ՀՀ ԳԱԱ ԻՆՖՈՐՄԱՏԻԿԱՅԻ ԵՎ ԱՎՏՈՄԱՏԱՑՄԱՆ ՊՐՈԲԼԵՄՆԵՐԻ ԻՆՍՏԻՏՈՒՏ

ԴԱՆԻԵԼՅԱՆ ՎԻԿՏՈՐ ՄԿՐՏԻՉԻ

ԾՐԱԳՐԱՅԻՆ ԱՊԱՀՈՎՄԱՆ ԹԱՔՆԱԳՐԱՅԻՆ ՊԱՇՏՊԱՆՈՒԹՅԱՆ ՀԱՄԱԼԻՐ ՀԱՄԱԿԱՐԳԻ ՀԵՏԱԶՈՏՈՒՄ ԵՎ ՄՇԱԿՈՒՄ

Ե.13.04-<<Հաշվողական մեքենաների, համալիրների, համակարգերի և ցանցերի մաթեմատիկական և ծրագրային ապահովում>> մասնագիտությամբ տեխնիկական գիտությունների թեկնածուի գիտական աստիճանի հայցման ատենախոսության

Ս Ե Ղ Մ Ա Գ Ի Ր

ԵՐԵՎԱՆ 2013

ИНСТИТУТ ПРОБЛЕМ ИНФОРМАТИКИ И АВТОМАТИЗАЦИИ НАН РА

ДАНИЕЛЯН ВИКТОР МКРТЫЧЕВИЧ

ИССЛЕДОВАНИЕ И РАЗРАБОТКА КОМПЛЕКСНОЙ СИСТЕМЫ СТЕГАНОГРАФИЧЕСКОЙ ЗАЩИТЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

А В Т О Р Е Ф Е Р А Т

диссертации на соискание ученой степени кандидата технических наук по специальности 05.13.04 – «Математическое и программное обеспечение вычислительных машин, комплексов и сетей»

ЕРЕВАН 2013

Ատենախոսության թեման հաստատվել է Հայաստանի Պետական Ճարտարագիտական Համալսարանում (Պոլիտեխնիկ)

Գիտական ղեկավար՝	տեխ.գիտ.թեկնածու	Գ.Ի. Մարգարով
Պաշտոնական ընդդիմախոսներ՝	տեխ.գիտ.դոկտոր	Հ.Հ. Հարությունյան
	Ֆիզ.մաթ.գիտ.թեկնածու	Ա.Ս. Վարոսյան

Առաջատար կազմակերպություն՝ Երևանի մաթեմատիկական մեքենաների գիտահետազոտական ինստիտուտ

Պաշտպանությունը կայանալու է 2013թ. հունիսի 13-ին, ժ. 15:00-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 2013թ. մայիսի 13-ին:

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

Тема диссертации утверждена в Государственном инженерном университете Армении (Политехник)

Научный руководитель:	кандидат тех. наук	Г.И. Маргаров
Официальные оппоненты:	доктор тех.наук	Г.А. Арутюнян
	кандидат физ.мат.наук	А.С. Варосян

Ведущая организация: Ереванский нвучно-исследовательский институт математических машин

Защита состоится 13 июня 2013г. в 15:00 на заседании специализированного совета 037 «Информатика и вычислительные системы» Института проблем информатики и автоматизации НАН РА по адресу: 0014, г. Ереван, ул. П. Севака 1.

С диссертацией можно ознакомиться в библиотеке ИПИА НАН РА.

Автореферат разослан 13 мая 2013г.

Ученый секретарь специализированного
совета, д.ф.м.н. А. Г. Саруханян

ԱՇԽԱՏԱՆՔԻ ԸՆԴՀԱՆՈՒՐ ԲՆՈՒԹԱԳԻՐԸ

Թեմայի արդիականությունը. Ծրագրային ապահովման (ԾԱ) նախագծման բնագավառում տեղեկատվական անվտանգության առավել արդիական խնդիրներից է պաշտպանությունը չթույլատրված օգտագործումից, ձևափոխումից, պատճենումից և ԾԱ արտադրողի հեղինակային իրավունքների այլ խախտումներից:

Ներկայումս գոյություն ունեն ծրագրային ապահովման պաշտպանության բազմաթիվ մեթոդներ և ալգորիթմներ, որոնք տարբերվում են միմյանցից թե պաշտպանության ապահովման տեխնոլոգիայով, թե տրամադրվող պաշտպանվածության մակարդակով: Ակնհայտ է, որ ծրագրային ապահովման բոլոր պաշտպանական մեխանիզմները ժամանակի ընթացքում կորցնում են իրենց արդիականությունը, ինչը պայմանավորված է հարձակումների մեթոդների և չարամիտ օգտագործվող ծրագրային ապահովման զարգացմամբ: Ծրագրային ապահովման պաշտպանության տարածված մեխանիզմները (գաղտնաբառային պաշտպանություն, կոնկրետ ապարատային միջավայրում ԾԱ աշխատանքի կազմակերպում, գաղտնաբառի կամ յուրօրինակ տեղեկատվության ակնհայտ ստուգում ցանցի միջոցով) հնարավոր է շրջանցել՝ կիրառվող պաշտպանական ալգորիթմի հայտնաբերման պարագայում: Արդիական է համարվում այն պաշտպանական համակարգը, որը շրջանցելու համար հարկավոր է ավելի շատ ժամանակ և ռեսուրս ծախսել, քան դրանով պաշտպանված ծրագրային ապահովումը օրինական գնելու համար: Ծրագրային ապահովման պաշտպանության ստանդարտ մեխանիզմները, բացի իրենց բնորոշ բացասական կողմերից, ունեն նաև մեկ ընդհանուրը՝ դրանց օգտագործումը (գաղտնաբառի մուտքագրում, բետա-տարբերակի ժամանակավոր օգտագործման մասին ծանուցում, Ինտերնետի պարտադիր առկայություն առցանց նույնականացման պարագայում և այլն) հասանելի է օգտագործողի համար, հետևաբար՝ հարձակվողի:

Այդ իսկ պատճառով արդիական է ծրագրային ապահովման նոր, առավել կատարյալ պաշտպանական համակարգների նախագծումը՝ հիմնված քիչ ուսումնասիրված թաքնագրային մեթոդների վրա, որոնք կոչված են թաքցնել հակառակորդից պաշտպանական մեխանիզմի, կամ նրա մի մասի գոյությունը, և դրանով իսկ կանխել նրա քայլերը: Թաքնագրային պաշտպանության մոտեցումների կիրառության արդիականությունը պայմանավորված է նաև ծրագրային ապահովմանը առանձին մոդուլների տեսքով կցված պաշտպանական մեխանիզմների ցածր հուսալիության վրա:

Նույնպես հետաքրքրություն է առաջացնում սկզբունքորեն միմյանից տարբերվող պաշտպանական ալգորիթմների համալիր օգտագործումը, հակառակորդի համար օգտագործվող պաշտպանության ակնհայտությունը նվազեցնելու նպատակով:

Վերը նշվածից հետևում է, որ թաքնագրային պաշտպանության համալիր համակարգի նախագծումը հանդիսանում է արդիական խնդիր:

Աշխատանքի նպատակն է հետազոտել և մշակել ծրագրային ապահովման պաշտպանության կայուն և վստահելի մեխանիզմներ՝ հիմնված թաքնագրային մեթոդներ վրա: Նշված նպատակին հասնելու համար աշխատանքում լուծվել են հետևյալ խնդիրները՝

- Ծրագրային ապահովման պաշտպանության գոյություն ունեցող մեթոդների և մեխանիզմների հետազոտություն, տեղեկատվության և ծրագրային ապահովման պաշտպանության ժամանակակից թաքնագրային մեթոդների հետազոտում:
- Կատարվող ֆայլերում և IP հաղորդակարգի փաթեթներում թաքնագրման միջոցների նախագծում, և դրանց հիման վրա ծրագրային ապահովման ընդհանուր սպառնալիքներից պաշտպանության մեխանիզմների նախագծում:
- Նախագծված մեխանիզմների վրա հիմնված պաշտպանական համակարգ իրականացնող ալգորիթմների և ծրագրային միջոցների ստեղծում:

Հետազոտման օբյեկտ է հանդիսանում ծրագրային ապահովման թաքնագրային պաշտպանության համալիր համակարգերի կայունությունն ու դիմացկունությունը

Հետազոտման մեթոդներ. Աշխատանքի ընթացքում օգտագործվել են թաքնագրման, գաղտնագրման, գծային հանրահաշվի, փորձարկման և վիճակագրական հանրահաշվի մեթոդներ:

Արդյունքների գիտական նորությունը:
- Առաջարկվել է Portable Executable ձևաչափի կատարվող ֆայլերում տեղեկատվության ներդրման միջոց, որն հնարավորություն է ընձեռում թաքնագրելու մեծ ծավալով տեղեկատվություն՝ չազդելով ֆայլի աշխատունակության վրա:
- Առաջարկվել է IP հաղորդակարգի փաթեթներում տեղեկատվության ներդրման միջոց, որը չի ազդում հաղորդակարգի աշխատանքի վրա և չի հանգեցնում փաթեթների կորստին՝ փախանցման ընթացքում:
- Առաջարկվել է ծրագրային ապահովման չթույլատրված օգտագործումից, ձևափոխումից և պատճենումից պաշտպանության համակարգ՝ հիմնված գոյություն ունեցող և աշխատանքի ընթացքում առաջարկվող պաշտպանական մեխանիզմների համալիր օգտագործման վրա:

Ստացված արդյունքների կիրառական նշանակությունը:
- Մշակվել են ալգորիթմներ և նախագծվել են Portable Executable ձևաչափի ֆայլերում և IP հաղորդակարգի փաթեթներում տեղեկատվության թաքնագրման իրարից ֆունկցիոնալապես անկախ մոդուլներ՝ որոնք ապահովում են համապատասխանաբար մինչև 2,5 անգամ դիմացկունության աճ համեմատած կատարվող ֆայլերում թաքնագրման գոյություն ունեցող միջոցների հետ և մինչև 40% աճ համեմատած IP փաթեթներում թաքնագրման միջոցների հետ:
- Նախագծվել է ՕԱ պաշտպանության SProtect համակարգը, որը ապահովում է ծրագրային ապահովման պաշտպանություն չթույլատրված օգտագործումից, ձևափոխումից և պատճենումից, ինչպես նաև ՕԱ-ի հեղինակային իրավունքների մասին վկայող տեքստային տեղեկատվության ներդրում կատարվող ֆայլերի մեջ:

Ներդրումներ. Ատենախոսության արդյունքները կիրառվում են “Telasco Communications” LTD կազմակերպության հայաստանյան ներկայացուցչությունում: SProtect համակարգը օգտագործվում է կազմակերպության ներքին կիրառման և վաճառվող/վարձու ծրագրային ապահովման պաշտպանության համար:

Բացի այդ, աշխատանքի հիմնական արդյունքները կիրառվում են կրթա-հետազոտական ծրագրային միջոցում՝ ՀՊՃՀ ՏԱՕԱ ամբիոնի կրթական պրոցեսի գործընթացում:

Ներդրման վկայականները բերված են թ.2 և թ.3 հավելվածներում:

Պաշտպանությանը ներկայացվում են հետևյալ դրույթները.

- Կատարվող ֆայլերում տեղեկատվության ներդրման գոյություն ունեցող միջոցներից մինչև 2,5 անգամ առավել դիմացկուն միջոցը:
- IP հաղորդակարգի փաթեթներում տեղեկատվության ներդրման, մինչև 40% դիմացկունության աճ ապահովվող միջոցը:
- Տեղեկատվության ներդրման նախագծված միջոցների և ՕԱ պաշտպանության գոյություն ունեցող մոտեցումների համալիր օգտագործման վրա հիմնված պաշտպանական մեխանիզմը:
- ՕԱ չթույլատրված օգտագործումից, ձևափոխումից և պատճենումից պաշտպանության մոդուլները իրականացնող ալգորիթմները և ծրագրային միջոցները:

Ստացված արդյունքների ապրոբացիան. Աշխատանքի հիմնական արդյունքները ներկայացվել և քննարկվել են XXXVI "Гагаринские чтения" միջազգային երիտասարդական գիտական կոնֆերանսին (2010թ., ք. Մոսկվա), Applications of Information Theory, Coding and Security WAITS2010 սեմինարին (2010թ., ք. Երևան), “Computer Science and Information Technologies (CSIT)” միջազգային կոնֆերանսին (2011թ., ք. Երևան), Կասպերսկու լաբորոտորիայի կողմից անցկացվող “It Security For The New Generation 2010” (2010թ., ք. Մոսկվա) և “It Security For The New Generation 2013” (2013թ., ք. Երևան) ուսանողական կոնֆերանսներին, վերջինի ընթացքում աշխատանքը ստացել է հատուկ մրցանակ՝ Microsoft կազմակերպության կողմից: Աշխատանքը նաև քննարվել է ՀՊՃՀ ՏԱՕԱ ամբիոնի գիտական սեմինարների ընթացքում (2010-2013թ., ք. Երևան):

Հրապարակումներ. Աշխատանքի հիմնական արդյունքները հրապարակվել են 4 գիտական աշխատություններում, որոնք բերված են սեղմագրի վերջում:

Ատենախոսության կառուցվածքը և ծավալը. Ատենախոսական աշխատանքը կազմված է ներածությունից, չորս գլուխներից, վերջաբանից, օգտագործված գրականության ցանկից և երեք հավելվածներից: Աշխատանքի ընդհանուր ծավալն է 107 էջ, ինչը ներառում է 32 նկար և 6 աղյուսակ: Գրականության ցանկը կազմում է 74 վերնագիր, իսկ հավելվածների ընդհանուր ցավալն է 16 էջ:

ԱՇԽԱՏԱՆՔԻ ԲՈՎԱՆԴԱԿՈՒԹՅՈՒՆԸ

Ներածության մեջ հիմնավորված է թեմայի արդիականությունը, ձևակերպված են աշխատանքի նպատակը, գիտական նորույթները, պաշտպանության ներկայացվող հիմնական դրույթները:

Առաջին գլխում քննարկվել են ծրագրային ապահովման պաշտպանության գոյություն ունեցող մեթոդների կազմակերպման ձևերը, մասնավորապես՝ գաղտնաբառերի համեմատման վրա հիմնված մեթոդները, գաղտնագրման և կոդավորման մոտեցումների կիրառմամբ մեթոդները, ինչպես նաև օբֆուսկացիայի և էլեկտրոնային բանալու կիրառությունը:

Անց է կացվել ծրագրային ապահովման պաշտպանության ստանդարտ մեթոդների և մեխանիզմների առավելությունների և թերությունների վերլուծություն, ուսումնասիրվել են դրանց շրջանցման մոտեցումները:

Ուսումնասիրվել են ժամանակակից թաքնագրության հնարավորությունները, զարգացման միտումները, ծրագրային ապահովման պաշտպանության տեսանկյունից թաքնագրային մոտեցումների կիրառության հնարավորությունները [1]:

Ծրագրային ապահովման պաշտպանության գոյություն ունեցող մեթոդների և մեխանիզմների ուսումնասիրության արդյունքների հիման վրա առաջարկվել է գոյություն ունեցող մեթոդների և ԾԱ բնագավառում քիչ հայտնի թաքնագրային տեխնոլոգիաների համատեղ կիռառման մոտեցում՝ ԾԱ պաշտպանական համակարգի նախագծման ընթացքում [2]:

Թաքնագրային մոտեցումների և ԾԱ պաշտպանության գոյություն ունեցող մեթոդների համատեղ օգտագործման վրա հիմնված հուսալի համակարգի նախագծման համար նպատակահարմար է բուն ծրագրայի ապահովման կատարվող ֆայլերի, ուսումնասիրությունը թաքնագրման տեսանկյունից ՝ տեղայնային պաշտպանության նախագծման նպատակով, և ցանցային տեխնոլոգիաների ուսումնասիրությունը՝ առցանց թաքնագրային պաշտպանության նախագծման նպատակով:

Առաջին գլխի վերջում, կատարված ուսումնասիրությունների հիման վրա, ձևակերպվել է ատենախոսական աշխատանքի նպատակը և դրվել են դրան հասնելու խնդիրները:

Երկրորդ գլխում անց է կացվել Windows օպերացիոն համակարգերի ընթանիքում կատարվող ֆայլերի Portable Executable (PE) ձևաչափի կառուցվածքի վերլուծությունը՝ կատարվող ֆայլերը որպես թաքնագրային կրիչ օգտագործելու նպատակով:

Հետազոտության արդյունքում առաջարկվել են PE ձևաչափի ֆայլերում տեղեկատվության թաքնագրման միջոցներ՝

- *Ներդրում վերնագրի մեջ:* PE ձևաչափի ֆայլերի կառուցվածքի հետազոտման արդյունքում հայտնաբերվել են վերնագրային դաշտեր, որոնց տեղեկատվությունը չի օգտագործվում ֆայլի բեռնման և աշխատանքի ընթացքում, և այդ տեղեկատվության փոփոխությունը չի հանգեցնում կատարվող ֆայլի աշխատանքի խափանմանը: Այդպիսի տեղեկատվություն պարունակող դաշտերը տասներեքն են: Այս միջոցի կիրառման դեպքում PE ֆայլի թաքնագրային տարողունակությունը (S)

հաշվարկվում է ներդրման համար օգտագործվող բոլոր վերնագրային դաշտերի տարողունակությունների (C_i) գումարով՝

$$S = \sum_{i=1}^{13} C_i = 50 \text{ բայթ} \quad (1)$$

Այս միջոցի առավելությունն այն է, որ տեղեկատվության ներդրումը չի անդրադառնում կատարվող ֆայլի չափի վրա:

- *Ներդրում սեկցիոն շեղումից առաջացած տարածքում:* Սեկցիաների աղյուսակի վերջին և առաջին սեկցիայի սկզբում առաջանում է տարածքային բացատ՝ ֆայլային հավասարեցման պատճառով: Թաքնագրման այս միջոցը ենթադրում է այդ տարածքի կիրառումը տեղեկատվության ներդրման համար: Այս մոտեցման կիրառման դեպքում թաքնագրային տարողունակությունը կազմում է 512 բայթ այն դեպքում, երբ առաջին սեկցիաի չափը փոքր կամ հավասար է 512 բայթի, իսկ հակառակ դեպքում հաշվարկվում է հետևյալ բանաձևով՝

$$S = ([C / 512] + 1) \times 512 - C \quad (2)$$

որտեղ C-ն՝ առաջին սեկցիայի չափն է:

Այս միջոցը, նախորդի պես, չի ազդում թաքնագրային կրիչի չափերի վրա՝ տեղեկատվության ներդրման դեպքում:

- *Տեղեկատվության ներդրում վերջին սեկցիային վերջից:* Այս միջոցի գաղափարը հանդիսանում է վերջին սեկցիայի վերջում թաքնագրվող տեղեկատվության ավելացման մեջ՝ ֆայլի վերնագրերում և սեկցիաների աղյուսակում փոփոխված սեկցիայի մասին համապատասխան տեղեկատվությունը ձևափոխելով: Այս մոտեցման առավելությունն այն է, որ ներդրվող տեղեկատվության ծավալը սահմանափակվում է միայն համապատասխան ֆայլային համակարգում ֆայլի առավելագույն թույլատրված չափով: Սակայն այս մոտեցմամբ թաքնագրման դեպքում ավելանում է կատարվող ֆայլի ծավալը՝ ներդրվող տեղեկատվության չափով:
- *Տեղեկատվության ներդրում նոր սեկցիայի ավելացմամբ:* Ներդրման այս միջոցը ենթադրում է տեղեկատվության թաքնագրման նպատակով կատարվող ֆայլում նոր սեկցիայի կամ սեկցիաների ավելացում: Նոր սեկցիայի ավելացմամբ տեղեկատվության ներդրման միջոցը օժտված է նույն առավելությամբ ինչ նախորդը, ինչպես նաև առավել ճկուն է կիրառության տեսանկյունից, սակայն ունի ավելի բարդ իրականացում, քանի որ այս միջոցի կիրառման դեպքում հարկավոր է ստեղծել նոր գրառում կամ գրառումներ սեկցիաների աղյուսակում և փոփոխել համապատասխան վերնագրային դաշտերի արժեքները:

Հիմնվելով կատարվող ֆայլերում տեղեկատվության թաքնագրման առաջարկված միջոցների վրա, նախագծվել է ծրագրային ապահովման չթույլատրված օգտագործումից և ձևափոխումից պաշտպանության մեխանիզմ [2] (Նկ. 1)

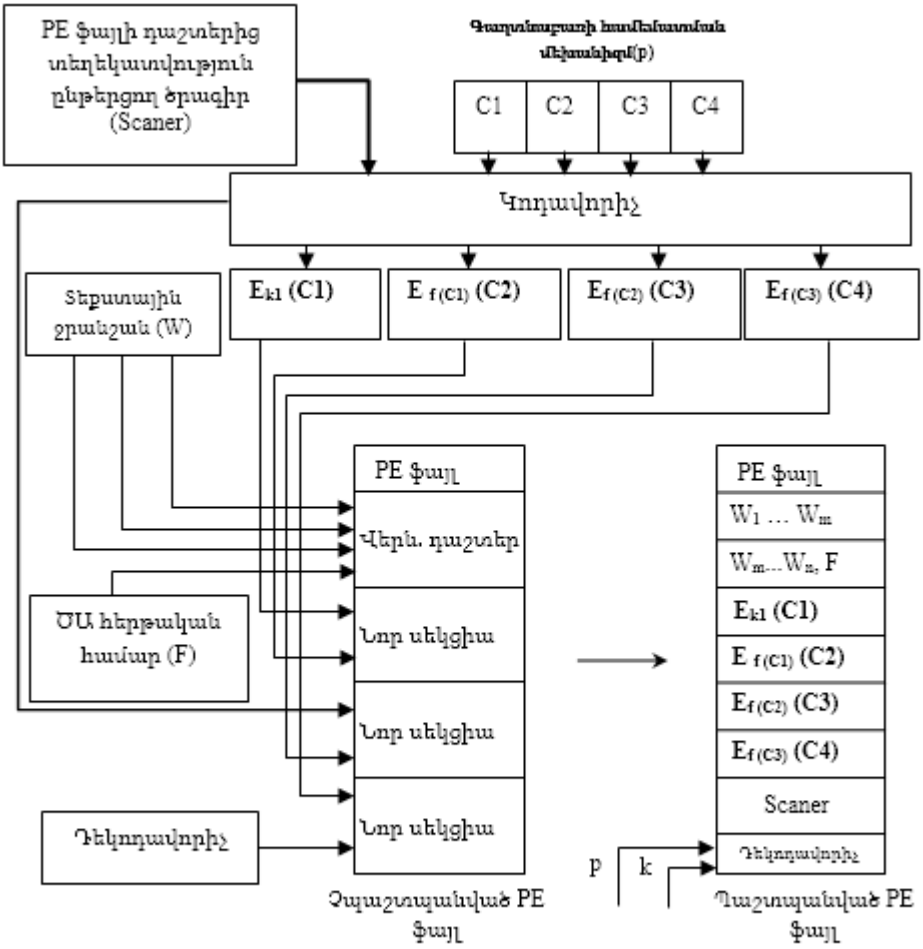
Գաղտնաբառի համեմատման մեխանիզմը իրենից ներկայացնում է ֆունկցիա, որը մուտքագրած (p) գաղտնաբառը ձևափոխում է իրեն համապատասխան հեշ ֆունկցիայի և իրականացնում է հեշ-գաղտնաբառի և հեշ-ստուգանմուշի համեմատում: Հեշավորման համար կիրառվում է MD5 ալգորիթմը:

Գաղտնաբառի հեմեմատման մեխանիզմի կատարվող կոդը բաժանված է չորս բլոկների: Կոդավորիչը իրենից ներկայացնում է մի ծրագիր, որը նախատեսված է

գաղտնաբառը համեմատող ծրագրի կոդի բլոկների և PE-ֆայլի դաշտերից տվյալները կարդացող ծրագրի (Scaner) գաղտնագրման համար: Կոդի առաջին բլոկը (C1) գաղտնագրվում է "բաց բանալիով գաղտնագրման" ձևափոխված ալգորիթմով՝

$$C1_E = E_{k1}(C1) \quad (2)$$

Դա նշանակում է, որ կոդի բլոկը գաղտնագրվում է փակ բանալու միջոցով (k1), որը հայտնի է միայն պաշտպանական համակարգի հեղինակներին, իսկ բաց բանալին (k) թաքնագրվում է կատարվող ֆայլի սեկցիոն շեղումից առաջացող տարածքում: Դա կատարվում է հետևյալ նպատակով՝ հակարակորդի կողմից կոդի նույնիսկ հաջող դեկոդավորման դեպքում, կոդի ձևափոխման համար նրան հարկավոր կլինի իմանալ նաև փակ բանալին:



Նկ. 1: Գաղտնվող ֆայլերում թաքնագրության վրա հիմնված ծրագրային ապահովման պաշտպանության մեխանիզմի կառուցվածք

Ամեն մի հաջորդ բլոկը գաղտնագրվում է համաչափ մեթոդով և որպես բանալի օգտագործվում է նախորդ բլոկի հսկող գումարը:

$$C2_E=E_{f(C1)}(C2),\ C3_E=E_{f(C2)}(C3),\ C4_E=E_{f(C3)}(C4) \quad (3)$$

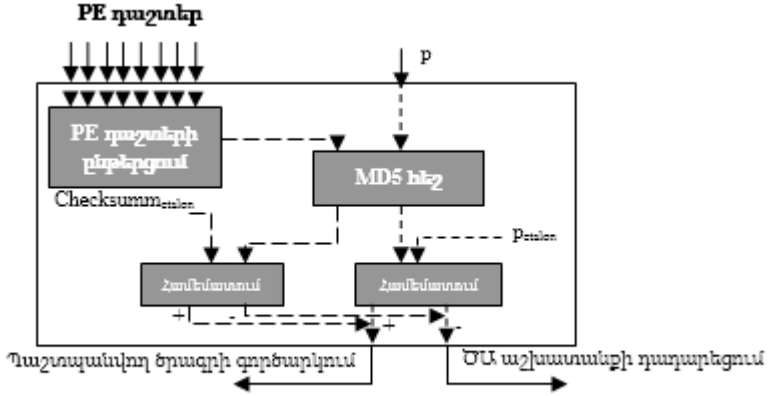
Դրանով ախահովվում է գաղտնաբառը համեմատող մեխանիզմի պաշտպանությունը ձևափոխումից: Այսինքն, բլոկերից մեկի կոդի փոփոխությունը կհանգեցնի հսկող գումարի փոփոխությանը և վերծանումը հաջող չի անցնի, դա կհանգեցնի պաշտպանվող ծրագիրային ապահովման աշխատանքի խափանմանը:

Կոդի բլոկների գաղտնագրված տարբերակների ստացումից հետո, դրանք թաքնագրվում են կատարվող ֆայլում՝ նոր սեկցիաների ստեղծման միջոցով:

Կատարվող ֆայլի վերնագրային դաշտերում գրանցվում են թվային ջրանշանը՝ հեղինակային իրավունքը գնած կազմակերպության անվան և գնման տարեթվի տեսքով և թվային մատնահետքը՝ ծրագրային ապահովման տվյալ պատճենի համարի տեսքով:

Բացի դրանից PE ձևաչափի կատարվող ֆայլի մեջ թաքնագրվում է նաև ֆայլի վերնագրային դաշտերում և սեկցիոն շեղումից առաջացած տարածքում թաքնագրված տեղեկատվությունը ընթերցող ծրագիրը (Scaner) և դեկոդավորող ծրագիրը՝ որը ստանալով համապատասխան գաղտնաբառը (k) Scaner ծրագրից իրականացնում է հիմնական գաղտնաբառի համեմատման ֆունկցիաների վերծանում:

Scaner ծրագրի գործառույթները հետևյալն են՝ այն ընթերցում է ֆայլի վերնագրային դաշտերում և սեկցիոն շեղումից առաջացած տարածքում թաքնագրված տեղեկատվությունը և ստուգում է նրա ամբողջականությունը: Դա ստուգվում է համեմատելով այդ տարածքներում գտնվող տեղեկատվության հսկող գումարը ստուգանմուշի հետ: Ոճրագործի կողմից, ծրագրի կոդի փոփոխությունից խուսափելու համար (հատկապես ստուգանմուշային հսկող գումարի), նպատակահարմար է ստուգանմուշ արժեքների հետ համեմատման գործողությունը համատեղել գաղտնաբառը համեմատող ծրագրի հետ (Նկ. 2).



Նկ. 2: Գաղտնաբառի և PE-ֆայլի դաշտերի հսկող գումարի համեմատման մեխանիզմ

Առաջարկված մեխանիզմը ապահովում է ծրագրային ապահովման պաշտպանությունը չթույլատրված օգտագործումից և ձևափոխումից [2]:

Երրորդ գլուխը նվիրված է ծրագրային ապահովման հեռավար թաքնագրային պաշտպանության նախագծմանը: Քանի որ ԾԱ հեռավար պաշտպանության հիմքում ընկած է օգտագործողների կամ ԾԱ համապատասխան պատճենների նույնականացումը հեռավար սերվերի վրա, նպատակահարմար է նախագծել պաշտպանական մեխանիզմ, որտեղ հակառակորդից կթաքցվի վավերացման մոտեցումը:

Իրականացվել է IP հաղորդակարգի փաթեթի կառուցվածքի վերլուծություն, և առաջարկվել է փաթեթների մեջ տեղեկատվության թաքցման միջոց: Թաքնագրման համար կիրառվել են Identification և Timestamp դաշտերը:

Սակայն IP փաթեթի Identification և «Ընտրանքներ» բաժնի Timestamp դաշտերի կիրառմամբ IP փաթեթների մեջ տեղեկատվության թաքնագրման համար հարկավոր է ապահովել հետևյալ պայմանները՝

- IP հաղորդակարգի մասնագիրը պահանջում է, որպիսի նույնացուցիչը չկրկնվի համապատասխան դատագրամի կյանքի տևողության ընթացքում (Time to live): Այս պահանջը բավարարվում է ավելացնելով նույնացուցիչի արժեքը մեկով՝ ամեն հաջորդ դատագրամի համար:
- Չնայած այն փաստին, որ հնարավոր է օգտագործել ոչ ստանդարտ ժամանակային կնիք, հարկավոր է ամեն հաջորդ դատագրամի համար մեծացնել այդ դաշտի արժեքը, արժեքի մեծացումը կիրառվում է իրար հաջորդող դատագրամների թիվը պարզելու համար:
- Ըստ հաղորդակարգի մասնագրի, ուղարկողը պետք է ստեղծի ժամանակային կնիքի ընտրանքը այնպես, որ ժամանակային կնիքների դաշտերը բավական լինեն ամբողջ սպասվող տեղեկատվությունը տեղավորելու համար: Ընտրանքի դաշտի չափը չի փոփոխվում ժամանակային կնիքներ ավելացնելուց: Եթե ժամանակային կնիքների դաշտը լցվել է, ապա դատագրամը փոխանցվում է առանց որևէ կնիքի, իսկ գերլցման հաշվիչը մեկով ավելանում է: Գերլցման հաշվիչի գերլցման դեպքում փաթեթը մերժվում է:

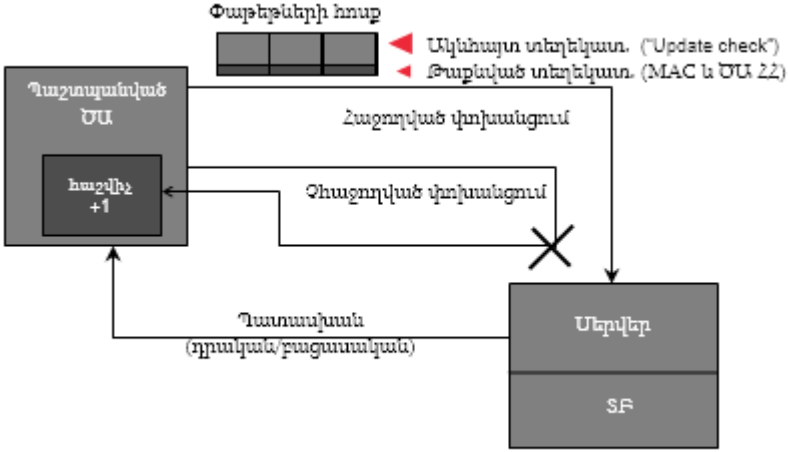
Հաշվի առնելով այս երեք պայմանները առաջարկվում է IP փաթեթներում տեղեկատվության թաքնագրման հետևյալ մոտեցումը՝

- IP-դատագրամի ID դաշտի ծավալը 16 բիթ է, որի 10 բիթը օգտագործվում է գաղտնի տեղեկատվություն թաքնագրելու նպատակով, հաջորդող 5 բիթը օգտագործվում է IP հաղորդակարգի առաջին պայմանին բավարարելու նպատակով, իսկ կրտսեր բիթը՝ թաքնագրային կրիչում թաքնագրված տեղեկատվությունը որպես ընդհանուր փոխանցվող հաղորդագրության վերջին հատված նշելու նպատակով: IP հաղորդակարգի առաջին պահանջը կատարվում է նշված 5 բիթերում գտնվող արժեքը, ամեն հաջորդ ուղարկվող դատագրամի համարը «1»-ով բարձրացնելով: Այդ գործողությունը հարկավոր է կրկնել փաթեթի կայքի տևողության ընթացքում (TTL):
- IP դատագրամի վերնագրում ստեղծվում է մեկ ժամանակային կնիք պարունակող Internet Timestamp ընտրանք, որը իրենից ներկայացնում է 32-բիթանոց դաշտ: Ժամանակային կնիքի ցածր կարգի 6 բիթը օգտագործվում է IP հաղորդակարգի երկրորդ պայմանին բավարարելու նպատակով, հաջորդ 25 բիթը օգտագործվում են թաքնագրվող տեղեկատվության հաջորդական հատվածի ներդրման նպատակով, ամենաբարձր կարգի բիթը տեղակայվում է 1, ինչպես պահանջում է IP հաղորդակարգը՝ ոչ ստանդարտ ժամանակային կնիքը բնորոշելու համար::

Այսպիսով IP հաղորդակարգի յուրաքանչյուր փաթեթի թաքնագրային

տարողունակությունը կազմում է 35 բիթ:

IP հաղորդակարգի փաթեթների վերնագրերում տեղեկատվության թաքնագրման միջոցի հիման վրա առաջարկված է ծրագրային ապահովման պաշտպանության համակարգ, որը ապահովում է ԾԱ պաշտպանությունը չթույլատրված օգտագործումից և պատճենահանումից՝ թաքցնելով գաղտնի տեղեկատվությունը բացահայտ փոխանցվող տեղեկատվության մեջ (Նկ. 3):



Նկ. 3: IP թաքնագրության վրա հիմնված պաշտպանական մեխանիզմ

Նախագծված մեխանիզմը թաքցնում է հակարակորդից օգտագործողի նույնականացման մեթոդը և նույնականացման համար օգտագործվող տեղեկատվությունը և համապատասխան պատճենի գործարկման համար չեն կիրառվում օգտագործողի նույնականացման հայտնի մոտեցումներ, ինչպիսին են լոգինների, գաղտնաբառերի, հերթական համարների ներմուծումը և այլն: Մեխանիզմը հիմնված է տեղեկատվության փոխանցման բաց ուղղում գաղտնի տեղեկատվության ներդրման վրա՝ օգտագործելով IP հաղորդակարգի փաթեթները [3]:

Պաշտպանված ծրագրային ապահովման առաջին գործարկման ժամանակ, մեխանիզմը օգտագործողին առաջարկում է ստուգել ծրագրի թարմացման առկայությունը: Օգտագործողի դրական պատասխանի դեպքում մեխանիզմը ընթերցում է օգտագործողի ցանցային քարտի MAC հասցեն և թաքնագրում է այն IP փաթեթների վերնագրային դաշտերում՝ ԾԱ համապատասխան պատճենի հերթական համարի հետ մեկտեղ: Ստեղծված IP փաթեթ թաքնագրային կրիչները, որպես բացահայտ տեղեկատվական ծանրաբեռնվածություն, պարունակում են ԾԱ թարմացման կեղծ հարցում, իսկ թաքնագրման համար օգտագործվող դաշտերում՝ նույնականացման համար անհրաժեշտ տեղեկատվությունը:

Հեռավար սերվերի վրա գտնվում է նույնականացման գրառումների տվյալների բազան և պաշտպանական մեխանիզմի սերվերային մասը, որը ընթերցում է բռնոտված փաթեթների Identification և Timestamp դաշտերը և ԾԱ յուրաքանչյուր պատճենի համար ստեղծում է համապատասխան գրառումը տվյալների բազայում:

Պաշտպանված ծրագրային ապահովման հաջորդ գործարկումների ժամանակ պաշտպանական մեխանիզմի սերվերային մասը կրկին փորձում է ստեղծել օգտագործողի ցանցային քարտից և ԾԱ հերթական համարից կազմված գրառումը տվյալների բազայում, սակայն, եթե համապատասխան ԾԱ հերթական համարը արդեն առկա է բազայում, ստուգվում է IP փաթեթները ուղարկող ցանցային քարտի MAC հասցեն, եթե փաթեթները ուղարկվել են նույն հասցեից, որը գրանցված է տվյալների բազայում, ապա նույնականացումը համարվում է անցած և ուղարկվում է հաստատող պատասխան ծրագրային ապահովմանը և նրա գործարկումը շարունակվում է [5].

Այն դեպքի համար, երբ օգտագործողը որոշում է բաց թողնել «ԾԱ թարմացման առկայության ստուգումը», պաշտպանվող ծրագրային ապահովման մեջ ներդրվում է ոչ առցանց գործարկումների հաշվիչ: Հաշվիչի արժեքը որոշվում է ծրագրի պաշտպանության փուլում և ամեն ոչ առցանց գործարկման դեպքում նվազեցվում է մեկով: Հաշվիչի արժեքը զրոյին հավասարվելու դեպքում, օգտագործողը զգուշացվում է «պարտադիր թարմացման» առկայության մասին, առանց որի ծրագրի հետագա աշխատանքը անհնարին է:

Հիմնվելով այն փաստի վրա, որ ցանցային քարտի MAC հասցեն ունի 48 բիթ երկարություն և ԾԱ պաշտպանության համակարգը պետք է հաշվարկված լինի առնվազն մեկ միլիոն ծրագրային պատճենների գրանցման համար, առաջարկվում է տեղեկատվության բլոկերի (հետագայում S-բլոկներ) հետևյալ հերթականությունը՝ IP փաթեթներում ներդնելու նպատակով (Նկ. 4):

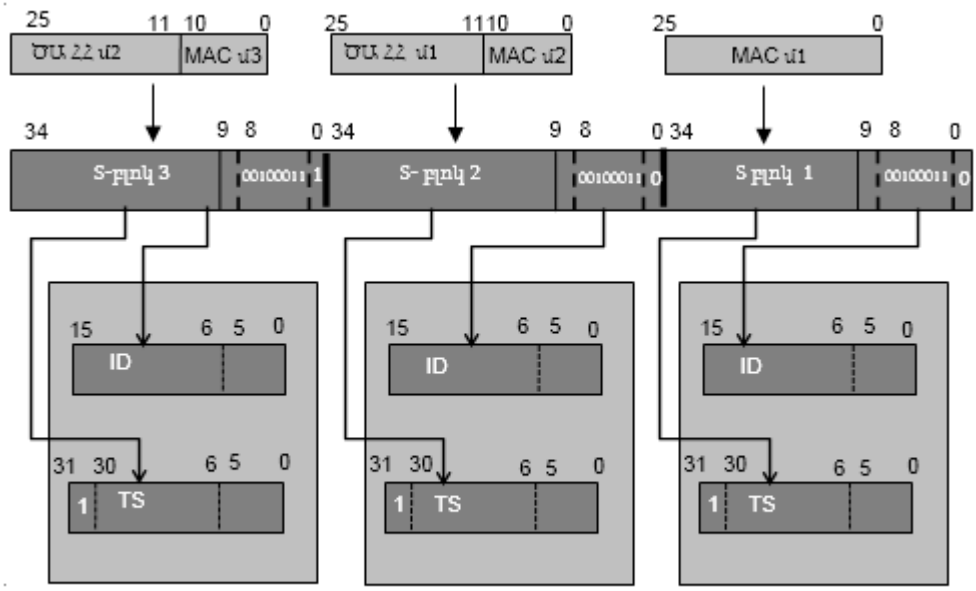
S-բլոկների հերթականությունը կազմված է երեք 35 բիթ ծավալով բլոկներից:

$$S_i = x_{i,34}..x_{i,2}x_{i,1}x_{i,0} \quad (4)$$

$x_{i,0}$ բիթը բնորոշում է, թե թաքնագրված տեղեկատվության որ հատվածն է պարունակում բլոկը, «1» արժեք տեղակայվում է միայն վերջին S-բլոկի կրտսեր բիթը:

$x_{i,8}...x_{i,1}$ հերթականությունը պարունակում է թաքնագրային-բանալին, որը ստուգվում է ստացող կողմում՝ ստացված փաթեթը որպես թաքնագրային կրիչ տարբերակելու նպատակով, բանալու դերում օգտագործվում է «##» սիմվոլային հերթականության երկուական ներկայացումը.

- *Առաջին բլոկի* x_0 բիթը տեղակայվում է «0», $x_8...x_1$ բիթերում գրանցվում է «00100011» բանալիային հերթականությունը, իսկ $x_{34}...x_9$ բիթերը պարունակում են ուղարկողի ցանցային քարտի MAC հասցեի առաջին 26 բիթը:
- Երկրորդ S-բլոկի կրտսեր բիթը նույնպես տեղակայվում է «0», իսկ թաքնագրային-բանալու զբաղեցրած դաշտերին հաջորդում են MAC հասցեի հաջորդ 11 բիթը և ԾԱ պատճենի հերթական համարի առաջին կեսը՝ համապատասխանաբար x19..x9 և x34..x20:
- Երրորդ S-բլոկը երկրորդից տարբերվում է միայն կրտսեր բիթով, որը տեղակայվում է «1» և բնորոշում է համապատասխան թաքնագրային կրիչը, որպես վերջինը տվյալ հերթականության մեջ:



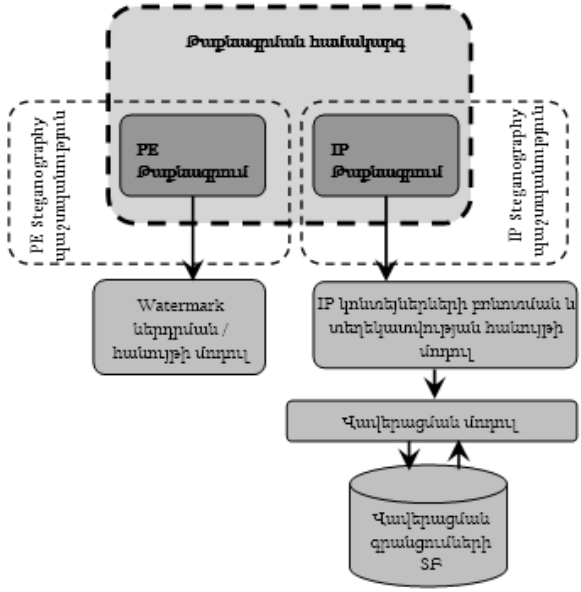
Նկ. 4: IP փաթեթներում վավերացման տեղեկատվության թաքնագրման մեխանիզմ

Օգտագործողի ցանցային քարտի և ծրագրային ապահովման հերթական համարի ստուգման համար երեք փաթեթի օգտագործումը թույլ է տալիս ստուգել և գրանցել մինչև «1073741823» պատճեն:

IP հաղորդակարգի փաթեթների մեջ տեղեկատվության ներդրման վրա հիմնված մեխանիզմը ապահովում է ՕԱ պաշտպանություն չթույլատրված օգտագործումից և պատճենների տարածումից:

<u>Չորրորդ գլուխը</u> նվիրված է առաջարկված թաքնագրման միջոցների և պաշտպանական մեխանիզմների ծրագրային իրականացմանը: Նախորդ գլուխներում նկարագրված ՕԱ պաշտպանական մեխանիզմները միավորված են ընդհանուր SProtect պաշտպանական համակարգի մեջ: Ծրագրային ապահովման պաշտպանության SProtect համակարգը բաղկացած է 5 միմյանցից ֆունկցիոնալապես անկախ մոդուլներից (Նկ. 5).

- “PE steganography” մոդուլ, որը իրականացնում է գլուխ 2-ում նախագծված պաշտպանական մեխանիզմը:
- “IP steganography” մոդուլ, որը իրականացնում է գլուխ 3-ում նախագծված պաշտպանական մեխանիզմը:
- IP թաքնագրային կրիչից տեղեկատվության հանույթը իրականացնող սերվերային մոդուլ:
- Տվյալների բազայի հետ համագործակցող և ՕԱ պատճենի վավերականացում իրականացնող սերվերային մոդուլ:
- PE ձևաչափի ֆայլերի մեջ տեքստային ջրանշանի ներդրման/հանույթի մոդուլ:



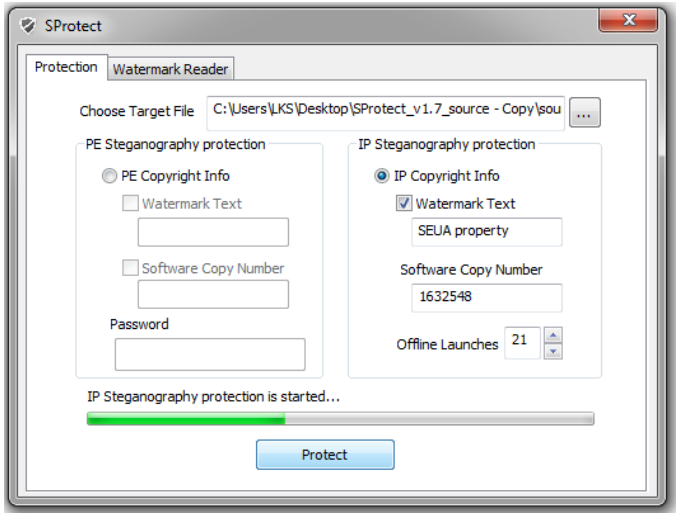
Նկ. 5: ԾՄ-ի պաշտպանական SProtect համակարգի մոդուլային կառուցվածքը

Առաջին երկու մոդուլը հանդիսանում են ԾԱ պաշտպանության մոդուլներ, ցանկացած մոդուլի ընտրության դեպքում SProtect համակարգը թույլ է տալիս լրացուցիչ ընտրել տեքստային ջրանշանի ներդրման մոդուլի գործարկումը: Ջրանշանի ներդրումը թույլ է տալիս, ծրագրային ապահովման գողության, կամ կորստի դեպքում, ապացուցել արտադրողի հեղինակային իրավունքները՝ օգտագործելով պաշտպանված ծրագրից ջրանշանի հանույթի մոդուլը:

SProtect ծրագրային համակարգի ինտերֆեյսը բերված է Նկար 6-ում.

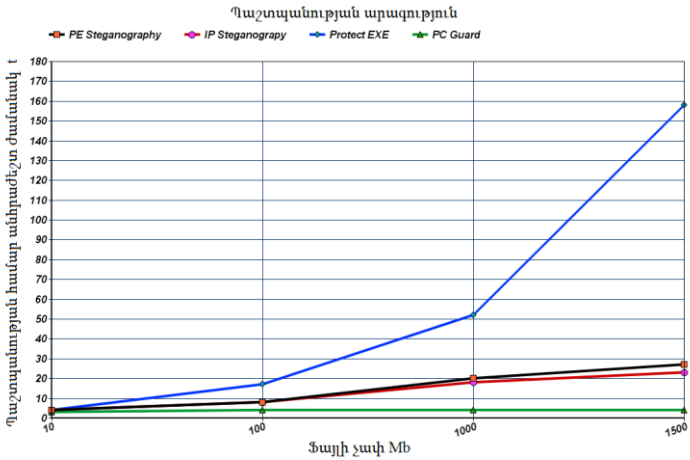
Այս գլխում իրականացվել է նաև ծրագրային ապահովման պաշտպանության համակարգի արագագործության գնահատում: Համեմատվել է այն ժամանակը, որի ընթացքում համակարգը պաշտպանում է մուտքին տրված կատարվող ֆայլը: Համեմատվել են «PE steganography (SProtect 1.7)», «IP steganography (SProtect 1.7)», «Protect EXE» և «PC Guard» պաշտպանական մեխանիզմները: Ստացվել է Նկար 7-ում բերված գրաֆիկը, որը բնութագրում է պաշտպանության վրա ծախսվող ժամանակի կտրուկ աճը «Protect EXE» համակարգի մոտ՝ պաշտպանվող ֆայլի չափի մեծացման դեպքում: Ատենախոսական աշխատանքի ընթացքում առաջարկվող մեխանիզմների ժամանակային բնութագրերը ունեն կայուն աճ և մնում են ընդունելի միջակայքում:

«PC Guard» համակարգի կողմից պաշտպանության վրա ծախսվող ժամանակի աճի բացակայությունը՝ կատարվող ֆայլերի ծավալի աճի դեպքում, պայմանավորված է նրանով, որ կատարվող ֆայլին կցվում է առանձին պաշտպանական մոդուլ, ինչը հեշտ շրջանցվող մոտեցում է՝ պաշտանության տեսանկյունից (Նկ. 7):



Նկ. 6: ՕՍ-ի պաշտպանական SProtect համակարգի ինտերֆեյս

Ինչպես նաև իրականացվել է առաջարկված թաքնագրային միջոցների դիմացկունության գնահատում օգտագործելով Խի-քառակուսու չափանիշի վրա հիմնված տեսական-վիճակագրական մեթոդը: Գնահատման արդյունքները ցույց տվեցին, որ «PE steganography» պաշտպանական մեխանիզմի հիմքում ընկած թաքնագրման մոտեցումը ապահովում է մինչև 2.5 անգամ դիմացկունության աճ համեմատած «File Handle change" steganography» և «Section slack" steganography» մոտեցումների հետ: Իսկ «IP steganography» պաշտպանական մեխանիզմի հիմքում ընկած թաքնագրային մոտեցումը մինչև 40% դիմացկունության աճ է ապահովում՝ համեմատած «Retransmission steganography» և «Packet Length Based Steganography» մոտեցումների հետ:



Նկ. 7: Պաշտպանության ընթացքի ժամանակային ցուցանիշի համեմատման գրաֆիկ

Աշխատանքի հիմնական արդյունքները հետևյալն են՝

1. Հետազոտվել են ծրագրային ապահովման պաշտպանության գոյություն ունեցող մեթոդները և մեխանիզմները, հետազոտությունների արդյունքում առաջարկվել է պաշտպանության բազային և թաքնագրական մեթոդների համատեղ օգտագործման մոտեցում, որը ապահովում է պաշտպանության և կայունության նոր մակարդակ [1]:
2. Առաջարկվել է Portable Executable ձևաչափում ինֆորմացիաի ներդրման թաքնագրային միջոց, որը ապահովում է մինչ 2.5 անգամ դիմացկունության աճ համեմատած կատարվող ֆայլերում տեղեկատվության ներդրման այլ մոտեցումների հետ և որը հետագայում կիրառվել է ԾԱ պաշտպանության մեխանիզմի նախագծման մեջ [1, 2]:
3. Առաջարկվել է IP հաղորդակարգ փաթեթի վերնագրում ինֆորմացիայի ներդրման թաքնագրային միջոց, որը ապահովում է դիմացկունության աճ 40%-ով՝ համեմատած ցանցային հաղորդակարգերի կիրառմամբ թաքնագրության այլ մոտեցումների հետ, և որը հետագայում կիրառվել է ԾԱ պաշտպանության մեխանիզմի նախագծման մեջ [3]:
4. Մշակվել է գաղտնաբառային պաշտպանության, գաղտնագրության և կատարվող ֆայլերում թաքնագրության մոտեցումների համալիր օգտագործման վրա հիմնված, ծրագրային ապահովումը չթույլատրված օգտագործումից և ձևափոխումից պաշտպանող մեխանիզմ և համապատասխան ծրագրային մոդուլ [1, 2, 4]:
5. Մշակվել է Ինտերնետ նույնականացման, կոնկրետ ապարատային միջավայրում ԾԱ աշխատանքի կազմակերպման և IP փաթեթներում թաքնագրության համալիր օգտագործման վրա հիմնված, ծրագրային ապահովումը չթույլատրված օգտագործումից և պատճենումից պաշտպանող մեխանիզմ և համապատասխան ծնագրային մոդուլ [3, 4]:
6. Նախագծվել է ծրագրային ապահովման պաշտպանության SProtect ծրագրային համակարգը [1, 2, 3]:

ԱՏԵՆԱԽՈՍՈՒԹՅԱՆ ԹԵՄԱՅԻ ՇՐՋԱՆԱԿՆԵՐՈՒՄ ՀՐԱՊԱՐԱԿՎԱԾ ԱՇԽԱՏՈՒԹՅՈՒՆՆԵՐԻ ՑԱՆԿ

[1] Даниелян В.М. «Защита от несанкионированного использования и копирования: комплексный метод против базовых», Научные труды Международной научной конференции «XXXVI ГАГАРИНСКИЕ ЧТЕНИЯ», Том 4, с. 83-85, 2010г, Москва.

[2] V. Markarov, V. Danielyan, “Software protection against unauthorized use and copying based on data hiding”, Proceedings of the Workshop “Applications of Information Theory, Coding and Security”, p. 63-66, 2010, Yerevan, Armenia.

[3] V. Danielyan, “Software protection based on IP steganography”, Proceedings of the conference Computer Science and Information Technologies (CSIT), p. 359-361, 2011, Yerevan, Armenia.

[4] Даниелян В.М. "Методы оценки систем защиты программного обеспечения" «Լրաբեր» գիտական հոդվածների ժողովածու 2 մասով, - Մաս. 1, Երևան, Ճարտարագետ, 2012, էջ. 250-254.

Виктор Даниелян

ИССЛЕДОВАНИЕ И РАЗРАБОТКА КОМПЛЕКСНОЙ СИСТЕМЫ СТЕГАНОГРАФИЧЕСКОЙ ЗАЩИТЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

РЕЗЮМЕ

Одной из наиболее актуальных проблем информационной безопасности в сфере проектирования программного обеспечения (ПО) является защита от несанкционированного использования, модификации, копирования и других нарушений авторских прав производителя ПО.

В настоящее время существует множество защитных алгоритмов и методов, которые отличаются как технологией обеспечения защиты, так и степенью защищенности. Вполне естественно, что многие защитные механизмы со временем теряют свою актуальность в связи с совершенствованием способов атак и применяемого вредоносного программного обеспечения. Распространенные механизмы защиты (защита паролем, связка с аппаратурой, шифрование, проверка пароля или уникальной информации через сеть) возможно обойти в случае определения алгоритма защиты. Важно помнить, что практически любой механизм защиты хотя бы теоретически можно взломать или обойти. И, следовательно, адекватным можно считать тот механизм, на взлом которого потребуется на много больше затрат времени и средств, чем на легальную покупку программного обеспечения. Кроме своих конкретных недостатков стандартные механизмы защиты программного обеспечения имеют один общий - их использование бывает видимым для пользователя (введения пароля, уведомление о временных рамках использования бета-версии, обязательное подключение к Интернету при онлайн аутентификации и т.д.) и, следовательно, для злоумышленника.

В этой связи представляется актуальным разработка новых, более совершенных методов защиты, в частности основанных на мало изученных стеганографических методах, которые призваны скрыть от противника сам факт наличия механизмов защиты или их части, и тем самым исключить провоцирование противника на активные хакерские действия. Так же актуальность использования стеганографических подходов основывается на слабой защищенности ПО, где защитный механизм реализован как отдельный модуль и не внедрен в само программное обеспечение.

Так же особый интерес представляет комплексное использовании нескольких принципиально различных защитных алгоритмов для уменьшения очевидности используемых методов защиты ПО.

Из вышеизложенного следует, что разработка средств комплексной защиты программного обеспечения от несанкционированного использования, модификации и копирования является актуальной задачей.

Цель и задачи работы – исследование и разработка стойких и надежных механизмов защиты программного обеспечения, основанных на стеганографических методах. Для достижения указанной цели в работе ставятся и решаются следующие основные задачи:

- Анализ существующих методов и механизмов защиты программного обеспечения, исследование возможностей современных стеганографических методов защиты информации и программного обеспечения в частности
- Разработка способов внедрения информации в исполняемые файлы и пакеты протокола IP, и на этой основе, разработка механизмов защиты программного обеспечения от типичных угроз.
- Создание алгоритмов и программных средств, реализующих защитную систему на основе разработанных механизмов.

Основные результаты и выводы

1. Проведен анализ существующих методов и механизмов зашиты программного обеспечения и предложен подход комплексного использования стеганографических методов защиты ПО вместе с базовыми методами, что предоставит новый уровень зашиты и стойкости [1].
2. Предложен стеганографический способ внедрения информации в исполняемые файлы Portable Executable, обеспечивающие повышение стойкости до 2.5 раза в сравнении со существующими способами внедрения в исполняемые файлы, который поставлен в основу разработки механизма защиты ПО [1, 2].
3. Предложен стеганографический способ внедрения информации в заголовок пакетов IP протокола, со стойкостью выше на 40%-ов в сравнении с существующими способами внедрения в IP, который поставлен в основу разработки механизма защиты ПО [3].
4. Разработаны механизм и соответствующий программный модуль для защиты ПО от нелегального использования и модификации, основанный на взаимном использовании методов парольной зашиты, криптографии и стеганографии в исполняемых файлах [1, 2, 4].
5. Разработаны механизм и соответствующий программный модуль для защиты ПО от нелегального использования и дистрибуции нелегальных копий, основанный на взаимном использовании методов Интернет верификации, привязки ПО к аппаратуре и стеганографии в IP пакетах [3, 4].
6. Спроектировано программное средство SProtect, обеспечивающее защиту программного обеспечения от нелегального использования, модификации и копирования [1, 2, 3].

Viktor Danielyan

RESEARCH AND DEVELOPMENT OF THE COMPLEX SYSTEM FOR SOFTWARE STEGANOGRAPHIC PROTECTION

RESUME

The software protection from unauthorized use, modification, copying and other copyright violations is the most important subjects of the information security in the sphere of software development.

Currently, there are a lot of protective algorithms and methods, which differ in both security technology and the degree of protection. Naturally, many protection mechanisms eventually lose their relevance by reason of improving the methods of hacker attacks and malware software they use. In case of protection algorithm detection, common software protection mechanisms (password protection, hardware linking, password or unique information verification through a network) are possible to break or avoid.

Therefore, we may assume that an adequate mechanism will require much more time and costs for breaking it, than for purchasing the protected software.

In addition to their own disadvantages the standard software protection mechanisms, have a common characteristic, the use of protection algorithms (password entry, software beta-version usage time notifications, Internet availability for online identification, etc.) is visible to the user, therefore, to the enemy.

It seems urgent to develop new, more sophisticated methods of protection, in little studied steganographic methods of protection, which are intended to hide the very existence of protective mechanisms from the enemy and thereby prevent hacking activities. Relevance of the work is also based on the need to avoid the wildly used software weak protection where the protective mechanism is implemented as a separate module and is not embedded in the software itself.

Of particular interest is the integrated use of several fundamentally different security algorithms, to reduce the obviousness of protection methods used.

From the above it follows that the development of the means of complex software protection against unauthorized use, modification and copying is an urgent goal.

The purpose and objectives

The main purpose of the work is to research and develop a stable and reliable software protection mechanism, based on steganographic methods. To achieve this goal, the following tasks must be solved:

- Analyze the existing software protection mechanisms and make a decision on their stability.
- Research the possibility of the steganography in executable files and network protocols. Develop the mechanisms that will protect software from common threats and based on researched steganographic approaches.
- Develop protection systems that implements algorithms and software tools based on researched mechanisms.

The main results are:

1. The examination of the existing software protection methods and mechanisms is performed, The protection based on the complex use of common protection methods along with the steganographyc methods is suggested, which will provide a new level of protection and stability [1].
2. The steganographic technique for information storing in Portable Executable (PE) files is proposed, that provides durability increase up to 2.5 times comparing to existing methods, this technique was later used during the design of software protection mechanism [1, 2].
3. The steganographic technique for information storing in IP protocol packets' headers is proposed, that provides durability increase up to 40% in compare to existing techniques and was later used during the design of software protection mechanism [3].
4. The software protection mechanism and software module, which performs protection from unauthorized use and modification, are proposed. The mechanism is based on password protection, cryptography and steganography in PE files [1, 2, 4].
5. The software protection mechanism and software module, which performs protection from unauthorized use and distribution of illegal copies, are proposed. The mechanism is based on Internet based identification, hardware linking and steganography in IP packets' headers [3, 4].
6. Software protection SProtect software tool is developed, that provides software protection from unauthorized use, modification and distribution [1, 2, 3].

Ծավալը՝ 21 էջ: Տպաքանակը՝ 100:
ՀՀ ԳԱԱ ԻԱՊԻ կոմպյուտերային պոլիգրաֆիայի լաբորատորիա:
Երևան, Պ. Սևակի 1