**План действий для реализаии курсового проекта:**

**1) Метод защиты от копирования: объекты привязки**

**Пример использования: Adobe Acrobat Pro DC.**

**Теория:**

Привязка программы к компьютеру базируется на процедуре активации, которую необходимо произвести при его первом открытии:

* При первом открытии программы система просит конечного пользователя ввести серийный номер – уникальную комбинацию цифр и латинских букв, дающую право открыть документ на данном компьютере.
* Конечный пользователь вводит серийный номер, полученный при покупке защищённой программы.
* Система защиты собирает информацию о программных и аппаратных параметрах компьютера и вместе с серийным номером передаёт её серверу управления лицензиями (необходимо подключение к интернету).
* Сервер управления лицензиями проверяет, что данный серийный номер корректен и ещё не использовался для запуска программы на другом компьютере, после чего возвращает **активационный ключ**, содержащий информацию о лицензии и об аппаратных параметрах компьютера.
* Система защиты проверяет соответствие активационного ключа аппаратным и программным параметрам компьютера и разрешает открытие документа.
* При последующих запусках система защиты проверяет соответствие активационного ключа аппаратным и программным параметрам компьютера и, при успешном прохождении проверки, разрешает запуск программы.

**Активационный ключ —** генерируется при помощи какого-либо метода хеширования (md5, sha1) из данных о физической комплектации компьютера и/или данных от установленных программ (серийный номер ОС).

**Задача:**

* Реализовать защищаемую программу. Можно взять программу курсового проекта из предыдущего семестра.
* Реализовать сервер генерации активационного ключа. Адрес сервера: ip 127.0.0.1 port 55555.
* Реализовать для программы модуль который собирает информацию о системе. При первом запуске программы, информация вместе с серийным номером отправляется на сервер. В ответ программа получает активационный ключ, который сохраняется в файл.
* Реализовать модуль для программы, который при каждом запуске сравнивает сохраненный активационный ключ с «на лету» сгенерированным ключем.

Ссылки:

* StartForce. Методы защиты от копирования: объекты привязки. URL:https://www.sfcontent.com/ru/education-pdf-protection?id=0
* Привязка программы к железу + онлайн верификация. URL:https://habrahabr.ru/sandbox/43435/

**2)Обход защиты от копирования: объекты привязки.**

Исхоные данные:

* Опонент получает бинарные файлы программы, программы сервера и серийный номер программы.
* Опонент произваодит регистрацию своей программы по серийному номеру.
* **Опоненту запрещается исследовать бинарный файл сервера, т.к. предполагается, что он запущен удаленно.**

Варианты обхода защиты:

* Создание бинарного патча для программы, отключаеющего проверку активационного ключа.
* Создание программы генератора активационных ключей по серийному номеру.

**3)Защита при помощи электронных ключей**

**Теория:**

Электронный ключ — это устройство которое подсоединяется к одному из портов компьютера. Обычно это специально разработанное прогамно-аппаратное устройство, которое содержит в себе информацию для работы программы.

**Задача:**

* Реализовать защищаемую программу. Можно взять программу курсового проекта из предыдущего семестра.
* На основе USB Flash создать электронный ключ.
* Встроить в защищаемую программу методы работы с электроным ключем.

**Электронный ключ:**

В программе могут быть использованы VID и PID флеш карты, так как эти параметры не изменяются.

На флеш какрту должны быть записаны данные которые проверяются в исполняемой программе. Так как файлы настоящего электронного ключа нельзя просто прочитать, то предполагаем что нашу флеш карту так же нельзя просто открыть и прочитать.

Ссылки:

* Guardant. Электронный ключ — средство предохранения софта

URL: <https://www.guardant.ru/press-center/publication/2001-08-01.html>

* <https://xakep.ru/2011/11/30/57901/#toc06>
* <http://www.cyberforum.ru/cpp-builder/thread553816.html>

**4) Обход использования электронного ключа**

Исходные данные:

* Опонент получает бинарный код программы и электронный ключ.

Варианты:

* Создание бинарного патча для программы. Отключение проверки ключа.
* Создание дампа памяти электронного ключа с использованием специальных утилит.

Ссылки:

* <http://rus.cadshop.ru/%D0%BF%D1%83%D0%B1%D0%BB%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D0%B8/%D1%81%D0%BD%D1%8F%D1%82%D0%B8%D0%B5-%D0%BB%D0%BE%D0%B3%D0%B0-%D0%BA%D0%BB%D1%8E%D1%87%D0%B0-%D0%B7%D0%B0%D1%89%D0%B8%D1%82%D1%8B-c-%D0%BF%D0%BE%D0%BC%D0%BE%D1%89%D1%8C%D1%8E-usbtrace/>

**5)Защита на основе голоса пользователя**

**Теория:**

Программа при запуске записывает голос пользователя, затем сравнивает его с эталонной записью в БД или на удалённом сервере. В случае совпадения на определённый % программа должна запуститься.

**Задача:**

* Реализовать защищаемую программу.
* Создать БД голосов пользователей.
* Встроить в программу метод распознования голоса.

Ссылки: <http://www.programmersforum.ru/showthread.php?t=225751>

**6) Обход защиты на основе голоса пользователя**

Исхоные данные:

* Опонент получает бинарные файлы программы, программу сервера или БД голосов.
* **Опоненту запрещается исследовать бинарный файл сервера или БД, т.к. предполагается, что они запущены удаленно.**

Варианты обхода защиты:

* Перебор различных звуков, до нахождения подходящего.

Создание виртуального ответа от сервера с разрешением запуска программы.