ESMART®

ESMART Token PKCS#11

Содержание

1. Требования к операционной системе

- *Windows XP 32 и 64 бита;*
- *Windows Vista 32 и 64 бита;*
- *Windows 7 32 и 64 бита;*
- *Windows 8 или 8.1 32 и 64 бита;*
- *Windows Server 2003 / 2008 / 2012 32 и 64 бита;*
- *Linux (SUSE, AltLinux, Ubuntu, Debian);*
- *Mac OS X 10.7 и выше.*

Поддержка 64-битных систем реализована только в режиме 32 бита (т.е. только для работы с 32-битными приложениями).

1.1 Рекомендуемые считыватели смарт-карт

- *ACR38U;*
- *ACR38K;*
- *ACR128;*
- *ACR1281.*

Для всех считывателей должны быть установлены PC/SC драйвера. Если драйвер не может быть установлен автоматически через Windows Update, установить драйвера вручную из папки drivers для выбранной ОС. Для USB-ключа ESMART Token драйвера устанавливаются из папки drivers/ ESMART Token USB 64K.

2. Автоматическая установка в Windows

*Криптопровайдер для ESMART Token устанавливается автоматически при установке ESMART PKI Client. Подробно установка при помощи программы-инсталлятора описана в документе **ESMART PKI Client – Руководство администратора**. Программа помещает библиотеки .dll в соответствующие системные папки для 32-битных и 64-битных систем. Вносятся изменения в реестр.*

*Также при установке программы в папке **X:\Program Files\ESMART** создаются директории с файлами изменения реестра для установленных библиотек.*

3. Ручная установка в Windows

*Скопируйте isbc_pkcs11_main.dll и isbc_esmart_token_mod.dll из папки **SystemFolder** в системную папку Windows (X:\Windows\System32).*

*Запустите файлы реестра из директории pkcs11 **esmarttoken x86.reg** для 32-битной версии Windows или **esmarttoken x64.reg** для 64-битной версии Windows.*

В Windows Vista и выше для импорта файлов реестра требуются права администратора.

Если библиотеки были помещены не в системные папки, скорректируйте файлы изменения реестра.

4. Установка Linux

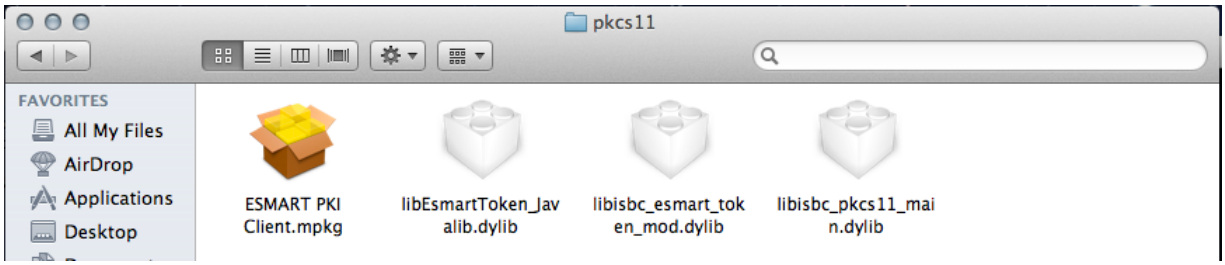
*Установите библиотеки PKCS#11 с помощью rpm-пакета **Linux/pkcs11/isbc-pkcs11-x.x.x-x.i586.rpm***

```
rpm -ivh isbc-pkcs11-x.x.x-x.i586.rpm
```

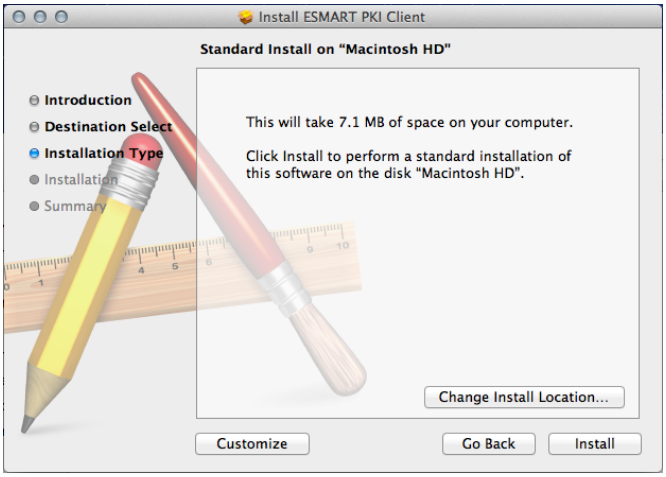
или скопируйте so-файлы вручную в папку /usr/lib.

5. Установка Mac OS X

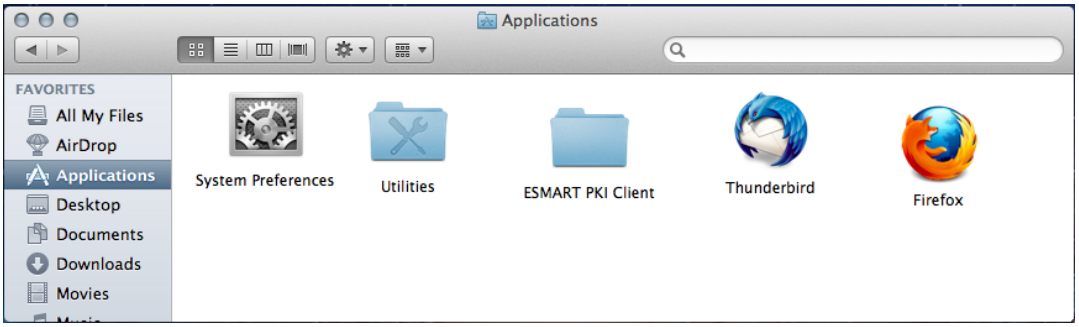
*Откройте папку MacOS/pkcs11 и запустите программу-инсталлятор **ESMART PKI Client.mpkg**. Следуйте подсказкам.*



*Укажите место установки, нажав **Change Install Location...** или оставьте значение по умолчанию.*

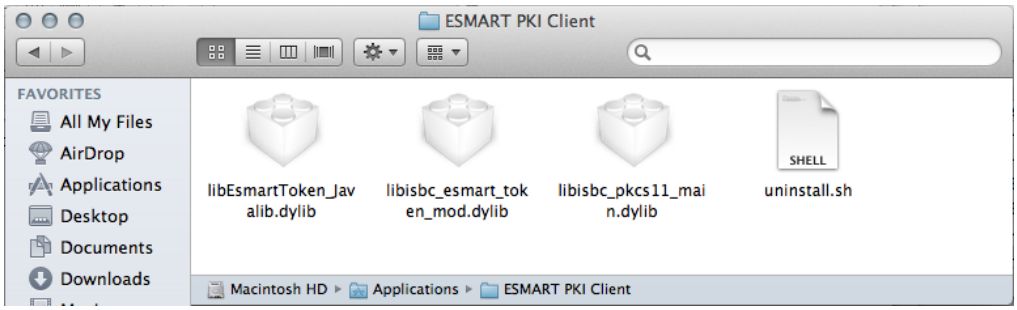


*Дождитесь окончания установки и сообщения **The Installation was successful**.*



Если при установке было выбрано место установки по умолчанию, в разделе Application (Приложения) появится папка ESMART PKI Client, содержащая файлы:

- *libisbc_esmart_token_mod.dylib;*
- *libisbc_pkcs11_main.dylib;*
- *libEsmartToken_Javalib.dylib;*
- *uninstall.sh.*



Установка завершена. Настройка и использование Mozilla Firefox и Mozilla Thunderbird описано в руководстве пользователя ***ESMART Token – ЭЦП и шифрование****.*

6. Использование утилит

В комплект ПО входит набор бесплатных утилит для работы с PKCS#11 от http://www.opensc-project.org/opensc.

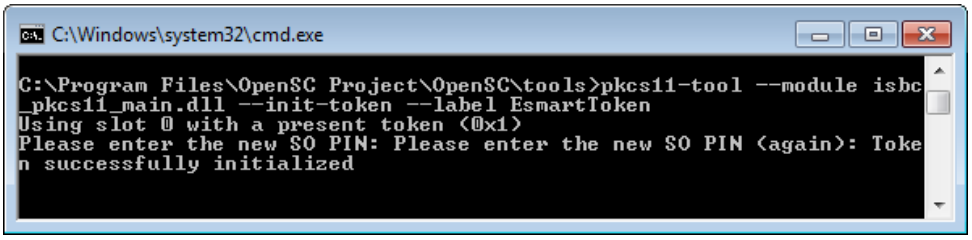
После их установки вы получаете возможность проверить работу PKCS#11. Для составления запроса на сертификат требуется пакет OpenSSL.

6.1 Примеры использования PKCS11-tools

Инициализация токена под Windows

```
pkcs11-tool.exe --module isbc_pkcs11_main.dll --init-token --label EsmartToken
```

Введите SO-PIN[1] карты 2 раза или передайте его в качестве параметра `--so-pin`



Инициализация токена под Linux

```
pkcs11-tool --module /usr/lib/libisbc_pkcs11_main.so  --init-token
--label EsmartToken
```

Введите SO-PIN карты 2 раза или передайте его в качестве параметра `--so-pin`

1 SO-PIN по умолчанию 12345678

Все команды под Linux и Windows аналогичны и отличаются только указанием пути к библиотеке (--module). Далее примеры будут приводиться только для Windows.

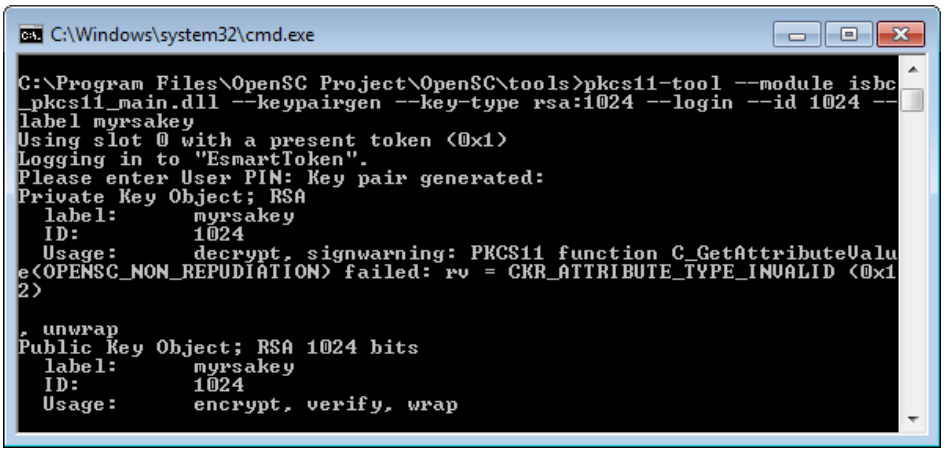
Получение информации об установленных токенах в системе

```
pkcs11-tool --module isbc_pkcs11_main.dll -L
```

Создание ключевой пары RSA 1024

```
pkcs11-tool --module isbc_pkcs11_main.dll --keypairgen --key-type rsa:1024
--login --id 1024 --label myrsakey
```

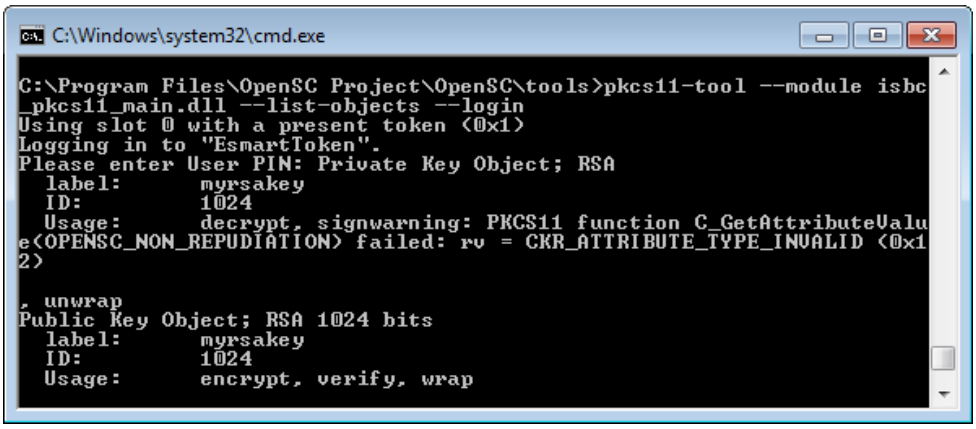
Для генерации ключей будет запрошен ПИН-код (по умолчанию при инициализации задается ПИН-код 12345678).



Просмотр объектов на карте

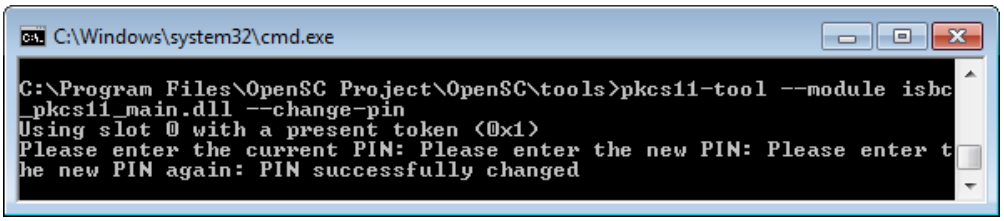
```
pkcs11-tool --module isbc_pkcs11_main.dll --list-objects --login
```

Введите ПИН-код для авторизации на карту или передайте его в качестве параметра `--pin 12345678`.



Смена PIN-кода

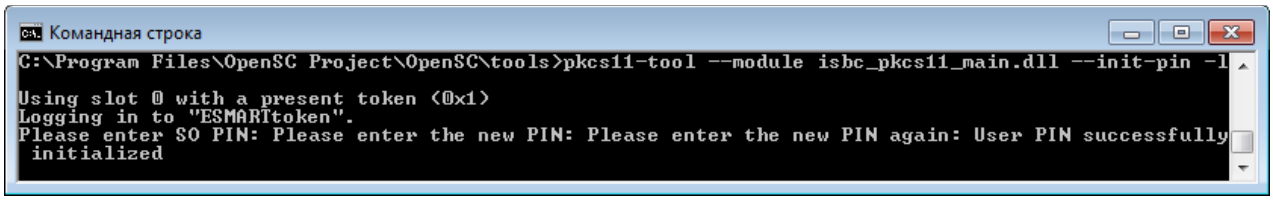
```
pkcs11-tool --module isbc_pkcs11_main.dll --change-pin
```



Разблокировка ПИН-кода

Требуется ввод SO PIN.

```
pkcs11-tool --module isbc_pkcs11_main.dll --init-pin -l
```



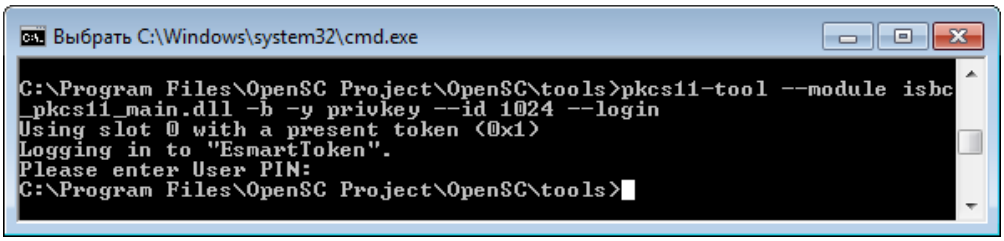
Удаление объекта

Для удаления объекта необходимо указать его тип и идентификатор (id) или название (label). Открытый и закрытый ключ удаляются отдельно.

```
pkcs11-tool --module isbc_pkcs11_main.dll -b -y privkey --login --id 1024
```

Типы объектов:

- ***privkey*** *– закрытый ключ;*
- ***pubkey*** *– открытый ключ;*
- ***cert*** *– сертификат.*



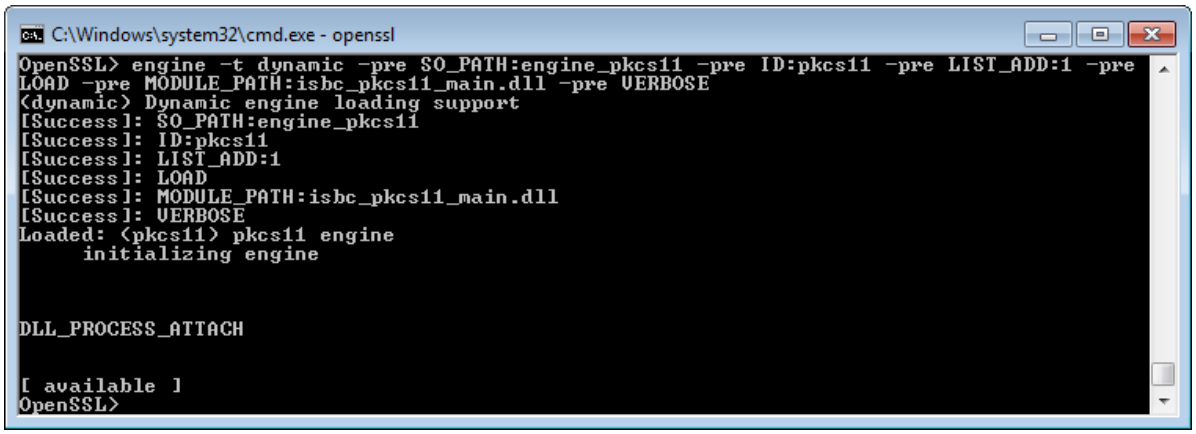
Внимание! *OpenSC не выдает сообщения при успешном удалении объекта с карты. Для проверки используйте команду* `--list-objects`.

6.2 Использование OpenSSL

Создание запроса на сертификат

1. *Для выпуска запроса на сертификат требуется использование OpenSSL с подключенной библиотекой для работы с PKCS#11 (engine PKCS#11) и модулем isbc_pkcs11_main.dll.*
2. *Для работы OpenSSl в Windows требуется библиотека MSVCR100.dll, входящая в распространяемый пакет Microsoft Visual C++ 2010. Скачать дистрибутив можно с сайта Microsoft:*
   32-битная версия: http://download.microsoft.com/download/5/B/C/5BC5DBB3-652D-4DCE-B14A-475AB85EEF6E/vcredist_x86.exe.
   64-битная версия: http://www.microsoft.com/ru-ru/download/details.aspx?id=14632
3. *Для подключения engine pkcs11 требуются библиотеки.*
   - ***engine_pkcs11.dll*** *и* ***libp11.dll*** *– библиотеки, реализующие взаимодействие OpenSSL через OpenSC;*
   - ***isbc_pkcs11_main.dll*** *и* ***isbc_esmart_token_mod.dll*** *– библиотеки, выполняющие роль подключаемого модуля (аналогично pkcs11-tool) для работы с ESMART Token;*
4. *Если OpenSSL не может открыть файл конфигурации, выполните в командной строке:*
   set OPENSSL_CONF="путь к openssl.exe"
5. *Команда для проверки* `echo %OPENSSL_CONF%`

6. *Запустите OpenSSL, желательно с правами администратора.*
7. *Подключите PKCS#11 engine с модулем isbc_pkcs11_main.dll:*
8. *engine -t dynamic -pre SO_PATH:engine_pkcs11 -pre ID:pkcs11 -pre LIST_ADD:1 -pre LOAD -pre MODULE_PATH:isbc_pkcs11_main.dll -pre VERBOSE*
   где:
   - ***SO_PATH:engine_pkcs11*** *– путь к библиотеке engine-pkcs11.dll (расширение .dll не указывается);*
   - ***MODULE_PATH:isbc_pkcs11_main.dll*** *– путь к библиотеке isbc_pkcs11_main.dll (расширение .dll обязательно указывается);*
9. *В примере библиотеки для сокращения записи пути помещены в ту же папку, что и OpenSSL.*



10. *Запрос на сертификат выполняется командой:*
req -engine pkcs11 -new -key slot_1-id_1024 -keyform engine -out cert.csr
где:
    - ***- engine pkcs11 –*** *указывается идентификатор библиотеки, заданной на предыдущем этапе командой* ***ID:pkcs11****;*
    - ***slot_1-id_1024 –*** *номер слота и идентификатор ключа, которые ранее были сгенерированы на карте утилитой pkcs11-tool;*
    - ***cert.csr –*** *название файла, в который будет сохранен запрос.*

7. Сертификаты

7.1 Получение сертификата

Запрос на сертификат необходимо подписать в аккредитованном удостоверяющем центре или в корпоративном центре сертификации на базе Windows Server.

7.2 Запись сертификата на карту

Подписанный сертификат необходимо записать на карту.

```
pkcs11-tool --module isbc_pkcs11_main.dll -w cert.cer -y cert --login
--id 1024 --label certificate
```

OpenSC требует сертификаты в двоичном формате (DER). При попытке записать файл в кодировке base64 (PEM), появляется сообщение об ошибке:

```
error: OpenSSL error during X509 certificate parsing
Aborting.
```

Сертификат можно переконвертировать при помощи OpenSSL.

PEM-->DER

```
Openssl> x509 -inform PEM -in cert.pem -outform DER -out cert.cer
```

DER-->PEM

```
Openssl> x509 -inform DER -in cert.cer -outform PEM -out cert.pem
```

7.3 Чтение сертификата

Если на карте имеется сертификат, его можно прочитать командой:

```
pkcs11-tool --module isbc_pkcs11_main.dll -r cert.cer -y cert --login
--id 0000 --label certificate
```

7.4 Использование сертификатов

Сертификаты на карте ESMART Token при работе по стандарту PKCS#11 могут использоваться различными приложениями, включая:

- *Интернет-браузер Mozilla Firefox;*
- *Почтовый клиент Mozilla Thunderbird;*
- *Adobe Acrobat;*
- *VPN-клиент OpenVPN.*

8. Удаление компонентов

При использовании автоматической установки воспользуйтесь панелью управления Windows, раздел Удаление программ. Если использовалась ручная установка, удалите файлы библиотек. В Windows запустите файлы изменения реестра ***remove esmarttoken x86.reg*** *или* ***remove esmarttoken x64.reg****, входящие в комплект установки.*

Список библиотек для удаления в Windows:

- *isbc_esmart_token_mod.dll;*
- *isbc_pkcs11_main.dll;*
- *EsmartToken_Javalib.dll.*

Список библиотек для удаления в Linux:

- *libisbc_esmart_token_mod.so;*
- *libisbc_pkcs11_main.so;*
- *libEsmartToken_Javalib.so.*

Список библиотек для удаления в Mac OS X в папке Приложения/Applications:

- *libisbc_esmart_token_mod.dylib;*
- *libisbc_pkcs11_main.dylib;*
- *libEsmartToken_Javalib.dylib.*

Опытным пользователям Mac OS X рекомендуется удалить компонент, запустив в консоли:

```
sudo /Applications/ESMART\ PKI\ Client/uninstall.sh
```

и ввести пароль администратора.