МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

ЦЕНТР ЗАОЧНОЇ, ДИСТАНЦІЙНОЇ ТА ВЕЧІРНЬОЇ ФОРМ НАВЧАННЯ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

ВИПУСКНА РОБОТА

на тему:

«Аудит безпеки модуля ліцензування прикладного програмного забезпечення»

Завідувач випускаючої кафедри **Довбиш А.С.**

Керівник роботи **Кузіков Б.О.**

Студента групи ІНЗ – 73 – 9с **Марчук А.П.**

СУМИ 2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

Центр заочної, дистанційної і вечірньої форм навчання

Кафедра комп'ютерних наук

Затверджую_____

Зав. кафедрою Довбиш А.С.

"_____" _____2021 р.

ЗАВДАННЯ

до випускної роботи

Студента четвертого курсу, групи ІНз-73-9с спеціальності "Комп'ютерні науки" заочної форми навчання Марчук Артема Павловича.

Тема: "**Аудит безпеки модуля ліцензування прикладного програмного забезпечення** "

Затверджена наказом по СумДУ

№_____ от _____2021 р.

Зміст пояснювальної записки: 1) аналітичний огляд матеріалів по захисту програмного забезпечення; 2) постановка завдання; 3) огляд програми з точки зору устрою захисту ; 4) розробка утиліти, яка вираховує код активації ; 5) рекомендації до посилення захисту програми.

Дата видачі завдання "_____"_____2021 р.

Керівник випускної роботи _____ Кузіков О.Б.

Завдання приняв до виконання _____ Марчук А.П.

РЕФЕРАТ

Записка: 42 стор., 25 рис., 1 додаток, 6 джерел.

Об'єкт дослідження — Модуль ліцензування програми «Підприємець 4.2».

Мета роботи — Проаналізувати літературу по темі захисту програмних продуктів, також видів обходу захисту. Дослідити програму «Підприємець 4.2», зокрема, модуль ліцензування. Реалізувати утиліту, яка буде вираховувати код активації із вхідних даних.

Методи дослідження — метод ревер-інженерінгу, або т.н. зворотного проектування.

Результати — Проаналізована література по темі захисту програмних продуктів, також видів обходу захисту. Досліджена програма «Підприємець 4.2», зокрема, модуль ліцензування. Реалізована утиліта, яка вираховує код активації із вхідних даних. Дані рекомендації стосовно посилення захисту програми.

ЗМІСТ

ВСТУП

Одним із аспектів, що шкодить сталому розвитку ІТ індустрії є проблеми пов’язані із дотриманням законодавства, щодо прав інтелектуальної власності. Так за даними Frontier Economics, шкода правовласникам що може бути нанесена до 2022 року складає 2,3 трлн. Доларів США. [7] Одним із видів таких збитків є використання неліцензійних копій ПЗ. Для вирішення цієї проблеми індустрія реалізує низку програмно-технічних та організаційних заходів (Шифрування, SaaS, реклама у безкоштовних додатках, недорогі підписки на контент). У випадку десктопного програмного забезпечення найбільш розповсюдженою проблемою є неліцензійне використання.

Актуальністю даної теми є те, що захист програм треба робити дуже надійним. Одна із складових безпеки продукту – захист від тиражування та не ліцензійного користування. Порушення безпеки у цій частині продукту, можуть призвести до значних фінансових втрат.

Виходячи з актуальності описаної проблеми, метою роботи буде аудит модуля ліцензування програми «Підприємець 4.2»

Для досягнення поставленої мети сформульовані наступні завдання роботи:

1. Проаналізувати літературу по темі захисту програмних продуктів, також видів обходу захисту.
2. Дослідити програму «Підприємець 4.2», зокрема, модуль ліцензування.
3. Реалізувати утиліту, яка буде вираховувати код активації із вхідних даних.

1 МЕТОДИ ТА ВИДИ ЗАХИСТУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ВАРІАНТИ ЇХ ОБХОДУ

1.1 Методи захисту програмного забезпечення в інформаційному середовищі умовно можна розділити на три типи:

- «**Захист програмного забезпечення** – це заходи, спрямовані на захист програмного забезпечення від несанкціонованого придбання, використання, поширення, модифікування, вивчення і відтворення аналогів.»[4]

- «**Захист від несанкціонованого використання програм** – заходи, спрямовані на протидію нелегальному використанню програмного забезпечення. При захисті можуть застосовуватися різні засоби.»[4]

- «**Захист від копіювання** до програмного забезпечення застосовується рідко, в зв'язку з необхідністю його поширення та установки на комп'ютери користувачів. Однак, від копіювання може захищатися ліцензія на додаток (при поширенні на фізичному носії) або його окремі алгоритми.»[4]

Існують наступні види захисту програмного забезпечення:

- «**Локальний програмний захист** - вимога введення серійного номера (ключа) при установці / запуску програмного забезпечення. Історія цього методу почалася тоді, коли додатки поширювалися тільки на фізичних носіях (компакт-дисках). На коробці з диском був надрукований серійний номер, що підходить тільки до даної копії програми.»[4]

- «**Мережевий програмний захист** - ділиться на локальний - сканування мережі виключає одночасний запуск двох програм з одним реєстраційним ключем на двох персональних комп'ютерах в межах однієї локальної мережі та глобальний - якщо програма працює з якимось централізованим сервером і без нього марна (наприклад, сервери оновлень антивірусів , сервери оновлення правових програм, таких як антивірус Касперського).»[4]

- «**Захист за допомогою компакт-дисків** - програма може вимагати оригінальний компакт-диск. Як правило, цей спосіб захисту застосовується

для захисту програм, записаних на цьому ж компакт-диску, який є одночасно ключем;»[4]

- «**Захист за допомогою електронних ключів** - вставлений в один з портів комп'ютера (з інтерфейсом USB, LPT або COM) носій, що містить ключові дані, називається також ліцензією, записані в нього розробником;»[4]

- «**Прив'язка до параметрів комп'ютера і активація** - прив'язка до інформації про користувача / серійним номерам компонентів його комп'ютера і подальша активація програмного забезпечення в даний момент використовується досить широко. В процесі установки програмне забезпечення підраховує код активації - контрольне значення, однозначно відповідає встановленим комплектуючих комп'ютера і параметрам встановленої програми. Це значення передається розробнику програми.»[4]

- «**Захист програм від копіювання шляхом перенесення їх в мережу Інтернет** - стрімко набирає популярність метод захисту, який полягає в наданні функціоналу програм (всього або частини), як сервісу онлайн, в мережі Інтернет. При цьому код програми розташований і виповнюється на сервері, доступному в глобальній мережі.»[4]

- «**Захист коду від аналізу** - засоби захисту безпосередньо коду програми від аналізу та використання в інших програмах. Зокрема, застосовуються обфускатор - програми для заплутування коду з метою захисту від його аналізу, модифікації та несанкціонованого використання.»[4]

1.2 Варіанти обходу захисту програмного забезпечення

Для початку потрібно встановити яким саме чином захищене програмне забезпечення.

- **Локальний програмний захист –** найпростіший варіант пошукати серійний номер в інтернеті, але не завжди його можна знайти, тоді в нагоді стануть програми для реверс-інженерінгу або дисассемблеру.

Реверс-інжинерінг (або зворотне проектування) - це процес вилучення знань з того, що коли-небудь було зроблено людиною.

Найпопулярніші програми для цього OllyDbg (Рисунок 1.1) або IDA Pro (Рисунок 1.2), краще користуватись IDA Pro, IDA - абревіатура, яка розшифровується як "інтерактивний дизассемблер". Ключовим і революційним свого часу було саме "інтерактивний". Це означає, що в результаті роботи ви отримуєте не просто довжелезний асемблерний лістинг, а щось, де ви можете залишити свої замітки, в якому можна зробити "заміточки на полях", і вони змінюються по ходу всього лістингу. Можна ще порівняти з функцією рефакторінга в сучасних програмерских IDE - перейменували функцію в одному місці, вона перейменувалася всюди, змінну - аналогічно і т.д. Ваше завдання як дослідника - з величезного масиву, який складно аналізується, залишити винятково важливу інформацію і надати їй форму добре зрозумілу для людини. Саме інтерактивність IDA Pro і дозволяє робити це дуже ефективно, і на сьогоднішній день ніхто її в цьому не перевищує.

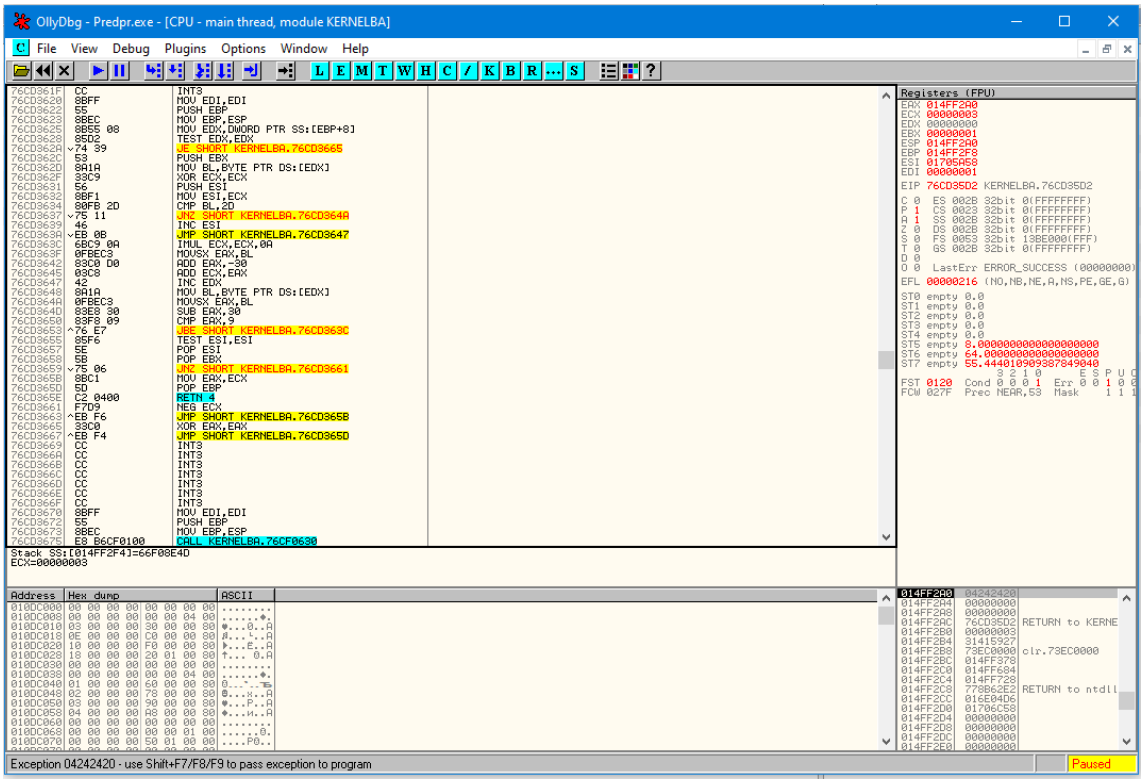

Рисунок 1.1. Загальний вигляд програми «Olly Dbg»

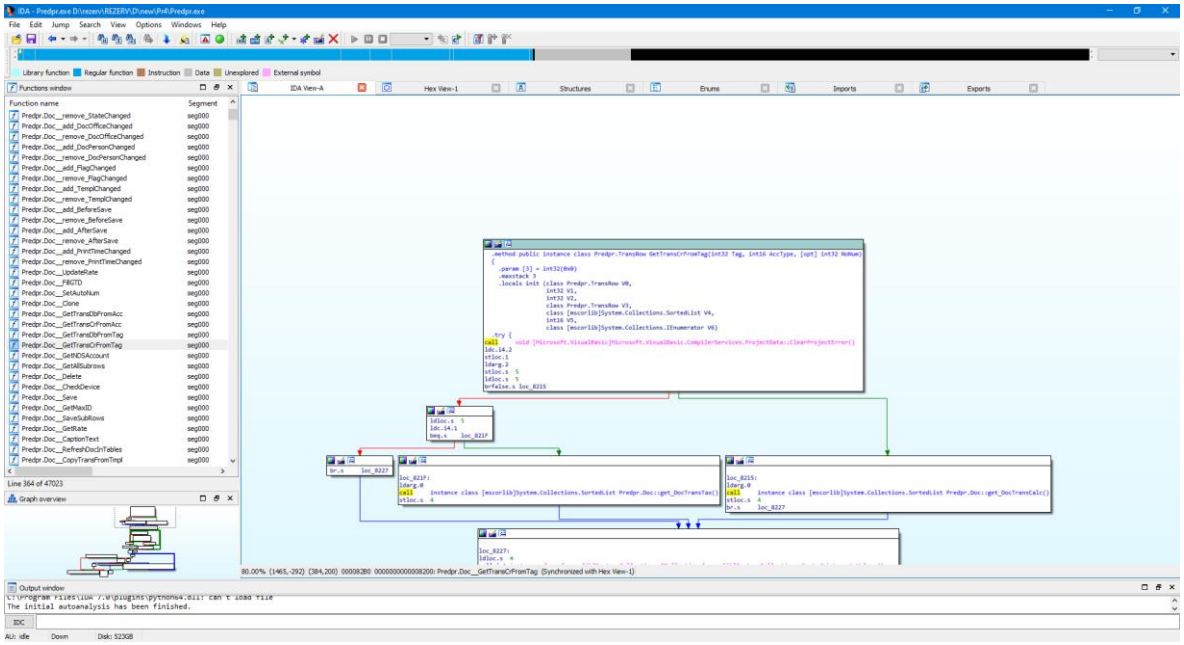

Рисунок 1.2. Загальний вигляд програми «IDA Pro 7.0»

Для того щоб обійти захист із вводом серійного коду, треба знайти точку у коді, де порівнюється введений користувачем серійний номер з якимось еталонним значенням, і після цього переходить до якоїсь іншої процедури, треба прослідкувати куди веде процедура порівняння, при невірному коді, та спробувати знайти ту процедуру на яку веде правильно введений серійний номер, також можна змінити умову порівняння. Якщо все вийде, при вводі невірного серійного коду ви отримаєте повідомлення, що код вірний і програма встановиться. Потім можна зберегти змінений файл, и при подальших встановленнях не потрібно буде знов шукати ці процедури.

- **Мережевий програмний захист –** найпростіший спосіб це блокування програмі доступу до мережі, або інтернету, або пере-адресація на емульований сервер.

- **Захист за допомогою компакт-дисків -** обійти захист за допомогою ключового диска можна декількома способами. Найпростіший спосіб - копіювання ключового диска. Більшість традиційних методів розраховано на неможливість для стандартної програми-копіювальника створення копії ключового диска. Однак поряд зі стандартними копіювальниками існують і спеціальні програми, які роблять копіювання диска методом "біт в біт" - так звані

побітові копіювальники. При копіюванні такі програми не залежать від стандартних параметрів форматування, тобто не перевіряють ні загальне число, ні нумерацію доріжок, ні кількість і нумерацію секторів на доріжці, ні інші особливості організації диска. Вони безпосередньо побітно відтворюють одну доріжку за одною, ігноруючи неотфармотовані доріжки. Таким чином, всі методи, які використовують нестандартне форматування ключового диска, можуть бути "обійдені" за допомогою побітових копіювальників. В даний час доступна також і спеціальна апаратура для аналогового копіювання дисків (наприклад, плати для побітового копіювання).

Інший спосіб зняття захисту з допомогою ключового диска, який може бути застосований в тому випадку, якщо побітовий копіювальник не справляється із завданням створення копії диска, - моделювання звернень до ключового диску.

Ідея полягає в імітації ключового диска. Спеціальна програма-імітатор побудована таким чином, що, будучи резидентно запущеною під час роботи захищеної програми, за допомогою підміни апаратних переривань повертає на запити захисту необхідні їй коди завершення і помилки.

За допомогою даного методу зазвичай знімають, наприклад, захист дисків з фізичним ушкодженням поверхні. Аналогічним моделюванням обходять захист дисків, що містять псевдосбійні кластери. Наприклад, читанням секторів диска можна попередньо визначити номери псевдосбійних кластерів, а далі створити програму-імітатор, що повертає механізму захисту необхідні номери секторів.

- **Захист за допомогою електронних ключів** – даний захист вважається самим надійним, але останнім часом також з’являються програми емулятори електронних ключів.

- **Прив'язка до параметрів комп'ютера і активація** – Принцип обходу такого захисту схожий із обходом локального захисту, але тут ще можна розібратись на якій мові написана програма і яким компілятором зібрана, це дасть змогу розібратись у логіці утворення ключа, та зробити «кейген» (про-

граму яка розраховує код активації), це один із найважчих способів обходу захисту активації.

Захист програм від копіювання шляхом перенесення їх в мережу Інтернет – способів обходу немає, так як немає самої програми, а тільки доступ до її функцій.

- **Захист коду від аналізу –** можна спробувати використання деобфускатора, все залежить від розміру програми та типу захисту.

1.3 Постановка задачі

Виходячи з аналізу літератури по темі захисту програмних продуктів, також видів обходу захисту треба проаналізувати механізми захисту програми за допомогою IDA Pro та PEiD. Реалізувати спосіб обходу захисту. Реалізувати утиліту, яка буде вираховувати код активації із вхідних даних та перевірити її в роботі. Надати рекомендації до посилення захисту програми.

2 ОГЛЯД ПРОГРАМИ «ПІДПРИЕМЕЦЬ 4.2» З ТОЧКИ ЗОРУ УСТРОЮ ТА ЗАХИСТУ

Для аудиту безпеки модуля ліцензування була обрана комерційна програма для бухгалтерського обліку «Підприємець 4.2», ця програма вже на ринку більше 10 років, та має дуже багато користувачів. На офіційному сайти можна скачати дистрибутив програми з безкоштовним періодом на 30 діб.

Після встановлення програми та створення тестової бази бачимо повідомлення про те скільки днів залишилось до завершення тестування. (Рисунок 2.1)

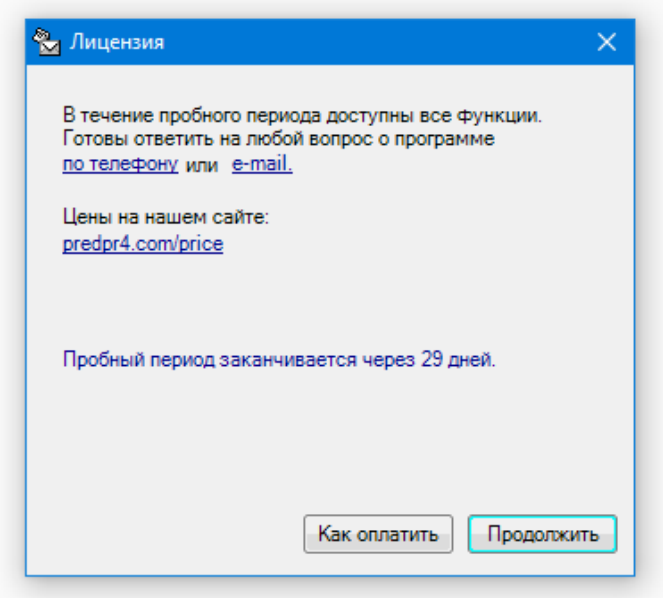

Рисунок 2.1. Діалогове вікно «Ліцензія»

На офіційному сайті знаходимо інформацію про ліцензування. Рисунок 2.2

Варианты		Код активации	USB-ключ	
		программа привязывается к одному компьютеру	переносится с одного компьютера на другой	Переход на SQL Server (необязательно)
Название	Описание	Первая покупка		
Стандарт	Устанавливается на каждый компьютер в сети.	2300 грн	4700 грн	+1200 грн
На сервере терминалов	Позволяет работать онлайн через интернет. Устанавливается на сервер, число пользователей неограниченно.		11900 грн	+5000 грн
На сервере терминалов + SQL Server	Работа через интернет, максимальная скорость вычислений и надежность. Число пользователей неограниченно.		16000 грн	
С ограниченной функциональностью				
в режиме "Товары"	Только для учета операций с товаром	1500 грн	-	-
в режиме "Документы"	Только для выписки первичных документов	1200 грн	-	-

Рисунок 2.2. Інформація про ліцензування с сайту «https://predpr4.com/»

Тобто бачимо, як мінімум 8 видів ліцензії

Подивимось як виглядає меню ліцензія. Рисунок 2.3, Рисунок 2.4

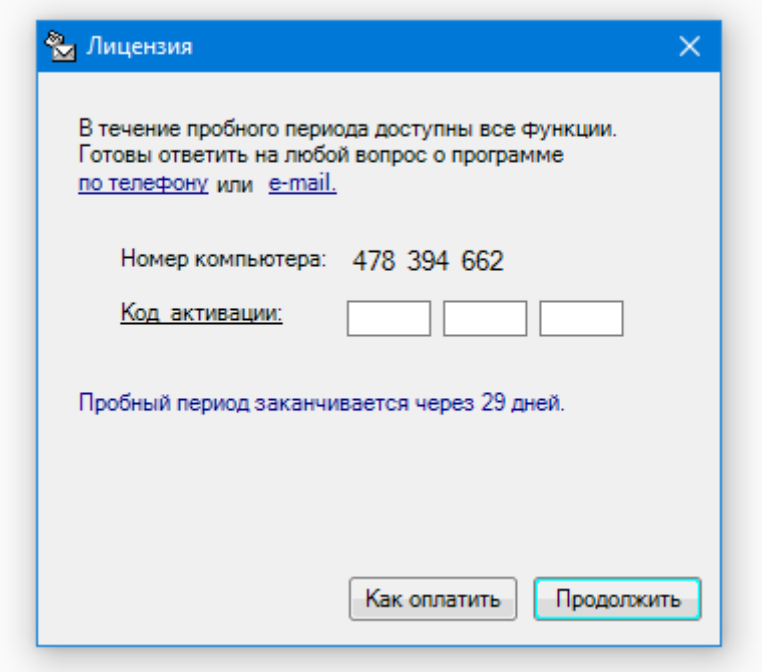

Рисунок 2.3. Діалогове вікно «Ліцензія»

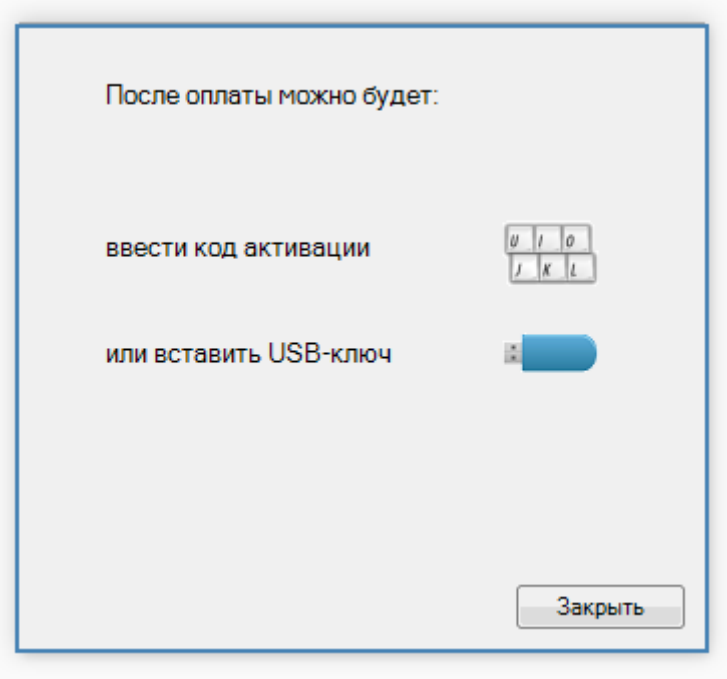

Рисунок 2.4. Діалогове вікно «Ліцензія»

Бачимо що номер комп’ютера тут унікальний і звідкись береться, і на різних комп’ютерах він різний. Звідси розуміємо що програма захищена видом захисту «**Прив'язка до параметрів комп'ютера і активація»**

Активація можлива за допомогою коду та USB ключа.

З офіційного сайту видно, що за допомогою коду програма активується для одного комп’ютера для звичайної повної версії, версії з режимом «документи», версії з режимом «Товари» та повної версії з доступом до SQL бази.

Версії «Сервер терміналів» та «Сервер терміналів SQL» активуються тільки за допомогою USB ключа.

Спробуємо дізнатися за допомогою якої мови написана ця програма, та який компілятор використовувався для її компіляції.

Для цього будемо використовувати програми PEiD.0.95 та DiE.0.65 Рисунок 2.5 та Рисунок 2.6

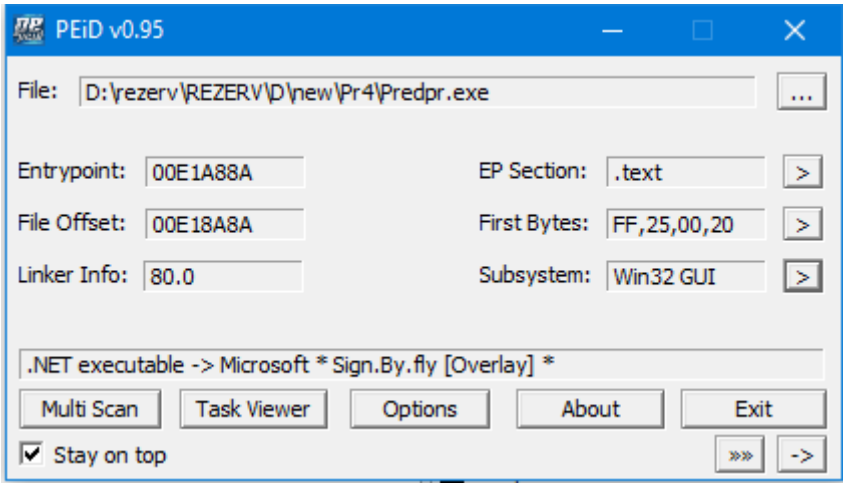

Рисунок 2.5. Інформаційне вікно «PEiD v0.95»

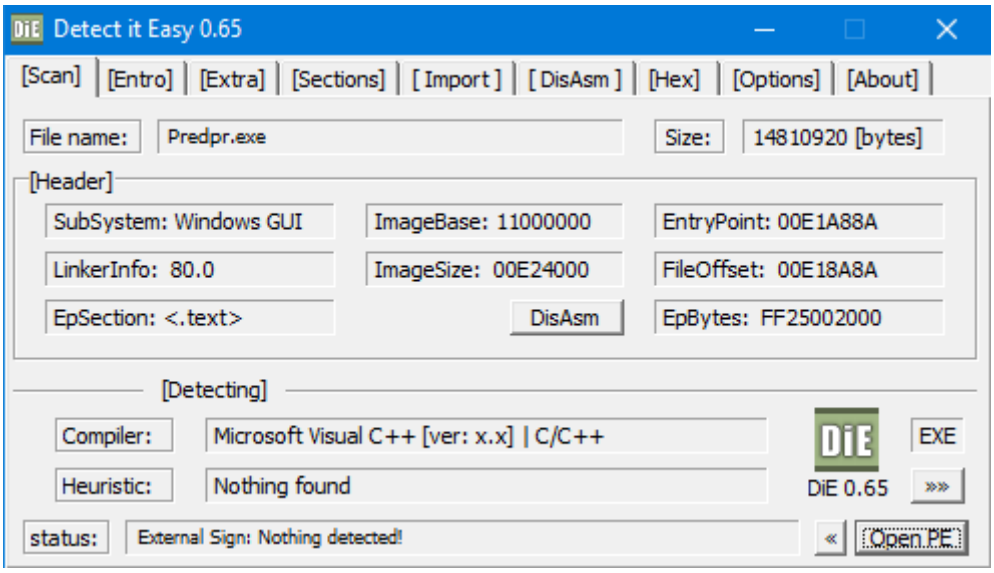

Рисунок 2.6. Інформаційне вікно «DiE v0.65»

За допомогою цих програм бачимо, що програма написана на мові Microsoft Visual C++ та скомпільована у середовищі .NET скоріш за все використовувалась Microsoft Visual Studio, також бачимо що при компіляції та після не використовувалось ніяких програм захисту коду.

Будемо тестувати безпеку модуля активації. Наскільки він добре захищен. Спочатку спробуємо скинути дату безкоштовного періоду, для цього видаляємо з реєстру данні про програму. Відкриваємо реєстр командою regedit з командної строки, та шукаємо данні за допомогою пошуку за ключем Predpr. Рисунок 2.7

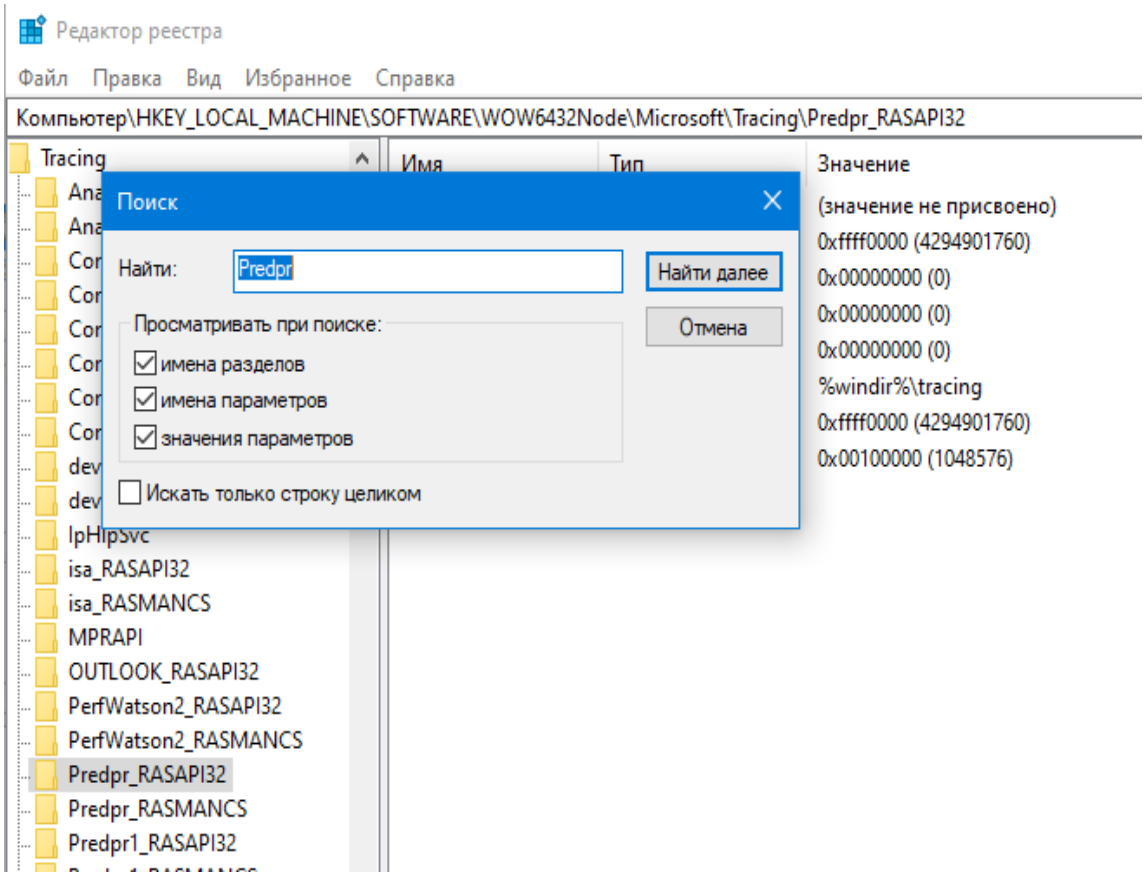

Рисунок 2.7. Вікно редактору реєстру операційної системи Windows

Після видалення всіх даних знайдених за назвою програми з реєстру, при запуску програми, безкоштовний період не змінився, але всі настройки зкинулись та файл з базою треба було вибирати заново. З цього робимо вивід що в реєстр записується прихований ключ, з датою початку використання безкоштовного періоду.

Так як очищення реєстру не допомогло, спробуємо запустити програму у середовищі відладника та дізасемблера IDA Pro. Рисунок 2.8, Рисунок 2.9.

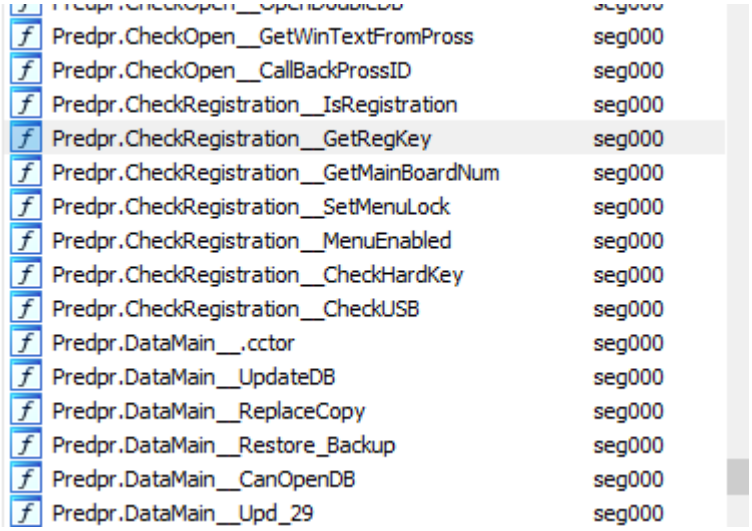

Рисунок 2.8.Вікно функцій програми «IDA Pro 7.0»

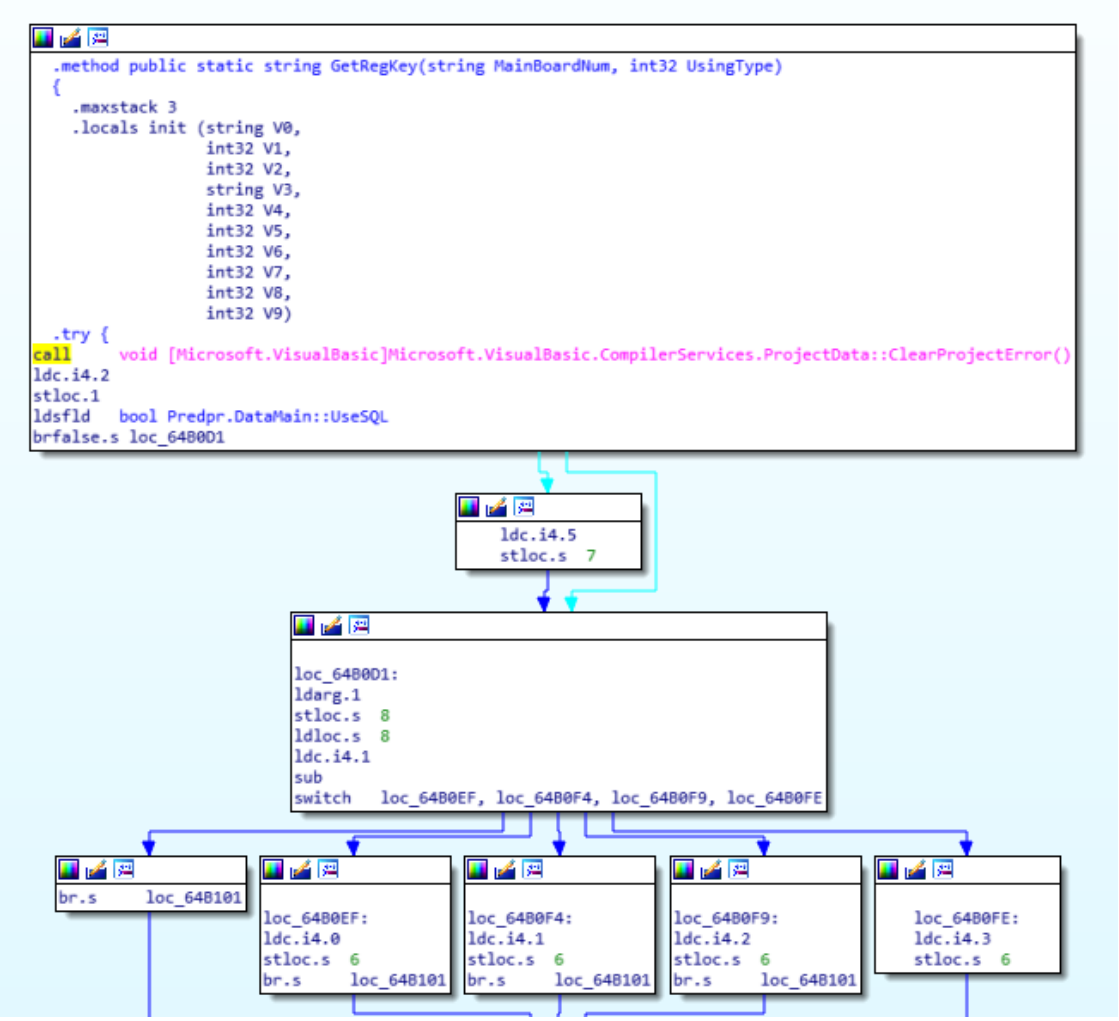

Рисунок 2.9. Вікно аналізу програми «IDA Pro 7.0»

IDA Pro, з модулем розпаковки .NET програм відмінно розпакувала програму, знайшов всі її функції, та наглядно демонструє, що та куди іде, логіку функцій.

Тут бачимо функцію GetRegKey з якої зрозуміло, що унікальний реєстраційний номер програма вираховує з серійного номеру материнської плати.

Також була знайдена функція прихованої реєстрації ключа реєстру у функціях GetHiddenRegistryKey та IsRegistration з яких зрозуміло що при першому відкритті програми, записуються ключі у реєстр с шифрованою датою першого запуску, а при подальших запусках перевіряються ці ключі, та вираховується кількість днів безкоштовного періоду. Рисунок 2.10, Рисунок 2.11

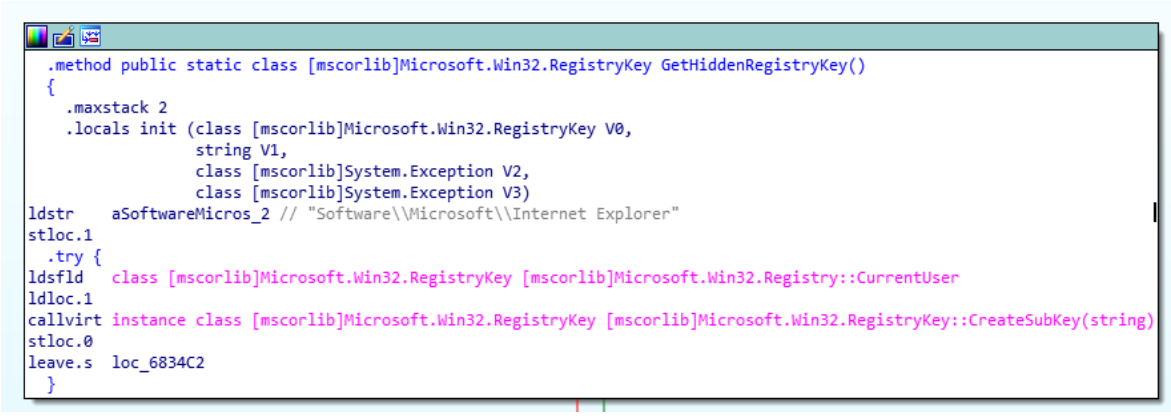

Рисунок 2.10. Вікно аналізу програми «IDA Pro 7.0»

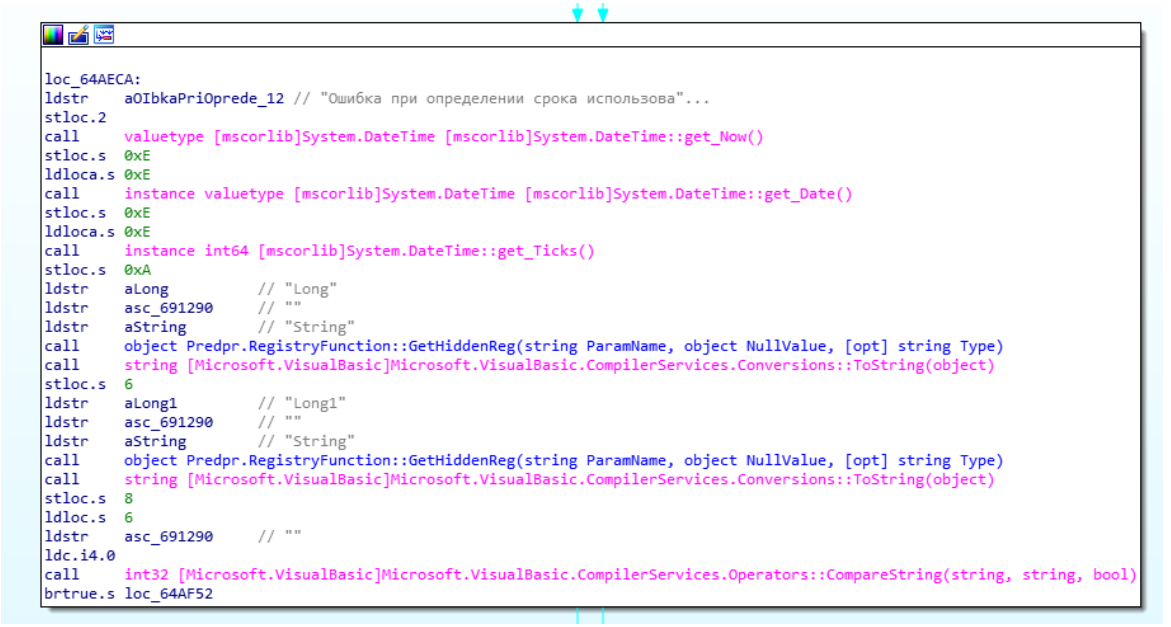

Рисунок 2.11. Вікно аналізу програми «IDA Pro 7.0»

Йдемо у реєстр та шукаємо ключі за адресою [HKEY_CURRENT_USER\Software\\Microsoft\\Internet Explorer] . Рисунок 2.12

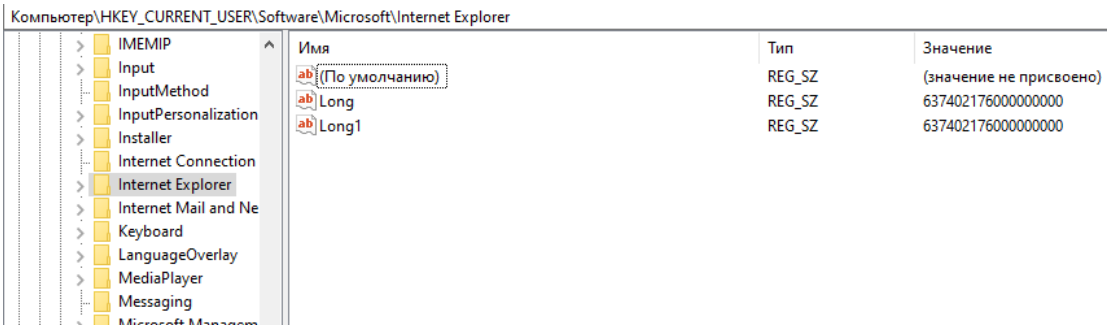

Рисунок 2.12. Вікно редактору реєстру операційної системи Windows

Бачимо ключі про які йшлось у коді відладника, та видаляємо їх для перевірки, після чого запускаємо програму та перевіряємо скільки днів залишилось безкоштовного періоду. Рисунок 2.13

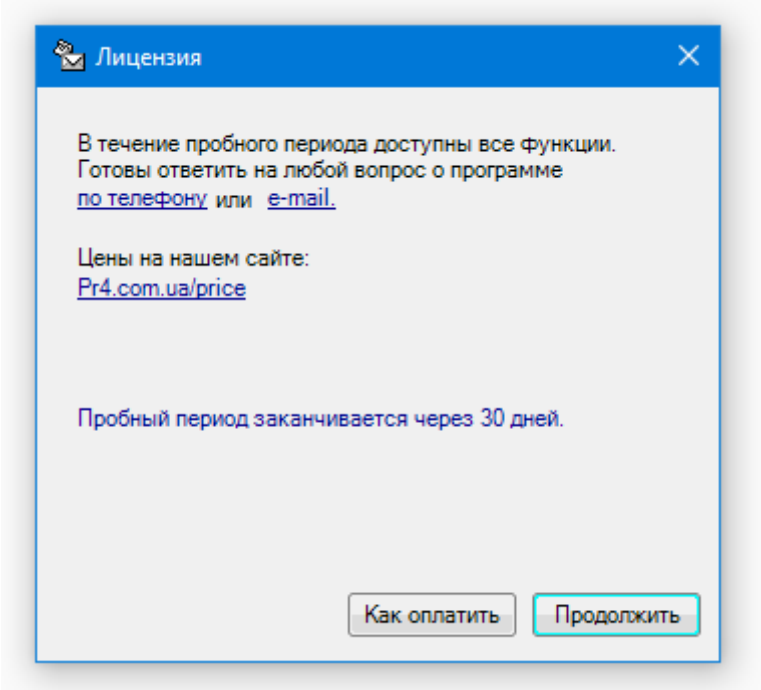

Рисунок 2.13. Діалогове вікно «Ліцензія»

І дійсно при видалені цих ключів, всі настройки програми збереглися, а безкоштовний період знову став 30 діб. Отже це перша знайдена можливість обходу модуля ліцензування програми. Для того щоб постійно не шукати це місце та якось автоматизувати скидання тестового періоду, можна зробити файл Predpr.reg з наступним кодом:

Windows Registry Editor Version 5.00

[HKEY_CURRENT_USER\Software\\Microsoft\\Internet Explorer]

"Long"=-

"Long1"=-

Зберегти та добавити у планувальник завдань на автоматичне виконання кожні 29 діб, і таким чином користуватись програмою безкоштовно та постійно.

Тепер спробуємо знайти функцію, яка відповідає за розрахунок коду активації. Для цього знаходимо у формі реєстрації, функцію кнопки «ОК», на яку натискаємо після вводу коду активації, та прослідкуємо які функції вона використовує. Рисунок 2.14, Рисунок 2.15, Рисунок 2.16

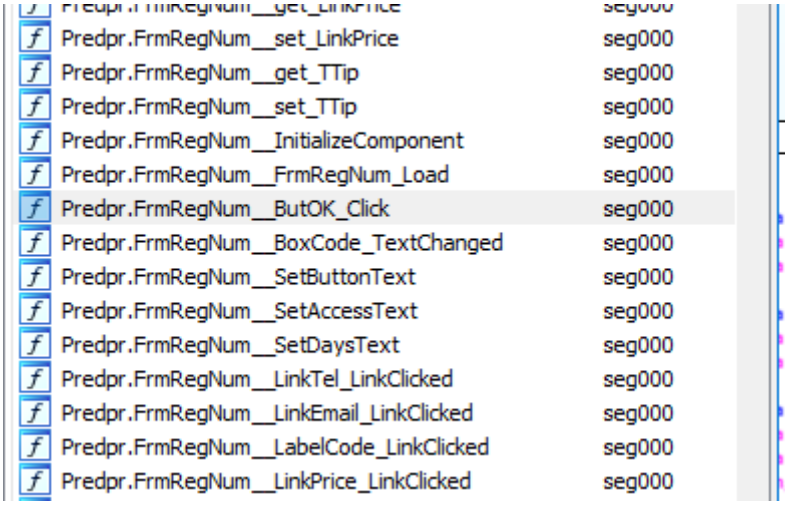

Рисунок 2.14.Вікно функцій програми «IDA Pro 7.0»

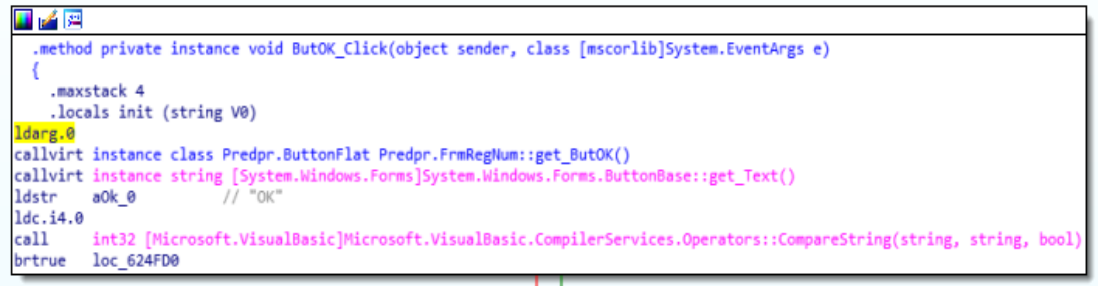

Рисунок 2.15. Вікно аналізу програми «IDA Pro 7.0»

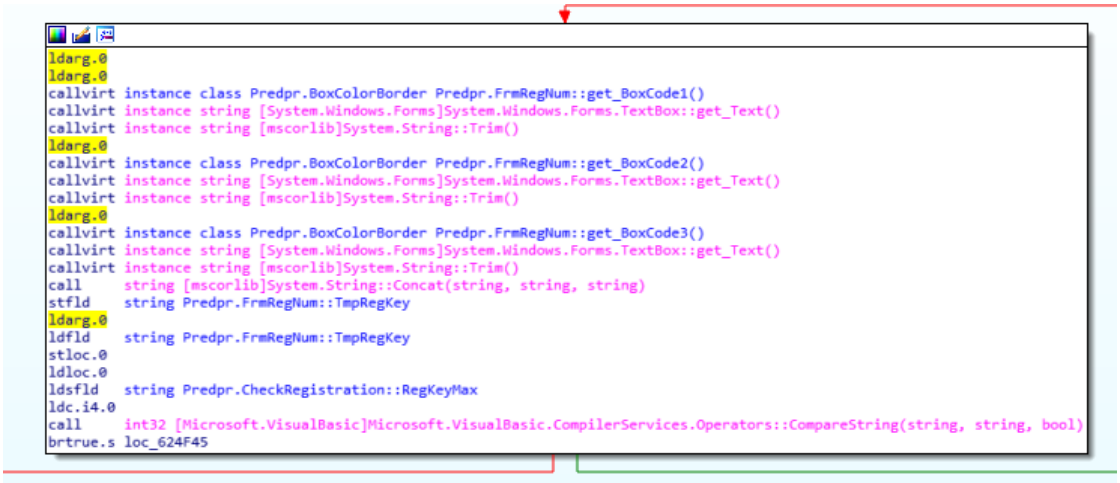

Рисунок 2.16. Вікно аналізу програми «IDA Pro 7.0»

Бачимо, що використовується функція Predpr.CheckRegistration, тому нам потрібно декомпілювати додаток, щоб отримати вихідний код цієї функції. Для цього знадобиться додаток .NET Reflector 8. Рисунок 2.17

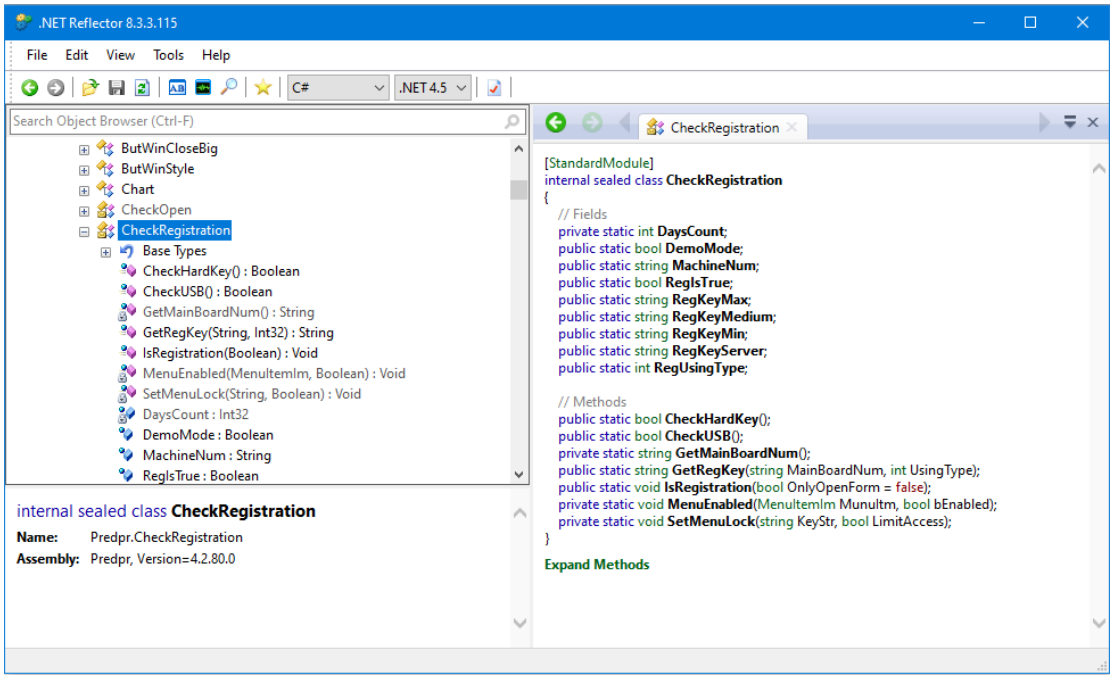

Рисунок 2.17. Вікно аналізу програми «.NET Reflector 8.3.3.115»

І дійсно у функції CheckRegistration є код який вираховує з номеру материнської плати, номер активації GetRegKey. Рисунок 2.18

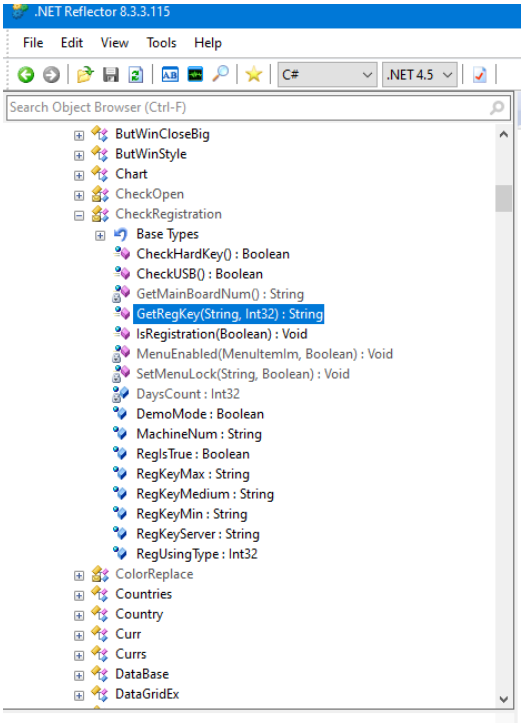

Рисунок 2.18. Вікно аналізу програми «.NET Reflector 8.3.3.115»

Дійсно у цій функції, є формула за якою вираховується код реєстрації для різних типів ліцензії. Знаючи ці данні можемо спробувати реалізувати утиліту, яка буде вираховувати код активації для різних типів ліцензії.

3 РЕАЛІЗАЦІЯ УТИЛІТИ ТА РЕКОМЕНДАЦІЇ ДО ПОСИЛЕННЯ ЗАХИСТУ

3.1. Реалізація утиліти типу КЕЙГЕН

Для реалізації утиліти обираємо середовище Microsoft Visual Studio та створюємо додаток Windows Forms (.NET Framework) на мові C#, так як декомпілятор показує вихідний код на цій мові. Рисунок 3.1

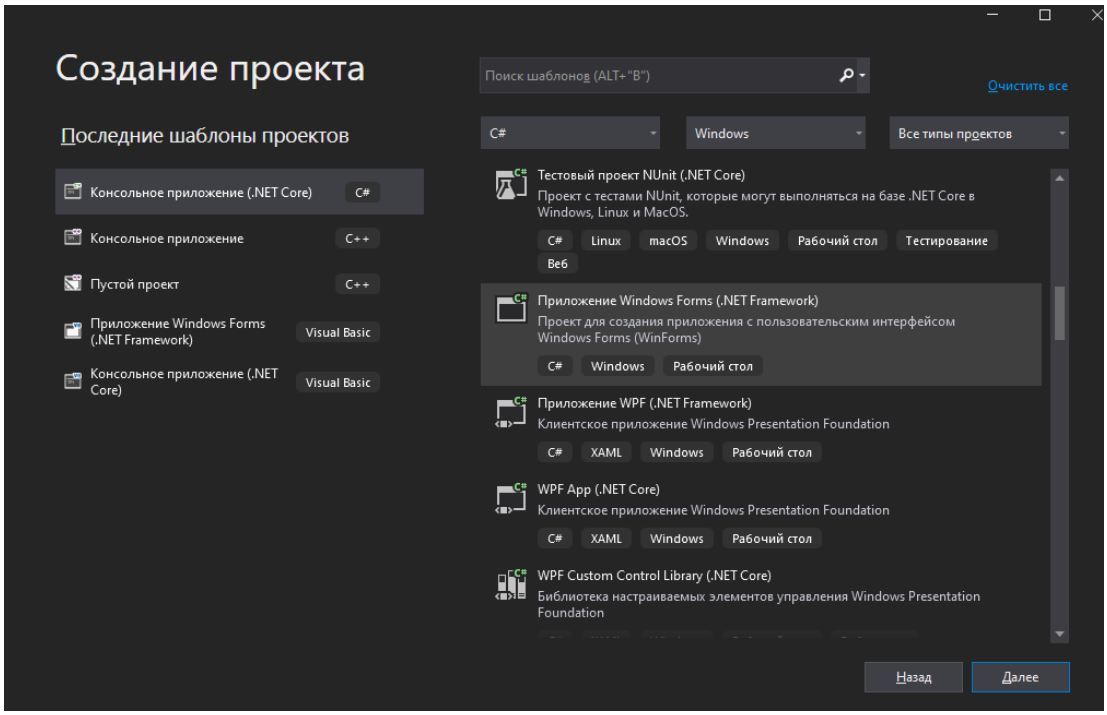

Рисунок 3.1. Вікно створення проекту «Microsoft Visual Studio»

Створюємо форму де буде вікно для внесення номеру комп’ютера, а також 8 полів для виведення ключів під різні типи ліцензії, та кнопка до якою прив’яжемо функцію розрахунку ключів. Рисунок 3.2

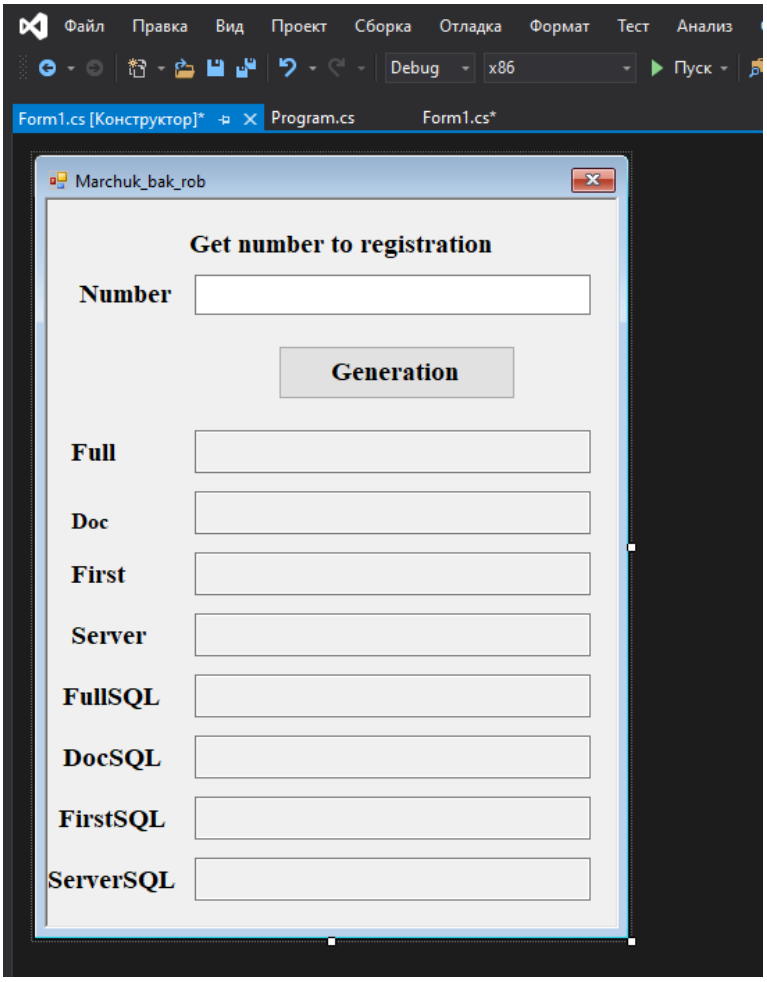

Рисунок 3.2. Вікно створення форми «Microsoft Visual Studio»

До кнопки приєднаємо дію на розрахунок ключів, з вихідного коду бачимо, що код розраховується по формулі:

Де num5 – це змінна, яка відповідає за тип ліцензії – повна, первинні документи, документи, та сервер. Тоді як змінна num6 – відповідає за версію локальну чи sql. Для цього робимо для кожного вікна виводу свою формулу з різними змінними, а MainBoardNum беремо із програми «ПІДПРИЕМЕЦЬ 4.2». Рисунок 3.3

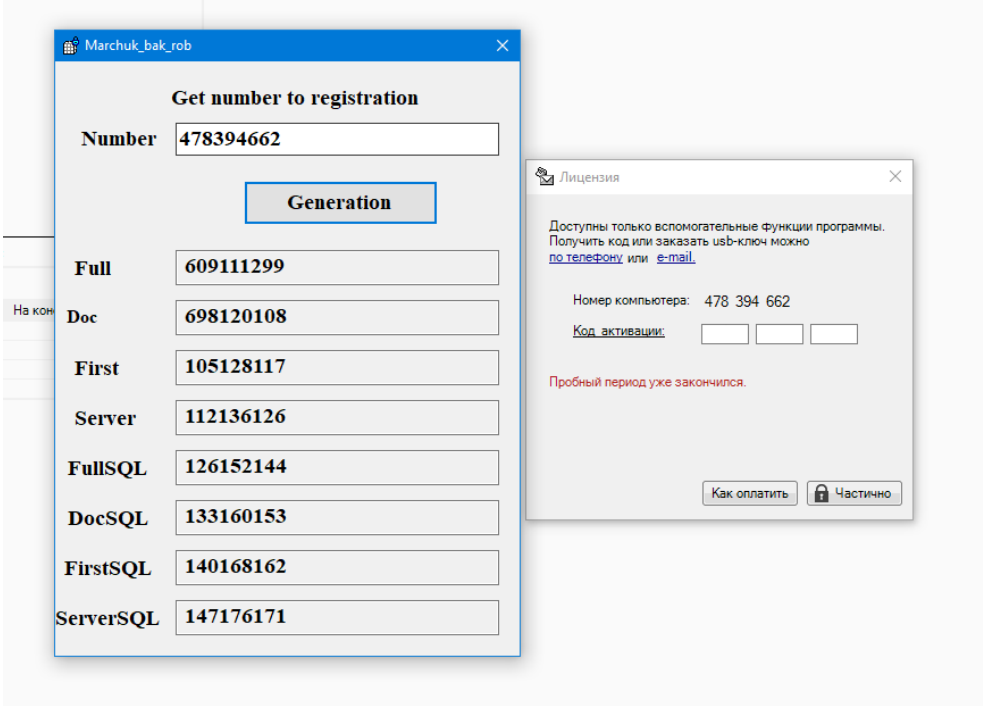

Рисунок 3.3. Вікно утиліти та діалогове вікно «Ліцензія»

Перевіримо чи вірно розраховується код активації. Рисунок 3.4, Рисунок 3.5

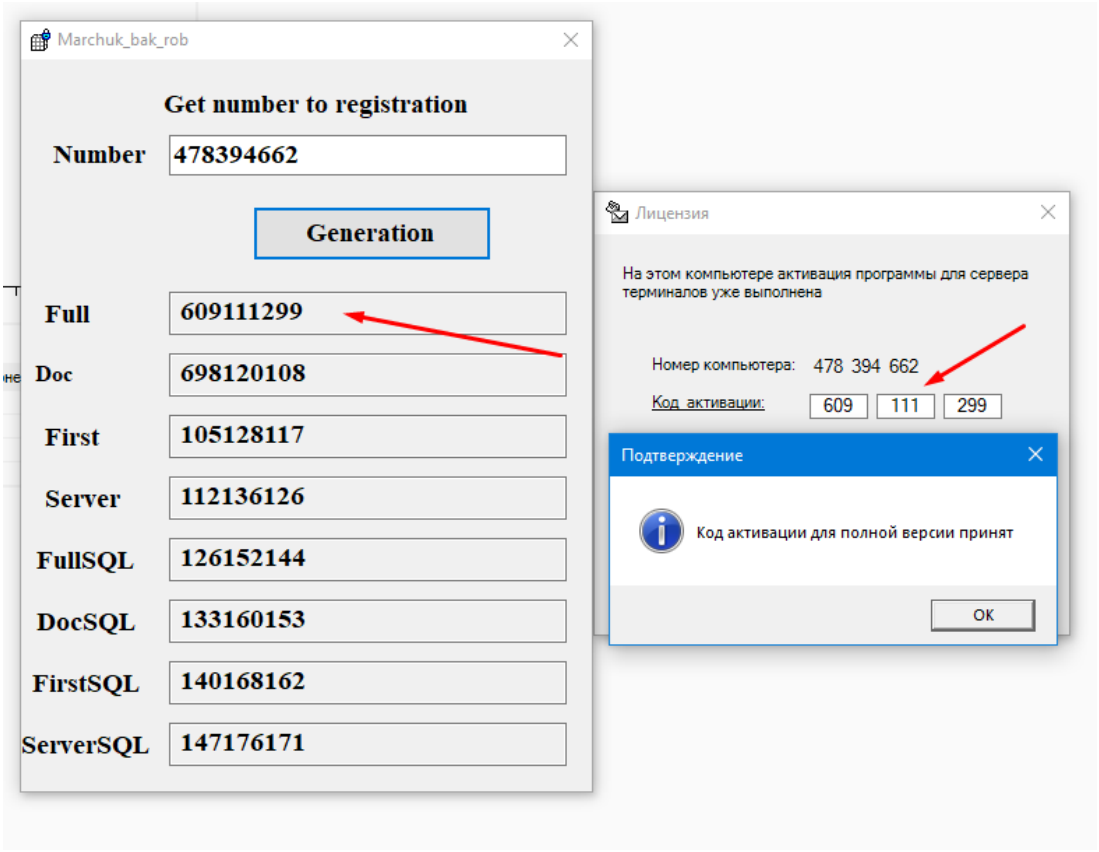

Рисунок 3.4. Вікно утиліти та діалогове вікно «Ліцензія»

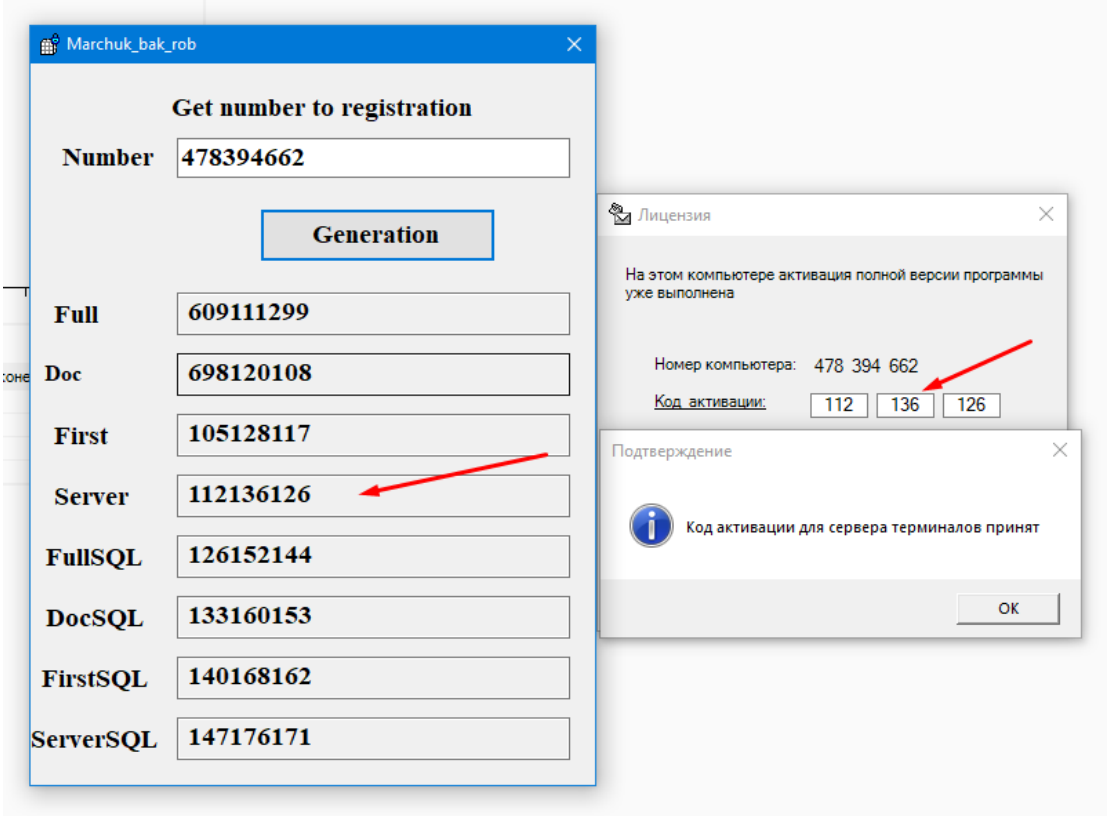

Рисунок 3.5. Вікно утиліти та діалогове вікно «Ліцензія»

Також всі інші ключі теж підходять, навіть ліцензія до якої офіційно не продають код активації, тільки USB ключ, теж приймається і програма робить, як сервер терміналів. Тож ця утиліта повністю обходить захист модуля ліцензування. Повний код утиліти додається у Додатку А.

3.2 Рекомендації до посилення захисту

Ліцензування програми це хороший спосіб захисту свого продукту, але якщо ця програма написана у середовищі .NET треба потурбуватись про захист вихідного коду, тому що є декомпілятори які дуже добре розкривають його. Треба використовувати обфускатор, який буде заплутувати імена, заплутувати потік виконання, шифрування ресурсів та констант, захищати від відладника та зняття дампу. Для цього можна використовувати безкоштовні версії з відкритим вихідним кодом, такі як ConfuserEx. Відкритий вихідні код дозволяє модифікувати систему захисту, змінювати сигнатуру обфускатора, ускладнюючи тим самим роботу програм де-обфускаторів і ручний реверс инженеринг. Як би код було неможливо прочитати, то неможливо було б

легко дізнатись куди записуються приховані ключі для реєстру та логіку розрахунку коду активації.

Висновок

У ході даної роботи були розглянуті основні методи та види захисту прикладних програм та способи їх обходу. Проаналізований механізм захисту програми за допомогою IDA Pro та PEiD. Реалізовано спосіб обходу захисту. Реалізована утиліта, яка вираховує код активації із вхідних даних та перевірена в роботі. Надані рекомендації до посилення захисту програми.

Загрози безпеці програмного забезпечення комп'ютерних систем виникають як в процесі їх експлуатації, так і при їх створенні, тому необхідно вносити захисні функції в програмне забезпечення на всьому протязі його життєвого циклу. Найбільш ефективним є використання комплексу програмно-технічних і правових засобів захисту програмного забезпечення. Правові засоби захищають інтелектуальні права, а також права власності розробників і власників програмного забезпечення. Правовий захист містить законодавство про авторське право і передбачає відповідальність за порушення авторських прав. Програмно-технічні засоби здійснюють протидію нелегальному використанню програмного забезпечення, шляхом використання різних технічних засобів.

Можна сказати, що не існує одного абсолютно надійного методу захисту. Найбільш повну безпеку можна забезпечити тільки при комплексному підході до цього питання. Необхідно постійно стежити за новими рішеннями в цій галузі. Це ми і побачили на прикладі програми «Підприємець 4.2» проаналізувавши її модуль ліцензування.

Список використаної літератури

1. Казарин О.В. Безопасность программного обеспечения компьютерных систем. Монография. – М.: МГУЛ, 2003. – 212 с.
2. Скляров Д.В. Искусство защиты и взлома информации. – СПб.: БХВ-Петербург, 2004. – 288 с.: ил.
3. Терехов А.В., Чернышов В.Н., Селезнев А.В., Рак И.П. Защита компьютерной информации: Учебное пособие. Тамбов: Изд-во Тамб. гос. техн. ун-та, 2003. 80 с
4. Захист програмного забезпечення [Електронний ресурс] // wiki. – 2019. – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Захист_програмного_забезпечення.
5. Фундаментальные основы хакерства. Мастер-класс по анализу исполняемых файлов в IDA Pro [Електронний ресурс] // ХАКЕР.РУ. – 2019. – Режим доступу до ресурсу: https://xakep.ru/2019/08/23/nezumi-hacking-guide-5/.
6. Yashechka. Введение в реверсинг с нуля, с использованием IDA PRO [Електронний ресурс] / Yashechka // ХАБР. – 2019. – Режим доступу до ресурсу: https://habr.com/ru/post/458354/.
7. ИССЛЕДОВАНИЕ: К 2022 ГОДУ УБЫТКИ ПРАВООБЛАДАТЕЛЕЙ ИЗ-ЗА ПИРАТСТВА ДОСТИГНУТ $2,3 ТРЛН [Електронний ресурс] // «Теле-Спутник». – 2017. – Режим доступу до ресурсу: https://telesputnik.ru/materials/video-v-internete/news/issledovanie-k-2022-godu-ubytki-pravoobladateley-iz-za-piratstva-dostignut-2-3-trln/.

Додаток А

Вихідний код утиліти типу КЕЙГЕН для програми «Підприємець 4.2»

```
namespace WindowsFormsApp3
{
    using Microsoft.VisualBasic.CompilerServices;
    using System;
    using System.ComponentModel;
    using System.Drawing;
    using System.Windows.Forms;

    public class Form1 : Form
    {
        private Button button1;
        private Label label1;
        private Label label10;
        private Label label2;
        private Label label3;
        private Label label4;
        private Label label5;
        private Label label6;
        private Label label7;
        private Label label8;
        private Label label9;
        private string MainBoardNum;
        private TextBox textBox1;
        private TextBox textBox2;
        private TextBox textBox3;
        private TextBox textBox4;
        private TextBox textBox5;
        private TextBox textBox6;
        private TextBox textBox7;
        private TextBox textBox8;
        private TextBox textBox9;
        public string TmpRegKey;

        public Form1()
        {
            this.InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
```

```
        private void InitializeComponent()
        {
            this.textBox1 = new System.Windows.Forms.TextBox();
            this.textBox2 = new System.Windows.Forms.TextBox();
            this.textBox3 = new System.Windows.Forms.TextBox();
            this.textBox4 = new System.Windows.Forms.TextBox();
            this.textBox5 = new System.Windows.Forms.TextBox();
            this.label1 = new System.Windows.Forms.Label();
            this.label2 = new System.Windows.Forms.Label();
            this.label3 = new System.Windows.Forms.Label();
            this.label4 = new System.Windows.Forms.Label();
            this.label5 = new System.Windows.Forms.Label();
            this.textBox6 = new System.Windows.Forms.TextBox();
            this.textBox7 = new System.Windows.Forms.TextBox();
            this.textBox8 = new System.Windows.Forms.TextBox();
            this.textBox9 = new System.Windows.Forms.TextBox();
            this.label6 = new System.Windows.Forms.Label();
            this.label7 = new System.Windows.Forms.Label();
            this.label8 = new System.Windows.Forms.Label();
            this.label9 = new System.Windows.Forms.Label();
            this.button1 = new System.Windows.Forms.Button();
            this.label10 = new System.Windows.Forms.Label();
            this.SuspendLayout();
            //
            // textBox1
            //
            this.textBox1.Location = new System.Drawing.Point(106, 54);
            this.textBox1.Margin = new System.Windows.Forms.Padding(6,
5, 6, 5);
            this.textBox1.MaxLength = 9;
            this.textBox1.Name = "textBox1";
            this.textBox1.Size = new System.Drawing.Size(285, 29);
            this.textBox1.TabIndex = 0;
            this.textBox1.TextChanged += new
System.EventHandler(this.textBox1_TextChanged);
            //
            // textBox2
            //
            this.textBox2.Location = new System.Drawing.Point(106, 254);
            this.textBox2.Margin = new System.Windows.Forms.Padding(6,
5, 6, 5);
            this.textBox2.MaxLength = 9;
```

```
                this.textBox2.Multiline = true;
                this.textBox2.Name = "textBox2";
                this.textBox2.ReadOnly = true;
                this.textBox2.Size = new System.Drawing.Size(285, 31);
                this.textBox2.TabIndex = 1;
                this.textBox2.TextChanged += new
System.EventHandler(this.textBox2_TextChanged);
                //
                // textBox3
                //
                this.textBox3.Location = new System.Drawing.Point(106, 298);
                this.textBox3.Margin = new System.Windows.Forms.Padding(6,
5, 6, 5);
                this.textBox3.MaxLength = 9;
                this.textBox3.Multiline = true;
                this.textBox3.Name = "textBox3";
                this.textBox3.ReadOnly = true;
                this.textBox3.Size = new System.Drawing.Size(285, 31);
                this.textBox3.TabIndex = 2;
                this.textBox3.TextChanged += new
System.EventHandler(this.textBox3_TextChanged);
                //
                // textBox4
                //
                this.textBox4.Location = new System.Drawing.Point(106, 210);
                this.textBox4.Margin = new System.Windows.Forms.Padding(6,
5, 6, 5);
                this.textBox4.MaxLength = 9;
                this.textBox4.Multiline = true;
                this.textBox4.Name = "textBox4";
                this.textBox4.ReadOnly = true;
                this.textBox4.Size = new System.Drawing.Size(285, 31);
                this.textBox4.TabIndex = 3;
                this.textBox4.TextChanged += new
System.EventHandler(this.textBox4_TextChanged);
                //
                // textBox5
                //
                this.textBox5.Location = new System.Drawing.Point(106, 166);
                this.textBox5.Margin = new System.Windows.Forms.Padding(6,
5, 6, 5);
                this.textBox5.MaxLength = 9;
```

```
                this.textBox5.Multiline = true;
                this.textBox5.Name = "textBox5";
                this.textBox5.ReadOnly = true;
                this.textBox5.Size = new System.Drawing.Size(285, 31);
                this.textBox5.TabIndex = 4;
                this.textBox5.TextChanged += new
System.EventHandler(this.textBox5_TextChanged);
                //
                // label1
                //
                this.label1.AutoSize = true;
                this.label1.Location = new System.Drawing.Point(19, 57);
                this.label1.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label1.Name = "label1";
                this.label1.Size = new System.Drawing.Size(76, 22);
                this.label1.TabIndex = 5;
                this.label1.Text = "Number";
                //
                // label2
                //
                this.label2.AutoSize = true;
                this.label2.Location = new System.Drawing.Point(13, 171);
                this.label2.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label2.Name = "label2";
                this.label2.Size = new System.Drawing.Size(42, 22);
                this.label2.TabIndex = 6;
                this.label2.Text = "Full";
                //
                // label3
                //
                this.label3.AutoSize = true;
                this.label3.Font = new System.Drawing.Font("Times New
Roman", 12F, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, ((byte)(204)));
                this.label3.Location = new System.Drawing.Point(13, 222);
                this.label3.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label3.Name = "label3";
                this.label3.Size = new System.Drawing.Size(36, 19);
                this.label3.TabIndex = 7;
```

```
                this.label3.Text = "Doc";
                //
                // label4
                //
                this.label4.AutoSize = true;
                this.label4.Location = new System.Drawing.Point(13, 259);
                this.label4.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label4.Name = "label4";
                this.label4.Size = new System.Drawing.Size(49, 22);
                this.label4.TabIndex = 8;
                this.label4.Text = "First";
                //
                // label5
                //
                this.label5.AutoSize = true;
                this.label5.Location = new System.Drawing.Point(13, 303);
                this.label5.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label5.Name = "label5";
                this.label5.Size = new System.Drawing.Size(64, 22);
                this.label5.TabIndex = 9;
                this.label5.Text = "Server";
                //
                // textBox6
                //
                this.textBox6.Location = new System.Drawing.Point(106, 342);
                this.textBox6.Margin = new System.Windows.Forms.Padding(6,
5, 6, 5);
                this.textBox6.MaxLength = 9;
                this.textBox6.Multiline = true;
                this.textBox6.Name = "textBox6";
                this.textBox6.ReadOnly = true;
                this.textBox6.Size = new System.Drawing.Size(285, 31);
                this.textBox6.TabIndex = 10;
                this.textBox6.TextChanged += new
System.EventHandler(this.textBox6_TextChanged);
                //
                // textBox7
                //
                this.textBox7.Location = new System.Drawing.Point(106, 474);
```

```
                this.textBox7.Margin = new System.Windows.Forms.Padding(6,
5, 6, 5);
                this.textBox7.MaxLength = 9;
                this.textBox7.Multiline = true;
                this.textBox7.Name = "textBox7";
                this.textBox7.ReadOnly = true;
                this.textBox7.Size = new System.Drawing.Size(285, 31);
                this.textBox7.TabIndex = 11;
                this.textBox7.TextChanged += new
System.EventHandler(this.textBox7_TextChanged);
                //
                // textBox8
                //
                this.textBox8.Location = new System.Drawing.Point(106, 430);
                this.textBox8.Margin = new System.Windows.Forms.Padding(6,
5, 6, 5);
                this.textBox8.MaxLength = 9;
                this.textBox8.Multiline = true;
                this.textBox8.Name = "textBox8";
                this.textBox8.ReadOnly = true;
                this.textBox8.Size = new System.Drawing.Size(285, 31);
                this.textBox8.TabIndex = 12;
                this.textBox8.TextChanged += new
System.EventHandler(this.textBox8_TextChanged);
                //
                // textBox9
                //
                this.textBox9.Location = new System.Drawing.Point(106, 386);
                this.textBox9.Margin = new System.Windows.Forms.Padding(6,
5, 6, 5);
                this.textBox9.MaxLength = 9;
                this.textBox9.Multiline = true;
                this.textBox9.Name = "textBox9";
                this.textBox9.ReadOnly = true;
                this.textBox9.Size = new System.Drawing.Size(285, 31);
                this.textBox9.TabIndex = 13;
                this.textBox9.TextChanged += new
System.EventHandler(this.textBox9_TextChanged);
                //
                // label6
                //
                this.label6.AutoSize = true;
```

```
                this.label6.Location = new System.Drawing.Point(7, 347);
                this.label6.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label6.Name = "label6";
                this.label6.Size = new System.Drawing.Size(80, 22);
                this.label6.TabIndex = 14;
                this.label6.Text = "FullSQL";
                //
                // label7
                //
                this.label7.AutoSize = true;
                this.label7.Location = new System.Drawing.Point(7, 391);
                this.label7.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label7.Name = "label7";
                this.label7.Size = new System.Drawing.Size(81, 22);
                this.label7.TabIndex = 15;
                this.label7.Text = "DocSQL";
                //
                // label8
                //
                this.label8.AutoSize = true;
                this.label8.Location = new System.Drawing.Point(4, 435);
                this.label8.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label8.Name = "label8";
                this.label8.Size = new System.Drawing.Size(87, 22);
                this.label8.TabIndex = 16;
                this.label8.Text = "FirstSQL";
                //
                // label9
                //
                this.label9.AutoSize = true;
                this.label9.Location = new System.Drawing.Point(-4, 479);
                this.label9.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label9.Name = "label9";
                this.label9.Size = new System.Drawing.Size(102, 22);
                this.label9.TabIndex = 17;
                this.label9.Text = "ServerSQL";
                //
                // button1
```

```
                //
                this.button1.Location = new System.Drawing.Point(166, 105);
                this.button1.Margin = new System.Windows.Forms.Padding(6, 5,
6, 5);
                this.button1.Name = "button1";
                this.button1.Size = new System.Drawing.Size(171, 39);
                this.button1.TabIndex = 18;
                this.button1.Text = "Generation";
                this.button1.UseVisualStyleBackColor = true;
                this.button1.Click += new
System.EventHandler(this.button1_Click);
                //
                // label10
                //
                this.label10.AutoSize = true;
                this.label10.Location = new System.Drawing.Point(98, 21);
                this.label10.Margin = new System.Windows.Forms.Padding(6, 0,
6, 0);
                this.label10.Name = "label10";
                this.label10.Size = new System.Drawing.Size(228, 22);
                this.label10.TabIndex = 19;
                this.label10.Text = "Get number to registration";
                //
                // Form1
                //
                this.AutoScaleDimensions = new System.Drawing.SizeF(11F,
22F);
                this.AutoScaleMode =
System.Windows.Forms.AutoScaleMode.Font;
                this.ClientSize = new System.Drawing.Size(409, 523);
                this.Controls.Add(this.label10);
                this.Controls.Add(this.button1);
                this.Controls.Add(this.label9);
                this.Controls.Add(this.label8);
                this.Controls.Add(this.label7);
                this.Controls.Add(this.label6);
                this.Controls.Add(this.textBox9);
                this.Controls.Add(this.textBox8);
                this.Controls.Add(this.textBox7);
                this.Controls.Add(this.textBox6);
                this.Controls.Add(this.label5);
                this.Controls.Add(this.label4);
```

```
                this.Controls.Add(this.label3);
                this.Controls.Add(this.label2);
                this.Controls.Add(this.label1);
                this.Controls.Add(this.textBox5);
                this.Controls.Add(this.textBox4);
                this.Controls.Add(this.textBox3);
                this.Controls.Add(this.textBox2);
                this.Controls.Add(this.textBox1);
                this.Cursor = System.Windows.Forms.Cursors.Hand;
                this.Font = new System.Drawing.Font("Times New Roman",
14.25F, System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point,
((byte)(0)));
                this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Fixed3D;
                this.Margin = new System.Windows.Forms.Padding(6, 5, 6, 5);
                this.MaximizeBox = false;
                this.MinimizeBox = false;
                this.Name = "Form1";
                this.Text = "Marchuk_bak_rob";
                this.TopMost = true;
                this.ResumeLayout(false);
                this.PerformLayout();

            }

            private void maskedTextBox1_MaskInputRejected(object sender,
MaskInputRejectedEventArgs e)
            {
            }

            private void textBox1_TextChanged(object sender, EventArgs e)
            {
            }

            private void textBox2_TextChanged(object sender, EventArgs e)
            {
            }

            private void textBox3_TextChanged(object sender, EventArgs e)
            {
            }
```

```
        private void textBox4_TextChanged(object sender, EventArgs e)
        {
        }

        private void textBox5_TextChanged(object sender, EventArgs e)
        {
        }

        private void textBox6_TextChanged(object sender, EventArgs e)
        {
        }

        private void textBox7_TextChanged(object sender, EventArgs e)
        {
        }

        private void textBox8_TextChanged(object sender, EventArgs e)
        {
        }

        private void textBox9_TextChanged(object sender, EventArgs e)
        {
        }
    }
}
```