**Задание TrueCrypt.**

1) Графическая оболочка (первоначально пишется в консольном варианте). Должна выполнять:

a) Считывание файлов с диска («загрузка»), с которыми будет работать наша программа (возможно просто указывать ссылку на то что надо шифровать)

b) «Кнопка» дающая команду на генерацию мастер-ключа (токена на флешку) и шифрующая предоставленные файлы

c) «Кнопка» дающая команду на расшифровку(открытие) файлов.

d) «Кнопка» дающая команду на удаление зашифрованной информации.

2) Демон (программа работающая в фоновом режиме) – должна запускаться автоматически. Должна выполнять:

a) При получение команды на генерацию мастер-ключа и шифрование, должна делать:  
 1. Вычислять данные о вставленной флешке (ее ID, Имя и тд.)

2. Вычислять данные о машине на которой запущена программа(опять же какие-то уникальные характеристики.

(1 и 2 пункт необходимы, чтобы токен подходил только для этой машины, а также, чтобы для этой машины подходил только один токен. Они могут быть вынесены из Демона в отдельную программу, которая будет запускаться при нажатие на «кнопку», и передавать выше указанною инфу через сокет)

3. Генерировать, используя данные о первых двух пунктах, мастер-ключ, используя библиотеку криптографии (скорее всего с помощью хэш-функции (не уверен)). После этого сохранять мастер-ключ на флешку.

4. Выделить некий объем памяти под зашифрованные данные.

5. Используя ссылку, зашифровать требуемые файлы, используя крипто-библиотеку и блочное шифрование. Для каждого блока используя мастер-ключ, создавать уникальный ключ, необходимый для шифрования. Помещать полученные блоки в выделенную память.

b) При получение команды на открытие: Должна проверять, является ли файл шифрованным, если нет, то возможно просто его выводить, если да, то при наличие правильного токена производить расшифровку.

1.Проверка правильности ключа: демон должен брать информацию о железе (желательно сохраненную при генерации мастер-ключа, чтобы не находить ее повторно), вычислять информацию о вставленной флешке (и есть ли она вообще), генерировать с помощью той же функции подходящий мастер-ключ для данного железа и данной флешки, и при совпадении токена с только что сгенерированным мастер-ключом, выдавать сообщение о корректности токена.

2. Расшифровка производится обратным образом к зашифровке. (Однако зашифрованные данные должны оставаться в выделенной памяти, но расшифрованные данные должны выводиться в графическую оболочку(к примеру с помощью сокета), и удаляться из демона.

c) При команде на удаление, опять же при наличие корректного токена, удалять все данные из выделенной области и саму область.

Также Демон должен постоянно проверять вставлена ли флешка в машину, при ее изъятии должен возникать сигнал (или его аналог), прерывающий работу.

3) Криптографические библиотеки (существует иностранные пакет Openssl, с готовыми функциями шифрования, первоначально пользоваться им), а потом переделать под российский аналоги (не до конца понял, но вроде они различаются методами, но суть одна).

Комментарии:

1)взаимодействие между демоном и графической оболочкой (по моему мнению может быть организовано с помощью сокетов и сигналов, но не уверен)

2)каждую функцию надо тестить отдельно (юнитест вроде)

3)в основном все занимаются Демоном, сказали, что на библиотеку человека два.

4)необходимое к концу выходных предоставить описание нашей программы, если я ни где сильно не ошибся , то достаточно будет исправить это.

Состав группы:

Storage

USB

Демон

Данные

Графическая оболочка

1)Абрамова Ирина

2)Зенкин Николай

3)Панченко Иван

4)Разина Наталья

5)Труфанова Юля

6)Щербаков Виктор