ESMART®

Описание ESMART Token

Содержание

1. Введение

Программно-аппаратный комплекс ESMART Token предназначен для повышения уровня информационной безопасности организаций, использующих пароли. ESMART Token позволяет производить двухфакторную авторизацию с использованием смарт-карт или USB-ключей.

Недостатки решений на основе паролей:

- *Пользователи задают слишком простые пароли, которые можно угадать перебором;*
- *Число попыток ввода пароля не всегда ограничено;*
- *Пользователи забывают сложные пароли;*
- *Однофакторная авторизация по паролю не достаточно защищена;*
- *Компрометация паролей может быть не заметна в течение длительного периода времени, а утеря карты или ключа обнаруживается практически сразу.*

Комплекс ESMART Token позволяет использовать все преимущества двухфакторной авторизации: предъявление физического носителя (карты или USB-ключа) и ввод ПИН-кода, защищенного от взлома методом перебора.

1.1 Инфраструктура открытых ключей

Обеспечение информационной безопасности предприятия может быть основано на Инфраструктуре открытых ключей PKI (англ. Public Key Infrastructure). Инфраструктура открытых ключей подразумевает использование ключевой пары: открытого и закрытого ключа, математически связанных друг с другом. Основополагающим принципом PKI является то, что закрытый ключ известен только его владельцу и должен быть надежно защищен.

В качестве носителей рекомендуется использовать не обычные флеш-накопители, а криптографические устройства в форме карты или USB-ключа. Преимущество хранения закрытых ключей на криптографической карте состоит в том, что генерация ключевой пары и операции, требующие предоставления закрытого ключа, выполняются криптографическим процессором карты или USB-ключа.

Помимо ключевой пары на карте или USB-ключе хранится сертификат пользователя. Сертификат является фиксированным набором сведений о пользователе, ключевой паре и самом сертификате. ESMART Token поддерживает сертификаты типа X.509.

1.2 Аппаратно-программный комплекс

Аппаратная часть:

- *Смарт-карты ESMART Token 64K;*
- *рекомендуемые считыватели Advanced Card Systems Ltd.:*
  - *ACR38U, ACR38K, ACR128, ACR1281*
  - *USB-ключ ESMART Token 64K;*

Программная часть:

- *Пользовательское приложение ESMART PKI Client;*
- *ESMART Token CSP;*
- *ESMART Token PKCS11;*
- *Инструменты разработчика:*
  - *JNI-wrapper для Java с примерами;*
  - *Примеры для C++;*
- *Plug-in для КриптоПро CSP;*
- *Драйверы оборудования.*

2. ESMART Token

Принцип работы смарт-карты и USB-ключа ESMART Token одинаков, следует выбирать наиболее удобный и подходящий вариант. Для офисного использования прекрасно подходит сочетание небольшого настольного считывателя и смарт-карты, которая помещается в кошелек. Носитель в виде USB-ключа фактически совмещает в себе миниатюрный считыватель и чип в одном корпусе, USB-ключ можно порекомендовать тем, кто часто бывает в разъездах.

В описании использован стандартный дизайн карты и ключа ESMART Token. Возможность, цены, условия и сроки поставки карт и USB-ключей ESMART Token с нестандартным дизайном зависят от объема заказа и оговариваются отдельно.

2.1 Смарт-карта ESMART Token

Модель ESMART Token в виде смарт-карты размера ID-1.

Лицевая сторона:

Опционально смарт-карта ESMART Token может поставляться со встроенной RFID-меткой[1] для использования в системах контроля и управления доступом (СКУД).

Для работы со смарт-картой ESMART Token требуется наличие считывателя контактных смарт-карт и свободный порт USB или RS-232 для подключения считывателя. На ПК пользователя должны быть установлены драйвера для выбранной модели считывателя и ПО для работы с картой посредством CSP (только Windows) и PKCS#11 (Windows, Linux, Mac OS).

Для выполнения основных операций с ESMART Token разработана программа ESMART PKI Client с графическим интерфейсом.
См. раздел ***Приложение ESMART PKI Client****.*

2.2 USB-ключ ESMART Token

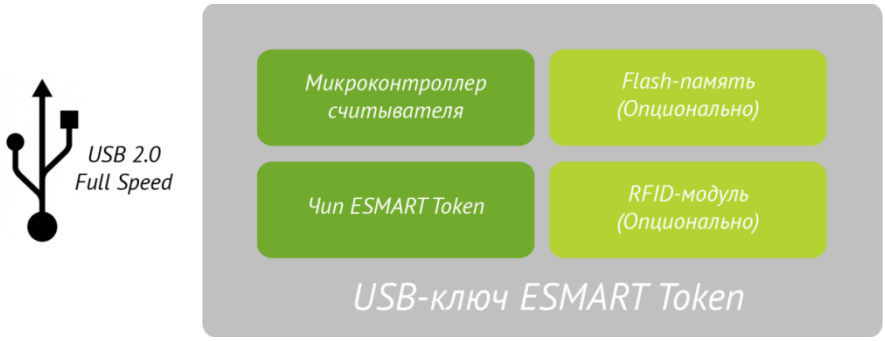
USB-ключ ESMART Token представляет собой комбинацию считывателя и чипа смарт-карты в виде миниатюрного устройства. Устройство подключается к ПК напрямую, отдельный считыватель смарт-карт не требуется. На каждый ПК должен быть установлен драйвер устройства для соответствующей операционной системы из папки drivers/ESMART Token USB 64K.

Ключ ESMART Token может быть оснащен дополнительной Flash-памятью и/или встроенной RFID-меткой[2].

---

[1] *Цены и условия поставки ESMART Token с RFID-меткой оговариваются отдельно*

[2] *Цены и условия поставки ESMART Token с Flash-памятью и/или RFID-меткой оговариваются отдельно*



2.3 Основные характеристики

Форм-фактор	Смарт-карта ID-1	USB-ключ
Подключение	Требуется считыватель	USB Plug&Play
Интерфейс	ISO 7816-2	USB 2.0 Full Speed
Размер	ID-1 (85,6 × 53,98 мм)	53,5 × 15,7 × 7,8 мм
Объем защищенной памяти EEPROM	80 КБ	
Объем памяти, доступной пользователю	64 КБ	
Срок хранения данных	10 лет	
Кол-во циклов перезаписи	500 000 циклов	
Криптографические алгоритмы	Асимметричные: RSA 512, 1024, 2048, 3072, 4096 (аппаратная генерация) ГОСТ Р 34.10-2001[3] Симметричные: DES, 3DES, AES 128, 192, 256, ГОСТ 28147-89 [3] Хеширование: SHA-1, SHA-256, MD5, ГОСТ Р 34.11-94 [3] Генератор случайных чисел в соответствии с FIPS 140-2	
Стандарты	PKCS#11 версии 2.20[4], Microsoft CryptoAPI, PC/SC, Сертификаты X.509 v3, SSL v3, IPSec/IKE, Microsoft CCID, ISO 7816 части 1,2,3,4,8,9	
Температура хранения	-65°C ~ +150°C	0°C ~ +50°C
Рабочая температура	-25°C ~ +85°C	0°C ~ +50°C

Спецификация может быть изменена без предварительного уведомления

[3] Поддержка российских криптографических алгоритмов осуществляется на программном уровне, требуется наличие криптопровайдера КриптоПро CSP (ООО «КРИПТО-ПРО»), ViPNet CSP (ОАО «ИнфоТеКС») или др.

[4] В том числе поддерживается расширенный функционал, такой как: функция формирования запроса сертификата в форматеPKCS#10, функции формирования и проверки ЭП в формате PKCS#7, а также функция загрузки на карту контейнера в формате PKCS#12.

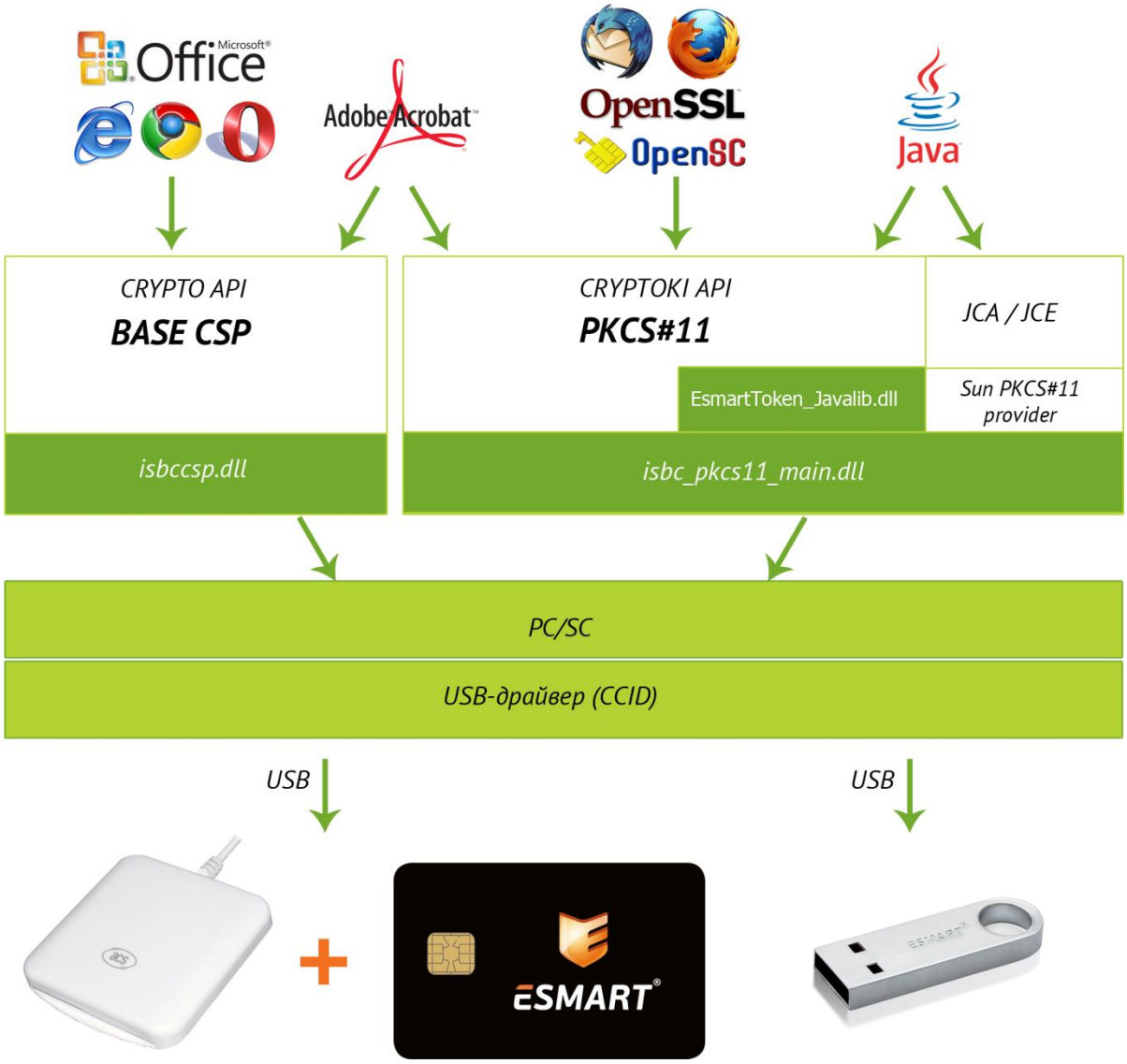
2.4 Преимущества ESMART Token

- Поддержка алгоритмов RSA с ключом до 4096 байт;
- Использования в качестве ключевого носителя для СКЗИ, поддерживающих российские криптографические алгоритмы (ГОСТ Р 34.10-94, ГОСТ Р 34.11-94, ГОСТ Р 34.10-2001, ГОСТ 28147-89)[5];
- Два форм-фактора: карта и USB-ключ;
- Поддержка Windows, Linux и Mac OS;
- Генерация ключей на чипе – невозможно извлечь закрытый ключ;
- Большой объем памяти для хранения нескольких пар ключей и сертификатов;
- Возможность записи и безопасного хранения произвольных данных;
- Пользовательское приложение с графическим интерфейсом ESMART PKI Client;
- Полный цикл управления ключом или картой при помощи бесплатных утилит OpenSC и OpenSSL;
- Встроенная RFID-метка (опционально);
- Дополнительная Flash-память (опционально, только для USB-ключей).

[5] Для поддержки российских криптографических алгоритмов дополнительно необходимо установить КриптоПро CSP, СигналКом CSP или ViPNet CSP.

3. Архитектура

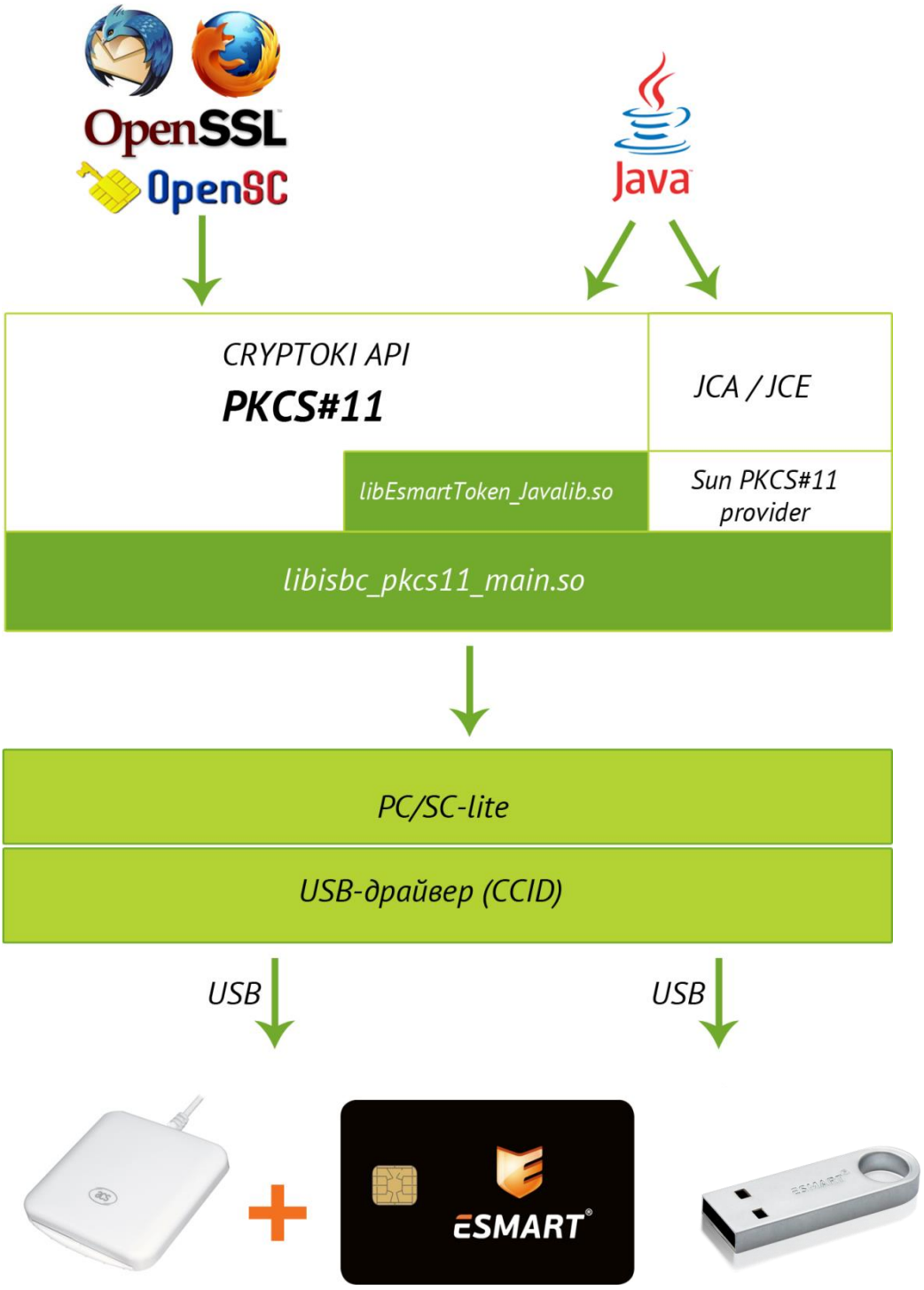
3.1 Windows



Примечания:

1) *JCA/JCE – см. ESMART Token Java API;*
2) *Темно-зеленым цветом выделены библиотеки, входящие в состав комплекса ESMART Token;*
3) *Base CSP для Windows XP устанавливается отдельно, См. раздел ESMART Token CSP.*

3.2 Linux



Примечания:

1) *JCA/JCE – см. ESMART Token Java API;*
2) *Темно-зеленым цветом выделены библиотеки, входящие в состав комплекса ESMART Token.*

4. Объекты

Ключевая пара

ESMART Token позволяет хранить одну или несколько ключевых пар, т.е. открытый ключ и соответствующий ему закрытый ключ. Одновременно можно хранить на карте ключи RSA и ГОСТ.

Сертификат

К каждой ключевой паре может быть загружен сертификат открытого ключа в формате X.509. Дополнительно на карту могут быть загружены сертификаты без закрытых ключей, например, корневой сертификат Удостоверяющего Центра.

Произвольные данные

*ESMART Token обладает достаточно большим объемом памяти для устройств своего класса, поэтому является универсальным и удобным носителем для хранения логинов и паролей в текстовом виде. Данные можно хранить в открытом виде или в защищенном виде и показывать только после предъявления ПИН-кода. Дополнительная информация по хранению данных на ESMART Token представлена в руководстве **ESMART Token – PKCS11**, руководстве пользователя и руководстве администратора к программе ESMART PKI Client.*

5. ПИН-код

ПИН-код предназначен для безопасного доступа к информации, хранящейся на карте или USB-ключе. ПИН-код задается ASCII-символами и должен иметь длину от 4 до 8 знаков. Не рекомендуется использовать слабые ПИН-коды (например 1234) и ПИН-коды по умолчанию[6].

ESMART Token использует следующие типы ПИН-кода:

- ***ПИН-код пользователя (User PIN)**, используется для смены ПИН-кода пользователя, авторизации на карте для получения доступа к ключам, генерации объектов, операциях подписания и шифрования из пользовательских программ.*
- ***ПИН-код администратора (SO[7] PIN)**, используется для инициализации карты, для очистки карты, для смены ПИН-кода пользователя.*

Два ПИН-кода позволяют обеспечить более высокую степень безопасности за счет разделения полномочий. Рекомендуется сменить все ПИН-коды сразу после получения карты.

Смарт-карты и USB-ключи защищены от взлома методом перебора ПИН-кода. ПИН-код блокируется после 10 неудачных попыток набора подряд. Сменить ПИН-код пользователя, когда карта заблокирована, можно только при помощи ПИН-кода администратора. Карту также можно повторно инициализировать, но в этом случае все сгенерированные ключевые пары, сертификаты и данные будут потеряны. Повторная инициализация может использоваться, например, если ключевая пара и сертификат записываются на карту из файла PKCS#12 или PFX.

Внимание! Количество неверных попыток набора ПИН-кода администратора (SO PIN) задается при инициализации карты. По умолчанию количество попыток 10. Если неверно набрать ПИН-код администратора указанное количество раз, карта блокируется без возможности восстановления или повторной инициализации.

5.1 Смена ПИН-кода

В целях безопасности рекомендуется регулярно менять ПИН-код, минимум один раз в 3 месяца. Пользователь может сменить ПИН-код самостоятельно, если помнит текущий ПИН-код и карта не заблокирована.

---

[6] *ПИН-код пользователя по умолчанию «12345678», ПИН-код администратора по умолчанию «12345678»*

[7] *Security Officer - сотрудник по вопросам информационной безопасности*

Для смены ПИН-кода можно использовать программу ESMART PKI Client или бесплатную кроссплатформенную утилиту OpenSC. Подробное описание работы с приложениями представлено в руководствах ***ESMART Token – PKCS11*** *и руководствах* ***ESMART PKI Client***.

5.2 Разблокировка ПИН-кода

Для разблокировки ПИН-кода можно использовать программу ESMART PKI Client или бесплатную кроссплатформенную утилиту OpenSC. Чтобы разблокировать ПИН-код пользователя, требуется ПИН-код администратора.

6. Возможности использования ESMART Token

6.1 ESMART Token CSP

CSP[8] – криптопровайдер, т.е. модуль, позволяющий производить криптографические операции в ОС Windows через CryptoAPI. Приложения не работают непосредственно с криптопровайдером, они вызывают функции CryptoAPI из библиотек Advapi32.dll и Crypt32.dll.

Для работы в Windows XP требуется установка пакета Microsoft Base CSP http://www.microsoft.com/en-us/download/details.aspx?id=4670

Модуль предназначен для авторизации в Windows в домене и работы ESMART Token с приложениями, включая:

- *Программы пакета MS Office;*
- *Программы пакета Open Office;*
- *Почтовый клиент MS Outlook;*
- *Adobe Acrobat;*
- *Internet Explorer.*

Подробно работа с приложениями описана в руководствах ***ESMART Token – CSP*** *и* ***ESMART Token – Настройка пользовательских приложений в Windows***.

6.2 ESMART Token PKCS#11

PKCS#11 – открытый кроссплатформенный стандарт взаимодействия операционной системы и смарт-карты. В отличие от CSP, сертификаты не хранятся в стандартном хранилище Windows. Стандарт PKCS#11 является основным средством работы со смарт-картами в Linux и Mac OS.

Модуль предназначен для работы ESMART Token с приложениями, включая:

- *Mozilla Firefox;*
- *Mozilla Thunderbird;*
- *Adobe Acrobat;*
- *OpenVPN;*
- *OpenSSL.*

Подробно работа с приложениями описана в руководствах ***ESMART Token – PKCS11*** *и* ***ESMART Token – Настройка пользовательских приложений в Windows***.

[8] *Cryptographic service provider*

6.3 ESMART Token Java API

Java API предназначено для вызова некоторых функций библиотеки PKCS#11 из Java. Вызовы функций библиотеки PKCS#11 осуществляются через технологию JNI (Java Native Interface). Возможно использование библиотеки в JavaScript с использованием технологии Java Applet.

Для использования ESMART Token в инфраструктурах, поддерживающих Java Cryptography Architecture (JCA) и Java Cryptography Extension (JCE), можно использовать Java wrapper над PKCS#11 от компании Oracle. Подробнее: http://docs.oracle.com/javase/7/docs/technotes/guides/security/p11guide.html.

Подробно работа с Java API и демонстрационным апплетом описана в руководстве ***ESMART Token – PKCS11 Java****.*

6.4 Plug-in ESMART Token для КриптоПро CSP

Плагин предназначен для использования USB-ключей и карт ESMART Token с программным обеспечением, разрабатываемым ООО «КРИПТО-ПРО» и поддерживающим российские криптографические алгоритмы: ГОСТ Р 34.10-94, ГОСТ Р 34.11-94, ГОСТ Р 34.10-2001, ГОСТ 28147-89.

Подробно установка и назначение плагина, а также процедура получения тестового сертификата описаны в руководстве ***ESMART Token – CryptoPro****.*

6.5 Криптопровайдер ViPNet CSP

Для совместимости с криптопровайдером ViPNet CSP необходимо инициализировать карту с профилем ViPNet, ViPNet2 или VipNet3 при помощи приложения ESMART Token PKI Client. Подробно профили описаны в Руководстве Администратора ESMART PKI Client.

6.6 Криптопровайдер SignalCOM SCP

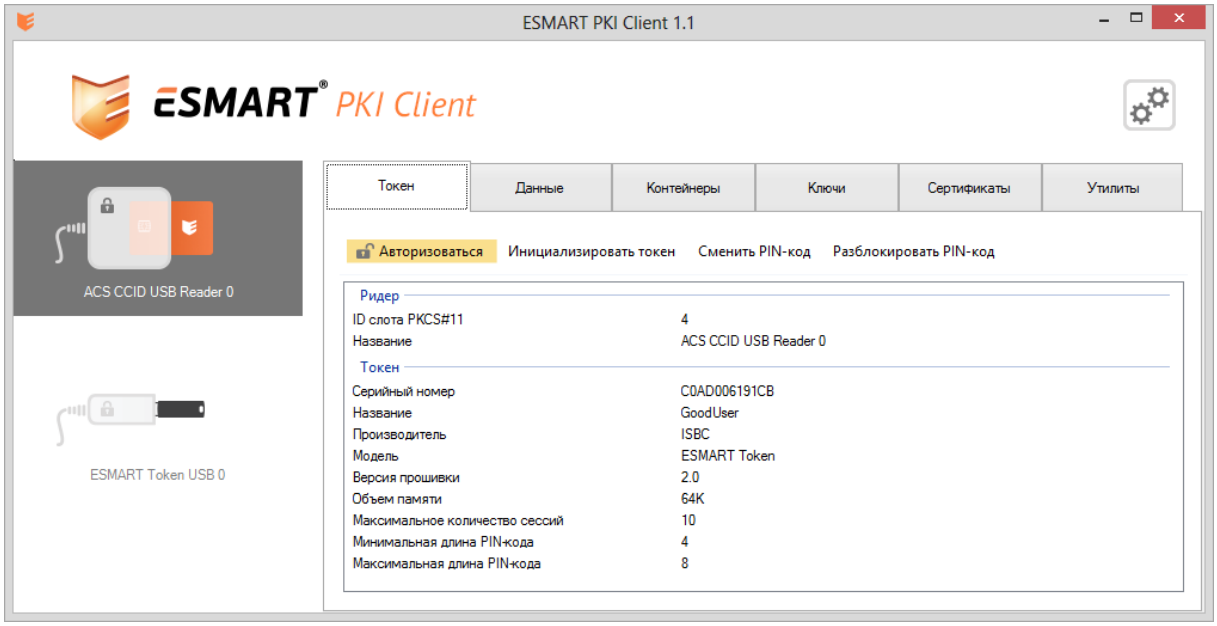
Чтобы использовать карты или токены ESMART Token с криптопровайдером SignalCOM SCP установите бесплатное приложение ESMART Token PKI Client (версии не ранее 2015 года). Все необходимые изменения будут внесены в систему автоматически при работе программы установщика. Если криптопровайдер SignalCOM SCP не может работать с ESMART Token, установите или переустановите приложение ESMART PKI Client.

7. Приложение ESMART PKI Client

Программа настройки смарт-карт ESMART Token с удобным графическим интерфейсом позволяет выполнить все операции с картой:

- *Просмотр информации о ESMART Token;*
- *Инициализация карты или USB-ключа ESMART Token;*
- *Смена и разблокировка ПИН-кода пользователя и ПИН-кода администратора;*
- *Просмотр объектов на ESMART Token;*
- *Генерация ключей;*
- *Создание запроса на сертификат;*
- *Запись сертификата на ESMART Token;*
- *Запись произвольных данных на ESMART Token;*
- *Импорт файлов PKCS#12 или PFX на ESMART Token;*
- *Поддерживается работа сразу с несколькими считывателями смарт-карт и/или USB-ключами.*

7.1 Интерфейс приложения



7.2 Сторонние утилиты для работы с ESMART Token

Для работы с ESMART Token по стандарту PKCS#11 можно использовать бесплатные кроссплатформенные утилиты OpenSC и OpenSSL.

*Подробно использование утилит описано в руководстве **ESMART Token – PKCS11**. В описание также включены примеры команд.*

8. Работа с ESMART Token

8.1 Задачи администратора

В задачи администратора могут входить:

- *Установка драйверов оборудования;*
- *Установка ПО для работы с ESMART Token;*
- *Инициализация ESMART Token;*
- *Генерация ключей и создание запроса на сертификат;*
- *Запись сертификата на ESMART Token;*
- *Разблокировка ПИН-кода;*
- *Настройка приложений для использования с ESMART Token.*

8.2 Задачи пользователя

В задачи пользователя могут входить:

- *Генерация ключей;*
- *Выписка запроса на сертификат;*
- *Запись сертификата на ESMART Token;*
- *Смена ПИН-кода;*
- *Запись произвольных данных, например, логинов и паролей;*
- *Использование в приложениях.*

Приложение. Состав дистрибутива

- *Documentation:*
  - *Тексты стандартов;*
  - *Руководства администратора;*
  - *Руководства пользователя;*
- *AnyPlatform:*
  - *Плагин для КриптоПро JCP;*
- *Linux:*
  - *Пакет драйверов оборудования;*
  - *Установочный файл и библиотеки PKCS#11;*
  - *Утилита OpenSC;*
  - *Плагин для КриптоПро CSP;*
- *Mac OS*
  - *ESMART PKI Client;*
  - *Пакет драйверов оборудования;*
  - *Установочный файл и библиотеки PKCS#11;*
  - *Утилита OpenSC;*
  - *Плагин для КриптоПро CSP;*
- *SDK:*
  - *Средства для использования ESMART Token в Java; Примеры на Java и JavaScript;*
  - *Примеры на C++;*
- *Windows:*
  - *ESMART PKI Client;*
  - *Пакет драйверов оборудования;*
  - *Библиотеки для установки вручную;*
  - *Утилита OpenSC с плагинами для OpenSSL;*
  - *Плагин для КриптоПро CSP.*