УДК 004.057.4

М.І. Главчев, О.І. Баленко

Національний технічний університет «Харківський політехнічний інститут», Харків

ФОРМУВАННЯ ПРОГРАМНОГО КОМПЛЕКСУ ЗАХИСТУ КОМЕРЦІЙНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПЕРСОНАЛЬНОГО ВИКОРИСТАННЯ

Розглянуто формування програмного комплексу захисту комерційного програмного забезпечення на основі послідовності рівнів захисту, що включають формування ключової інформації, організацію прихованої мітки, захист від відлагодників, захист від дизасемблювання, online-підтримка контролю запуску.

Ключові слова: *захист програмного забезпечення, захист програмного коду, мітка захисту, захист від відлагодників, захист від дизасемблювання.*

Постановка проблеми

Інформаційно-комп‘ютерне суспільство розвивається все більш швидкісними темпами, обсяги використання результатів розробок професійних та аматорських програмістів вже декілька останніх років мають стабільну тенденцію к збільшенню. Слід зазначити, що визначаючи загальний спад економіки України, галузь інформаційних технологій і розробки програмного забезпечення виказує значні темпи постійного росту (у 2016р. 15-20%)[1].

Однак для більш ефективного отримання прибутку від реалізації програмних розробок необхідно встановити якісні припони «комп‘ютерним піратам» та неліцензійному програмному забезпеченню. Якщо на світовому ринку у 2015 році частка неліцензійного програмного забезпечення (ПЗ) складала 39%, то на українських персональних комп‘ютерах (ПК) ця частка визначена у розмірі 82% [2]. Зазначимо, що присутня тенденція до зменшення цього показника на один відсоток у рік, але у сучасний час це не є вирішенням проблеми порушення авторських прав на програмні продукти.

Наявність законодавчої бази для захисту ліцензійного ПЗ не може захистити виробника від використання його розробок у особистих («тестових», «дослідних») цілях на персональних домашніх комп‘ютерах.

Вирішення питання ліцензійного використання ПЗ можливо лише за рахунок додання до програмних розробок якісного комплексу захисних програмних модулів, які максимально ускладнять роботу «піратів» з отримання кодів доступу, реєстраційних ключів, тощо.

Загальний підхід формування захисту програмного продукту

На підставі проведено розгляду існуючих методів захисту програмного забезпечення і багаторічного досвіду виконання робіт у даному напрямку та викладання дисциплін присвячених захисту інформації [3, 4] можливо представити загальну структуру схему комплексного програмного захисту ПЗ (рис.1). Ця структурна схема містить розподілення виконання захисту на декілька рівнів, послідовність яких забезпечить високий ступень захисту програмної розробки при використанні якісних складових на кожному рівні.

Рис. 1. Структура програмного захисту

Розглянемо складові наведеної структури і визначимо з них ті, що будуть використані при формуванні програмного комплексу захисту.

Визначення складових програмного комплексу захисту

Рівень 1 – уявляє основну, найбільш значну частину системи захисту, яка відповідає за формування і перевірку ключової інформації. Ця частина додається до оригінального коду програмної розробки і є її складовою.

Рівень 1 присутній у всіх ліцензійних програмних розробках, що призначені для комерційного використання.

Забезпечення цього рівня можливо різноманітними засобами, наприклад:

- парольний захист;
- визначення характеристик ПК;
- внесення відповідних значень у файл реєстру операційної системи (ОС);
- додаткові електронні ключи;
- магнітні мітки;

тощо.

Найбільш популярний і звичайний для користувача – це парольний захист[5]. Існує багато варіантів його організації у напрямках видів паролю і алгоритмів формування перевірочної інформації. Визначення характеристик ПК використовується для визначення унікальності апаратних засобів і зазначення зв‘язку з програмною розробкою. Обмеження даного засобу визначається неможливістю переносу ліцензійної копії на інший ПК з відмінними характеристиками.

Також даний засіб вимагає попереднього визначення характеристик і подальшої їх постійної перевірки. Цей засіб все ж використовується разом з парольним захистом. Слід зазначити, що електронні ключи є додатковими пристроями, які не завжди є можливість передати ліцензованому користувачу, і використовуються вони звичайно з спеціалізованими проектними системами, або з базами даних обмеженого доступу.

Магнітні мітки – це занесення у певні області носія ідентифікаційної інформації. Однак засоби ОС та антивірусні засоби стежать за такими змінами і в цілому ці засоби не уявляють проблемі їх виявлення.

Звернемо увагу на організацію файлової прихованої мітки за рахунок «округлення кластеру». Цей засіб використовує обсяг вільного місця останнього кластеру будь-якого файлу. В наслідок того, що довжина файлу не кратна розміру кластера, то у останньому кластері є достатньо місця для розміщення додаткової інформації. Існує проблема вибору файлів до яких буде додана прихована мітка. Пропонується використовувати файли не програмної розробки, а файли ОС. Слід виключити ті файли стан яких контролюють антивірусні засоби. Треба створити перелік файлів, що відповідають визначеним вимогам, і обрати певний з них на підставі ключа підтвердження і мережних характеристик персонального комп‘ютера, що забезпечить певну унікальність файлу з списку для конкретного користувача програмного продукту.

Рівень 2 виконує делікатну функцію недопущення використання засобів відлагодження програмного коду для визначення особливостей ключового захисту Рівня 1.

Зауважимо, що ці засоби не забороняють використовувати програми-відлагодники, але роблять роботу фахівця з аналізу програмного коду дуже складною [6].

Додання засобів цього рівня виконується за рахунок виконання спеціальних перевірок у засобах Рівня 1, бажано у моменти перевірки ключової інформації. До основних засобів цього рівня слід віднести наступне:

- застосування конвеєру команд процесора для зміни програмного коду;
- перевірку відповідних бітів регістру прапорів процесору для визначення роботи під програмною-відлагодником;
- контроль використання окремих переривань ОС, які можуть застосовувати програми-відлагодники.

Використання цих засобів неоднозначне. Перевірка бітів регістру прапорів процесору є досить простою задачею і обхід цього засобу нескладний. Засіб контролю використання переривань може бути обмежений ОС, що не дозволить у повному обсязі застосувати цю можливість. Застосування конвеєру команд центрального процесору, навпаки, поширюється з розвитком обсягу внутрішнього кешу команд процесору.

Основа використання цього засоби лежить у зміні коду програми під час її виконанню у оперативній пам‘яті і неможливості зміни цього ж фрагменту у коді, який вже завантажений у кеш процесору (конвеєр команд).

На лістингу 1 наведений тестовий приклад, який виводить повідомлення «Test» у разі роботи без відлагодника, а при роботі під управлінням відлагодника модифікує різними засобами програмний код (заміна коду команди, заміна операнду команди, заміна фрагменту програми), що приводить к значний його зміні і зависанні самої програми.

Використання засобів Рівня 2, а особливо конвеєру команд, є дуже якісним методом, але побудова відповідного коду вимагає від розробника певного рівня знань системного програмування та архітектури процесору.

Лістинг 1

Тестовий приклад
використання конвеєру команд

```
RADIX 16
code1 segment para
assume cs:code1
pr1  proc far
       ret
pr1  endp
code1 ends
code  segment para
assume cs:code, ds:code
       mes db 'Test',0dh,0a,'$'
Start:
          push cs
          pop ds
          mov  di, word ptr m0
          and byte ptr m1+1, 0fbh
          mov ax, 5101
          mov cx, 51
          and di, 0FF
          mov bx, offset m0-9
          push bx
          jmp near ptr m0
          call far ptr pr1
          mov ah, 4c
          int 21h
m0:
          aaa
          or ax,0ea
          push ds
          pop es
          cld
          jmp short m1
m1:
          repz scasw
          ret
code ends
          end start
```

На Рівні 3 у програмному комплексі захисту бути виконане перетворення деяких фрагментів коду для забезпечення неефективного статичного вивчення складових захисту [7]. Основні методи Рівня 3 наступні:

- шифрування усього коду програмної розробки або окремих критичних елементів;
- динамічне перетворення програмного коду під час роботи ПЗ;
- заміна команд на еквівалентні по функціоналу, але виконується порушення «логічного» розуміння коду;
- боротьба з фахівцем за рахунок ускладнення програмного коду з великою кількістю програмних переходів.

Заміна команд на еквіваленти вимагає попереднього виділення додаткових байт в залежності від довжини команд заміни або довжини фрагментів коду, що будуть вставлені. Ускладнення коду – засіб, розширює загрузочну частину програмного продукту і не є складним засобом, але вимагає значних додаткових затрат часу для аналізу.

Шифрування коду звичайно виконується за рахунок створення окремого бінарного модуля, який виконує первинне завантаження основної частини, яка вже зашифрована, розшифровує її та передає керування.

Шифрування коду також дозволяє захиститися від антивірусних засобів. Проблема у тому, що деякі елементи захисту можуть мати схожість з діями комп‘ютерних вірусів.

Більш складним засобом є динамічне перетворення, яке виконується різними засобами: поетапне дешифрування коду в залежності від коректності отриманих попередніх результатів; модифікацію основного коду на підставі накладення за допомогою логічної операції «виключне АБО» з іншим підготовленим фрагментом коду; завантаження на «некоректний код» блоку «вірного коду» наприклад з стеку та інше.

Рівень 3 – є зовнішнім рівнем, який приховує інші. Для запобіганню зміни коду файлів програмної розробки рекомендуємо виконувати перевірку цілісності файлів.

Ця перевірка повинна виконувати двома або більшою кількістю контрольних розрахунків, що пов‘язано з можливістю компенсації виконаних змін.

Наприклад, контрольна сума командою додавання без переносу байтів, контрольне значення двобайтових значень операцією «виключне АБО», контрольна відмінність без позику байтів після виконання операція «логічне заперечення», тощо. Як варіант перевірки контрольного значення можливо зберігати контрольні розрахунки певних (важливих) блоків файлів для зменшення часу перевірки.

Для забезпечення ефективного функціювання програмного комплексу захисту ліцензійного ПО у сучасному Інтернет-просторі необхідно організувати online-підтримку перевірки ліцензійності ПЗ. Для виконання цього запропонована послідовність дій реєстрації програмного продукту, яка наведена на рис. 2.

На підставі ліцензійного ключа і даних користувача на сервері сертифікації (розробника) виконується формування ключа підтвердження. Рекомендується для формування ключа підтвердження використовувати хеш-перетворення і криптографічні алгоритм з відкритими ключами.

На етапі запуску програмного продукту виконується зчитування прихованої мітки і перевірка її з еталоном на сертифікаційному центрі. У якості рекомендації, слід перевіряти IP- і MAC-адреси підключення і при частої їх зміні тимчасово блокувати запуск програмного продукту для запобігання «піратському» використанню розробки.

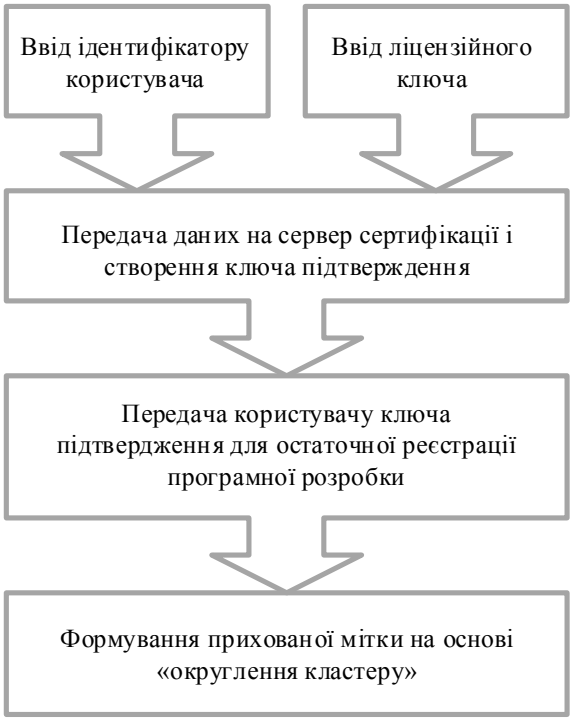

Рис. 2. Реєстрація програмного продукту

Запропонований комплекс не враховує використання у віртуальних мережах, але у майбутньому це буде вирішене за рахунок унікальних характеристик персональних комп'ютерів. Для цього треба використовувати характеристики, які найбільш стабільні у конфігурації, до яких слід віднести номер центрального процесору, характеристики материнської плати, номер жорсткого диску.

Висновки

Забезпечення перевірки ліцензійності ПЗ є обов'язковою дією для захисту авторських прав розробника. Запропонований програмний комплекс захисту програмного продукту і є засобом перевірки ліцензії ПЗ, який реалізує у повній мірі збалансовану і повноцінну послідовність засобів захисту. Ця послідовність включає: формування ключа підтвердження на сервері розробника на підставі ліцензійного ключа і даних користувача; організація прихованої мітки за рахунок «округлення кластеру»; перевірка прихованої мітки; захист коду процедур організації і перевірки прихованої мітки засобом використання конвеєра команд центрального процесору; динамічне перетворення фрагментів коду, які відповідають за роботу з організацією и перевірку ліцензійності ПЗ. Все це в цілому значно знизить нелегальне використання програмного продукту.

Список літератури

1. Игорь Беда, GlobalLogic Украина: К 2020 году экспорт украинского программного обеспечения может вырасти до $10 млрд [Елетронний ресурс]. URL: http://hubs.ua/starter/i-beda-globallogic-ukraina-k-2020-godu-eksport-ukrainskogo-po-mozhet-vy-rasti-do-10-mlrd-97379.html

2. Украинские пираты держат позиции: доля нелицензионного ПО в стране в 2015 году составила 82%/ [Елетронний ресурс]. URL:https://ain.ua/ukrainskie-piraty-derzhat-pozicii-dolya-nelicenzionnogo-po-v-strane-v-2015-godu-sostavila-82

3. Главчев М.І. и др. Захист інформації. Навчальний посібник. Ч. 1. Захист інсталяційних програм. Харків: НТУ "ХПІ", 2007. 164 с.

4. Методы защиты от исследования программ [Елетронний ресурс]. - Режим доступа: URL: http://mf.grsu.by/UchProc/livak/arxiv_22102010/kursi/zaschita/lections/zlec12.htm

5. Главчев М.І., Аннануров А.Д. К вопросу создания парольной защиты. Международная научная конференция MicroCAD : Секція №21 - Інформатика і моделювання - НТУ "ХПИ", 2013. – С.12

6. Панов А. С. Реверсинг и защита программ от взлома. – БХВ-Петербург, 2006.

7. Алейников С. И., Богатов А. О. Защита программ от дизассемблирования //Труды Института системного программирования РАН. – 2006. – Т. 11.

Надійшла до редколегії 18.09.2016

Рецензент: д-р техн. наук, с.н.с. С.Г. Семенов, Національний технічний університет «Харківський політехнічний інститут», Харків.

ФОРМИРОВАНИЕ ПРОГРАММНОГО КОМПЛЕКСА ЗАЩИТЫ КОММЕРЧЕСКОГО ПЕРСОНАЛЬНОГО ИСПОЛЬЗОВАНИЯ

М.И. Главчев, А.И. Баленко

Рассмотрено формирование программного комплекса защиты коммерческого программного обеспечения на основные последовательности уровней защиты, включающих формирование ключевой информации, организацию скрытой метки, защита от отладчиков, защита от дизассемблирования, online-поддержка контроля запуска.

***Ключевые слова**: защита программного обеспечения, защита программного кода, метка защиты, защита от отладчиков, защита от дизассемблирования.*

CREATING PROGRAM COMPLEX FOR THE PROTECTION OF COMMERCIAL PERSONAL USE

M.I. Glavchev, O.I. Balenko

The formation of complex software protection software to provide commercial-tion on a-new sequences layers of protection, including the formation of the key information, the organization of the hidden mark, protection against debuggers, anti-disassembling, of online-support to start monitoring.

***Keywords:** software protection, protection of software code, security label, for on-board debugger, from the disassembly protection.*