ՀՀ ԳԱԱ ԻՆՖՈՐՄԱՏԻԿԱՅԻ ԵՎ ԱՎՏՈՄԱՏԱՑՄԱՆ ՊՐՈԲԼԵՄՆԵՐԻ ԻՆՍՏԻՏՈՒՏ

Սոխակյան Տիգրան Վահանի

ԵՐԿՈՒ ՄԱՍՆԱԿՑՈՎ ԱՆՎՏԱՆԳ ՀԱՇՎԱՐԿՆԵՐԻ ՀԱՄԱԿԱՐԳԻ ՆԱԽԱԳԾՈՒՄ և ԻՐԱԿԱՆԱՑՈՒՄ ԳԱՂՏՆԻՈՒԹՅՈՒՆ ՊԱՀԱՆՋՈՂ ԿԻՐԱՌՈՒԹՅՈՒՆՆԵՐԻ ՀԱՄԱՐ

Ե.13.05 «Մաթեմատիկական մոդելավորում, թվային մեթոդներ և ծրագրերի համալիրներ» մասնագիտությամբ տեխնիկական գիտությունների թեկնածուի գիտական աստիճանի հայցման ատենախոսության

ՍԵՂՄԱԳԻՐ

Երևան – 2016

INSTITUTE FOR INFORMATICS AND AUTOMATION PROBLEMS OF NAS RA

Tigran Vahan Sokhakyan

DESIGN AND IMPLEMENTATION OF SECURE TWO-PARTY COMPUTATION FRAMEWORK FOR PRIVACY-PRESERVING APPLICATIONS

ABSTRACT

For obtaining candidate degree in technical sciences in specialty 05.13.05 “Mathematical modeling, numerical methods and software complexes”

Yerevan - 2016

Ատենախոսության թեման հաստատվել է Հայ-Ռուսական համալսարանում

Գիտական ղեկավար՝	տեխ.գիտ.դոկտոր Գ. Հ. Խաչատրյան
Պաշտոնական ընդդիմախոսներ՝	տեխ.գիտ.դոկտոր Դ. Գ. Ասատրյան
	տեխ.գիտ.թեկնածու Վ. Գ. Մարկարով
Առաջատար կազմակերպություն՝	Երևանի մաթեմատիկական մեքենաների գիտահետազոտական ինստիտուտ

Պաշտպանությունը կայանալու է 2016թ. հունիսի 8-ին, ժ. 17:00-ին ՀՀ ԳԱԱ Ինֆորմատիկայի և ավտոմատացման պրոբլեմների ինստիտուտում գործող 037 «Ինֆորմատիկա և հաշվողական համակարգեր» մասնագիտական խորհրդի նիստում հետևյալ հասցեով՝ Երևան, 0014, Պ. Սևակի 1:

Ատենախոսությանը կարելի է ծանոթանալ ՀՀ ԳԱԱ ԻԱՊԻ գրադարանում: Սեղմագիրը առաքված է 2016թ. մայիսի 8-ին:

Մասնագիտական խորհրդի
գիտական քարտուղար, ֆ.մ.գ.դ. Հ. Գ. Սարուխանյան

The subject of the dissertation has been approved in the Armenian-Russian University

Scientific advisor:	Doctor of Tech. Sci. G. H. Khachatryan
Official opponents:	Doctor of Tech. Sci. D. G. Asatryan
	Candidate of Tech. Sci. V. G. Markarov
Leading organization:	Yerevan Computer Research and Development Institute

The defense of thesis will take place on 8 June, 2016 at 17:00 in Institute for Informatics and Automation Problems of NAS RA, during the session of the specialized council 037 “Informatics and Computing Systems”, address: 0014, Yerevan, P. Sevak str. 1.

The thesis is available in library of IIAP of NAS RA.
The abstract is sent on 8 May 2016.

Scientific Secretary of the specialized council

math. sciences Doctor of phys. H.G.Sarukhanyan

ԱՇԽԱՏԱՆՔԻ ԸՆԴՀԱՆՈՒՐ ԲՆՈՒԹԱԳԻՐԸ

Թեմայի արդիականությունը

Կապի ենթակառուցվածքների վիթխարի զարգացումը վերջին տասնամյակներում առաջ է բերել կազմակերպությունների, էլեկտրոնային ծառայությունների և անհատների համագործակցության այնպիսի սցենարներ, որոնք ներգրավվում են մասնակիցների կողմից տրամադրվող գաղտնի ինֆորմացիա: Տրամադրվող ինֆորմացիայի գաղտնիությունը ժամանակ առ ժամանակ հենվում է երրորդ վստահված կողմի առկայութան վրա, օրինակ. էլեկտրոնային աճուրդների ժամանակ eBay համակարգը կարող է հանդես գալ որպես այդպիսի վստահված կողմ: Չնայած նրան, որ այդպիսի կազմակերպությունների վարքը հստակ կանոնակարգվում և վերահսկվում է առկա օրենսդրական դաշտի շրջանակներում, ոչ բոլոր օգտատերերն են պատրաստ տրամադրել գաղտնի ինֆորմացիան այլ, թեկուզ և վստահված կողմին, հատկապես, երբ այդ ինֆորմացիան մեծ արժեք է ներկայացնում կամ կարող է գործածվել օգտատիրոջը վնաս հասցնելու նպատակով:

Երկու մասնակցով անվտանգ հաշվարկների հաղորդակարգերը թույլ են տալիս միմյանց չվստահող մասնակիցներին կատարել հաշվարկներ յուրաքանչյուրի գաղտնի մուտքային տվյալների վրա, այնպես, որ հաղորդակարգի կատարման վերջում մասնակիցներին հայտնի է դառնում միայն կատարվող հաշվարկների արդյունքը՝ առանց մուտքային տվյալների բացահայտման: Անցած դարի ութանասունականներին առաջարկվել են երկու մասնակցով անվտանգ հաշվարկների մի քանի հաղորդակարգեր, որոնցում մուտքային տվյալների վրա կատարվող հաշվարկները ներկայացում են բուլյան սխեմայի միջոցով:

Առաջարկված անվտանգ հաշվարկների հաղորդակարգերը մինչև 2004 թ. դիտարկվում էին միայն տեսական հետազոտությունների շրջանակներում: 2004 թ. առաջարկվեց Յաոյի հաղորդակարգի ծրագրային առաջին իրականուցումը՝ Fairplay համակարգի տեսքով, որը ապացուցեց երկու մասնակցով անվտանգ հաշվարկների գործնական իրագործելիությունը:

Անվտանգ հաշվարկների առաջին ինդուստրիալ կիրառությունը տեղի ունեցավ 2008 թ. սկզբին, երբ Դանիայի ավելի քան հազար ֆերմերների կողմից այն կիրառվեց վստահված կողմի բացակայության պայմաններում ֆերմերային ապրանքների գների աճուրդը իրականացնելու համար՝ խուսափելով մասնակիցների առաջարկած առքի և վաճառքի գները հանրայնացնելուց:

2004 թ. ի վեր, առաջարկվել են երկու մասնակցով անվտանգ հաշվարկներ իրականանող համակարգերի մի շարք լավարկումներ, որոնց արդյունքում հնարավոր են դարձրել անվտանգ հաշվարկների որոշ գործնական կիրառություններ:

Արդյունքում, պարզ է դառնում նախկին իրականացումների համեմատ հաշվողական տեսանկյունից ավելի էֆեկտիվ համակարգի մշակման անհրաժեշտությունը, որը կլայնացնի երկու մասնակցով անվտանգ հաշվարկների կիրառելիության շրջանակը՝ հնարավոր դարձնելով նախկինում չհաշվարկված չափերի բույլան սխեմաների մշակումը:

Աշխատանքի նպատակը

Աշխատանքի հիմնական նպատակն է նախագծել և իրականացնել երկու մասնակցով անվտանգ հաշվարկներ իրագործող ծրագրային համակարգ, որը թույլ կտա կատարել հաշվարկներ մասնակիցների մուտքային տվյալների վրա՝ ապահովվելով դրանց գաղտնիությունը: Համակարգը պետք է օգտագործի այսպես կոչված սպիտակ-արկղի ծածկագրության ինովացիոն մեթոդներ, խուսափելով հաշվողական տեսանկյունից աշխատատար բաց բանալիով գործողություններից, միևնույն ժամանակ՝ գործնականում ապացուցելով դրանց կիրառելիությունը երկու մասնակցով անվտանգ հաշվարկների բնագավառում: Ցանկալի է, որ իրականացվող համակարգը հարմար լինի օգտագործողի տեսանկյունից և արդյունավետությամբ գերազանցի նախկինում իրականացված նմանատիպ համակարգերին:

Հետազոտման մեթոդները

Աշխատանքում օգտագործված են բաց բանալիով գաղտնագրություն, ծրագրային ճարտարապետության, սպիտակ արկղի գաղտնագրության և թարգմանությունների տեսության արդիական մեթոդներ:

Գիտական նորույթը

Երկու մասնակցով անվտանգ հաշվարկների համակարգի նախագծում և իրականացում, որը

- ունակ է կառուցել և հաշվարկել ավելի շատ հանգույցներ պարունակող բույլան սխեմաներ՝ համեմատած նախկին իրականացումների հետ
- նմանատիպ համակարգերի իրականացման համար առաջին անգամ օգտագործում է սպիտակ արկղի ծածկագրության մեթոդները
- նմանատիպ իրականացումների հետ համեմատած էապես ավելի արագագործ է:

Աշխատանքի արդյունքների հավաստիությունը հիմնավորվում է իրականացված ծրագրային համակարգի կիրառմամբ ստացված մի շարք փորձնական արդյունքներով:

Ստացված արդյունքների կիրառական նշանակությունը

Ատենախոսության շրջանակներում նախագծվել և իրականացվել ծրագրային համակարգ, որը

- օգտագործողին հնարավորություն է տալիս նկարագրել մուտքային տվյալների նկատմամբ կատարվող հաշվարկները և կարող է օգտագործվել երկու մասնակցով անվտանգ հաշվարկներ պահանջող գործնական կիրառություններում
- ներառում է վերջին տարիներին առաջարկված տարբեր տեսական մեթոդների ծրագրային իրականացումներ և հետազոտողին հնարավորություն է տալիս կատարել տարատեսակ փորձարկումներ դրանց հետ, ինչպես նաև գործնականում հաշվարկել նոր մշակվող մոտեցումների արդյունավետության տարատեսակ բնութագրիչներ:

Աշխատանքի արդյունքների ներդրումը

Աշխատանքի արդյունքում ստեղծված ծրագրային համակարգը ներդրվել է Qube ընկերության «Liquid files» նախագծի իրականացման շրջանակներում, օգտատիրոջ կողմից տրամադրվող տվյալների գաղտնիությունը պահպանող մշակման պրոցեսը էապես ավելի արագագ դարձնելու համար և գտնվում է փորձնական շահագործման փուլում:

Պաշտպանությանը ներկայացվող դրույթները

- Նախկին իրականացումների համեմատ որոշ խնդիրների ավելի արդյունավետ իրականացում, այդ թվում Հեմինգի և Լեվենշտեյնի հեռավորության գաղտնիությունը պահպանող հաշվարկները
- Երկու մասնակցով անվտանգ հաշվարկների ոլորտում արդիական մեթոդների իրականացում մեկ ծրագրային համակարգի շրջանակում
- Սպիտակ արկղի ծածկագրման մեթոդների օգտագործում երկու մասնակցով անվտանգ հաշվարկների ծրագրային համակարգի իրականացման համար
- Երկու մասնակցով անվտանգ հաշվարկների համակարգի բաղկացուցիչ մաս կազմող կոնֆիգուրացվող կոմպիլյատոր

Ապրոբացիա

Ատենախոսության հիմնական արդյունքներն ու դրույթները զեկուցվել և քննարկվել են

- ՀԱՀ «Applied Cryptography Laboratory» հետազոտական լաբորատորիայի սեմինարների ընթացքում (2013-2016 թթ., ք. Երևան)

- «Computer Science and Information Technologies» 10-րդ միջազգային գիտաժողովում (CSIT, 2015 թ., ք. Երևան)
- Հայ-Ռուսական (Սլավոնական) Համալսարանի 10-տարեկան գիտական ժողովի ընթացքում (2015 թ., ք. Երևան)
- ՀՀ ԳԱԱ ինֆորմատիկայի և ավտոմատացման պրոբլեմների ինստիտուտի ընդհանուր սեմինարում

Հրատարակումներ

Ատենախոսության հիմնական արդյունքները տպագրված են 4 գիտական աշխատություններում, որոնք թվարկված են սեղմագրի վերջում:

Աշխատանքի կառուցվածքը և ծավալը

Ատենախոսությունը բաղկացած է ներածությունից, 3 գլուխներից, եզրակացությունից և օգտագործված գրականության ցանկից: Աշխատանքի ընդհանուր ծավալն է 105 էջ՝ ներառյալ 99 անուն օգտագործված գրականության ցանկում:

ԱՇԽԱՏԱՆՔԻ ԲՈՎԱՆԴԱԿՈՒԹՅՈՒՆԸ

Աշխատանքի առաջաբանում հիմնավորված է թեմայի արդիականությունը, ձևակերպված են նպատակներն ու խնդիրները, ինչպես նաև պաշտպանությանը ներկայացվող հիմնական դրույթները: Նշված են ստացված արդյունքների գիտական նորույթը և նրանց կիրառման գործնական արժեքը:

Ատենախոսության **առաջին գլխի** սկզբում բերված են աշխատանքի ընթացքում օգտագործված նշանակումները և սահմանումները: Նկարագրված են համակարգի նախագծման և իրականացման համար կարևոր դեր կատարող հասկացությունները: Առանձնակի ուշադրություն է դարձվել բուլյան սխեմաների աղավաղման արդիական մեթոդների տարբեր տեսանկյուններից հետազոտությանը: Նկարագրվել է Յաոյի հաղորդակարգը այսպես կոչված ազնիվ, բայց հետաքրքրասեր (honest but curious) անվտանգության մոդելում, որի ենթադրությամբ կատարվել է ներկայացվող համակարգի նախագծումն ու իրականացումը:

Հետազոտվել են երկու մասնակցով անվտանգ հաշվարկների նախկինում կատարված իրականացումները, մատնանշվել են այդ համակարգերի իրականացումներում առկա խնդիրները, և հիմնավորվել են տվյալ աշխատանքի նպատակները: Գլխի վերջում բերված են գործնական հետաքրքրություն ներկայացնող երկու մասնակցով կիրառությունների օրինակներ, որոնցում ցանկալի է մասնակիցների մուտքային տվյալների գաղտնիությունը:

Աշխատանքի **երկրորդ գլխում** ներկայացված է ծրագրային համակարգի մաս կազմող կոմպիլյատորի իրականացման մանրամասները: Կոմպիլյատորը համակարգի կարևոր բաղադրիչներից մեկն է, քանի որ այն օգտագործողին հնարավորություն է ընձեռնում նկարագրել կատարվող հաշվարկները բարձր մակարդակի ծրագրավորման լեզվով և ավտոմատացնում է այդ նկարագրությունից բուլյան սխեմաների կառուցման պրոցեսը:

Գլուխի սկզբում բերվում է նախկինում իրականացված նմանատիպ կոմպիլյատորների հետազոտություն: Կատարված հետազոտության արդյունքում հիմնավորվել է նոր կոմպիլյատորի իրականացման անհրաժեշտությունը:

Այնուհետև նկարագրված է կոմպիլյատորի մուտքային լեզուն, որը հիմնված է Fairplay համակարգում ներմուծված SFDL լեզվի վրա: Հաշվարկվորող ֆունկցիայի նկարագրության հարմարավետության բարձրացման նպատակով, համեմատած SFDL լեզվի հետ, կատարվել են մի շարք լեզվական կառուցվածքների ընդլայնումներ, որոնցից ամենաուշագրավը *include* հայտարարությունն է, ինչը հնարավորություն է տալիս օգտագործողին նկարագրվող ծրագրային մոդուլի մեջ օգտագործել արդեն մշակված ծրագրային կոմպոնենտներ:

Կոմպիլյատորի ելքում ստացված բուլյան սխեմայի նկարագրությունը պահվում է ֆայլի մեջ: Այդ Ֆայլը, կախված օգտագործողի կարգավորումներից, կարող է ունենալ երկու ներկայացում՝ երկուական կամ տողային, որը օգտագործողի համար հեշտ ընկալելի է: Երկուական ֆայլը համակարգչի հիշողությունում զգալիորեն քիչ տեղ է զբաղեցնում համեմատած տողային ներկայացման հետ: Մյուս ներկայացման գլխավոր առավելությունը կայանում է նրանում, որ այն կարող է ծառայել կառուցված բուլյան սխեմայի իսկությունը օգտագործողի կողմից ստուգելու համար: Անկախ օգտագործվող ֆորմատից, ելքային ֆայլում գրված են կառուցված բուլյան սխեմայի հանգույցները ճիշտ տոպոլոգիական դասավորությամբ: Ելքային բուլյան սխեման նկարագրող ֆայլը բաղկացած է հետևյալ տրամաբանական կտորներից, տրված հերթականությամբ՝

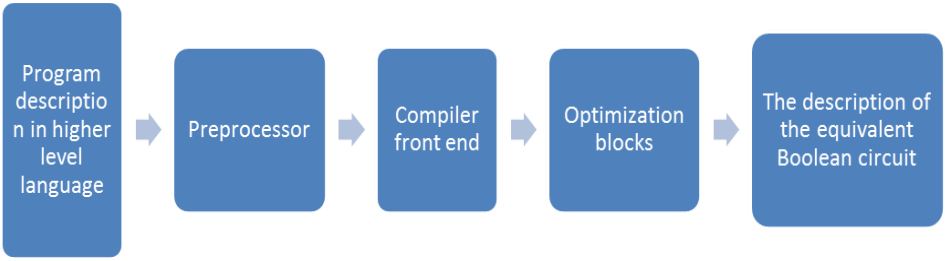
- առաջին մասնակցի մուտքային լարեր
- երկրորդ մասնակցի մուտքային լարեր
- միջանկյալ հաշվողական հանգույցներ
- առաջին մասնակցի գաղտնի ելքային լարեր
- երկրորդ մասնակցի գաղտնի ելքային լարեր:

Աղյուսակ 1-ում բերված են յուրաքանչյուր հանգույցին համապատասխան ելքային ֆայլում գրվող դաշտերը և նրանց ծավալները: Ելքային ֆայլում բուլյան սխեման նկարագրվում է հանգույցների տոպոլոգիական ճիշտ հաջորդականության միջոցով:

Աղյուսակ 1. Յուրաքանչյուր բուլյան հանգույցի համար ելքային ֆայլում գրված հատկությունները և դրանց ծավալները

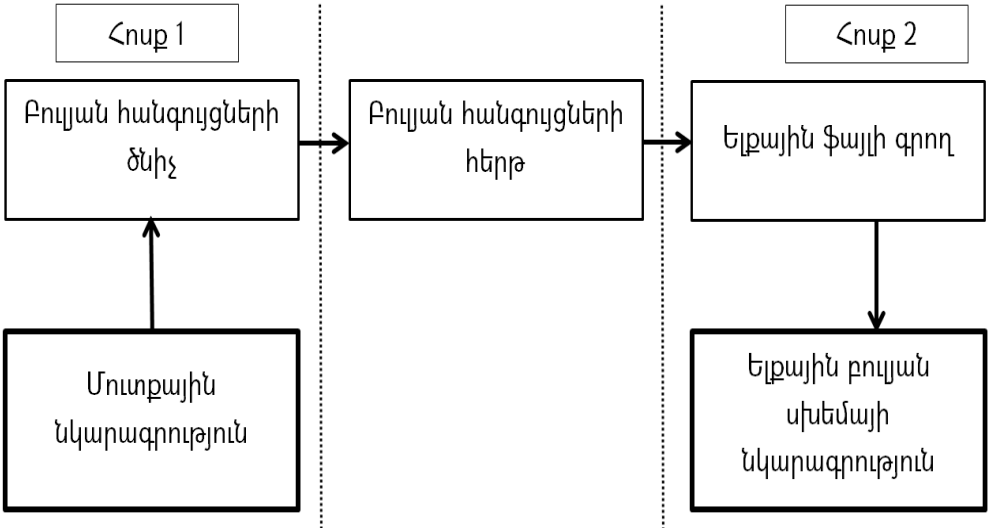
Նկարագրություն	Ծավալ (բիթ)
Նույնականացման թիվ	64
Մուտքային լարերի քանակ	1
Իսկության աղյուսակ	$2^{<մուտք>}$
Տիպ	2
Մուտքային հանգույցների նայնականացման թվերը	$<մուտք> \times 64$

Այնուհետև բերվում է մուտքային բարձր մակարդակի ծրագրավորման լեզվի միջոցով հաշվարկվող ֆունկցիայի նկարագրությանը համարժեք բուլյան սխեմայի կառուցման պրոցեսի (կոմպիլյացիայի) նկարագրությունը: Նկար 1-ում ներկայացված են կոմպիլյացիայի փուլերի համառոտ հաջորդականությունը:



Նկար 1. Կոմպիլյացիայի փուլերի հաջորդականությունը

Բերվում է կոմպիլյատորի դիմային մասի մանրամասն նկարագրությունը: Այս ծրագրային միավորը հանդիսանում է կոմպիլյատորի ամենակարևոր բաղադրիչը, քանի որ կոմպիլյատորը կկատարի իր առջև դրված խնդիրը, նույնիսկ նախնական մշակման և լավարկման միավորների բացակայության պայմաններում: Նմանատիպ կոմպիլյատորների նախկին իրականացումների հետազոտությունը հանգեցրել է այն եզրակացությանը, որ միլիարդավոր հանգույցներ պարունակող բուլյան սխեմայի կառուցման համար ֆայլային մուտքի և ելքի գործողությունները զգալի ժամանակ են պահանջում: Հաշվի առնելով, որ բազմապրոցեսորային միջավայրերը լայն տարածում ունեն, կոմպիլյատորի դիմային մասի նախագծումը կատարվել է այնպես, որ ֆայլային մուտքի և ելքի գործողությունները հնարավոր լինի կատարել առանձին հոսքում: Նկար 2-ում բերված է կոմպիլյատորի դիմային մասի նկարագրությունը:



Նկար 2. Կոմպիլյատորի դիմային մասի նկարագրություն

Բուլյան հանգույցների ծնիչը իրականացվել է օգտագործելով flex և bison ծրագրերը: Մուտքային նկարագրության հերթական գործողությանը համարժեք կառուցված բուլյան սխեմայի յուրաքանչյուր հանգույց ավելացվում է բուլյան հանգույցների հերթում, որի առավելագույն ծավալը վերահսկվում է օգտատիրոջ կողմից: Հերթում առկա հանգույցները գրվում են ելքային ֆայլում և հանվում հերթից: Բազմահոսք միջավայրում հերթի հետ ճիշտ աշխատանքը ապահովելու համար իրականացվել են մի քանի սինքրոնացման մեթոդներ, մասնավորապես, լռությամբ իրականացված է շրջանաձև հերթ՝ օգտագործելով C++14 լեզվի ատոմիկ գործողությունները:

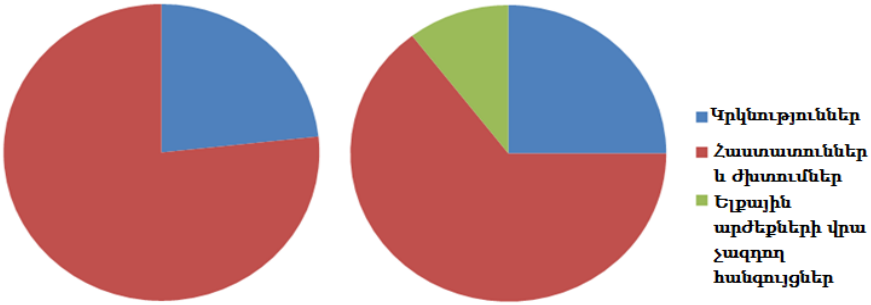
Այնուհետև նկարագրվում են ծրագրային համակարգում իրականացված լավարկման մեթոդների իրականացման մանրամասները, որոնց կիրառումը էապես կրճատում է կառուցվող բուլյան սխեմայում հանգույցների քանակը:

Նկար 3-ում բերված է սխեմայի ելքային արժեքի վրա ազդեցություն չունեցող հանգույցների կրճատման ալգորիթմը, որի իրականացման ընթացքում, յուրաքանչյուր հանգույցի համար հաշվվում է այն հանգույցների քանակը, որոնց համար տվյալ հանգույցը հանդիսանում է մուտքային:

```
Input: file f containing description of Boolean circuit C
Output: 1. description of equivalent Boolean circuit with eliminated gates no
           influence on output
        2. a file containing usage count for each gate
read output gates of the circuit C into O
foreach g in O do
    foreach G in pred(g) do
        incrementUsageCount(G)
    end
end
foreach g in { read the gates in backward topological order } do
    if getUsageCount(g) != 0 then
        foreach G in pred(g) do
            incrementUsageCount(G)
        end
    endif
end
output usage count file
output all input gates
foreach g in { read the gates in topological order } do
    if getUsageCount(g) != 0 then
        output(g)
    endif
end
output all output gates
```

Նկար 3. Բուլյան սխեմայի արժեքի վրա ազդեցություն չունեցող հանգույցների հեռացման և յուրաքանչյուր հանգույցի օգտագործման քանակի հաշվման ալգորիթմ:

Նկար 4-ում բերված են նշված լավարկման մեթոդների համեմատական արդյունավետությունները դիտարկված երկու խնդիրների համար: Լավարկման մեթոդների կիրառման հետևանքով, սկզբնական կառուցված AES ծածկագրումը հաշվող սխեմայի հանգույցների քանակը կրճատվել է մոտ 50 տոկոսով, իսկ Լեվենշտեյին հեռավորությունը հաշվող սխեման՝ մոտ 20 տոկոսով:



Նկար 4. Բուլյան սխեմայի լավարկման մեթոդների հարաբերական արդյունավետությունը անվտանգ AES ծածկագրումը (ձախ) և Լեվենշտեյնի հեռավորությունը հաշվող (աջ) սխեմաների համար

Բերվել է կոմպիլյատորի աշխատանքի ճշտությունը թեստավորող ավտոմատացված համակարգի նկարագրությունը, որը կառուցված բուլյան սխեմայի արժեքը հաշվում է մի քանի մուտքային արժեքների համար և դրանք համեմատում նախապես տրված ելքային արժեքների հետ:

Կոմպիլյատորի աշխատանքի որոշ փուլերում օգտագործվող հիշողություն ծավալը կարող է գերազանցել առկա ֆիզիկական հիշողության չափը: Դրանից խուսափելու համար՝ միջանկյալ արդյունքները մշակելու համար օգտագործվել է արտաքին հիշողութունը, որի հետ աշխատանքի արդյունավետության բարձրացման նպատակով օգտագործվել են օպերացիոն համակարգին բնորոշ համակարգային կանչեր, մասնավորապես հիշողության մեջ արտապատկերված ֆայլեր:

Գլուխը եզրափակվում է մուտքային լեզվում օգտագործվող գործողություններին համարժեք բուլյան սխեմաների մանրամասն նկարագրությամբ:

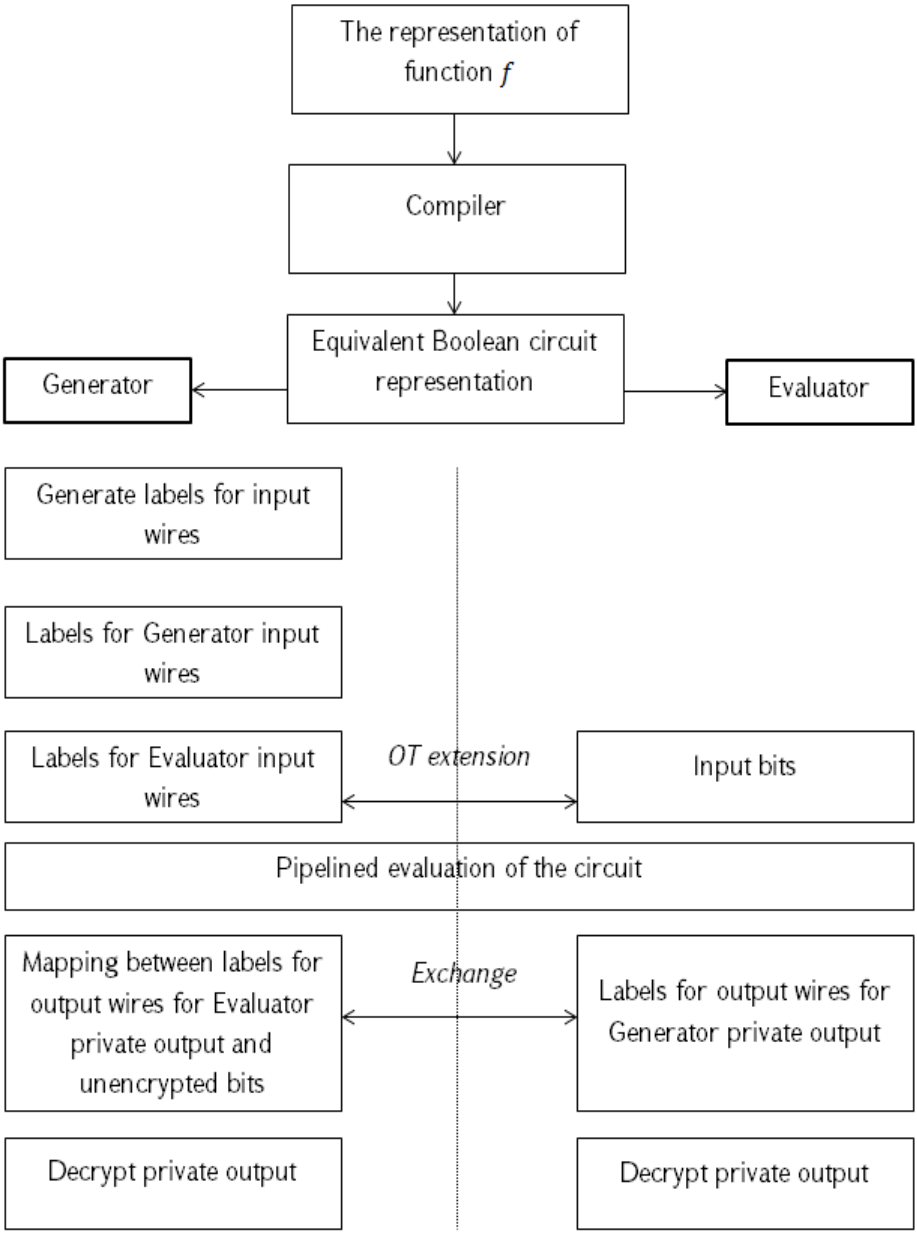
Ատենախոսության **երրորդ գլխում** նկարագրվում է կառուցված բուլյան սխեմայի հաշվարկը կատարող ծրագրային համակարգը:

Ошибка! Источник ссылки не найден.-ում բերված է իրականացված ծրագրային համակարգի օգտագործման ժամանակ մասնակիցների կատարած քայլերի հաջորդականությունը: Մասնակիցները նկարագրում են հաշվարկվող ֆունկցիան՝ օգտագործելով կոմպիլյատորի մուտքային լեզուն: Այնուհետև, կոմպիլյատորի միջոցով, կառուցվում է նկարագրված ֆունկցիան իրացնող բուլյան սխեմա: Կառուցված բուլյան սխեմայի հաշվարկը մասնակիցների գաղտնի մուտքային տվյալների վրա կատարվում է հիմնվելով Յաոյի հաղորդակարգի վրա, որում արդունավետ իրականացման համար կատարվել են որոշ փոփոխություններ: Հիմնական փոփոխությունը կայանում է նրանում, որ առաջին մասնակիցը բուլյան սխեմայի յուրաքանչյուր լարի համար

պատահականորեն ընտրում է 128 բիթ երկարությամբ տողեր, և իր մուտքային տվյալներին համապատասխան տողերը ուղարկում է երկրորդ մասնակցին՝ մինչև աղավաղված սխեմայի կառուցումը: Այնուհետև, անտեղյակ փոխանցման հաղորդակարգի միջոցով առաջին մասնակիցը երկրորդ մասնակցին է փոխանցում վերջինիս գաղտնի մուտքային տվյալներին համապատասխանող տողերը: Այսինքն, առաջին մասնակիցը հանդես է գալիս որպես ուղարկող կողմ՝ երկրորդ մասնակցի մուտքային լարերին համապատասխանող տողերի զույգերի հաջորդականությամբ, իսկ երկրորդ մասնակիցը որպես ստացող կողմ՝ իր գաղտնի մուտքային արժեքի բիթերով: Անտեղյակ փոխանցման հաղորդակարգի իրականացումից կատարվում է բուլյան սխեմայի հաշվարկը, որից հետո մասնակիցները վերծանում են ելքային արժեքները: Այսպիսով, անտեղյակ փոխանցման հաղորդակարգը մասնակիցների միջև իրականացվում է n անգամ, որտեղ n-ը երկրորդ մասնակցի մուտքային տվյալների բիթերի քանակն է: Ի տարբերություն երկու մասնակցով անվտանգ հաշվարկների նախկին իրականացումների, ներկայացվող համակարգում անտեղյակ փոխանցման համար օգտագործվել են սպիտակ արկղի ծածկագրության մեթոդների վրա հիմնված հաղորդակարգ, որը իր արագագործությամբ էապես գերազանցում է նախկինում մշակված անտեղյակ փոխանցման հաղորդակարգերին:

Այնուհետև, ներկայացվել է աղավաղված սխեմայի կառուցման և հաշվարկի հոսքագծային մեխանիզմի իրականացման մանրամասները, ինչը թույլ է տալիս էականորեն կրճատել ցանցի միջոցով մասնակիցների միջև փոխանակվող ինֆորմացիայի քանակը:

Տվյալ աշխատանքի նպատակներից մեկն է հանդիսանում այնպիսի համակարգի մշակումը, որը ունակ լինի մշակել նախկինում չհաշվարկված մեծությամբ բուլյան սխեմաներ: Յուրաքանչյուր աղավաղված հանգույցի կառուցման և հաշվարկի համար անհրաժեշտ են մուտքային հանգույցների աղավաղված արժեքները: Յուրաքանչյուր այդպիսի արժեք իրենից ներկայացնում է 128 բիթ երկարությամբ երկուական տող, որը պետք է պահվի հիշողության մեջ քանի դեռ այն անհրաժեշտ է հետագա հանգույցների մշակման համար: Այսպիսով, սխեմայի մշակման համար պահանջվող միջանկյալ արժեքների ծավալը կարող է զգալիորեն շատ լինել: Միջանկյալ արժեքների պահպանման համար պահանջվող հիշողության նվազեցման համար մշակվող համակարգում կիրառվել է հետևյալ մոտեցումը: Հաշվարկվող ֆունկցիան իրացնող բուլյան սխեմայից արժեքի վրա ազդեցություն չունեցող հանգույցների հեռացման



Նկար 5. Համակարգի օգտագործման սցենարը

ժամանակ՝ յուրաքանչյուր հանգույցի համար կոմպիլյատորը հաշվում է սխեմայում դրա օգտագործման քանակը: Աղավաղված սխեմայի կառուցման և հաշվարկի ժամանակ՝ հանգույցի համապատասխան օգտագործումների թիվը նվազեցվում է մեկով, երբ նրա արժեքը գործածվում է հերթական հանգույցի մշակման համար: Հերթական նվազեցման գործողությունից հետո, հանգույցին վերաբերվող ինֆորմացիան հեռացվում է հիշողությունից, երբ օգտագործումների քանակը հավասարվում է զրոյին: Այսպիսով, սխեմայի մշակման յուրաքանչյուր պահին, հիշողությունում պահվում է հետագայում անհրաժեշտ հանգույցների մասին ինֆորմացիա, ինչը զգալիորեն նվազեցնում է աղավաղված սխեմայի կառուցման և հաշվարկի համար անհրաժեշտ հիշողության քանակը: Աղյուսակ 2-ում բերված են որոշ խնդիրների համար կառուցված բուլյան սխեմայում հանգույցների քանակը և աղավաղված սխեմաների կառուցման և հաշվարկման ընթացքում հիշողությունում միաժամանակ պահվող առավելագույն հանգույցների քանակները:

Աղյուսակ 2. Որոշ բուլյան սխեմաների հանգույցների և մշակման ժամանակ հիշողությունում միաժամանակ պահվող հանգույցների քանակի համեմատություն

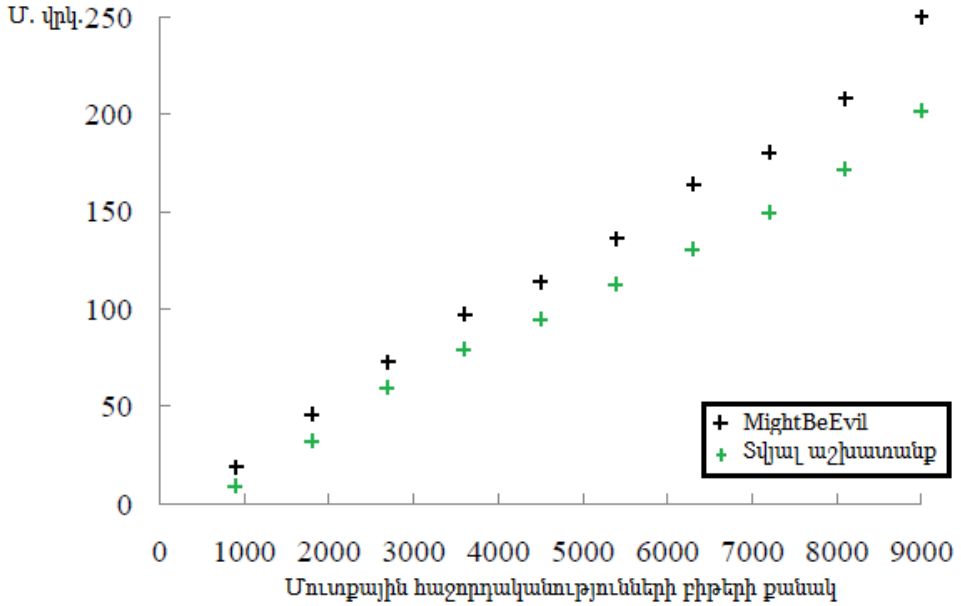
Խնդիր	Սխեմայի հանգույցների քանակը	Հիշողությունում պահվող հանգույցների քանակը
AES ծածկագրում (128 բիթ)	49,912	323
Խմբագրային հեռավորություն	15,540,196	2,829

Գլխի երկրորդ մասում ներկայացվել է մշակված համակարգի հաշվողական էֆեկտիվությունը ապացուցող փորձնական տվյալներ, որոնցում այն համեմատվել է նախկին լավագույն իրականացում հանդիսացող MightBeEvil համակարգի հետ: Համեմատության համար օգտագործվել են հետևյալ ֆունկցիաները՝

- մասնակիցների կողմից տրամադրված գաղտնի x և y բիթային հաջորդականությունների միջև Հեմինգի հեռավորության գաղտնիությունը պահպանող հաշվարկ
- մասնակիցների կողմից տրամադրված գաղտնի x և y հաջորդականությունների միջև այսպես կոչված խմբագրային հեռավորության (edit distance) անվտանգ հաշվարկ:

Ներկայացված խնդիրների իրականացման համար օգտագործվել երկրորդ գլխում ներկայացված կոմպիլյատորի մուտքային լեզվի կոմպոնենտներ:

Նկար 6-ում պատկերված է տվյալ աշխատանքի շրջանակներում մշակված և MightBeEvil համակարգերի միջոցով՝ Հեմինգի հեռավորության գաղտնիությունը պահպանող հաշվարկի տևողությունների համեմատությունը:

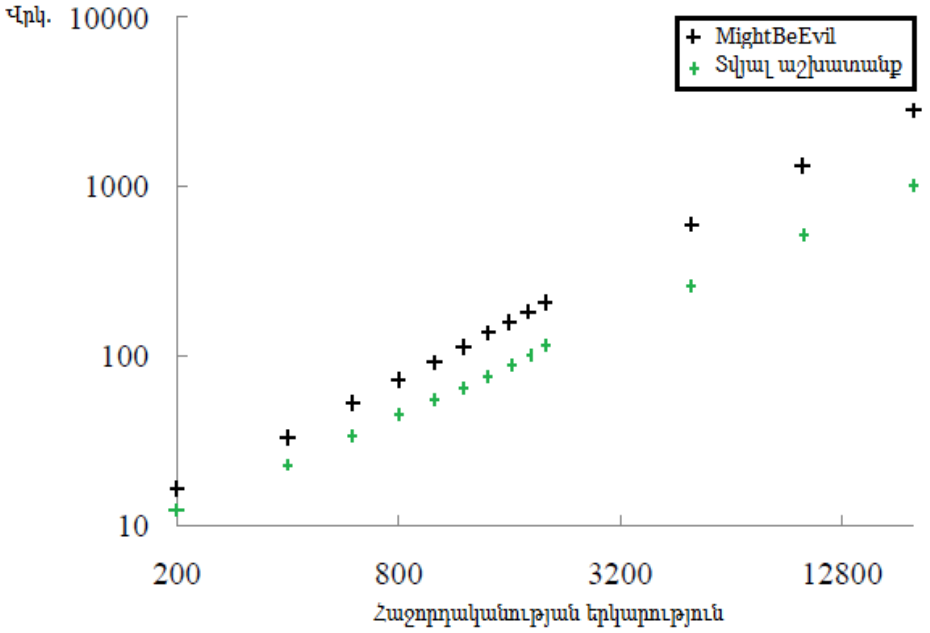


Նկար 6. Ներկայացվող և MightBeEvil համակարգերում Հեմինգի հեռավորության գաղտնիությունը պահպանող հաշվարկի տևողությունների կախումը մուտքային տվյալների երկարությունից:

Նկար 7-ում բերված է ներկայացվող և MightBeEvil համակարգերում խմբագրային հեռավորության գաղտնիությունը պահպանող հաշվարկի տևողությունների համեմատությունը:

Կատարված փորձերի արդյունքում պարզ է դառնում, որ տվյալ աշխատանքի շրջանակներում նախագծված և իրականացված համակարգը հաշվողական տեսանկյունից ավելի արդյունավետ է՝ համեմատած նախկին իրականացման հետ: Ընդ որում, ներկայացվող համակարգի հաշվողական տեսանկյունից արդյունավետությունը ավելի ակնառու է դառնում երկրորդ մասնակցի մուտքային տվյալների բիթերի քանակի աճին զուգահեռ: Վերջինս բացատրվում է նրանով, որ երկրորդ մասնակցի մուտքային տվյալների բիթերի քանակի աճը հանգեցնում է դրանց քանակին

հավասար անտեղյակ փոխանցման հաղորդակարգի իրականացման անհրաժեշտության, որը նախկին իրականացված համակարգերում օգտագործում է հաշվողական տեսանկունից թանկարժեք բաց բանալիով գործողություններ: Ի տարբերություն նախորդ իրականացումների, մեր համակարգում օգտագործվում է անտեղյակ փոխանցման հաղորդակարգ, որը բաց



Նկար 7. Ներկայացվող և MightBeEvil համակարգերում խմբագրային հեռավորության գաղտնիությունը պահպանող հաշվարկի տևողությունների կախումը մուտքային տվյալների երկարությունից:

բանալիով գործողությունները փոխարինում է ավելի արագագործ սպիտակ արկղի գաղտնագրման մեթոդներով:

Հիմնական արդյունքներն ու եզրակացությունները

- Սպիտակ արկղի գաղտնագրության մեթոդների վրա հիմնված անտեղյակ փոխանցման ընթացակարգը ներդրվել է երկու մասնակցով անվտանգ հաշվարկների համակարգի իրականացման համար, որը ծրագրային համակարգի միջոցով ապացուցում է [2] գաղտնագրման նոր մեթոդների

կիրառելիությունը երկու մասնակցով անվտանգ հաշվարկների կատարման համար [1]:

- Մշակվել է երկու մասնակցով անվտանգ հաշվարկների համակարգ, որը իր մեջ ներառում է արդիական լավարկման մեթոդներ, ինչը թույլ է տալիս հաշվարկել գաղտնիություն պահանջող որոշ կիրառություններ ավելի արդյունավետ համեմատած նմանատիպ իրականացումների հետ, մասնավորապես, Հեմինգի և Լեվենշտեյնի հեռավորության անվտանգ հաշվարկը [3]:
- Մշակվել է նախորդ իրականացումների հիման վրա արդիականացված ծրագրավորման իտերատիվ լեզու, որը թույլ է տալիս անվտանգ հաշվարկների մասնակիցներին նկարագրել կատարվող գործողությունները օգտագործողի տեսանկյունից հարմար ձևով և համապատասխանաբար իրականացվել է այդ նկարագրությունից բույլան սխեմա կառուցող թարգմանիչ, որը նմանատիպ թարգմանիչների համեմատ ավելի էֆեկտիվ է հաշվողական տեսանկյունից:

ԱՏԵՆԱԽՈՍՈՒԹՅԱՆ ԹԵՄԱՅՈՎ ՏՊԱԳՐՎԱԾ ՀՈԴՎԱԾՆԵՐԸ

1. A. Jivanyan, G.H. Khachatryan, T. V. Sokhakyan, D. H. Danoyan, “Acceleration of Secure Function Evaluation Protocol” In proceedings of CSIT2015 Tenth Internation Conference on Computer Science and Information Technologies, 2015, pp. 115-118.
2. D. H. Danoyan, T. V. Sokhakyan, “A Generic Framework for Secure Computations” In Proceedings of the Russian-Armenian (Slavonic) University 2015, N. 2, Physical-Mathematical Sciences. 2015, pp. 14-21.
3. T. V. Sokhakyan, “Optimization techniques for generic secure two-party computation platform” In Transactions of IIAP of the NAS RA, Mathematical Problems of Computer Science, vol. 45, pp. 90--98, 2016.
4. T. V. Sokhakyan, “A user configurable compiler for secure computation framework” In Proceedings of Engineering Academy of Armenia, Vol. 13, N. 1, 2016, pp. 138-142.

ABSTRACT

Tigran V. Sokhakyan

“Design and implementation of secure two-party computation framework for privacy-preserving applications”

Prodigious progress of communication infrastructure during recent decades interaction scenarios of practical interest between organizations, individuals and electronic services where the participants need to provide sensitive or private information. The confidentiality of provided private information relies on presence of a trusted third party presence, for example in case of electronic auctions at eBay can act as a trusted party. The behavior of such trusted organizations is clearly regulated and controlled by existing government rules, but not all users will reveal personal or sensitive information to anyone else, especially when this information is related with high stakes or potentially can be used to harm the user.

Secure two-party computation enables mutually distrustful parties to compute a function $f(x,y)$ on corresponding private inputs x_0 and y_0 while revealing nothing beyond the result $f(x_0,y_0)$. Since the middle of eighties secure two-party computations have gained the attention of many researchers in cryptography, but was widely believed to be far inefficient for practical privacy-preserving applications. The first software implementation of a generic secure tow-party computation framework has been introduced in 2004 in the scope of Fairplay project. This implementation is just the proof of concept and is not as efficient to be used in privacy-preserving applications of practical interest. But the influence of Fairplay implementation is huge: it justified the possibility of practical implementations of secure two-party computations and stimulated many researches aimed to implement efficient frameworks able to satisfy the needs of practical applications.

One of the notable industrial applications of secure computations took place in 2008. More than a thousand farmers from Denmark incorporated in a privacy-preserving manner to carry out the auctions for sugar beet prices without usage of a trusted party and without revealing offered buy and sell prices.

Since 2004 many optimizations have been offered resulting secure two-party computations applicable many applications of practical interest from many fields, including and not limited to medicine, facial recognition, economics, and social networks. Also, there are many applications, where secure two-party computations have huge potential to be used but are not used for now because the lack of efficiency.

Thus, it is obvious, that we need to further improve the efficiency of implementations of secure two-party computations and every single improvement is a step to widen the scope of its applicability.

Investigation of previous implementations highlights that their performance highly suffers from the usage of public key operations. This thesis has objective to replace computationally expensive public key operations with novel white-box cryptography based operations.

For secure two-party computations frameworks, it is common to provide a compiler offering constructing the Boolean circuit representation of the function being computed. Another goal of this thesis is the design and implementation of a compiler which constructs Boolean circuits from the given higher level description of the function, outperforms previous implementations from the computational perspective and has the capacity to generate Boolean circuits with sizes previous implementations could not handle.

The main results of the thesis are:

- Practical incorporation of white-box cryptography based oblivious transfer protocol and elimination of expensive public key operations usage for secure two-party computations. The applicability of white-box cryptography methods for secure two-party computations [1] is justified in practice [2].
- An efficient framework for secure two-party computations employing state of the art techniques and various optimizations allowing practical evaluation of some privacy-preserving applications more efficiently, including and not limited to secure evaluation of AES encryption and Levenshtein distance [3].
- A compiler being able to construct an equivalent Boolean representation of input description of functionality in an iterative programming language using less computational resources compared with previous implementations. The compiler is suited to be used as a part of the framework for secure two-party computations and is able to handle the creation of circuits consisting of billions of gates [4].

РЕЗЮМЕ

Тигран В. Сохакян

“Разработка и реализация платформы для безопасных вычислений с двумя участниками для приложений требующих конфиденциальность”

Огромный прогресс вычислительной инфраструктуры в течение последних десятилетий привёл к жизни множество сценариев взаимодействия практического характера между организациями, отдельными лицами и электронными сервисами, где участники должны предоставлять частную информацию. Конфиденциальность предоставленной информации часто опирается на наличие доверенной третьей стороны, например, в случае электронных аукционов eBay может выступать в качестве доверенной стороны. Поведение доверенных организаций четко регулируется и контролируется существующим законодательством, но не смотря на это, не все

пользователи желают раскрыть личную или конфиденциальную информацию кому-либо, особенно если эта информация имеет высокую цену или потенциально может быть использована против самого пользователя.

Безопасные вычисления с двумя участниками позволяют взаимно недоверчивых сторонам A и B вычислить значение функции двух переменных $f(a,b)$ для соответствующих конфиденциальных данных a_0 и b_0 участников, без раскрытия входных данных кроме результата $f(a_0, b_0)$. Начиная с середины восьмидесятых годов безопасные вычисления с двумя участниками были в центре внимания многих исследователей в области криптографии. Долгое время в этой области проводились только теоретические исследования, и бытовало мнение, что безопасные вычисления далеки от практических приложений требующих конфиденциальности входных данных. Первая программная реализация обобщённой платформы для безопасных вычислений с двумя участниками была разработана в 2004 году в рамках проекта Fairplay. Эта реализация является лишь доказательством концепции программной реализуемости безопасных вычислений с двумя участниками, и не является эффективным для использования в практических целях. Но влияние реализации Fairplay огромен: она обосновала возможность практической реализации безопасных вычислений с двумя участниками и стимулировала многие исследования, направленные на реализацию эффективных механизмов, способных удовлетворить потребности приложений практического характера. Одним из наиболее значимых применений безопасных вычислений имело место в 2008 году, когда более чем тысячи датских фермеров, применили безопасные вычисления для выявления цен для сельскохозяйственных продуктов без привлечения доверенной стороны и не раскрывая предлагаемых цен. В результате множества предложенных оптимизаций, безопасные вычисления с двумя участниками нашли практическое применение во многих областях, в том числе и не ограничивается медициной, распознаванием лиц, социальными сетями. Кроме того, существует множество приложений, где имеется огромный потенциал для применения безопасных вычислений с двумя участниками.

Таким образом, очевидна необходимость эффективной реализации безопасных вычислений с двумя участниками, и что каждое отдельное улучшение является расширением сферы их применения.

Исследования предыдущих реализаций подчеркивают влияние операций с открытым ключом на их производительность. Основной целью данной диссертационной работы является разработка и реализация безопасных вычислений с двумя участниками гарантирующий конфиденциальность входных данных, которая вместо трудоёмких операций с открытым ключом использует инновационные методы основанные на так

называемой криптографии белого ящика. Другим важным направлением данной работы является разработка и реализация компилятора для автоматического построения логической схемы вычисляемой функции, исходя из его описания на языке более высокого уровня, который превосходит предыдущие реализации с вычислительной точки зрения, а также способен строить логические схемы с размерами, которыми предыдущие реализации не способны оперировать.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ ДИССЕРТАЦИОННОЙ РАБОТЫ

- Практическое использование протокола забывчивой передачи основанной на методах криптографии белого ящика для устранения трудоёмких операций с открытым ключом. Применимость методов так называемой криптографии белого ящика для безопасных вычислений с двумя участниками [1] показана на примере разработанной программной системой [2].
- Разработана и реализована быстродействующая система для безопасных вычислений с двумя участниками, которая позволяет произвести вычисление некоторых приложений требующих конфиденциальности входных данных, включая, но не ограничиваясь конфиденциальным вычислением расстояний Хемминга и Левенштейна, быстрее по сравнению с предыдущими реализациями [3].
- На основе предыдущих реализаций разработан язык для описания вычисляемой функций участниками протокола, и реализован компилятор, способный построить булеву схему, реализующую описанную функцию, который использует меньше вычислительных ресурсов по сравнению с предыдущими реализациями. Реализованный компилятор приспособлен к системе безопасных вычислений и способен построить булевы схемы, состоящие из нескольких миллиардов логических элементов [4].