УДК 004.056.5
ГРНТИ 81.93.29

«HBC-256» ХЕШТЕУ АЛГОРИТМІНДЕГІ S-БЛОКТАРДЫҢ ДИФФЕРЕНЦИАЛДЫ КРИПТОТАЛДАУДАҒЫ КРИТИКАЛЫҚ НҮКТЕЛЕРІН ЗЕРТТЕУ

С. Нысанбаева[1], Қ.Сақан[1,2]
[1]Ақпараттық және есептеуіш технологиялар институты ҒК БҒМ, Алматы, Қазақстан,
[2]әл-Фараби атындағы Қазақ Ұлттық Университеті, Алматы, Қазақстан
E-mail: kairat_sks@mail.ru
[1]ORCID ID: https://orcid.org/0000-0002-5835-4958
[2]ORCID ID: https://orcid.org/0000-0002-6812-6000

Аңдатпа. Бұл жұмыста ҚР БҒМ КН Ақпараттық және есептеуіш технологиялар институтында жасалған хештеу алгоритмдерінің бірі «HBC-256» алгоритміне дифференциалдық криптоталдау жүргізу қарастырылған. HBC-256 алгоритмі қысу функциясының CF (Compression Function) блоктық шифры негізінде жасалған және ұзындығы 256 биттік хэш мәнін шығарады. Кез келген жаңа криптографиялық схема сияқты HBC-256 алгоритмі оның де криптографиялық қасиеттерінің оң екендігін растау үшін мұқият зерттеуді қажет етеді, атап айтқанда: қайтымсыздық және бірінші және екінші типтегі коллизияға төзімділік. Жұмыстың нәтиже алу үшін барысында C++ бағдарламалау тілдерін қолдана отырып, жаңа хэштеу алгоритмін бағдарламалық қамтамасыз ету жүзеге асырылды. HBC-256 хэштеу алгоритмі үшін сызықтық емес хештеу түйіндерінің (s-блоктарының) дифференциалдық қасиеттері қарастырылады. Раундтық сипаттамаларды құрудың әртүрлі нұсқалары қарастырылады. Дифференциалды криптоанализдің негізгі қасиеттерін ескере отырып және басқа ғылыми жұмыстардың қорытындыларына сүйене отырып, ашық және сәйкес жабық мәтіндер жұптарының критикалық нүктелердегі жай-күйі зерттелген. Мақаланың қорытындысында хештеу алгоритмнің S-блоктары айырымдар кестесіндегі критикалық нүктелер арқылы коллизия тудыруға мүмкін болатын жағдайларға төзімді деген тұжырым жасалады.

Кілттік сөздер: хештеу алгоритмдері, хештеу алгоритмдеріне қойылатын талаптар, криптоталдаудың негізгі әдістері, дифференциальды криптоталдау.

Кіріспе

Қазіргі ақпараттық әлемде ақпараттың сенімділігі мен қауіпсіздігін қамтамасыз ету басты құндылықтардың бірі болып табылады. Әртүрлі қауіпсіздік мәселелеріне қатысатын негізгі криптографиялық түрлендірулердің бірі болып хеш-функциялар саналады – кел-келген еркін ұзындықтағы кіріс деректер массивін бекітілген ұзындықтағы бірегей тізбегіне түрлендіретін бір жақты математикалық түрлендірулер. Қазіргі заманғы хеш-функциялар пайдаланушылар аутентификациясы [1-5], деректердің тұтастығын бақылау [1, 6, 7], электрондық қолтаңба [1, 8], криптовалюта операцияларын қалыптастыру [9-12] сияқты ақпараттық қауіпсіздіктің әртүрлі процедураларын жүзеге асыру үшін қолданылады. Пайдаланушы аутентификация мәселелерін хеш мәндерінің деректерге кездейсоқ компонентті («криптографиялық тұз») міндетті түрде қосу арқылы тексерушіде аутентификация туралы ақпаратты қауіпсіз сақтау үшін пайдаланылады. Бұрын есептелген хеш мәндерін үлкен деректер жиындарының тұтастығын жылдам тексеруді қамтамасыз ету немесе стегоконтейнерлердің тұтастығын тексеру үшін де пайдалануға болады. Электрондық қолтаңбаны генерациялау кезінде хеш-функциялар қол қойылған деректердің бекітілген ұзындықтағы бірегей бит тізбегіне салыстыруын генерациялау үшін пайдаланылады, бұл болашақта олардың өзгермейтіндігіне сенімділік берілген кепілдікпен деректерге қол қою жылдамдығын арттыруға мүмкіндік береді.

Криптографиялық хэш функцияларына қойылатын талаптар:

Өнімділік: кез келген M хабарламасы үшін нақты уақытта h хэш мәнін тиімді

есептеуге болады.

Қайтымсыздық (бірбағыттылық): h хеш мәнін ескере отырып, h=H(M) болатындай

M хабарламасын табу есептеуде қиын болуы тиіс.

Әлсіз мағынада беріктілік: бізге белгілі M хабарламасы берілгенде, h=H(M)=H(M') болатындай M' хабарламасын қалыптастыру (есептеу) қиын болуы тиіс.

Күшті мағынада беріктілік: H(M)=H(M') болатындай M және M' кездейсоқ хабарламаларын табу қиын болуы тиіс.

Қазіргі уақытта хештеу функцияларын жасау келесі архитектуралық нұсқаларды пайдалануға негізделген:

1) әртүрлі сызықты емес (биттік) бульдік функцияларды қолдану;

2) блоктық симметриялық шифрлау алгоритмдері түріндегі қысу функциясына негізделген Меркле-Дамгард құрылымын пайдалану;

3) «Криптографиялық губка» құрылымын пайдалау;

4) арнайлы бағытталған құрылымдар.

Ақпараттық қауіпсіздік саласындағы ұсыныстарға, стандарттарға және халықаралық тәжірибеге сүйене отырып, хеш-функциялардың криптографиялық алгоритмдерінің беріктілігін талдау бойынша зерттеулер өзекті болып табылады және осы білім саласындағы ағымдағы жағдайды бағалау бойынша қолданылатын әрбір алгоритмге немесе оның модификациясына қатысты үздіксіз жұмысты талап етеді деп айтуға болады.

Әлі күнде де блоктық шифр ақпараттың құпиялылығын қамтамасыз етудің маңызды құралы болып табылады. Симметриялы блокты алгоритмдердің құрылымы сызықты және сызықты емес түйіндерден тұрады. Тәжірибеде сызықты емес түйін мәселесін S-блок ауыстырулары көп пайдаланылуда. Қазіргі уақытта радиожиілікті сәйкестендіру жүйелері (RFID) сияқты шектеулі ресурстарға ие құрылғылардың қауіпсіздігін арттыруға үлкен мәселелер туындауда. Аз ресурсты құрылғыларда көбінесе 4 биттік S-блок ауыстырулары пайдаланылады.

Талдау мен нәтижелер

Біз ұсынып отырған HBC-256 хештеу алгоритміндегі жаңа CF шифрлау алгоритмінде оның бағдарламалы-аппараттық және аппараттық тұрғыда икемді жүзеге асырылу мақсатында және бұл алгоритмді блоктық шифрлар негізінде хеш алгоритмдерін жасауда пайдалануды ойластыра отырып, S-блок ауыстыруларын басқа жолмен іске асыру қарастырылған. Осы алгоритм құрамындағы қолданылатын төрт 4-биттік S-блоктарды матрица элементтерінің орналасу жаңдайына байланысты белгілі бір тәртіппен қолдану - ұсынылған алгоритмнің құрылысының, соның ішінде кілт жасау алгоритмінің де ажырамас бөлігі болып табылады [13].

HBC-256 хештеу алгоритмінде сызықты емес биективті түрлендіру S-блок SBOX процедурасы арқылы анықталады. S0, S1, S2, S3 төрт алмастыру берілген, мұнда $S_i: \mathbb{Z}_{2^4} \rightarrow \mathbb{Z}_{2^4}, i = 0, \dots, 3$.. Жұмыс үшін кестеге сәйкес төрт «алтын» S-блок таңдалды. Олар төменгі Кесте-1-де көрсетілген.

Кесте 2 – төрт «алтын» S-блоктар

Зерттеу жұмысы Кесте-2-дегі төрт S-блоктарға дифференциалдық талдау әдісі арқылы ұсынылған HBC-256 хештеу алгоритмінің коллизияларға төзімділігін бағалауға бағыттаймыз.

Дифференциалды криптоталдау әдісін алғаш рет DES шифрлау стандартын талдау үшін Э.Бихам мен А.Шамир ұсынған [14, 15].

Дифференциалды криптоталдау шифрлаудың әртүрлі кезеңдеріндегі шифрланған мәндер арасындағы айырмашылықтарды түрлендіруді зерттеуге негізделген. Айырмашылық ретінде, әдетте, разрядтық қосынды модулі 2 операциясы (xor операциясы) қолданылады, оның ішінде $2n$ модулінің айырмасын талдау да мүмкін. Дифференциалды талдау симметриялық блоктық шифрларды және басқа криптографиялық примитивтерді, атап айтқанда хеш-функцияларды және ағындық шифрларды криптоталдау әдісі болып табылады [16]. Талдаудың бұл түрін қолдану үшін алгоритмнің барлық сызықтық емес элементтері – S-блок алмастыру блоктары негізінде xor операциясы немесе 2n қосу модулі немесе басқалары үшін дифференциалдық қасиеттердің кестелерін (xor операциясы үшін – айырымдар кестелерін) құру қажет.

Жалпы жағдайда, кез-келген шифрлау немесе хештеу алгоритмдеріне қатысты дифференциалды криптоталдау әдісін қолдану келесі қадамдарды жүргізуге негізделген:

1. Сызықты емес элементтерді талдау және олар үшін ең ықтимал айырымдарын анықтау.

2. Қарапайымнан күрделіге, яғни 1 раундтан n раундқа дейін көп раундты сипаттаманы (кіріс айырымы – шығыс айырымы) тұрғызу. Құрылған сипаттаманың пайда болу ықтималдығын анықтау.

3. Мәтіндердің дұрыс жұптарын, яғни кіріс мәндерінің қосындысы кіріс айырмасының мәніне сәйкес келетін мәтіндерді, ал шығыс мәндерінің қосындысы шығыс айырымдарымен сәйкес келетін мәтіндерді іздеу процесін жүргізу.

Бірінші кезекте қарастырылып отырған төрт S-блок алмастыруының xor операциясына қатысты айырымдар кестесін тұрғызайық (Кестелерде критикалық нүктелер қызыл боямен ерекшеленген):

Кесте 2 – S0-блок айырымдар кестесі

Кесте 3 – S1-блок айырымдар кестесі

Кесте 4 – S2-блок айырымдар кестесі

Кесте 5 – S3-блок айырымдар кестесі

Енді критикалық нүктелерлің кейбіреулеріне талдау жасап көрейік. Мысалы, S0-блогі үшін (1,5) қиылысындағы 4 рет кездесетін айырымдар жұбы: d1=1 және d2=5 үшін критикалық нүктелер төмендегідей болады.

Кесте 5 – S0-блок айырымдар кестесі негізінде критикалық нүктелерді анықтау

Po+ P1=1= d1	S0(Po)+ S0(P1)= d2	Критикалық нүктелер
0001+0000	1111=F	
0010+0011	0111=3	
0100+0101	0101=5= d2	0100 және 0101
0110+0111	0101=5= d2	0110 және 0111
1000+1001	1100=C	
1010+1011	0110=6	
1101+1100	1101=D	
1110+1111	1011=B	

Біз талдауды жалғастырайық. Ол үшін біз критикалық нүктелер 4=0100, 5=0101, 6=0110, 7=0111 ескеріп, тек басында ғана ерекшеленетін 4 ашық мәтінді таңдайық:

P1=4030303030303030303030303030303130303030303030303030303030303230303030303030303030303030303031

P2=5030303030303030303030303030303130303030303030303030303030303230303030303030303030303030303031

P3=6030303030303030303030303030303130303030303030303030303030303230303030303030303030303030303031

P4=7030303030303030303030303030303130303030303030303030303030303230303030303030303030303030303031

НВС-256 алгоритмінің 1 циклін жүргізген соң сәйкес шифрмәтіндерді аламыз:

1-ші раундтан кейін 2-ші және 3-ші блоктардағы бірдей бөлімдер, өйткені олар өзгерістерге ұшырамайды (ерекше бояумен боялған бөліктер):

C1=DD9251EA6DC04DA02A2C4DEEEF69ECDDEF631837B7A0C6DE4CDF315B04046EFA48A13E7F1519D0E4301FC9CB1F095F87

C2=CB9251816DC0CDA02A7A4DEEB169EC6CEF630437B7D0C6DE38DF31A6040463FA48F23E7FF419D047301F74CB1FCC5F87

C3=029251AD6DC074A02AEC4DEEFB69EC65EF63ED37B78DC6DEEFDF31020404CFFA484A3E7F2D19D0C6301FBECB1FFC5F87

C4=D49251E26DC0D0A02AE54DEE7B69EC53EF630037B7D4C6DEE6DF314C040485FA486A3E7F3919D03F301FE2CB1F6F5F87

2-раундтан кейін:

C1=9CDEC25066B9042733D96A00DF1E41D2C925FE1A17E36DA0410F99D0FC8771EC1CD548511F9CE480B85D6545939291FE

C2=2F3A1FCE0D3EC0E03B8CFCA1F6C40DEF065E898CE8F7F7A0E6C981EB0E6A404EF6E4FC22F56317F9916381DEBD86854F4

C3=C2B0F90A4EE751A8A9343F39599CE898DB899BC9C7EC81088E02EAE155E36DEC26B28FD5EBA181F8FA296DD5196DE22D

C4=27EDE0D4201E4D1E86C3ECD02D00D1CB262957A70695EB9B0009F592862711314D0A44B0FD5AA76B5D42FA847A3ABCBE

3-раундтан кейін:

C1=F0175EB6D6E05EC15057A967AD9472462393E9228D2B6EC4BC92AE075F24EEC73E47BB3CC31FD8CFC89DAC792E41BF02

C2=35D28BDCEC9A548BBAFCD062F672AC04E315CE3FCB8A2717970B193B0BEDEDD0E0550F2AAB2D5E5EDCF85351F5AC98E1

C3=46F39EEF58E341DEB34D2337B1800FAEFD48F0DD92B1525C5077528D75785D540ABF4238D8D143D48146FD3E3E27A404

C4=4E408A76F2D99AC06E1B920362DA32C87EFB76C7EE4C9CEEAC9FAC7283972B513E40101EC9E7860C48EDF9F17A894DB4

4-раундтан кейін:

C1=C8CDDC1EE712FE904DD228FEE42679568D69B879389D74069682FC3248F2B512BAE37311E1492D05F8AEDB868D51647D

C2=CFBBE27FA5D8FACDD3F119548E3E3023F07A7B61783649D87730FE3B25CC76798DECA53A147A8B91B3240EE5D8E67147

C3=449C5018D2C2ED8BD10E83CCAF3F3D77D20F8FF5BFBCD3440CC17ABD42215DDE3E807F88E8C10B2FD05B1843F80566F3

C4=68A1B2F3150B831C1D2DFBE2E9A0FFBE7BADE06766B8D0BB4FA3F6EB00074CCAD264FB99021841634ED8082E0B78185D

Енді 4-раундты D21= C1+C2, D22= C3+C4 ескеріп талдайық:

D21=07763E6142CA045D9E2331AA6A1849757D13C31840AB3DDEE1B202096D3EC36B370FD62BF533A6944B8AD56355B7153A

D22=2C3DE2EBC7C96E97CC23782E469FC2C9A9A26F92D90403FF43628C5642261114ECE48411EAD94A4C9E83106DF37D7EAE

Нәтижедегі мәндерді салыстырсақ, 23-байты кездейсоқтық сәйкестік.

Қорытынды: D21≠ D22 болғандықтан, критикалық нүктелер хештеу процесінен кейін коллизия тудындатпайды.

Келесі критикалық нүктелерді қарастырайық. Ол үшін S0-блоктағы 4 рет кездесетін (6,8) қиылысын қарайық. d1=6=0110 и d2=8=1000 үшін критикалық нүктелерді табайық:

Значений Po =>

Po+ P1=1= d1	S0(Po)+ S0(P1)= d2	Критикалық нүктелер
0000+0110	0110=6	
0001+0111	1100=C	
0010+0100	0111=7	
0011+0101	0001=1	
1000+1110	1000=8	1000=8, 1110=E
1001+1111	1111=F	
1010+1100	1000=8	1010=A, 1100=C
1011+1101	0011=3	

Әрі қарай біз критикалық нүктелер 8=1000, E=1110, A=1010, C=1100 ескеріп, тек басында ғана ерекшеленетін 4 ашық мәтінді таңдайық:

P1=80303030303030303030303030303130303030303030303030303030303032303030303030303030303030303031

P2=E0303030303030303030303030303130303030303030303030303030303032303030303030303030303030303031

P3=A0303030303030303030303030303130303030303030303030303030303032303030303030303030303030303031

P4=C0303030303030303030303030303130303030303030303030303030303032303030303030303030303030303031

Сәйкес шифрмәтіндер 1-раундтан соң төмендегідей болады:

C1=639251F26DC0C8A02AA24DEEB969EC07EF636A37B797C6DE05DF314A040421FA48E33E7F7319D0CA301FCCCB1FED5F87

C2=A69251F16DC090A02A9C4DEE2569EC0EEF631D37B74BC6DE64DF31590404D9FA48253E7F1A19D055301F48CB1FF95F87

C3=8C9251266DC0EFA02A504DEE2269ECAEEF63F837B70DC6DE72DF311804044BFA48CD3E7FDF19D065301F2BCB1F4C5F87

C4=6992515F6DC04CA02AAA4DEE1769ECB5EF637E37B7F0C6DE16DF31690404C8FA48F03E7F3319D0B8301FEECB1F4E5F87

4-раундтан соңғы шифрмәтін:

C1=90C5CA858FB3DEB493AE795C1A2D6044B89F0A08334AAE8A636C408E2406DC83022D971DA1B7B8B5CA57EDC16C9784B7

C2=1DB82C755A45E8C6ADC8FB414869D8A1CC8DF69D5C2374400683965D59E62C811F01146B3109B5621C5BF8132D0C3DDE

C3=9BA8F543FA45553C425ED785949943CE75D2E778E7AD852DC200DD519E099E13024A23D3ECFF81217AA8FBE181017ACF

C4=14A83D9FFF052C972BE94A043DF63A7EA65D635B53E69ECA25D6D5C371D37ED487EE91C036782561361A364DEEED75E6

Осы нәтижені D21= C1+C2, D22= C3+C4 тексерейік, сонда:

D21=8D7DE6F0D5F636723E66821D5244B8E57412FC956F69DACA65EFD6D37DE0F0021D2C837690BE0DD7D60C15D2419BB969

D22=8F00C8DC054079AB69B79D81A96F79B0D38F8423B44B1BE7E7D60892EFDAE0C785A4B213DA87A4404CB2CDAC6FEC0F29

Бұл мысалда да D21≠D22 болғандықтан, қарастырылған критикалық нүктелер – 1000=8, 1110=E, 1010=A, 1100=C хештеу процесінен кейін коллизия тудындатпайды.

Қорытынды

Бұл мақалада жаңа CF симметриялық блокты шифрлау алгоритміне негізделген НВС-256 хештеу алгоритмінде қолданылатын сызықты емес криптографиялық приминив - төрт 4-биттік S-блок зерттелген. Қазіргі уақытта алгоритмнің криптоберіктілігін қамтамасыз ететін осы компоненттер дифференциялдық криптоталдау әдісі арқылы беріктілігі бағаланды. Айырымдар кестесіндегі критикалық нүктелерге тәжірибе жүзінде талдау жүргізіліп, алғашқы бейнелер арасында коллизия туындату жағдайы талданды. Осындай талдауды барлық төрт S-блоктағы критикалық нүктелер үшін жүргізіп шығуға болады. Егер де алынған нәтиже D21= C1+C2, D22= C3+C4 есептелініп, D21≠D22 орындалатын болса, хештеу алгоритмдегі сызықты емес түрлендіру S-тар тудыратын критикалық нүктелер коллизия туындауға мүмкіндік бермейді. Зерттеу барысында D21= C1+C2, D22= C3+C4 мәндерін теңдіктердің салыстырмалы түрде орындалуы

ықтималдығын да ескерген жөн.

Алғыс

Жұмыс OR11465439 «Электрондық цифрлы қолтаңба үшін еркін ұзындықтағы хэштеу алгоритмін құру мен зерттеу және олардың беріктілігін бағалау» бағдарламалық-нысаналық қаржыландыру ғылыми жобасы аясында жүргізілді.

Әдебиеттер тізімі:

[1] Nita, S.L., Mihailescu, M.I. Hash Functions. In: Cryptography and Cryptanalysis in Java. Apress, Berkeley, CA. 2022. https://doi.org/10.1007/978-1-4842-8105-5_8

[2] Kheshaifaty N., Gutub A., Engineering Graphical Captcha and AES Crypto Hash Functions for Secure Online Authentication. Journal of Engineering Research. https://kuwaitjournals.org/jer/index.php/JER/article/download/13761/2687

[3] Farshim, P., Tessaro, S. Password Hashing and Preprocessing. In: Canteaut, A., Standaert, FX. (eds) Advances in Cryptology – EUROCRYPT 2021. Lecture Notes in Computer Science. 2021. 12697. Springer, Cham. https://doi.org/10.1007/978-3-030-77886-6_3

[4] Herrera J., Ali M. L. Concerns and Security for Hashing Passwords. 2018 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON). 2018. 861-865. doi: 10.1109/UEMCON.2018.8796720.

[5] Brogada M. D., Sison A. M., Medina R. P. Head and Tail Technique for Hashing Passwords. IEEE 11th International Conference on Communication Software and Networks (ICCSN), 2019. 805-810. doi: 10.1109/ICCSN.2019.8905384.

[6] Fomichev V., Bobrovskiy, D., Koreneva, A. et al. Data integrity algorithm based on additive generators and hash function. J Comput Virol Hack Tech. 2022. 18. 31–41. https://doi.org/10.1007/s11416-021-00405-y

[7] Wang J., Luo W., Hu Y., Jiang H. PN-HASH: An Immune-Inspired Scheme for Data Integrity Check. 12th International Conference on Advanced Computational Intelligence (ICACI), 2020. 340-348. doi: 10.1109/ICACI49185.2020.9177796.

[8] Thomas Espitau. Mitaka: Faster, Simpler, Parallelizable and Maskable Hash-and-Sign Signatures on NTRU Lattices. In Proceedings of the 8th ACM on ASIA Public-Key Cryptography Workshop (APKC '21). Association for Computing Machinery, New York, NY, USA, 1. https://doi.org/10.1145/3457338.3458293

[9] Belej, O., Staniec, K., Więckowski, T. The Need to Use a Hash Function to Build a Crypto Algorithm for Blockchain. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds) Theory and Applications of Dependable Computer Systems. DepCoS-RELCOMEX 2020. Advances in Intelligent Systems and Computing. 2020. 1173. Springer, Cham. https://doi.org/10.1007/978-3-030-48256-5_6

[10] N, Y., P, M. Radial kernelized regressive merkle–damgård cryptographic hash blockchain for secure data transmission with IoT sensor node. Peer-to-Peer Netw. Appl. 2021. 14. 1998–2010. https://doi.org/10.1007/s12083-021-01135-0

[11] Chilambarasan, N.R., Kangaiammal, A. Matyas–Meyer–Oseas Skein Cryptographic Hash Blockchain-Based Secure Access Control for E-Learning in Cloud. In: Suma, V., Chen, J.IZ., Baig, Z., Wang, H. (eds) Inventive Systems and Control. Lecture Notes in Networks and Systems. 2021. 204. Springer, Singapore. https://doi.org/10.1007/978-981-16-1395-1_65

[12] Patil, Harsha Kundan. Blockchain Technology-Security Booster. Blockchain Applications in IoT Security, edited by Harshita Patel and Ghanshyam Singh Thakur, IGI Global, 2021. 128-139. https://doi.org/10.4018/978-1-7998-2414-5.ch008

[13] Sakan K., Nyssanbayeva S., Kapalova N., Algazy K., Khompysh A., Dyusenbayev D. Development and analysis of the new hashing algorithm based on block cipher. Eastern-European Journal of Enterprise Technologies. Ukraine. 2022. 2/9(116). 60–73. https://doi.org/10.15587/1729-4061.2022.252060.

[14] Biham, E., Shamir, A. Differential cryptanalysis of the full 16-round DES. Advances in cryptology, proceedings of CRYPTO'92. 1992. 487-496.

[15] EliBiham, AdiShamir Differential Cryptanalysis of Hash Functions. Differential Cryptanalysis

of The Data Encryption Standard, Springer, 1993, ISBN 978-1-4613-0314-6. 133 – 148.

[16] Eli Biham, Orr Dunkelman, Differential Cryptanalysis in Stream Ciphers. Cryptology ePrint Archive, Report. 2007.218. http://eprint.iacr.org/

STUDY OF CRITICAL POINTS IN THE DIFFERENTIAL CRYPTOANALYSIS OF S-BLOCKS IN HASHING ALGORITHM "HBC-256"

S.Nyssanbaeva[1], K.S .Sakan[1,2]
[1]Institute of Information and Computational Technologies SC MES RK, Almaty, Kazakhstan,
[2]Kazakh National University named after al-Farabi, Almaty, Kazakhstan
E-mail: kairat_sks@mail.ru
[1]ORCID ID: https://orcid.org/0000-0002-5835-4958
[2]ORCID ID: https://orcid.org/0000-0002-6812-6000

Abstract. In this work described a differential cryptanalysis of the "HBC-256" algorithm one of the hashing algorithms developed at the Institute of Information and Computing Technologies of the RK MES CS, is provided. The HBC-256 algorithm is based on the block cipher of the CF (Compression Functions) and generates a 256-bit hash value. Like any new cryptographic structure, the HBC-256 algorithm requires careful research in order to confirm its cryptographic properties, namely: irreversibility and resistance to collisions of the first and second kind. As a result of the work, a software implementation of a new hashing algorithm using the C++ programming language was obtained. Differential properties of nonlinear hashing nodes (S-blocks) are considered for the HBC-256 hashing algorithm. Various options for constructing round characteristics are considered. Taking into account the main properties of differential cryptanalysis and based on the conclusions of other scientific works, the situation of pairs of open and corresponding closed texts at critical points is studied. In the conclusion of the article, it is concluded that the S-boxes of the hashing algorithm are resistant to situations where it is possible to cause a collision through critical points in the table of differences.

Keywords: hash algorithms, requirements for hash algorithms, basic methods of cryptanalysis, differential cryptanalysis.

ИССЛЕДОВАНИЕ КРИТИЧЕСКИХ ТОЧЕК ПРИ ДИФФЕРЕНЦИАЛЬНОМ КРИПТОАНАЛИЗЕ S-БЛОКОВ В АЛГОРИТМЕ ХЕШИРОВАНИЯ "HBC-256"

С.Е. Нысанбаева[1], Қ.С. Сақан[1,2]
[1]Институт информационных и вычислительных технологий МОН РК,Алматы, Казахстан
[2]Казахский национальный университет им. аль-Фараби, Алматы, Казахстан,
E-mail: kairat_sks@mail.ru
[1]ORCID ID: https://orcid.org/0000-0002-5835-4958
[2]ORCID ID: https://orcid.org/0000-0002-6812-6000

Аннотация. В данной работе проводится дифференциальный криптоанализ алгоритма «HBC-256», одного из алгоритмов хеширования, разработанного в Институте информационных и вычислительных технологий МВОН Республики Казахстан. Алгоритм HBC-256 разработан на основе блочного шифра функции сжатия CF (Compression Function) и вырабатывает хэш-значение длиной 256 бит. Как любая новая криптографическая структура алгоритм HBC-256 требуют тщательного исследования с целью подтверждения его криптографических свойств, а именно: необратимости и устойчивости к коллизиям первого и второго рода. В процессе исследования алгоритма была использована программная реализация нового алгоритма хэширования с использованием языка программирования C++. Для алгоритма хэширования HBC-256 рассмотрены дифференциальные свойства нелинейных узлов хеширования (S-блоков). Рассмотрены различные варианты построения раундовых характеристик. С учетом основных свойств дифференциального криптоанализа и на основании выводов других научных работ исследуется ситуаций пар открытых и соответствующих им закрытых текстов в критических точках. В заключении статьи делается вывод о том, что S-блоки алгоритма хеширования устойчивы к дифференциальному криптоанализу и имеющие критические точки после хеширования не порождают коллизию.

Ключевые слова: симметричные алгоритмы шифрования, требования к алгоритмам шифрования, основные методы криптоанализа, атака методом бумеранга.

Авторлар жайында мәлімет:

Қаз: Сақан Қайрат – Әл-Фараби атындағы Қазақ ұлттық университетінің докторанты, kairat_sks@mail.ru

Рус: Сакан Кайрат – докторант Казахского национального университета им. аль-Фараби, kairat_sks@mail.ru

Англ: Sakan Kairat – a doctoral student at Al-Farabi Kazakh National University, kairat_sks@mail.ru

Қаз: Нысанбаева Сауле Екребулановна – Ақпараттық және есептеуіш технологиялар институты, т.ғ.д., профессор

Рус: Нысанбаева Сауле Екребулановна – Институт информационных и вычислительных технологий, д.т.н., профессор

Англ: Nysanbayeva Saule Ekrebulanovna – Institute of Information and Computing Technologies, Doctor of Technical Sciences, Professor