифрлау, кодты қаптау немесе түрлі ассемблерлік трюктар арқылы, дизассемблердің шығу кодын бұрмалауға бағытталған:

- жасырынды командалармен басқаруды беру (динамикалық түрде өзгеретін адрес бойынша өту, JMP-дан RET арқылы, RET және CALL JMP арқылы), дизассемблермен құруда басқаруды беретін графты күрделендіреді;
- жасыратын код;
- жүктелетін модульдың стандартты емес форматын, мысалы, сегменттер кодында стекті анықтау т.с.с.

Сонымен қатар, бұл тәсілдердің түрлі комбинациялары мүмкін.

Кодты шатастырудың қарапайым тәсілі оның шартсыз өтулермен шатасуы болып табылады. Өту командасынан кейін жасыратын командаға бірнеше мағынасыз командалар және операция коды не/және байттарда үлкен өлшемді команда префиксі қойылады. Әрбір нақты жағдайда ұзындық мынадай есеппен алынады: бұл фиктивті команданың соңы шын команданың біреуінің ортасына түссін. Бұл дизассемблер осы командадан бастап бұрыс реттіліктегі командаларды шығаруға әкеледі. Ал кейде ештеңені декодировать етпей, тек мәліметтерді хабарландыратын директивалар реттілігін жазады (DB, DW,…).

Бұл тәсілдің оңай және қысқа жолы – қорғалатын команданы ашуда шартты өту қолданылады. Алайда, бұл шарттың шындығы компилятор әсіресе дизассемблер үшін «түсінікті» болмау керек. Бұл тәсілдің артықшылығы – программа мәтініне қосымша, ешқашан орындалмайтын командаларды енгізуді талап етпейді. Бұл тәсіл бұзуға қарсы жақсы әдіс болуы мүмкін – бір уақытта дизассемблерді де, бұзушыны да адастырады.

Интерактивті дизассемблерлер шығатын мәтін/листингті программаның орындалатын кодын автоматикалық дизассемблерлер сияқты реттейді. Бірақ интерактивті дизассемблерлер автоматикалықтан мықты қолданбалы интерфейсімен (дизассемблерленген программа анализін жақсы жеңілдетеді) ерекшеленеді.

Интерактивті дизассемблерлер мынаған жол береді:

- айнымалылардың, таңбалардың, подпрограммалардың, т.с.с. атын өзгертуге, жаңа адрестер үшін атын енгізу, бар таңба/аттарды жоюға;
- нәтижелі мәтіндегі символдар реттілігін және орындалатын кодтағы байт реттілігін іздеуге;
- код учаскелерін ассемблерлік командалар немесе DB директивалар реттілігіне қайта дизассемблерлеуге;

- барлық сәйкес шақыруларға автоматты түрде қойылатын үзулерге, подпрограммаларға, т.с.с. түсініктеме (комментарий) беруге;
- программа сегменттерінің тізімін көруге;

дизассемблерлік мәтінді орындалатын кодтың автоматты модификациясымен немесе онсыз өңдеуге/жөндеуге.

Түрлі интерактивті дизассемблерлер сонымен қатар басқа да мүмкіншіліктерді береді. Ең соңғы интерактивті дизассемблерлер (IDA) дизассемблерленген кодты тек ауыстыруға ғана емес, сонымен қатар дизассемблерлеу процесінің өзіне араласуына мүмкіндік береді.

Жоғарыда айтылған тәсілдер автоматты дизассемблерлерге қарсы қолдануда жақсы. Бірақ IDA-ны шатастыру (немесе басқа интерактивті дизассемблер) олардың қолынан келмейді. Дәлірек айтсақ, шатастыра алады, бірақ біздің қарсылас қоқыстанатын байтты 'Undefined' ретінде, ал одан соң код ретінде көрсету керектігін түсінген уақытқа шейін. Бұдан кейін ол дизассемблер ортасында қорғалатын программа анализін алады.

Хакерлік ортада «динамикалық фуфель» деген атақ алған эффекті әрі күрделі тәсіл бар. Тәсілдің мағынасы мынада-қоқыстанатын байттар басқаруды беру командаларынсыз еш айналып кетпейді. Олар программаның орындалу барысында еш зияны жоқ командалармен (NOP, STI ,...) алмасады. «Фуфелі» бар подпрограммалардың алғашқы жіберілуіне дейін. Басқаша айтқанда, дизассемблерлеуден қроғалатын программа фрагменті, шынында да, дискідегі программа түрінде жіберілмейді, ол компьютердің дұрыс жұмыс істесмеуіне әкеледі. Алайда, жіберілген программа белгісіз жерден қоқыстанатын байттарды алу үшін мәліметтерді есептейді де, оларды программаның орындалуына еш әсерін тигізбейтін командаларға алмастырады.

Мұндай тәсілде қорғалған рпограмманы бұзу өте ұзақ әрі күрделі процесс.

14.2 Нақты режим бұзушыларына қарсы күрес

Бұзушының зерттеуінен қорғанудың екі тәсілі бар:

- бұзушыны анықтау және басқаруды бұзушыға қарсы бұтақ реакциясына беру;
- код фрагменттерімен программаны «қоқыстау», олар бұзушысыз қалыпты жұмыс істейді, ал бұзушы бар кезде апаттық жағдайға, компьютердің дұрыс жұмыс істемеуіне немесе программаның орындалу барысының бұрамлануына әкеледі.

Бұзушыны анықтау

Нақты режим бұзушыларын анықтау жетерліктей оңай. Оларды анықтаудың екі негізгі тобын атап айтуға болады:

- процессордың аппараттық ерекшеліктерін қолдану, әсіресе командалар тізбегінің бары, сонымен бірге кейбір нұсқауларды орындағаннан кейін трассирлі үзудің жоғалуы;
- операциялық ортаның өзгеруін үзу векторын тексеру жолымен, программаның жеке бөліктеріне кеткен уақытты тексеру, программаны жіберу кезінде регистлердің алғашқы жағдайын тексеру және т.с.с. анықтау.

Бұзушылар комьютердің мынадайресурстарын қолданады: INT1 сияқты үзулерді, INT 3 және TF трассировка жалауы. Мұның бәрі қорғалатын программа бұзушының зерттеуін анықтау үшін қолданылуы мүмкін. Мәселе мынада, Intel 80x86 сияқты процессорлар бір команданың трассировкасын «жоғалтады», егер оның алдындағы команда сегмент регистрінің мәнін өзгерткен болса. Сондықтан бұзу процесіндегі TF трассировкасының жалауының орнатылғандығын анықтауға болады.

486 тобындағы процессорларда бұзушыны команданы алдын ала таңдау буферін қолданып білуге болады. Таңдалған және сол тізбекте орналасқан команда кодының өзгеруі программаның орындалу барысына әсер етпейді.

Бұзушыны анықтаудың келесі тәсілін тек нашар сапалы бұзушыларға қарсы қолданамыз, мәселен CodeView немесе Turbo Debugger. Ол мынаған негізделген: программаны жүктеу барысында анықталған түрде регистрлердің инициализациясы жүреді. Программаны бұзушыға зерттеу аз емес қууларды міндетті етеді. CodeView және TD бірінші қуу кезінде AX, BX, CX, DX, SI, DI, BP регистрлерін нольдейді. Екінші қуу кезінде CodeView бұл регистрлерді тағы да нольге теңестіреді, ал Turbo Debugger алдындағы қуудан қалған «қоқысқа» мүлдем тиіспейді. Программа басында регистрлердің мәнін керектілермен салыстырып, бұзушыны анықтауға болады.

Нақты режимде бұзушының барында программа жұмысының бұрмалануы

Бұған бірнеше тәсілдерді атап кетуге болады:

- контрольді нүктенің құрылуына және программа кодының өзгеруіне қарсы әрекет жасау;
- қолданушымен бірге интерфейстің бұзылуы, мысалы, клавиатураны блоктау арқылы экранға нәтижені бұрмалау;
- код учаскесінің генерациясы, шифрлау, жүйені қорғаудағы басқа дар подпрограммаларды шақыру сияқты жауакершілікті талап ететін әрекеттерді жүзеге асыру үшін бұзу (кейде тек бұзу ғана емес) үзулерін қолдану;
- орындалатын код аймағында стекті анықтау және оны бұлжытпай ауыстыру.

14.3 Қорғалған режим бұзушыларымен күрес

Кейбір бұзушылар (қорғалған режимді) режимді қорғауға арналған арнайы бұзуға қарсы трюктерге түсіп қалады. Бұзушыларға арналған тағы бір тәсіл олардың ұсынатын API-ге негізделген. Мәселен, DeGlucker, API no INT 15h (OFFxx функциясы)-ды ұсынып, конструкцияда мәңгі тұрып қалады.

```
mov   ax, OFFOlh
int 15h
```

Ал қорғалған режимді бұзушылармен күрестегі үшінші тәсіл аппараттық бұзу құралдарының жағдайын нашарлату болып табылады. Мәселен, программаның трассировкасы үшін DR1 регистрі қолданатыны бізге белгілі. Әрине, онда оның не мәнін, не DR7 регистріндегі басқаратын биттердің мәнін дұрыс көрсетпейтіндей етуге болады. Алайда, қазіргі кезде бұзу регистрлерін бұзушылардың (DeGlucker 0.05) өздері де қолданады, ал бұзылатын программаға оларды қолдандырмайды.

Соңғы төртінші топқа қарсы күрес – бұзушылардың нақты қателіктеріне негізделген.

14.4 «Ерекшеленген» программалау

Қорғауды жасаушының міндеті – қарсыласының бұзуға кететін уақытын көбейту. Сонымен бірге, программаны тексеруді қиындату.

Ол үшін бізге жоғары сапалы, бірақ дизассемблер түрінде түсініксіздеу, ал бұзушыға шартты және шартсыз өту теруінде былықшылық көзқарасындағы программа жасау керек. Мұндай әрекет «изощренный» программалауды қолданғанда шешіледі.

Бірнеше негізгі бағыттарды бөліп көрсетуге болады:

- экзотикалық, процессордың немесе оның стандартты емес сәйкестіктерінің жиі кездеспейтін командаларын қолдана отырып, ерекше түрдегі алгоритм жасау;
- бір алгоритмдегі бірнеше толық эквивалентті нұсқаларды реализациялау, оған әрбір қатынау кезінде оны реализациялаудағы нұсқалардың біреуі алынады;
- коданың қоқысталуы – біздің мәліметтерді өңдеуде еш зәсерін тигізбейтін командалар.

Бұл тәсілдермен жақынырақ танысайық.

Алгоритмдерді экзотикалық жүзеге асыру .Мәселен, бізде бір жалау бар делік (немесе айнымалы) және оған 0-ге тексеру өте қажет. Бірақ біз CMP AX,0 командасын анық жазғымыз келмейді, немесе мүлдем басқаруды беру командаларын мүмкіндігінше қолданудан айналып кетпекпіз.

Ең бірінші 0-ге тексерудегі анық емес командаларды қолдану ойға келеді. Мысалы, екілік-ондық арифметикадағы командаларды қолдану:

Мысалы,

```
mov  ax,                OurFlag
daa
pushf
pop   ax      ; нольдік жалаудың анық емес тексеруі
and   ax, 40h
 jz    FlagIsZero
```

Әрине, шын программада жалауды алу және нольге анықтау сол баяғы тексеруді қиындатуда таратылуы керек. Алайда бұл нұсқа қолданбалы болса да бұзуда өте оңай болып табылады.

Эквивалентті бұтақтарды жүзеге асыру. Программаны тексеруді қиындату үшін бұл тәсілдің пайдасы анық. Шынында да, егер біз бұзушылықта болсақ, бірде бір командаға, бірде бірнеше командаға түссек, программа алгоритмін түсіну оңай болмайды.

Анализді қиындату үшін мүмкін:

- бұтақтар санын көбейту (барлық алгоритмдер үшін қолданылмайды);
- бұтақтарды үзу өңдеушілері түрінде беру (INT I, INT 3, INT 4, INT6 және т.б.) және оларға тура қатынамай, сәйкесінше жағдайларды жасау жолымен;
- кездейсоқ сандарды тексерудің санын көбейту.

Коданы қоқыстау. Коданы қоқыстау мағынасында оған командаларды қолдан енгізу деп білеміз, ол командалардың орындалатын алгоритмге еш қатысы

болмайды және олар алгоритмнің анализін не қиындатады, не ол анализді канителен етеді, сәйкесінше оған көбірек уақыт әрі күш керек.

Бұл жерде әрекетсіз регистрлерді манипуляциялау; кейбір флагтарды құру не алы тастау, бұл флагтарға еш қатысы жоқ бірнеше командаларды орындау келесіде шартты өтумен, ал оның өзі шын мәнінде орындала ма, орындалмай ма белгісіз; бірдей құрылымды өңдеу, олардың біреуінде ғана біздің алгоритмнің мәліметтері болады және т.б. бұның бәрі тек дизассемблерленген мәтіннің көлемін ғана көбейтпейді, сонымен бірге қорғалатын алгоритмнен көңілді бөледі.

Компьютерлік вирустар компьютерлік және ақпараттық даму үрдісінде пайда болған өзінше бір құбылыс. Бұл құбылыстың мәні - программа-вирустардың тірі ағзалаларға тән туылу, көбею және тіршілігін жою секілді қасиеттерді иеленуінде.

Компьютерге қатысты “вирус” деген терминді 1984ж. Фред Коэн ұсынған болатын. Ф. Коэннің вирусқа берген алғашқы анықтамасы: “Компьютерлік вирус - өзге программаларға өзінің немесе өзгертілген көшірмелері арқылы өзгеріс енгізу арқылы жұқтыратын программа, сондай-ақ, соңғысы әрі қарай көбею мүмкіндігін сақтап қалады.” Вирустардың көрсетілген қасиеттері тірі табиғаттағы биологиялық вирустардың жұқтыру әрекетіне пара-пар.

Әдетте вирус компьютер жүйесінде неғұрлым табылмай қала беретіндей етіп құрастырылады. Вирустардың алғашқы “ұйықтау” периоды оның өмір сүру механизмі болып табылады. Вирус мысалға жұма 13-і, белгілі бір күнде және т.с.с. шақыру оқиғасы пайда болғанда нақты сәтте әрекет жасайды.

Компьютерлік вирус өзін компьютерлік дискілерге құпия түрде жазуға тырысады. Вирустардың көбісінің әрекеті - вирус өз жұмысын компьютердің әрбір жүктемелену кезінде бастайтындай етіп оның жүйелік файлдарын өзгерту. Мысалға, жүктемелеу аймақтарын жұқтырушы вирустар тек операциялық жүйе және жіберу файлдарын сақтау үшін оқшауланған дискета мен қаттыл дискінің бөлігін жұқтырады. Бұлар компьютерді іске қосқан сайын жадыға жүктемеленіп отыратындықтан залалды вирустар болып табылады. Мұндай вирустардың көбею мүмкіндігі жоғары және олар үнемі жаңа дискілерге тарала алады.

Әдетте вирустар COM және EXE Файлдарға жақын жүреді. Кейбір вирустар компьютерлік жүйені жұқтыру үшін жүктемелеу аймағын, сондай-ақ файлдарды жұқтыру әдісін пайдаланады. Бұл - вирустарды арнайы программалармен іздеп табу мен айқындауды қиындатып, оның тез таралуына әрекет етеді.

Вирустардың басқа түрлері де бар. Компьютерлік вирустар көп түрлі көбейетіндігінен және өмір сүру ортасын бұзатындықтан жүйеге зиян келтіреді.

Жүйелік “құрт” глобальды торап бойынша таралып, магниттік тасуышта өз көшірмесін қалдырмайтын программа-вирустардың бір түрі. “Құрт” бос ресурстары бар компьютерлердің торабында дұрыс пайдаланса “құрт” технологиясы пайдаға асуы мүмкін. Мысалы: World Wide Web Worm “құрт”

Web бөлімдерін іздеу индексін қалыптастырады. Бірақ, “құрт” зиянды программаға тез айналады.

Зиян келтіруші программалардың белгілісінің бірі - UNIX жүйесінің командалық интерпретаторының кірме тілінде және Си тілінде 4000 жолдан тұратын Морристің “құрт” программасы. Бұл программа VAX және SUN компьютерлерінде UNIX операциялық жүйесіндегі қателерді пайдалану енгенде жұмыс істейді. Жүйелік “құрт” - зиянды программалардың ең қатерлісі. Себебі олардың жұқтыру объектісі Internet торабына қосылған миллиондаған компьютердің кез келгені болуы мүмкін.

“Троянский конь” (бұл терминді хакер Дан Эдварс ұсынған) программалық қыстырманың бір түрі. Ол өзінің ішіндегі жасырын түрдегі қатерін білдіртпей пайдаланушыға программаны жіберу жағдайын туғызатын алдау әдісін пайдаланады. Мұндай программалар әдетте пайдалы утилиталардың ішінде байқалмай жүреді. “Троялық ат” программасының қауіптілігі - зиянсыз программаға қосылған қосымша командалар блогында. Бұл командалар блогы белгілі бір шарттың орындалуымен немесе сыртқы команда бойынша өзінің қаскүнемдік жұмысын істей бастайды.

15.1 Вирустарды және басқа зиянды программаларды жіктеу

Зиянды программаларды былай жіктестіруге болады:

- қауіптілігінің дәрежесі бойынша;
- жұқтырылатын объекттер бойынша;
- жұқтыру әдісі бойынша;
- жүйеде бар болуын жасыру әдісі бойынша;
- программалау тілі боынша.

Қауіптілігінің дәрежесі бойынша жіктеу. Зиянды программаларды қауіптілігінің дәрежесі бойынша былай бөлуге болады:

- зиянсыз, яғни өзінде ешқандай бұзатын функцияларды сақтамайды және өзін тек көбеюмен білдіреді;
- қауіпсіз, яғни өзін хабармен, бейнеәсерімен және с.с. білдіреді;
- қауіпті, яғни есептеу жүйенің жұмысында маңызды жаңылысуылар пайда болады;
- өте қауіпті, яғни файлдарда, жүйелік аймақтарда, логикалық дискілерде ақпаратты жоюға жарамды, аппаратураны бұзуға болады.

Жұқтырылатын объекттер бойынша жіктеу. Тарату үшін пайдаланылатын объектілер бойынша зиянды программаларды былай бөлуге болады:

- файлдық вирустар, яғни қандай да бір тәсілмен файлдарға қосылатын программалар;
- жүктелетін вирустар, яғни өз кодасын дискілердің жүйелік аймақтарына жазатын программалар;
- тораптық вирустар, немесе “черви”, яғни есептеу тораптарда қандай да бір тәсілмен өз көшірмелерін жіберетін программалар;
- “троянский конь”, яғни қандай болса да зиянсыз программаларға астарланған программалар; вирус алгоритмдері бойынша файлдарға,

жүйелік аймақтарға немесе тораптық хабарларға жазылу мүмкін, бірақ осындай әрекеттерді жасау үшін арнайы программа керек; трояндылықтарда көбеуі мүмкіндігі жоқ;
- “логикалық бомбалар”, яғни нормалды программада құрастырушымен программаланған троялық компонентері (белгілі шарт бойынша орындалады, мысалы, винчестердің 0-ші жолында кілттік ақпарат болмағанда).

*Жұқтыру әдісі бойынша жіктеу***.** Әртүрлі зиянды программалар объекттің жұқтыру әдісі бойынша әртүрлі жіктеледі.

Осы параметр бойынша *Файлдық вирустар* былай бөлінеді:
- вирус-спутниктер, файл атын өзгертіп (әдетте кеңейтуін өзгертеді) және ескі атын сақтап жаңа файлға өзін жазып кетеді;
- орнын алатын вирустар, файлдың мазмұнын сақтамай үстіне өзін жазып кетеді (бұл вирустар өте қауіпті вирустарға жатады);
- жапсыранылатын (пристыковывающиеся) вирустар, немесе паразиттық вирустар программаларға қосылып жазылады, жұмысын бастағанда басқару алдымен вирус кодасына беріледі, содан кейін вирус жұқтырылған программаның кодасын шақырады.

Жүктелетін вирустар:
- сирек қолданатын секторға жүктеуіш кодасын сақтайды және оған басқаруды береді;
- жүктеуіш кодасының орнын алатын вирустар және оның барлық функцияларын өзі орындайды.

Троялықтар:
- дербес, яғни кез келген әдіспен пайдалы программаларға астарланған программа;
- жапсыратын (пристыковочные), яғни дроппер-программа көмегімен орындалатын файлдарға жазылатын программалар;
- “приваживаемые”, яғни қаскүнем жүйелік конфигурация файлдарына троялықтарды жандандыру командаларын қосу керек

Жүйеде бар болуын жасыру әдісі бойынша жіктеу.

Жүйеде бар болуын жасыру әдісі бойынша вирустарды былай бөлуге болады:
- жүйеде өз бар болуын жасырмайтын;
- шифрланатын, яғни кез келген кілтімен өз орындалатын кодасын шифлайтын, бірақта кері шифрлауыш әрқашанда бір;
- полиморфты, яғни әрбір жаңа файлға жұқтыру кезінде вирус кез келген тәсіл бойынша генерацияланған кілтімен шиФрланады және кері шифрлауышты өзгертеді;
- “көрінбейтін” (“стелс”) вирустар, яғни резидентті вирустар жүйелік үзулерді ұстап алыды және жүйеде бар болуын жасырады.

Программалау тілі бойынша жіктеу. Зиянды программалар келесі тілдерде:
- Ассемблер тілінде;
- жоғары деңгейдің тілінде;

- ОЖ командалық тілінде;
- қолданбалы программалық кешенінің құрамдас тілінде/макротілінде жазылу мүмкін.

15.2 Жұғу алгоритмдері.

Файлдық вирустармен стандартты жұғу алгоритмдері

Спутник-вирустар программаны жұқтырады, олар программалық файлдың мазмұнын өзгертпейді. Спутникті-вируспен жұғудың екі алгоритмі бар. Бірінші алгоритмнің негізі мынада: Dos программалық процессорына ат бергенде, кеңейтусіз программаны қосқанда, ол COM кеңейтілуі бар файлды іздейді, одан кейін EXE, соңында BAT. Бұндай спутник-вирусы EXE файлын тапқан кезде, сондай атпен COM кеңейтілуі бар файл құрады. Мұндай вирустардың белсенділігі толығымен VC/DN/NC типті жай файлдық бөлшектерді жұмыс істеуінен тоқтатады, олар ENTER клавишасын басқан кезде көрестілген кеңейтілуімен орындалатын файлды іске қосады.

Екінші жұқпалы спутник-вирусының алгоритмінің негізі, кеңейтуі орындалатын файлдың шектілігін тек командалық процессор қояды. Программаны іске қосатын немесе орныдайтын Dos функциялары, кез келген кеңейтілуі бар файлды қосуға мүмкіндік береді. Берілген алгоритмді іске асыратын спутник-вирусы EXE файлын іздейді, оның мазмұнын файлға тура сол атымен және кеңейтілуімен көшіреді, ал EXE файлға вирустың кодын жазады. Құрбан-программасында сақталатын кеңейтілуі стандартты болуы мүмкін, вирус кодынан кейін EXE файлдың соңына жазады.

Екінші алгоритмнің вариациясы қиындау, манипуляциялық файлдардың жұғуы болып табылады. Бұл жерде екі емес, үш кеңейтілу қолданылады.. Құрбан-программасына басқаруды бергенде, вирус өзін файлға үшінші түрдегі кеңейтілумен сақтайды, құрбан-программаны EXE файлға көшіреді, оны содан кейін іске қосады, құрбан-программасы аяқталғанда өзін қайта EXE файлға көшіреді. Үшінші түрдегі кеңейтілуі бар файл жойылады.

Осы алгоритмді қолданатын вирустар, кеңейтуді таңдау кезінде қауіпті немесе өте қауіпті болуы мүмкін.

Файлдың орнын басатын вирустар, өздерін орындалатын файлдың басына көшіреді, бірақ ескі мазмұнын сақтамайды. Бұндай файлдардың резервті көшірмесі болмаса, қалпына келтіру мүмкін емес.

Орындалатын файлға вирус жұққан кезде, ол өзінің кодын файлға жазу керек, құрбан-программасына кіру нүктесін есте сақтап, өзгерту керек, онда вирустың коды басқаруды алады. Мұны COM форматты Dos программасында істеген оңай, өйткені олар орындалатын кодтың екілік бейнесін оперативті жадыда ұсынады. COM файлдарды жұқтыратын үш стандартты алгоритм бар: соңына жазатын, басына және ортасына.

Стандартты түрде файлдың соңына жазылатын вирус, кіру нүктесін өзінде есептейді. Ол бірінші байттарды есте сақтайды (көбінесе 3, 5 немесе 6) және олардың орнына өзіне берілетін басқару командасын жазады. Істің соңында, вирус есте сақтаған командалрды орындайды, немесе CS: 100h адресімен байттарды қалпына келтіреді және соған басқаруды жібереді, мысалы:

```
mov     ax6 1000h
push    ax
retn
```

Стандартты түрде басына жазылатын вирус, файлдың соңына сол ұзындықтағы кесек кодты көшіреді. Содан, вируспен қоданылмайтын және кодсыз, берілген программасыз команда адресі соңғы программаны жадыда қалпына келтіреді, вирус бұл кесекті CS: 100h адресімен көшіреді және басқаруды осы адреске жібереді.

Файлдың ортасына жазылатын вирустар, бірдей мәнді файлдарды жұқтырады, көлемімен үлкен вирус файлды ажыратады, орын босаған жерге өзінің кодын жазады. Құрбан-программаға басқаруды жіберу, стандартты түрде соңына жазылатын вирустардыкіндей.

Структурозацияланған орындалатын файлдарды жұқтыратын вирустар (MZ EXE, LE EXE, ELF т.б.) өте қиын жұғу алгоритмдерін қолданады.

Структурозацияланған орындалатын файлды жұқтырғанда, вирус тақырып атының алаңын анализдейді, оны файлдың соңына жазады және тақырып атының алаңын модификациялайды. Бұл информация сосын құрбан-программаға жіберуді орындайды.

Іске қосылған вирустармен жұқтыратын стандартты алгоритмдер

Іске қосылған вирустар жүйелік аумақты үш түрлі стандартты алгоритммен жұқтырады.

Бірінші алгоритм, айталық “Brain” вирусымен қолданылсын, файл кластерінен бос құрбан аумағының жазылуында қорытындылады.

Екінші алгоритм, айталық “Stone” вирусымен қолданылсын, құрбан аумақ қолданылмайтын немесе кейде қолданылатын секторларға көшіріледі. Кейде вирус дискіде қосымша жолды форматтайды және сонда құрбан аумақпен құйрықты көшіреді.

Үшінші алгоритм, вирус құрбан аймақты ешқайда сақтамайды. Бұндай вирустар ОС-ты өздері іске қосады. Мысалы Troian.Surprise.456. НМД-ны жұқтырады.

Жұқтыру модификациясының алгоритмдері

Егер вирусты тазалау алгоритмі, стандартты алгоритмдердің файлдарын жұқтыртса, ол антивируста орындалған, онда жаңа вирусты тазалауды орындау бірнеше секунд ғана алады. Сондықтан вирус жазушылар әрдайым жаңа алгоритм ойлап табуға тырысады.

Ең оңай амалдың бірі – вирусты екіге бөлу. Вирус стандартты түрде файлды жұқтырады, ол басына немесе ортасына жазады, бірақ онда кодтың бір ғана бөлігін сақтайды, ал қалғанын соңына жазады. Негізінде бұндай вирусты табу қиынға түспейді, өйткені вирустың екі бөлігі жұқпалыдан жұқпалыға өзгермейді. Алдымен құйрығын кесіп, содан вирусты кәдімгідей тазарту керек.

Есеп кейде қиындайды, егер вирус бірнеше подпрограммаға бөлінсе, әр жаңа жұқтыру кезінде түрлі подпрограмма санын құйрыққа қояды. Бұндай вирус файлдың соңы мен басына байланысты фиксациялы сигнатурлары болмауы мүмкін, және ол кіру нүктесіне де байланысты. Мұндай вирусты

тазалау әрбір подпрограмманың сигнатруларын тексеруін, оның басы мен құйрығының ұзындығын табуын немесе кіру нүктесінен подпрограмманы шақыру командасына өтуін қамтамасыз етеді.

Псевдо іске қосатын вирустар, қосылатын секторларды жұқтырпайды, дискідегі фиксацияланған аумақтағы жүйелік файлдарды жұқтырады. Бұл файлдар басқа жерге жазылады, ал вирус жүйелік файлдың үстіне жазылады. Бұндай вирусты тазалау үшін тіке FAT-қа немесе тамыр каталогына бару керек.

Резидентті вирустармен жұғу алгоритмдері

Файлдарды іздегенде резидентті вирустар жұқпалы каталогтарды айналмайды. Олар жүйелік үзулерді жібермейді, ол аргументтерге файлдың аты кіреді (немесе басқа аргументтер бойынша анықтауға болады). Резидентті вирустар файлдарды ашқанда, жапқанда, атын өзгерткенде, іске қосқанда және т.б. жұқтыруы мүмкін. Теориялық түрде файлдарды жұқтыратын вирус істеуге болады, мысалы, файлдағы позицияны өзгерткенде, бірақ жұғу үшін мұндай функцияларды ұстау машинаның жұмысы төмендегенде білінетін еді. Файлдағы позицияны өзгерту типінде жүйелік функция резидентті «стелс»-вирустарды ғана ұстайды, ол басқа програмаларға вирус кодының үстіне бүлдіртпей жазылмас үшін істелінеді.

Жады көптеген рет жұқпалы болса, ол вирустың жұмысын тоқтатады. Шынында да, машинада бірнеше жұқпалы программаларда ашылған вирус бар делік. Резидентті вирустардың көшірмесі тез арада жадыны толтырады да, машинаның жұмысы төмендетеді немесе тоқтатып тастайды. Сондықтан резидентті вирус орнату уақытында қолда барын тексереді.

Бұндай тексерудің үш алгоритмі бар. Бірінші алгоритм – кейбір үзу кезіндегі жаңа «мен мұнда» функциясын ұстау, ол өзінің орындалуы кезінде вирусты шақырады. Екінші алгоритм - өзінің сигнатурларына компьютер жадысын көшіру - көбінесе вирустар қолданады, ол белгілі бір адрес бойынша резиденттік көшірмелерді жазады, мысалы, видеожадының нолі жоқ нөмірлер бетіне жазады. Ал үшінші алгоритм – резидентті вирустар Dos жадысындағы сөздердің өлшемін кішірейтеді, мысалы, 640-тың орнына 639К-ны қояды. Бұның орындалуы жадының жұқпалы екенін көрсетеді.

HLLP вирустраымен жұққан алгоритмдердің қаупі

HLLP (High Level Language, Prasitic) вирустары жоғарғы дәрежедегі тілде жазылған программаларды таныстырады. Осында олардың негізгі қаупі білінеді.

Бұл программалар EXE форматтағы файлдармен компиляцияланады. COM файлдарға қарағанда, программадағы домп коды жадыда болмайды. HLLP вирусы өзін жадыдан оқиды және кәдімгі түрде файлды жұқтырады, кіру нүктесін өзінің денесінде сақтайды, Exe тақырып атын модификациялайды, оны жасау өте қиынға түседі. Ол жоғарғы класификацияны және кодты генерациялайтын қолданылатын компиляторды жоғарғы деңгейде білу қажет. Бұл парадоксальды естілгенімен, кәдімгі түрде вируспен жұққан алгоритмді Паскаль мен Си-ға қарағанда, Ассемблерде жазған оңай. HLLP вирусының авторлары Ассемблерді білмейді және білгісі

де келмейді. Сондықтан олар өзінің басымдылығымен примитивті жұқпалы алгоритмдерді жасайды, негізінде олар деструктивті емес функцияларды көрсетеді. Оның себебі, жаңа файлды жұқтыртқан кезде типтік HLLP вирустары өздерін жадыда оқымайды, ол тек сол іске қосылған файлда оқылады.

Енді осы алгоритмдердің өзімен нені ұсынатынын және оларды қолданатын вирустардың неге теориялық түрде қауіпсіз болмауын қарастырайық.

HLLP вирустарында қоданылатын жұқпалы алгоритмдер. HLLP вирусы екі біртипті жұқпалы алгоритмдерін қолданады, олар бір-бірінен тек құрбан-файлының кодын түрлі жағдаймен көшірілуімен ғана ерекшеленеді.

Бірінші алгоритмде құрбан-файлының мазмұны түгелімен вирустың соңына жазылады. Екінші алгоритмде жұқпалы файлдың басынан вирустың көлеміндей фрагмент «кесіледі». Ол файлдың соңына жазылады, ал вирус өзін өзінің файл тасушысынан оқиды және босаған орынға жазады. Басқа сөзбен айтқанда, осы екі алгоритм құрбан-программадағы EXE тақырып атынның модификациялауын қарастырмайды, вирус денесінен CALL FAR, JMP FAR немесе RETF командаларына басқаруды да бермейді, жадыда келесі файлды жұқтыртқанда вирустың көшірмесін де есептемейді. Вирус жіберілген жерден вирустың көшірмесі жұқпалы файлдың басында есептелінеді. Орындалғаннан кейін алгоритмді жұқтыратын вирус кейбір орындалатын файлдарды жасайды да, онда жұққан программаның кодын көшіреді, одан соң өшіреді.

Осындай жұққан алгоритмдердің қандай деструктивті салдарға әкелетінін қарастырайық.

Орындалатын файлдардың жинақталуы мен ашылуы. HLLP вирусымен жұққан орындалатын файл, Dos-тың ойынша, «құйрығы» бар қысқа программаны береді, ол оверлей немесе басқа бір нәрсе болуы мүмкін. Сонымен қатар, заңды түрде вирус жинақталған болады. Орындалатын файлдардағы жинақтаушылардың бір бөлігі (PKlite, DIET6 WWPACK және т.б.), мұндай файлдарды байламайды,өйткені құйрығы бар файлды табады немесе жинақталған вирусты қыса аламайды. Бірақ осындай файлдарды жинақтайтын программалар бар.

Негізінде дұрыс істелінген HLLP вирустары (мысалы, HLLP.Yarik.7991) қосылғанда өзінің сигнатурын тексереді, таппаса, хабарлайды, файлды ашуын сұрайды және жұмыс істеуге қарсы болады. Бірақ бұл көбінесе көмектеспейді. Сол HLLP.Yarik.7991 осындай ашылудан кейін тазаланбайды.

Орындалатын бар файлдарды вирустармен алмастыру. HLLP вирусы, авторы деструктивті функцияларды программалау ойында болмағандықтан, тексергенде қауіпті болуы мүмкін. Керекті орындалатын файлдар немесе мәліметтер файлы атын дұрыс жазбағандықтан немесе кеңейтілуін дұрыс қоймағаннан осындай вирустармен жойылуы мүмкін.

15.3 Антивирустың әдісінің классификациясы

Антивирустық программалардың барлығы фагтарға, детекторларға, ревизорларға, вакциналарға және резидентті күзетшлерге бөлінеді.

Фаг – бұл программа өзіне белгілі сигнатур бойынша вирустарды анықтайды, яғни осы вирусқа байланысты кодтардың тармағында анықтайды. Вирусты анықтаған кезде, оны файлдан тазалайды немесе өшіреді. Егер вирус қайта орнына келген болса, оны «өлтірген» жөн.

Детектор вирустарды табады, бірақ фагқа қарағанда тазалай алмайды. Вирусты тапқанда фагты шақыратын детекторлар болады, оған тазалау үшін керекті мәліметтерді беріп тұрады.

Ревизор дискіде әрбір өзгертулерді қадағалау үшін қажет: іске қосылған аумақтың кодын өзгерту, дискті бөлуді өзгерту, файлдардың пайда болуы/өшірілуі/өзгертілуі және т.б. Жақсы жасалынған ревизорлар оперативті жадыдағы тексеру алгоритмдерінен тұрады, қосылған резидентті стелс-вирустардың жоқ болған жағдайда. Нақты вирустарды сигнатур бойынша анықтауға ревизор тағайындалмаған.

Программалық вакцина, вирустармен жұққан жүйеде, қолда барын немесе өңделген файлдарды тексеретін программаларды қорғауға арналған. Егер резидентті вакцина қолданбалы болса, қазіргі кезде файлдық ескіріп кеткен. Файлдарды вакциналаудың 1980 жылдары маңызы болды, ол кезде бірнеше вирустар жиындары болды, олар вирустарды сигнатур бойынша анықтаған жоқ, оның атрибуттары бойынша анықтады.

Резидентті күзетші – бұл программа жүйелік функцияларды қарауды байқайды, олар алгоритмдерге тән программаларды вирустармен жұқтыру және қолданушыдан оның орындалуын рұқсат сұраудан тұрады. Бірақ та бұл функциялар жай программалармен белсенді орындалады. Сондықтан барлық бар резидентті күзетшілерге бірдей кемшіліктер тән – көптеген өтірік урейленулер мен мазасыз хабардың жіберілуі. Оған қарамастан, резидентті күзетші өңай келісімге келеді немесе оперативті жадыдағы өңдеушілердің модификациялауымен немесе аппараттық ресурстарға тікелей көңіл бөледі.

15.4 Алгоритмдерді тазалау

Алгоритмдерді тазалауды бейнелеу

Алгоритмді тазалау негізінде жұғу алгоритіміне қарайды. Файлдың басына жазылатын COM-вирустары, HLLP вирустары мен сол алгоритмдармен жұқтыратын вирустар, HLLP сияқты кодтың орнын ауыстыруы арқылы тазаланады.

Спутник-вирустары EXE-ге қайта құрбан-файлынның атын өзгертумен тазаланады. Оның алдында вирустың денесіде кеңейтілуі саналуы мүмкін, оның астында құрбан-файлы сақталады, ал егер вирус шифрланса, онда оған кілт болу керек.

Программалық кодты көретін вирустар, қайтымсыз файлдар құртады. Бұндай файлдарды тек өшіру керек. Бір қателіктермен қалпына келген файлдан вирус қайта іске қосылмас үшін, оның басына INT 20h командасын жазған жөн.

Орындалатын файлдың сруктурасына негізделген вирустарға қарағанда , алгоритмдер қиынырақ тазалады. Файлдардан алгоритмдерді тазалау келесі қадамдарға келеді:

1) Егер вирус шифрланса, расшифровщиктен сигнатурды табу керек, кілтті есептеу, вирус денесінен шифрды ашу керек.
2) Вируста сигнатура бар екенін тексеру керек. Сигнатура табылмаса – шығу, қайта кодты есептеу.
3) Құрбан-программасының тақырып атының жолдарын вирус денесінен есептеу керек, тақырыптың атын қалпына келтіру.
4) Программадан вирустың кодын өшіру керек, программаның өзінің кодын қалпына келтіру керек.

Резидентті вирустарды файлдан тазалау мағынасыз, олар оперативті жадыда орналасса, файлдарды қайта-қайта жұқтыруы мүмкін. Егер вирус файлдарды жабу кезінде жұқтыртса, онда антивирусты іске қосудың еш көмегі болмайды. Оперативті жадыдан вирусты тазалау алгоритмінің келесі қадамдары бар:

1) «Мен мұнда» функциясын шақыру немесе жадыда вирустың сигнатурлары барын тексеру. Егер вирус жадыда табылмаса – шығу.
2) Вирус өңдеушіден үзулерді санау немесе нағыз өңдеушіден жүйелік өзгермелі адрестерді санау.
3) Өңдеуші үзулерін қалпына келтіру, вируспен орын алған жадыдағы аумақты босату.
4) Жадыда вирус кодын үстінен басу, мысалы, нольдермен. Өңдеуші вирустың басына өту командасын жазуға болады.

Үзулердің бірін ұстайтын резидентті вирустар болады, олар өңдеуші вирустардан үзу векторларының орын босатуынан қорғайды. Сондықтан, дәл осы векторлар, вирусты операциялық жүйеде тазалаған кезде, бірінші болып құтқарылуы керек. Осыған қарап, векторларды Dos 25h немесе 35h функцияларына жүктемей, тікелей векторлар кестесіне қарау керек.

Іске қосатын вирустардың тазалау алгоритмі келесі қадамдарға тән:

1) Жүйелік аумақты есептеу, вирус сигнатурының болуын тексеру. Таппаса – шығу.
2) Вируспен сақталған кодтарды MBR немесе BOOT-қа қайта жазу, ал егер вирус оларды ешқайда сақтамаса, антивирустағы стандартты кодты жазу керек.
3) Дискіде вирустың «құйрығының» үстін басу.

Вирустардың басқа вирустарда немесе маңызды программаларда жасырынуы.

Кез-келген компьютерлік вирусты табу есебі, оның жұққан программадағы сигнатурлардың барлығын көрсетеді. Алдында кодтың шифрын ашу қажет етіледі, полиморфты шифр ашушыны детектрлеу және шифрды ашу реттілігін генерациялайды, программа тақырыбының өзгеруінсіз вирусқа кіруі – кіру нүктесінен басқаруды анализ командасына беру.

Бірақ вирустардың авторлар, антивируспен берілген сигнатурларды және берілген А вирусында орналасуын біле тұра, Б вирусын жазуы мүмкін, ол да сол сигнатур бойынша басы/соңы/кіру нүктесі қосылу арқылы болады. Ескі полифаг түрі арқылы сыпайсыз файлдардың тазалығына әкеп соғады, А вирусын біле, Б вирусын білмейді.

Басқа да «алдамшылар» болуы мүмкін

Неге бұл болуы мүмкін? Бір себепті біз атағанбыз, заң бойынша сигнатура вирустың көшірмесі болмайды, және вирус табылған болып есептеледі. Жеткілікті ұзындықтағы сигнатура (8-16 байт) және оны дұрыс таңдауда, жұқпаған файлда жұмыс істеу ықтималдығы нольге ұмтылады. Жай сөзбен айтқанда, бұл жерде жұқпаған сигнатура файлы болмайды.

Бірақ та, осы айтылғанның бәрі жұқпаған файлдарға жатады. Ешкім вирус жазуға кедергі жасамайды, ол дәл сол ығысумен, сол сигнатураны қояды. Келесіде не болады? Антивирус оған белгілі сигнатура бойынша жұмыс істейді, ол кезде файл басқа вируспен жұқпалы болады, және оның басқа ұзындығы, байты сақтайтын ығысуы, мүмкін басқа алгроитмі болады. «Тазалау» қорытындысында вирустан программаның жұмыс істеу қабілеті тоқтайды.

«Технотышқанның» тағы бір амалы, стандартты тесттік файлдардың болуында, мысалы, EICAR. Осы сигнатурдан файлды өзінің вирусының басына қойса, мұндай эффект алуға болады: осы вирусты білмейтін полифаг жұқпалы программаларды «модификацияланған тесттік файл EICAR (Вирус емес)» немесе өзгертілмейтін тесттік файл ретінде табады.

Үшінші амалы – сапасыз антивирустар, вирустың сигнатурынан басқа, «біле тұра таза» файлдардың сигнатурынан тұрады. Мысалы, TBAV сигнатуры кіру нүктесінде бар болса, белгілі вирустардағы файлдарды мүлде тексермейді. Одан да «біле тұра таза файлдар» сигнатураны қолданбаған жөн, онсыз болмаса, онда олардың барын тексеру үшін барлық вирус сигнатурын тексергеннен кейін орындалу керек.

Бұндай вирустарға қалай қарсы тұруға болады. Авторларға мұндай вирустарға қарсы тұру белгілі:

- ұзын сигнатурларды қолдану, оларға орындалуға қажетті кесек кодтарын таңдау керек;
- біреуден көп сигнатура қолдану;
- стандартты тесттік файлдарға сигнатураны таңдау, оны қарсылас керексіз эффектерден қашу үшін, кодтың кесектерін модификациялауға болатындай ету;
- Антивирусты қоса, «тигізбейтін» сигнатурларды жүйеге қоспау; стандартты тесттік файлдарды жүйеде бейнелеу, егер олорынбасатын вирус болса, тазалау уақытында, оны өшіруіне әкеледі;
- Жаңа вирус пайда болу кезінде, сол өзгерісі мен байты бар, алдынғы белгілі антивирустардікіндей, алдын ала белгілі анитивирус жүйесіне қосу, одан да екі вирустың сигнатурларын өзгертіп немесе үлкейтіп тазалаған жөн;
- «Тигізбейтін» сигнатурларды ұстайтын антивирустар, біле тұра сапасыз деп санаймыз да, түгелімен қолданудан алып тастаймыз және жадыда ұстамаймыз.

Бақылау сұрақтары:

1. Программаны статистикалық тексеруден қалай қорғауға болады?
2. Программаны статистикалық тексерудің құралдарын ата.

3. Программаны динамикалық тексерудің құралдарын ата.
4. Зиянды программалардың классификациясының параметрін атаңыз.
5. Орындалатын файлдың структурасына негізделген алгоритмдерді вирустардан тазалау қандай қадамдардан тұрады.
6. Жұқпалы объектілерде классификацияланған зиянды программалардың түрлерін атаңыз.

Пайдаланылатын әдебиет: [4] – 207- 225 б., [4] – 225-244 б.

•

4 Негізгі және қосымша әдебиет тізімі

Негізгі әдебиеттер

1. Тұрым А.Ш., Мұстафина Б.М., Ақпарат қорғау және қауіпсіздендіру негіздері. – Алматы: Алматы энергетика және байланыс институты, 2002ж.
2. Романец Ю.В., Тимофеев П.А., Шаньгин В.Ф. Защита информации в компьютерных системах и сетях. –М.: РАДИО И СВЯЗЬ, 1999.
3. Алферов А.П., Зубов А.Ю., Кузьмин А.С., Черемушкин А.В. Основы криптографии: Учебное пособие. – М.: Гелиос АРВ, 2002. – 480 с.
4. Бурдаев О.В., Иванов М.А., Тетерин И.И. Ассемблер в задачах защиты информации. Под ред. Ю.В.Жукова. – М.: КУДИЦ-ОБРАЗ, 2002. – 320 с.
5. Венбо Мао. Современная криптография: теория и практика. – М.: Издательский дом “Вильямс”, 2005.- 768 с.
6. Анин Б.Ю. Защита компьютерной информации. – СПб.: БХВ – Санкт-Петербург, 2000. – 384 с.
7. Зегжда Д. П., Ивашко А. М. Основы безопасности информационных систем.– М.: Горячая линия – Телеком, 2000.– 452 с.
8. Брюс Шнайер. Прикладная криптография. . –М.: «Солон-Р», 2000.
9. Бабаш А.В., Шанкин Г.П. Криптография. –М.: «Солон-Р», 2002.
10. Ростовцев А.Г., Маховенко Е.Б. Теоретическая криптография. –С.-П.:

Қосымша әдебиеттер

1. Гундарь К.Ю., Гундарь А.Ю., Янишевский Д.А. Защита информации в компьютерных системах. – К.: “Корнейчук”, 2000. – 152 с.
2. Столлингс В. Криптография и защита сетей: принципы и практика. Пер с англ. – М.: Издательский дом “Вильямс”, 2001. – 672 с.
3. Иванов М.А. Криптографические методы защиты информации в компьютерных системах и сетях. –М.: КУДИЦ-ОБРАЗ, 2001.
4. Н. Фергюсон, Б.Шнайдер. Практическая криптография. – М.: Издательский дом “Вильямс”, 2005.- 424 с.
5. Список дополнительной литературы
6. Зегжда Д. П., Ивашко А. М. Основы безопасности информационных систем.– М.: Горячая линия – Телеком, 2000.– 452 с.
7. Грушо А. А., Тимонина Е. Е. Теоретические основы защиты информации.– М.: Издательств агентства “Яхтсмен”, 1996.– 71 с.

8. Хорев А. А. Защита информации от утечки по техническим каналам. Часть 1. Технические каналы утечки информации. Учебное пособие.– М.: Гостехкомиссия России, 1998.– 320 с.
9. Уолкер Б. Дж., Блейк Я.Ф. Безопасность ЭВМ и организация их защиты: Пер. с англ.– М.: Связь, 1980.– 112 с.
10. Ященко В.В. Введение в криптографию. Новые математические дисциплины. –М.: МЦНМО Питер, 2001.
11. Касперский К. Фундаментальные основы хакерства (искусство дизассемблирования). –М.: Солон-Р, 2002.
12. Расторгуев С.П. Программные методы защиты информации в компьютерах и сетях.-М.:"Яхтсмен",1993. с. 187.