



Реализация механизма лицензирования программного обеспечения с использованием системы безопасного удалённого обновления (RU-“Remote Update”).

Оглавление

Вступление.	2
Создание “мастер-ключа”.	3
Создание удалённо обновляемого “ключа-клиента”.	5
Создание “файла лицензии”.	7
Создание пакета обновления.	8
Удалённое обновление “ключа-клиента”.	11
Проверка результата.	13
Заключение.	14

Вступление.

Пример реализации механизма защиты для ограничения программ по времени работы или лизинговых систем продаж на основе системы безопасного удалённого обновления RU - “Remote Update”.

Для чего это нужно:

Обычно подобные механизмы защиты программ используются для обеспечения демонстрации Вашего продукта в полнофункциональном режиме работы в течение жестко ограниченного времени. Используя наши решения, Вы можете быть уверены, что исключена возможность неконтролируемого продления времени работы защищенного в ключе кода без Вашего ведома.

Примерная схема работы:

- Вы инициализируете ключ и предоставляете его вместе с защищенным программным обеспечением пользователю.
- Ваш пользователь делает первый взнос.
- Вы создаёте пакет обновления для ключа и передаёте его вашему пользователю.
- Ваш пользователь, используя систему RU, загружает пакет обновления в ключ.
- Ключ вашего пользователя продлевает время работоспособности на указанный в пакете обновления период времени.
- После истечения заданного времени и в случае неоплаты клиентом очередного взноса, ключ перестаёт выполнять свои функции, а вместе с ним теряется функциональность программы.

Что нужно для реализации:



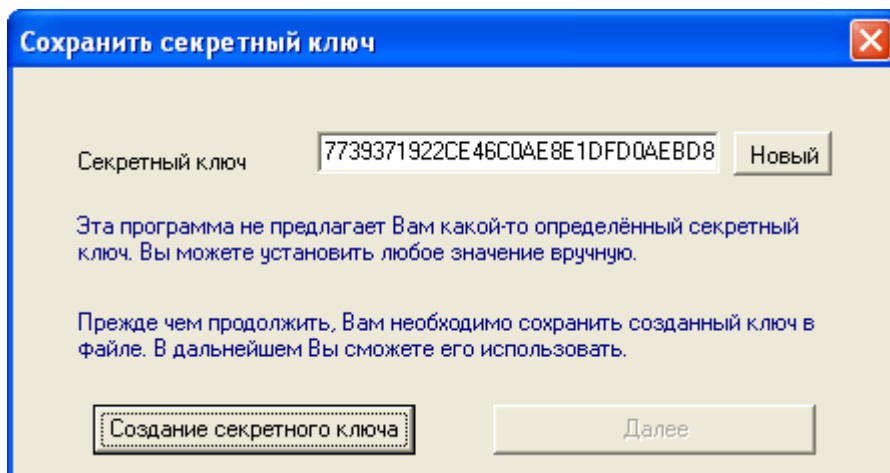
Один мастер-ключ, которым может быть любая модель серии “**Senselock EL**” и “**Senselock EL-RTC**” в качестве “ключа-клиента”.

“**Senselock EL-RTC**” наиболее подходит для реализации системы продаж в кредит/лизинг. На основе функций модели RTC вы можете получать наиболее достоверные данные о текущем времени и дате.

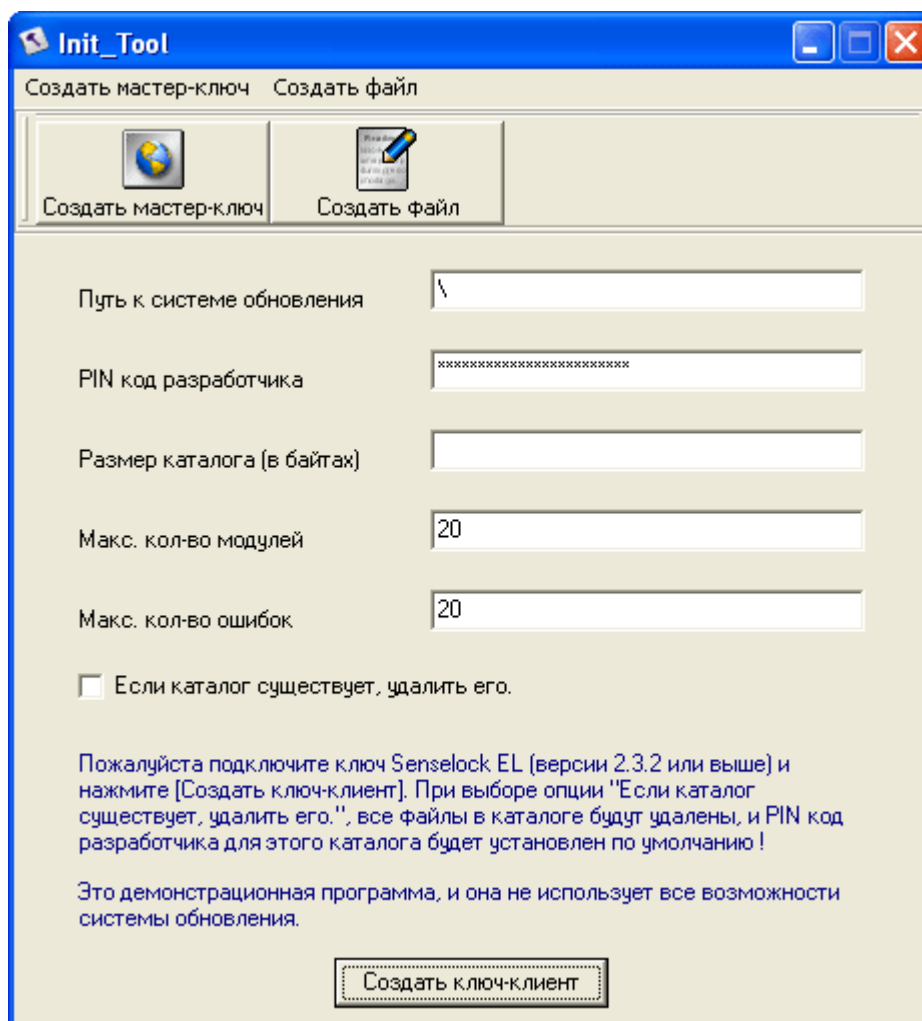
Также необходимо установить в ключ, предназначенный для пользователя, клиентскую часть системы RU.

Создание “мастер-ключа”.

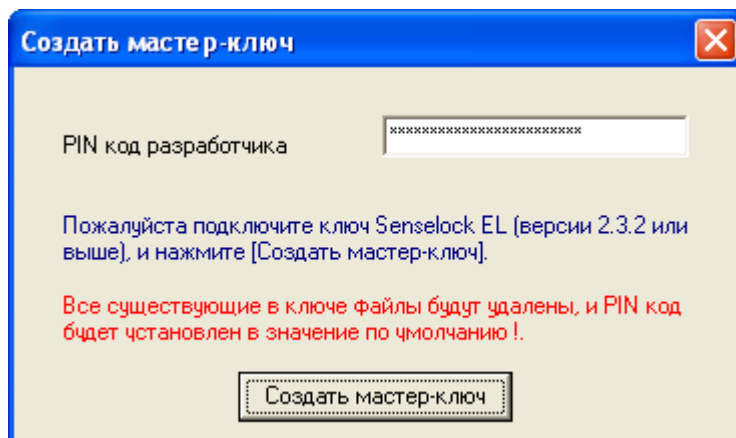
Запустите приложение “**InitTool.exe**”. При первом запуске вам необходимо сгенерировать и сохранить секретный ключ, который будет использоваться для шифрования пакетов обновления “мастер-ключом” и расшифровыванием пакета “ключом-клиентом”. Кнопкой “**Новый**” Вы можете сгенерировать ключ с помощью случайных чисел либо установить его вручную. После этого нажмите на “Создание секретного ключа” и Ваш ключ будет сохранён в файл “**InitTool.ini**”.



Установите чистый ключ **Senselock EL** в USB порт. Окно утилиты “**Init_Tool.exe**” должно выглядеть следующим образом:

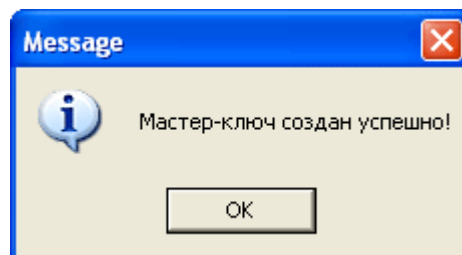


Оставьте все опции без изменения и нажмите “Создать мастер-ключ”. Следующее диалоговое окно выглядит так:



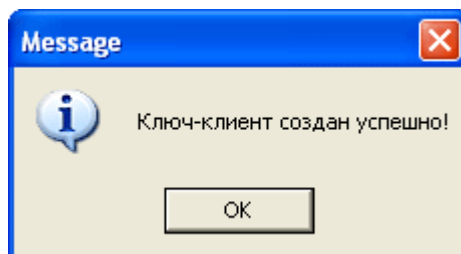
Если вы установили в USB порт чистый новый ключ, то PIN код разработчика должен быть установлен по умолчанию в значение “123456781234567812345678”, в противном случае введите Ваш PIN код. Далее нажмите на кнопку “Создать мастер-ключ”.

Если всё прошло успешно, Вы получите соответствующее сообщение:

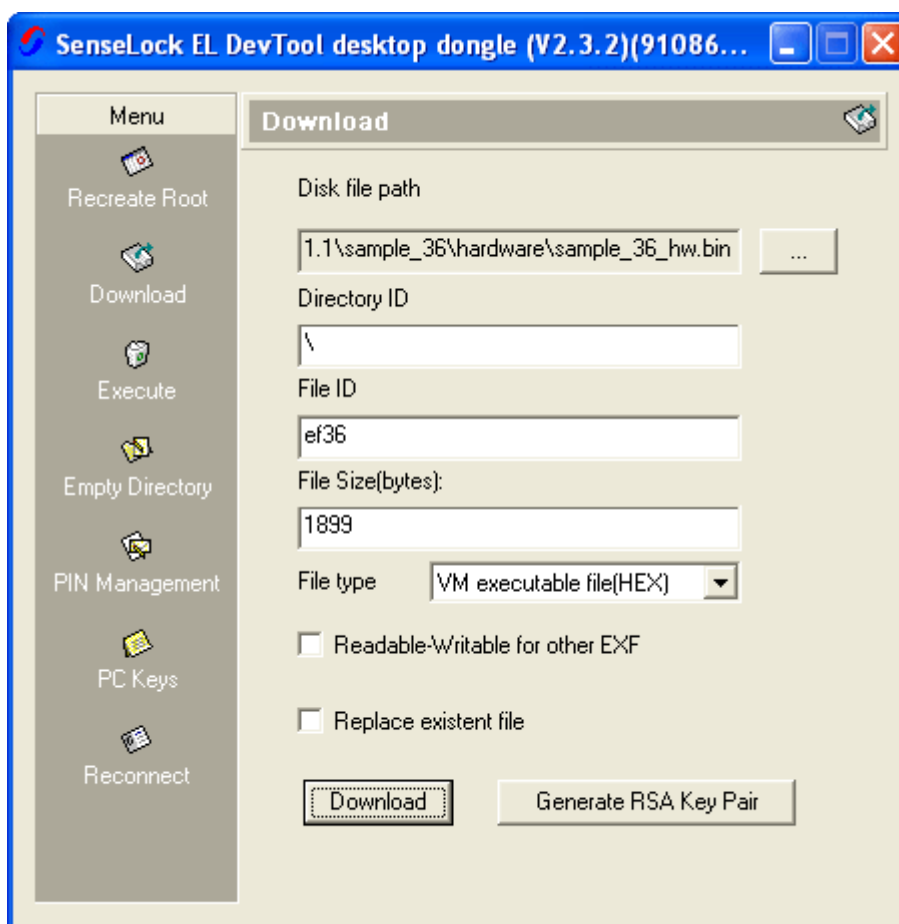


Создание удалённо обновляемого “ключа-клиента”.

Установите в USB порт чистый ключ **Senselock EL-RTC**. В главном окне демонстрационной утилиты “**Init_Tool.exe**”, нажмите “Создать ключ-клиент”. Подождите несколько секунд, и вы получите сообщение об успешном создании “ключа-клиента”.



С помощью утилиты DevTool загрузите скомпилированный исполняемый файл “**sample_36_hw.hex**” в “ключ-клиент”. Не забудьте установить этому файлу ID, равное EF36. Пример Вы можете посмотреть ниже:



Исходный текст программы для ключа Вы можете найти в том же каталоге, где и скомпилированный пример. Программа сделана так, что при запуске функции **foo()** происходит проверка лицензии и времени. Если время во внутреннем таймере RTC превышает установленное в лицензии, значит функция **foo()** выполняться не будет, и она вернёт код ошибки. Время, до которого программа будет выполняться и флаг режима работы программы (ограничена по времени/полная) хранятся в отдельном файле данных с ID: **0036**.

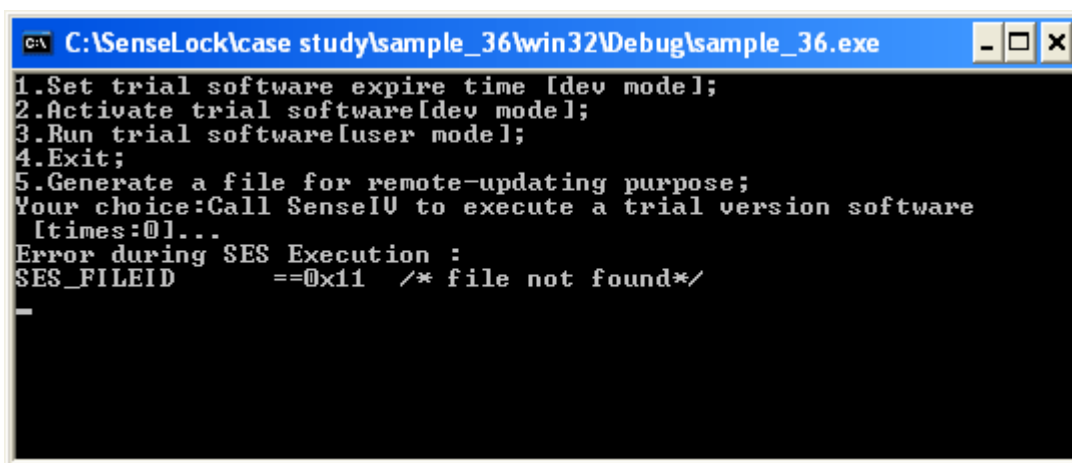
Этот файл мы будем называть “файл лицензии”. Ниже представлена его структура:

Значение	Описание
TrialFlag(1 byte)	флаг режима работы программы
Expiring time in time_t format (4 bytes)	Дата и время, до которой будет работать программа

Создание “файла лицензии”.

Для того, чтобы программа внутри ключа выполнялась нам необходимо создать файл лицензий и поместить его в ключ. То есть произвести активацию.

Без файла лицензии программа внутри ключа исполняться не будет. Вы можете проверить это запустив пример “**sample_36.exe**” и выбрав пункт “3”. Этот пункт запускает на выполнение программу внутри ключа и выводит на экран результат её работы.



```
C:\SenseLock\case study\sample_36\win32\Debug\sample_36.exe
1.Set trial software expire time [dev model];
2.Activate trial software[dev model];
3.Run trial software[user model];
4.Exit;
5.Generate a file for remote-updating purpose;
Your choice:Call SenseIU to execute a trial version software
[Times:0]...
Error during SES Execution :
SES_FILEID ==0x11 /* file not found*/
```

Запустите вновь “**sample_36.exe**”, и выберите пункт “5” чтобы создать новый файл лицензии, который разрешит программе находящейся внутри ключа работать “5 минут” от текущего времени ключа (это называется – “установить время окончания действия лицензии”). “5 минут” устанавливаются следующим образом (смотрите исходный текст программы “**sample_36.exe**”):

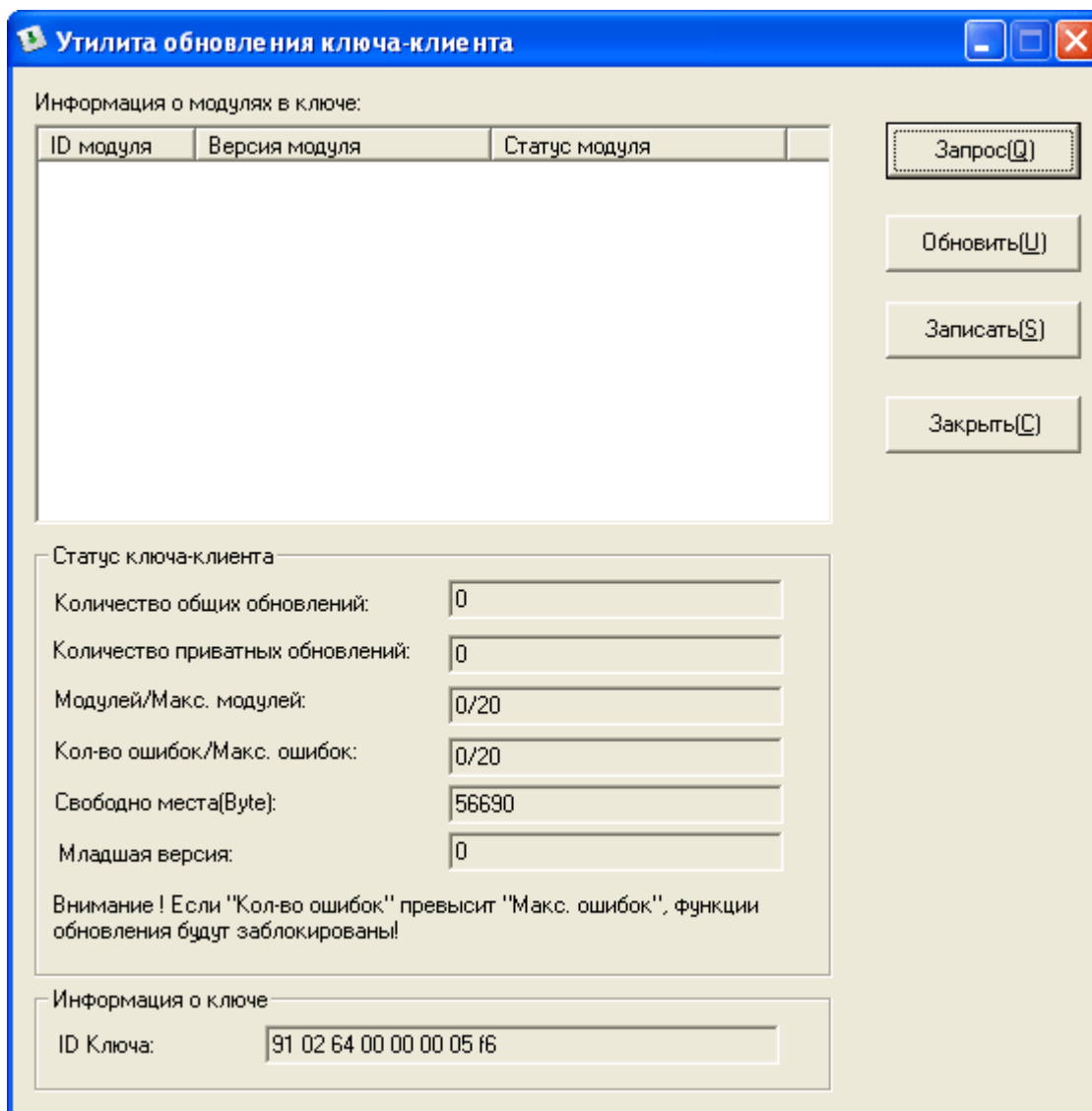
```
.....
iLastingMinutes = 5; // минут до окончания действия лицензии
tmp_time_t += 60*iLastingMinutes;
.....
```

Полученный файл лицензии находится в том же каталоге, где и приложение “**sample_36.exe**” и называется “**RU_EXP_TIME.dat**”. Теперь нам необходимо создать пакет обновления для “ключа-клиента” и загрузить с его помощью файл лицензии в ключ.

Создание пакета обновления.

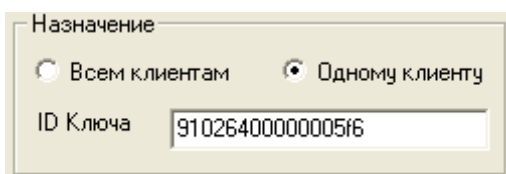
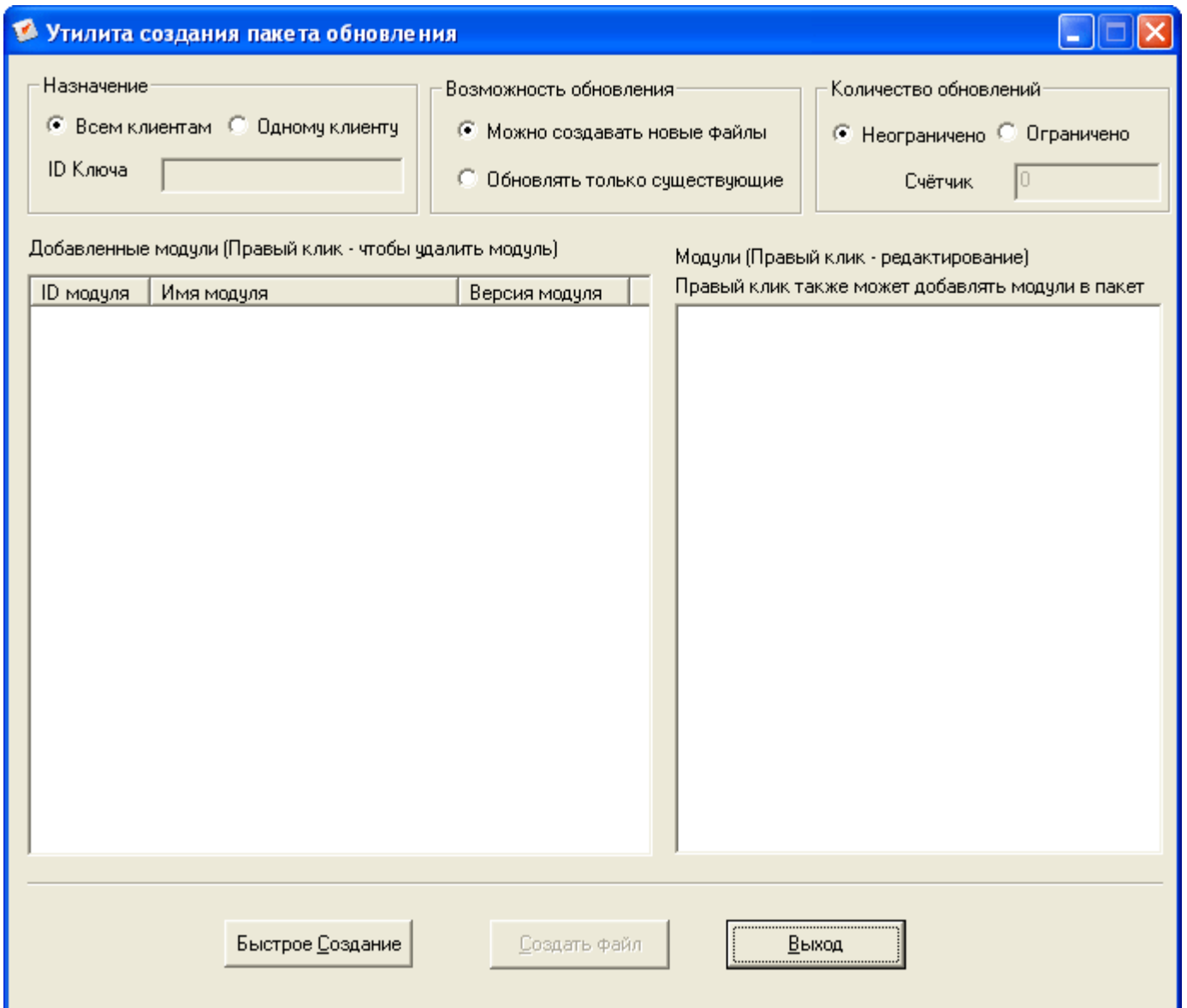
Теперь необходимо определиться с тем, будем ли мы активировать определённый “ключ-клиент” по его ID или же мы произведём активацию всей партии “ключей-клиентов”.

В первом случае нам необходимо получить ID определённого ключа. Для этого необходимо запустить утилиту “ClientTool.exe” и выполнить запрос информации о ключе (клавиша “Запрос”).



Таким образом, в поле “ID Ключа” будет показан уникальный серийный номер “ключа-клиента”. На его основе мы можем создать уникальный пакет обновления, который возможно применить только на этом “ключе-клиенте”.

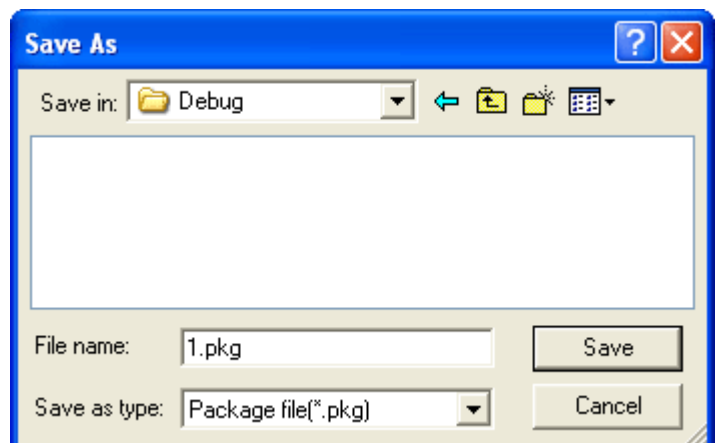
Перейдём к процедуре создания пакета обновления. Запустите утилиту создания пакета обновления (“IssuerTool.exe”).

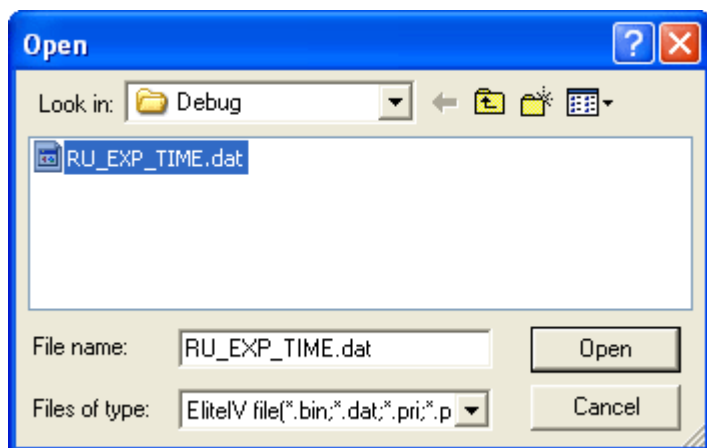


В левом верхнем углу Вам предлагается выбрать назначение обновления. В случае назначения только конкретному ключу-клиенту выберите опцию “Одному клиенту” и введите ID ключа клиента, полученного с помощью утилиты “ClientTool.exe”.

После выбора необходимых Вам опций нажмите кнопку “Быстрое создание”.

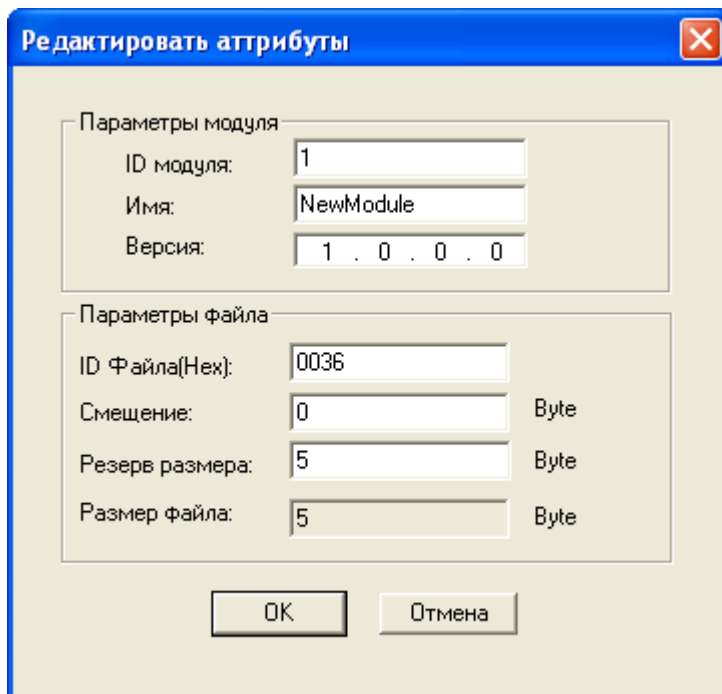
Выберите каталог, в котором будет сохранён пакет обновлений и укажите ему название. В нашем примере мы выбрали каталог, в котором находится пример “sample_36.exe” и имя пакета обновления - “1.pkg”.





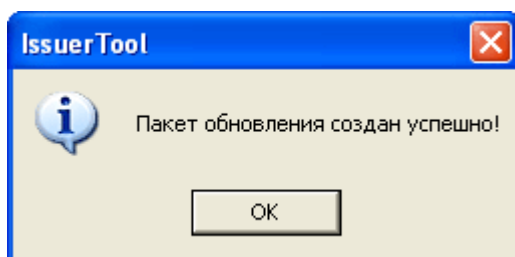
Следующим шагом нам необходимо указать файлы, которые мы должны поместить в пакет обновления. В нашем случае нужно поместить файл “RU_EXP_TIME.dat”, который мы получили на примере описанном в параграфе “Создание файла лицензии”.

Измените поле “ID Файла(Hex)” на значение “0036”. 0036 – ID внутреннего файла лицензии, который проверяет программа внутри ключа. И в соответствии с которым программа выбирает режим своей работы с “ограниченного по времени” до “полнофункционального”. Остальные опции в этом диалоге оставьте без изменений.



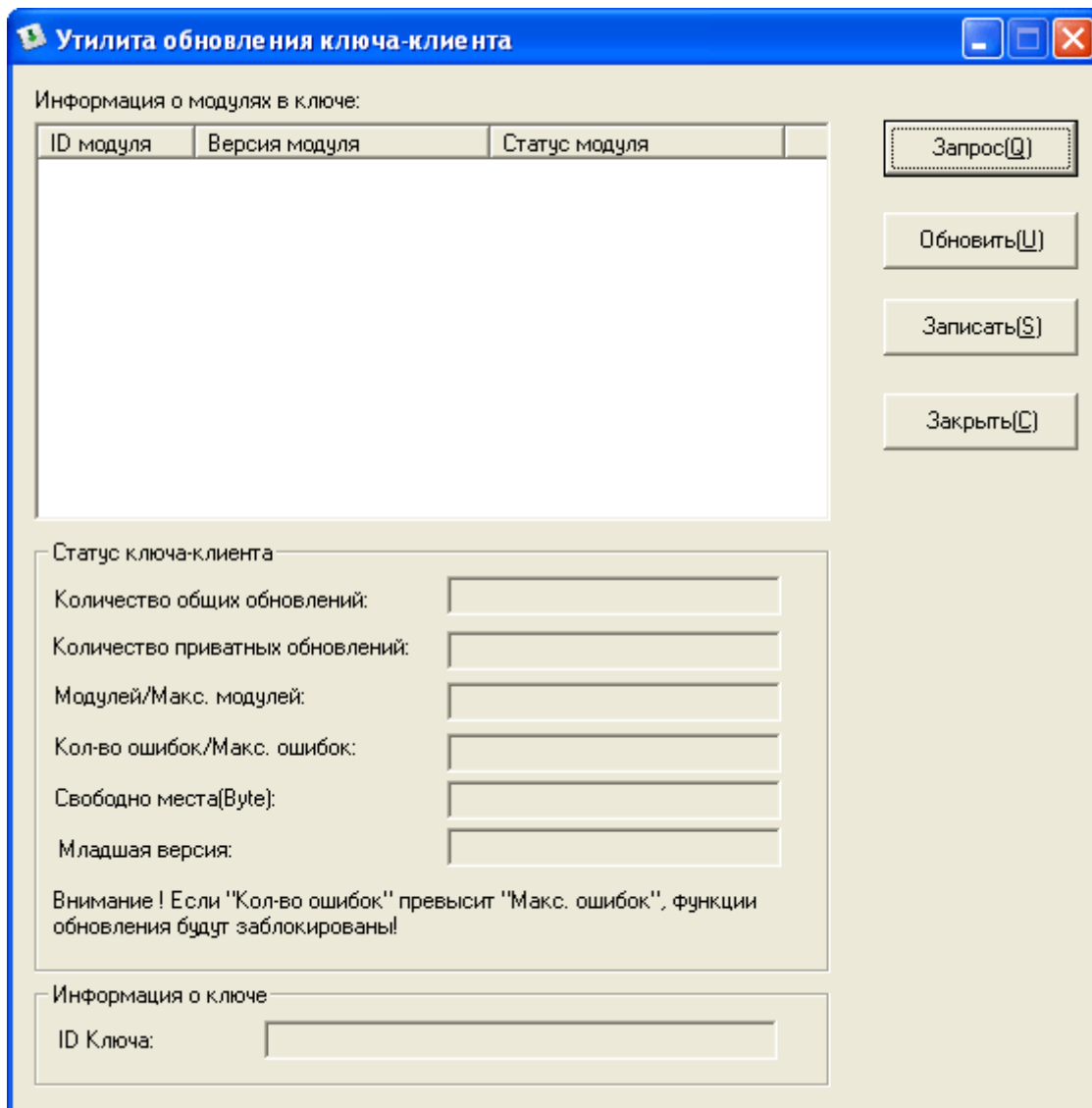
Теперь установить в USB порт “мастер-ключ”. И нажмите кнопку “OK” чтобы создать пакет обновления.

Если всё сделано верно, вы получите соответствующее сообщение.

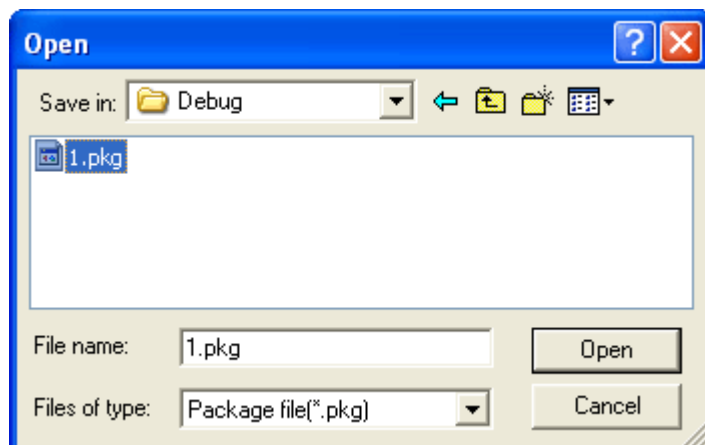


Удалённое обновление “ключа-клиента”.

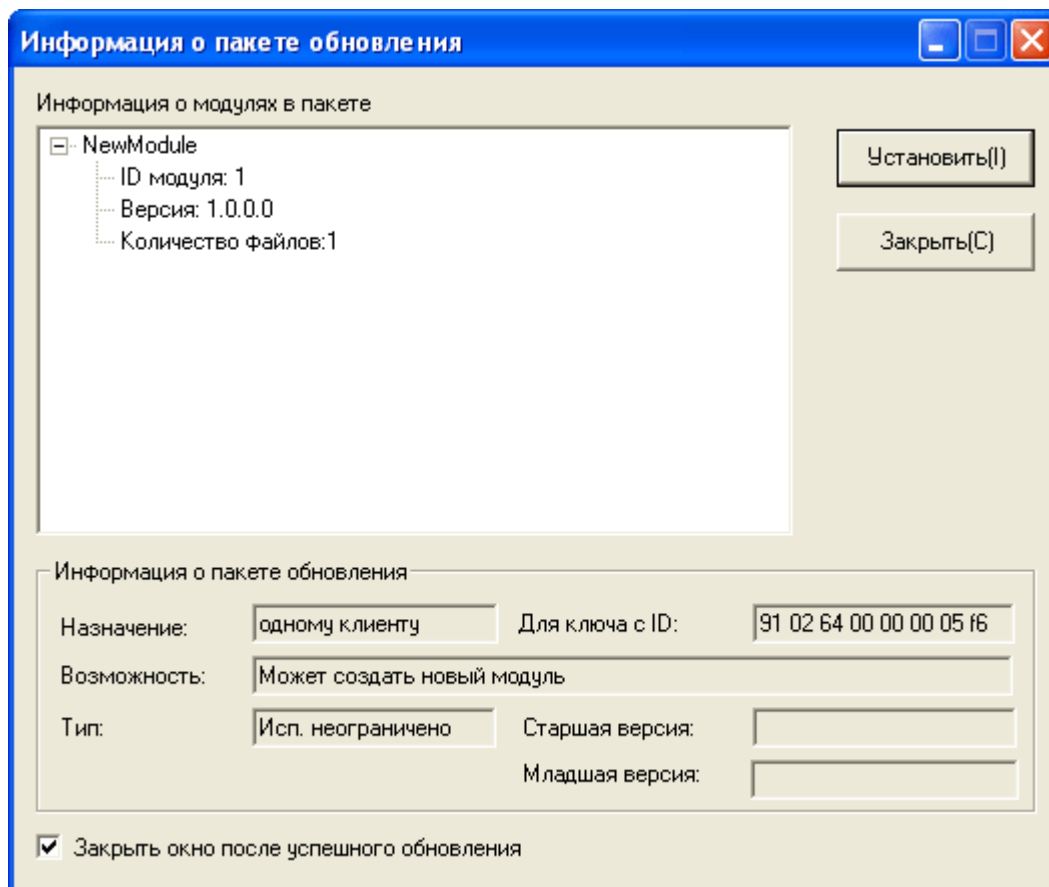
Созданный пакет обновления необходимо установить в ключ клиент. Запустите приложение “ClientTool.exe”, установите в порт USB ключ-клиент и выберите опцию “Обновить”.



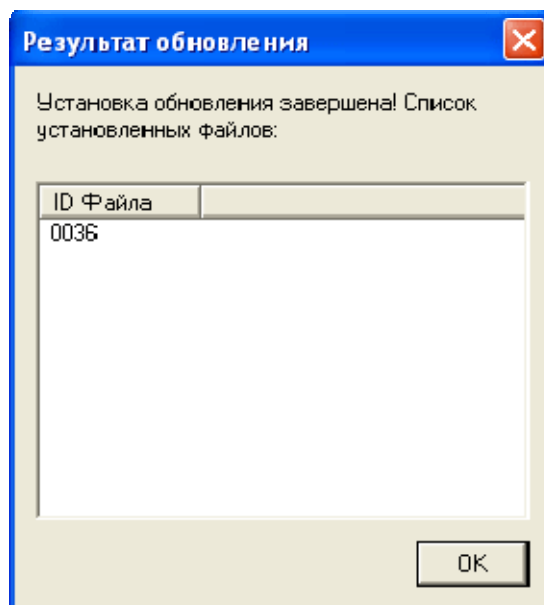
Откройте файл пакета обновления.



Вам будет представлена информация о пакете обновления. В случае назначения обновления определённому “ключу-клиенту” в поле “Для ключа с ID” будет указано ID, которому оно предназначено. Нажмите кнопку “Установить”.

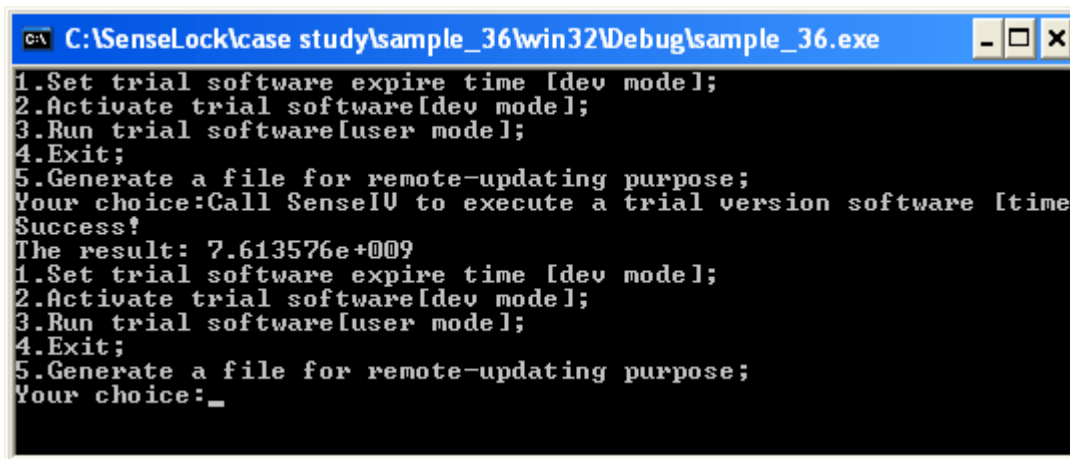


Если всё прошло без ошибок, Вы получите следующее диалоговое окно, которое показывает список успешно установленных в клиент-ключ файлов.



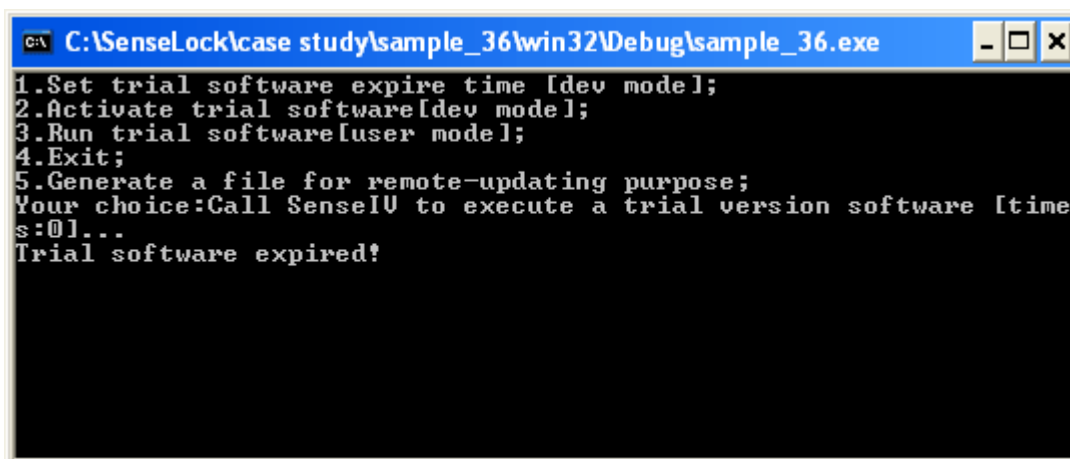
Проверка результата.

Запустите вновь пример “sample_36.exe” и выберите пункт “3”. Если файл лицензии установлен в ключ корректно и локальное время ключа (RTC) меньше времени указанного в файле лицензии, то программа внутри ключа будет выполняться корректно в течение пяти минут.



```
C:\SenseLock\case study\sample_36\win32\Debug\sample_36.exe
1.Set trial software expire time [dev model];
2.Activate trial software[dev model];
3.Run trial software[user model];
4.Exit;
5.Generate a file for remote-updating purpose;
Your choice:Call SenseIU to execute a trial version software [time
Success!
The result: 7.613576e+009
1.Set trial software expire time [dev model];
2.Activate trial software[dev model];
3.Run trial software[user model];
4.Exit;
5.Generate a file for remote-updating purpose;
Your choice:_
```

По истечении заданного в лицензии времени программа в ключе выполняться не будет и Вы получите сообщение об ошибке “Trial software expired”.

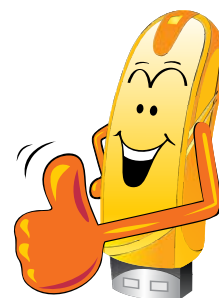


```
C:\SenseLock\case study\sample_36\win32\Debug\sample_36.exe
1.Set trial software expire time [dev model];
2.Activate trial software[dev model];
3.Run trial software[user model];
4.Exit;
5.Generate a file for remote-updating purpose;
Your choice:0]...
Trial software expired!
```

Для продления времени работы программы необходимо создать и загрузить в ключ новую лицензию.

Заключение.

В приведённом примере реализации защиты Вы ознакомились с современным и надёжным механизмом, который легко создать на основе ключей **Senselock-EL** серии. Реализованная подобным образом система защиты позволит осуществлять продажи дорогого программного обеспечения по схемам “лизинг” и “поэтапная оплата”. Ваши клиенты смогут полноценно использовать возможности Ваших программных продуктов и произвести его поэтапную оплату.



Вы можете полностью удаленно контролировать работу программы на стороне клиента, включать и выключать необходимые опции с периодическими обновлениями ключа, а также переключить Ваш продукт в полнофункциональный режим работы после полной оплаты.