

Trusted Bitrix

Руководство пользователя



ООО «Цифровые технологии»

Trusted Bitrix. Руководство пользователя. Версия 1.1.

Дата сборки документа 17.04.2013.

Этот документ является составной частью технической документации ООО «Цифровые технологии».

Сайт справки по продуктам ООО «Цифровые технологии» <http://www.trusted.ru>

© 2012-2013 ООО «Цифровые технологии». Все права защищены.

Контактная информация

ООО «Цифровые технологии»

<http://www.trusted.ru>

Содержание

Введение	4
Раздел 1. Описание программы	5
Раздел 2. Функциональные возможности программы.....	6
Раздел 3. Требования к программному окружению.....	7
Раздел 4. Технология аутентификации по протоколу TLS	8
НЕОБХОДИМЫЕ ЭЛЕМЕНТЫ АУТЕНТИФИКАЦИИ ПО TLS	8
СХЕМА АУТЕНТИФИКАЦИИ	8
Раздел 5. Установка модуля Trusted Bitrix	10
УСТАНОВКА МОДУЛЯ ИЗ MARKETPLACE.....	10
НАСТРОЙКА ПАРАМЕТРОВ МОДУЛЯ.....	12
НАСТРОЙКА СПРАВОЧНИКА ДОВЕРЕННЫХ ПРОКСИ-СЕРВЕРОВ	14
АДМИНИСТРИРОВАНИЕ ХРАНИЛИЩА СЕРТИФИКАТОВ	16
УСИЛЕННАЯ АВТОРИЗАЦИЯ ПО СЕРТИФИКАТУ	19
УДАЛЕНИЕ МОДУЛЯ.....	21
Раздел 6. Настройка TLS-соединений	23
ГЕНЕРАЦИЯ СЕРТИФИКАТОВ ЧЕРЕЗ OPENSSL	23
НАСТРОЙКА TLS НА СЕРВЕРЕ APACHE	27
НАСТРОЙКА TLS НА ПРОКСИ-СЕРВЕРЕ NGINX.....	29
НАСТРОЙКА TLS НА ПРОКСИ-СЕРВЕРЕ APACHE	30
Раздел 7. Настройки браузеров.....	33
НАСТРОЙКА БРАУЗЕРА MICROSOFT INTERNET EXPLORER	33
НАСТРОЙКА БРАУЗЕРА GOOGLE CHROME	34
НАСТРОЙКА БРАУЗЕРА SRWARE IRON.....	38
Раздел 8. Работа с токенами ruToken (модели S,CSP,ECP)	39
ПОДГОТОВКА КЛИЕНТСКОГО МЕСТА	39
ГЕНЕРАЦИЯ НА ТОКЕНЕ ЗАКРЫТОГО КЛЮЧА И ЗАПРОСА НА СЕРТИФИКАТ.....	39
УСТАНОВКА СЕРТИФИКАТА НА ТОКЕН.....	44
ПОДКЛЮЧЕНИЕ ОТЧУЖДАЕМОГО НОСИТЕЛЯ	46
НАСТРОЙКА БРАУЗЕРОВ ДЛЯ РАБОТЫ С ТОКЕНАМИ	49
Раздел 9. Коды ошибок модуля.....	52

Введение

Руководство предназначено для пользователей, администраторов и редакторов сайтов на базе продуктов компании «1С-Битрикс». В руководстве описаны основные действия по установке и настройке модуля **Trusted Bitrix**.

Использование модуля **Trusted Bitrix** может быть интересным для компаний, предоставляющих доступ к своим информационным веб-ресурсам, развернутым на базе продуктов от компании 1С-Битрикс, и желающих обеспечить не только конфиденциальность передаваемой информации, но и дополнительную возможность авторизации по сертификатам своих клиентов/партнеров на этих ресурсах.

Решение с использованием модуля **Trusted Bitrix** редакции StartPRO («Доступ пользователей по SSL сертификату»), StandartPRO рекомендуется компаниям, обязанным использовать сертифицированные в РФ средства криптографической защиты данных, как при их передаче по открытым каналам связи, так и при их обработке. Компаниям предоставляется возможность развернуть для своих сотрудников/клиентов/партнеров систему авторизации на информационных порталах по ГОСТ-сертификатам с возможностью использования механизмов передачи данных через прокси-серверы.

Раздел 1. Описание программы

Программа **Trusted Bitrix** является модулем для продуктов компании [1С-Битрикс](#) версии 12.x:

- “[1С-Битрикс: Управление сайтом](#)” в редакции «Старт» или “Бизнес” (для Trusted Bitrix Standart) и выше,
- “[1С-Битрикс: Официальный сайт государственной организации \(расширенный\)](#)” (решение на базе старшей редакции «Веб-кластер» продукта «1С-Битрикс: Управление сайтом»),
- “[1С-Битрикс: Корпоративный портал](#)”.

Модуль обеспечивает:

- аутентификацию пользователя на удаленном ресурсе (портале/сайте) по его цифровому сертификату, используемому при установлении защищенного (шифрованного) соединения с двухсторонней аутентификацией (клиента и сервера)
- обработку цифровых сертификатов клиентов, полученных как непосредственно из защищенного потока данных, так и от прокси-серверов.

Загрузка модуля осуществляется через [Маркетплейс](#) компании 1С-Битрикс.

Раздел 2. Функциональные возможности программы

Модуль **Trusted Bitrix** предоставляет следующие возможности:

- Управление сертификатами пользователей портала: регистрация (добавление), удаление, привязка к пользователям.
- Активация сертификата для проведения операции аутентификации, назначение периода его активности
- Установка запрета пользователю на проведение стандартной авторизации. Аутентификация пользователя только по сертификату
- Аутентификация пользователей по сертификату, полученному непосредственно от http-сервера из защищенного потока
- Установка запрета клиентам, прошедшим аутентификацию по сертификату, на авторизацию от имени другого пользователя.
- Управление справочником доверенных прокси-серверов: регистрация (добавление), удаление, активация прокси-сервера, назначение ему периода активности (за исключением редакции Start)
- Автоматическая регистрация настроенного прокси-сервера в справочнике доверенных прокси-серверов (за исключением редакции Start)
- Аутентификация пользователей по сертификату, переданному на http-сервер через прокси-сервер (за исключением редакции Start)

Раздел 3. Требования к программному окружению

Программа **Trusted Bitrix** является модулем к продуктам 1С-Битрикс, работающим под управлением


- «1С-Битрикс: Веб-окружение» 4.2- Linux
- «1С-Битрикс: Веб-окружение» 2.1 - Windows

Программа **Trusted Bitrix** поддерживается в следующих продуктах компании [1С-Битрикс](#) версии 12.x:

- “[1С-Битрикс: Управление сайтом](#)” в редакции «Старт» или “Бизнес” (для Trusted Bitrix Standart) и выше,
- “[1С-Битрикс: Официальный сайт государственной организации \(расширенный\)](#)” (решение на базе старшей редакции «Веб-кластер» продукта «1С-Битрикс: Управление сайтом»),
- “[1С-Битрикс: Корпоративный портал](#)”.

Поддерживается следующий список платформ

- Microsoft Windows Server 2003/2008 32/64
- CentOS 5/6 (i386, x86_64), Red Hat Enterprise Linux 5/6 (i386, x86_64)

 **Важно!** В случае необходимости использования ГОСТ алгоритмов на клиентском рабочем месте требуется установка дополнительного ПО – криптопровайдера КриптоПро CSP. На сервере в этом случае должны быть установлены продукты КриптоПро CSP, TrustedTLS, КриптоАРМ (ОС Windows) или Trusted Java (ОС Linux) для модуля Trusted Bitrix **Standart**.

Раздел 4. Технология аутентификации по протоколу TLS

Необходимые элементы аутентификации по TLS

Что необходимо настроить для использования TLS-соединений:

1. *Сертификат сервера.* При генерации запроса на создание сертификата серверу необходимо учитывать тот факт, что клиентский браузер проверяет соответствие доменного имени сервера полю из серверного сертификата «Common Name». Если такая проверка не проходит, то браузер сообщает об ошибке несоответствия. Поэтому необходимо заранее спланировать подходящее имя серверу, задаваемое в поле «CN» сертификата.
2. *Сертификат клиента.* Сертификат клиента или сертификат удостоверяющего центра (УЦ), издавшего данный сертификат, некоторым образом должен быть сохранен на сервере в списке доверенных. Иначе сервер не будет пропускать клиента до своих ресурсов.
3. *Доверие серверному сертификату.* Если в процессе установления TLS-соединения сервер присылает свой сертификат, который не заверен центром сертификации, известным компьютеру в статусе доверенного, клиентский браузер, в зависимости от настроек, как правило, спрашивает пользователя о предоставлении доверия этому сертификату. Возможность бездумного положительного ответа перечеркивает всю безопасность, предлагаемую протоколом TLS. Для отключения этой возможности сертификат сервера или УЦ, его издавшего, при необходимости желательно заранее занести в список доверенных серверов или УЦ. Появление запроса на доверие серверу должно побуждать клиента к отказу от сеанса обмена и к предварительной настройке доверия. Для уменьшения действий по настройке клиентского места желательно получать серверные сертификаты в УЦ, которые уже присутствуют в операционной системе (и/или добавляются при обновлениях) как доверенные.


 **Важно!** Быстродействие установления TLS соединения значительно отличается от обычного входа. Инициализация протокола требует трех посылок данных туда и обратно (т.е. это уже 3 пинга), еще достаточно времени требует генерирование цифровой подписи (мы рассматриваем сторону сервера), а также разворачивание ключей для симметричных алгоритмов шифрования, что в сумме может занять более полсекунды (кстати, надо заметить, что генерирование DSA-подписи при равном размере с RSA-ключом, происходит раза в 2-4 быстрее).

Схема аутентификации

На стороне главного сервера (Apache), на котором функционируют приложения 1С-Битрикс, или на прокси-серверах (Apache, Nginx) настраивается функционирование защищенного канала с опциональной или обязательной аутентификацией клиента по сертификату на ресурсах, на которых будет производиться аутентификация клиента в приложениях 1С-Битрикс. На (http-, прокси-) сервере при необходимости настраивается


- список доверенных УЦ, сужающий список клиентов, получающих доступ к ресурсам по признаку принадлежности их сертификатов указанным УЦ.
- получение списков отозванных сертификатов (СОС) для определения актуального статуса состояния сертификата клиента, значение которого используется при предоставлении доступа к ресурсам

Если на переднем плане на стороне сервера устанавливаются прокси-сервера, то они настраиваются на проброс сертификатов клиентов до главного сервера в специальных заголовках запросов.

На стороне главного сервера в рамках приложения при наличии в схеме взаимодействия прокси-серверов настраивается их список как доверенных, который используется при получении от них сертификатов клиентов.

При поступлении клиентского запроса приложению оно, прежде чем сформировать ответ, проверяет присутствие сертификата клиента в запросе, определяет источник сертификата (непосредственно из защищенного соединения с клиентом или от доверенного прокси-сервера). Затем производится аутентификация через поиск полученного сертификата клиента в справочнике ассоциаций сертификатов с зарегистрированными пользователями приложения. Если соответствие сертификата пользователю найдено и оно активно (**аутентификация** успешно пройдена), и если клиент еще не авторизован, то происходит его автоматическая **авторизация**. Если клиент уже был авторизован, то производится проверка принадлежности сертификата авторизованному пользователю и при отрицательном ее исходе производится авторизация пользователя, владеющего сертификатом. Такое поведение смены авторизации управляется настройками модуля Trusted Bitrix.

Исходя из методики авторизации по сертификату, рекомендуется (достаточно) защищать требованием клиентской авторизации ресурсы, где расположены встроенные стандартные скрипты авторизации, основанные на вводе логина и пароля. В этом случае данный скрипт уже не будет требовать ввода логина и пароля, так как авторизация будет проведена автоматически на основе клиентского сертификата, если, конечно, он будет предъявлен клиентом.

 **Важно!** Сложные (комплексные) скрипты авторизации (с использованием внешних источников авторизации) желательно размещать на ресурсах, свободных от требования клиентской авторизации на ресурсе, чтобы при их вызове не происходила автоматическая авторизация по сертификату при его предъявлении. Это замечание относится также к скриптам авторизации, которые не позволяют авторизоваться без выхода пользователя из системы. Это замечание относится также к случаю, когда допускается авторизоваться в системе пользователю – не владельцу сертификата.

Раздел 5. Установка модуля Trusted Bitrix

Установка модуля из Marketplace

Модуль «**TrustedBitrix**» можно установить из Marketplace следующим образом:

Перейдите на административную часть сайта и выберите на вкладке **Настройки** пункт **MarketPlace**.

В разделе **Решения для сайтов -> Безопасность** представлена информация о модуле «**Trusted Bitrix. Доступ пользователей по SSL сертификату**». Перейдите по ссылке **Установить**.

В открывшемся окне ознакомьтесь с соглашением об использовании (содержание лицензионного соглашения).

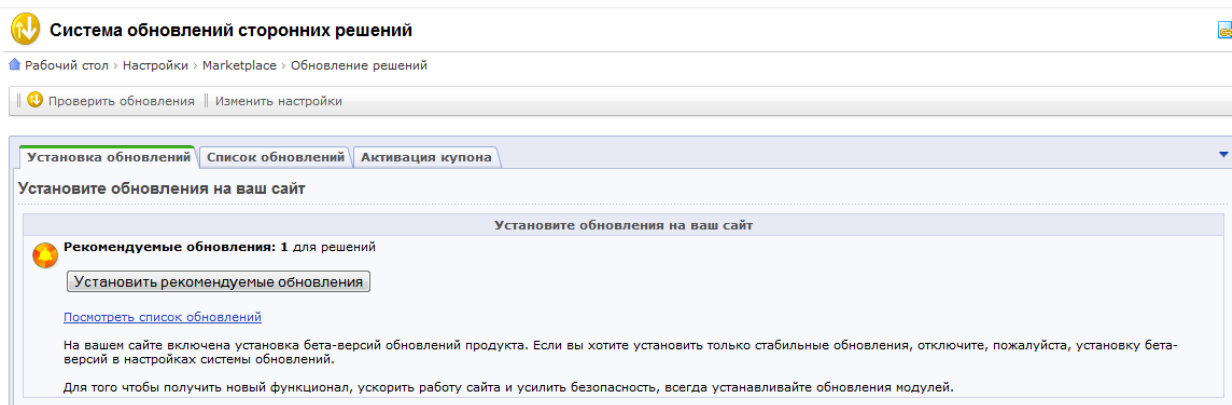


Рис. 5.1 Инициализация загрузки модуля на портал

После выбора действия по установке рекомендуемого обновления, исходный код модуля будет загружен на портал (рис. 5.1).

После успешной загрузки модуля его можно установить (рис. 5.2).

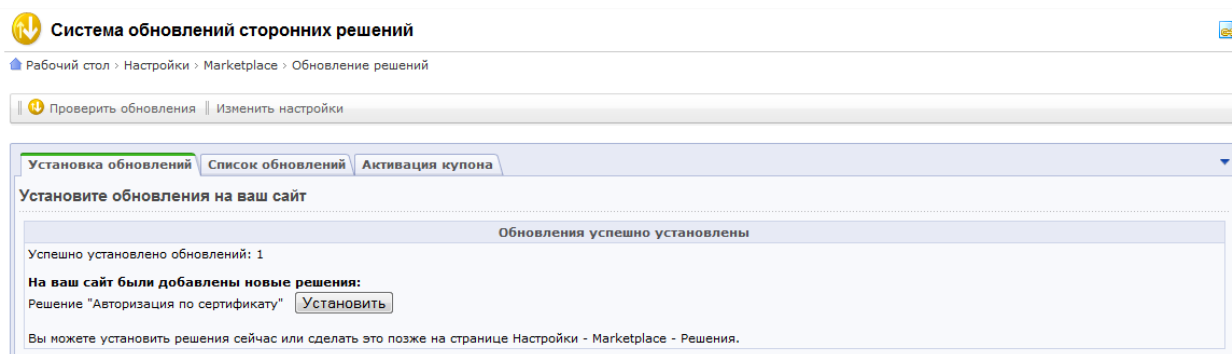


Рис. 5.2 Установка решения на портал

Установка модуля заключается в последовательном прохождении двух шагов мастера. На первом шаге предлагается выбор варианта организации инфоблоков хранилищ (рис. 5.3):

- в первом случае создаются новые инфоблоки с неизменными именами и заданными пользовательскими свойствами. Если инфоблоки модуля были созданы и использовались ранее, то произойдет уничтожения существующих в них данных.
- во втором случае инфоблоки не создаются, а используются существующие. Имена инфоблоков будут отражены в полях просмотра и остаются неизменными.

▲ Рабочий стол > Marketplace > Установленные решения

Установка модуля Trusted Bitrix ☆

✓ При установке модуля обнаружены ранее созданные инфоблоки с требуемой структурой. При создании новых инфоблоков произойдет удаление прежних данных.

☐ Создать инфоблоки?

☒ Использовать существующие?

*Инфоблок хранения атрибутов пользователей: TrustedBitrix_cert : CERT_STORE

*Инфоблок хранения сертификатов: TrustedBitrix_mapping : MAPPING_STORE

*Инфоблок хранения проху: TrustedBitrix_proxy : PROXY_STORE

Продолжить установку модуля

Рис. 5.3. Установка решения на портал

На следующем шаге мастера установки (рис. 5.4) должно появиться сообщение о завершении установки и регистрации модуля на портале.

▲ Рабочий стол > Marketplace > Установленные решения

Установка модуля Trusted Bitrix ☆

✓ Установка модуля успешно завершена

Вернуться в список

Рис. 5.4. Завершение установки

После щелчка по кнопке **«Вернуться в список»** будет выполнен переход в список установленных решений (рис. 5.5).

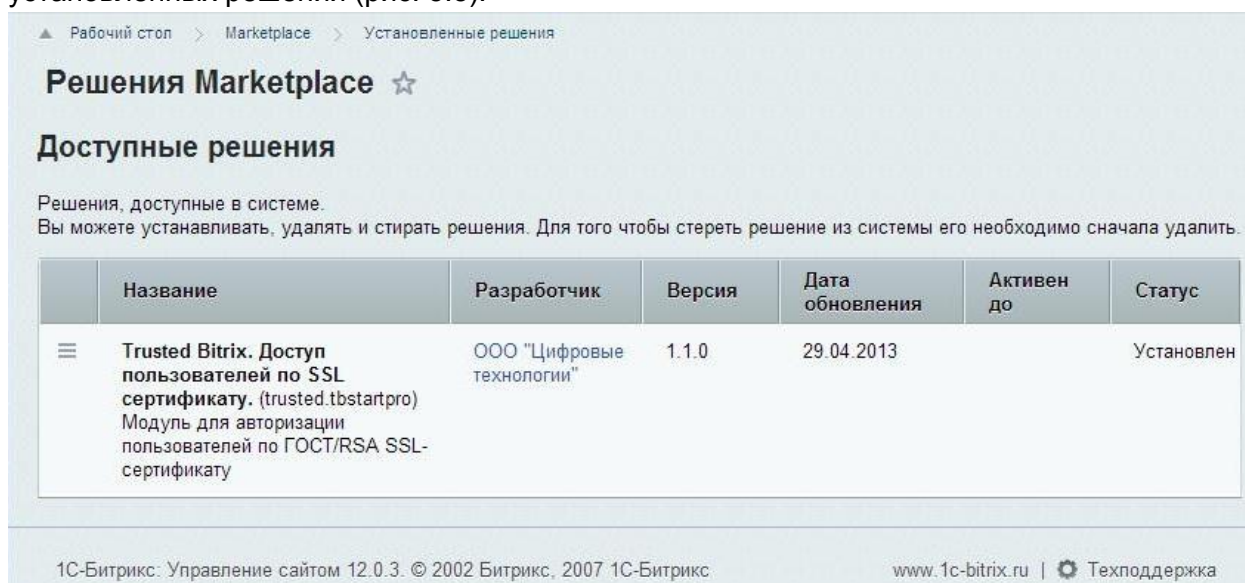


Рис. 5.5. Список установленных решений

Присутствие записи о решении в списке со статусом «Установлен», говорит о том, что модуль готов к использованию при условии настройки доступа по протоколу TLS.

Настройка параметров модуля

Настройка модуля осуществляется через панель (рабочий стол) администратора в разделе **Настройки** → **Настройки продукта** → **Настройки модулей**. И далее выбирается модуль **«Trusted Bitrix. Доступ пользователей по SSL сертификату»**.

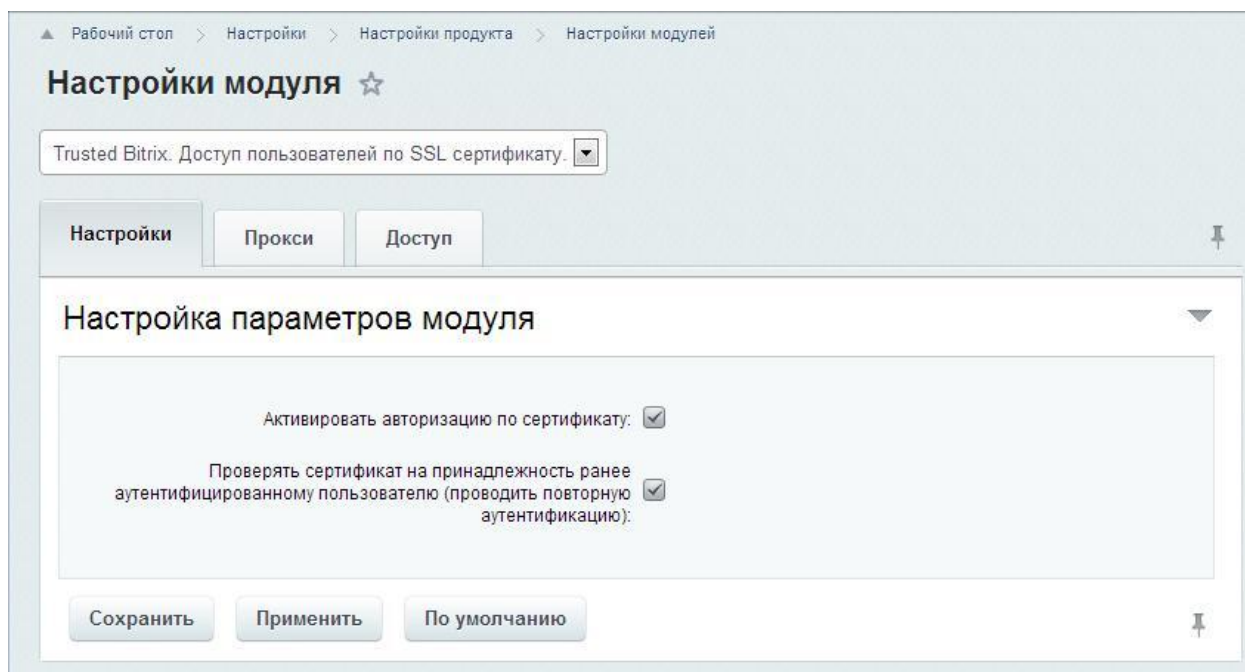


Рис. 5.6. Настройка модуля

На закладке **«Настройки»** устанавливаются общие параметры (рис. 5.6).

Если отключить опцию «**Активировать авторизацию по сертификату**», то можно (по различным причинам) полностью отключить механизм авторизации по сертификату. И, наоборот, включение данной опции дает возможность клиентам авторизоваться по личным сертификатам. По умолчанию после установки модуля данная опция отключена. Подразумевается, что ее включение производится после построения всей инфраструктуры доставки сертификата пользователя до веб-ресурса.

Вторая опция «**Проверять сертификат на принадлежность ранее аутентифицированному пользователю (проводить повторную аутентификацию)**» включает/отключает процесс верификации зарегистрированного пользователя на предмет владения им сертификатом аутентификации, который предъявляется им при просмотре защищенных соответствующим образом ресурсов. В данном случае подразумевается, что пользователь мог быть ранее зарегистрированным на ресурсе, где не требовался его сертификат, и при переходе на защищенный таким образом ресурс можно включить/отключить дополнительную проверку. Если при проверке обнаруживается, что аутентифицированный в системе клиент не владеет предъявленным сертификатом (данный сертификат не присвоен клиенту при настройке), то производится смена пользователя согласно владельцу сертификата.

На закладке «**Прокси**» устанавливаются параметры, описывающие конфигурацию прокси-серверов (рис. 5.7).

▲ Рабочий стол > Настройки > Настройки продукта > Настройки модулей

Настройки модуля ☆

Trusted Bitrix. Доступ пользователей по SSL сертификату. ▾

Настройки Прокси Доступ

Настройка параметров прокси-сервера ▾

Активировать авторизацию через прокси: ☒

Автоматически добавлять неизвестный прокси в список доверенных (без его активации): ☒

Имя HTTP-заголовка, содержащего клиентский сертификат:

Имя HTTP-заголовка, содержащего DNS-имя прокси:

Имя HTTP-заголовка, содержащего результат проверки сертификата на прокси:

Сохранить Применить По умолчанию

Рис. 5.7. Настройки прокси

Опция «**Активировать авторизацию через проху**» позволяет отключить/включить авторизацию по сертификатам, передаваемым всеми прокси-серверами. Данная опция не влияет на авторизацию по сертификату, получаемому напрямую с (backend) сервера, на котором непосредственно эксплуатируется конфигурация 1С-Битрикс.

Опция «**Автоматически добавлять неизвестный прокси в список доверенных (без его активации)**» включает возможность автоматической регистрации нового прокси-сервера в справочнике доверенных прокси-серверов. Это облегчает процесс настройки, когда успешная автоматическая регистрация индицирует правильность настройки прокси-сервера. Данную опцию можно отключить, когда все прокси-сервера уже подключены.

Поле «**Имя HTTP-заголовка, содержащего клиентский сертификат**» содержит имя заголовка HTTP-запроса, в котором прокси-сервер возвращает сертификат клиента.

Поле «**Имя HTTP-заголовка, содержащего DNS-имя прокси**» содержит имя заголовка HTTP-запроса, в котором прокси-сервер возвращает DNS-имя прокси-сервера, которое обычно используется для автоматической регистрации.

Поле «**Имя HTTP-заголовка, содержащего результат проверки сертификата на прокси**» содержит имя заголовка HTTP-запроса, в котором прокси-сервер возвращает результат собственной проверки сертификата клиента. Если данное поле формируется на прокси-сервере, то для успешной авторизации по сертификату результатом проверки должно быть значение «SUCCESS».

Настройка справочника доверенных прокси-серверов

Для настройки авторизации по клиентскому сертификату, доставляемому через прокси-сервер, необходимо заполнить информацию о доверенных прокси-серверах, если они не добавляются автоматически, и активировать их. Все эти действия можно проделать через панель (рабочий стол) администратора в разделе **Настройки → Trusted Bitrix → Справочник Proxu** (рис. 5.8).

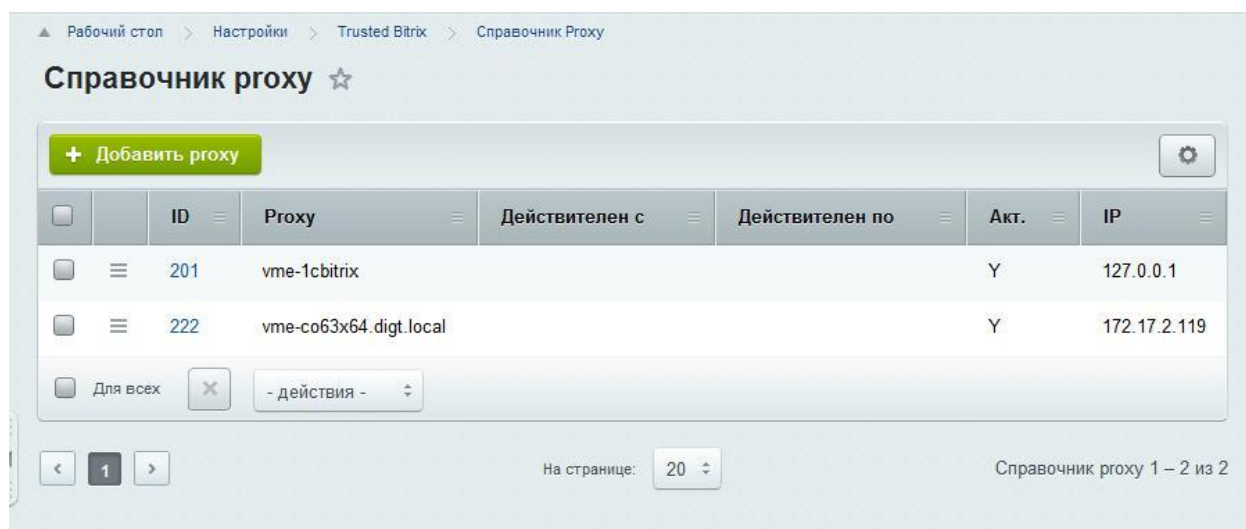


Рис. 5.8. Справочник прокси-серверов

При нажатии на кнопку «**Добавить проху**» или при переходе на редактирование существующего прокси-сервера (пункт «**изменить**» контекстного меню строки, нажатие «мышкой» на id строки или двойное нажатие «мышкой» на строке) открывается форма для заполнения его основных параметров (рис. 5.9):

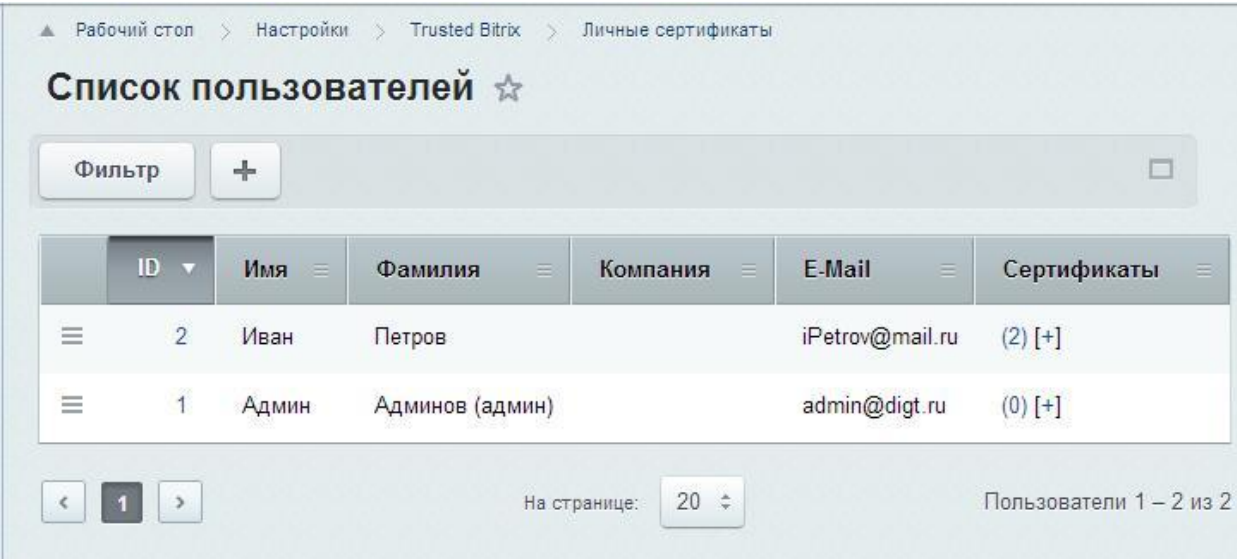
Рис. 5.9. Редактирование данных о прокси-сервере

- наименование (обязательное поле) - «**Имя проху**».
- ip-адрес (обязательное поле) - «**IP**». При обработке поступившего от прокси-сервера сертификата клиента всегда проверяется наличие в справочнике его ip-адреса.
- «**активность элемента**» выставляется, если требуется активировать авторизацию по сертификату, передаваемому через данный прокси-сервер.
- «**Начало активности**» и «**Окончание активности**» задают временной период активности прокси-сервера. Отсутствие значения в этом поле означает отсутствие соответствующего ограничения.

Администрирование хранилища сертификатов

Чтобы начать работу с модулем необходимо иметь сертификаты пользователей для их занесения в хранилище. Пользователи должны быть предварительно зарегистрированы на портале с необходимыми правами.

Управление сертификатами пользователей можно произвести через панель (рабочий стол) администратора в разделе **Настройки** → **Trusted Bitrix** → **Личные сертификаты** (рис. 5.10).



▲ Рабочий стол > Настройки > Trusted Bitrix > Личные сертификаты						
Список пользователей ☆						
Фильтр + <input type="text"/>						
	ID ▾	Имя ≡	Фамилия ≡	Компания ≡	E-Mail ≡	Сертификаты ≡
≡	2	Иван	Петров		iPetrov@mail.ru	(2) [+]
≡	1	Админ	Админов (админ)		admin@digl.ru	(0) [+]
< 1 > На странице: 20 Пользователи 1 – 2 из 2						

Рис. 5.10. Таблица со списком пользователей портала

В колонке Сертификаты имеются ссылка на список привязанных к пользователю сертификатов (обозначается их количеством) и ссылка на добавление нового сертификата. Переход по второй ссылке приводит к открытию формы добавления нового сертификата.

На форме добавления сертификата указывается, кому назначается данный сертификат, и предлагается выполнить выбор файла *.cer через диалог выбора файла (рис. 5.11). При включении отметки **Активация сертификата**, добавленный элемент немедленно активируется, и пользователь может использовать загруженный сертификат для входа. При отсутствии отметки сертификат будет добавлен в хранилище, но вход по нему будет временно блокирован.

Рис. 5.11. Форма загрузки сертификата аутентификации

С каждым пользователем в хранилище может быть связано сколько угодно сертификатов. По любому из них, при соблюдении условий активности и валидации он может авторизоваться на сайте. **В данной бесплатной версии модуля могут использоваться сертификаты, сгенерированные на RSA-алгоритмах. ГОСТ-сертификаты не будут загружаться в хранилище сертификатов.**

⚠ Важно! При удалении пользователя данные из хранилища сертификатов (сертификаты привязанные к ID пользователя) безвозвратно удаляются. При добавлении пользователя для него необходимо загрузить необходимые сертификаты. Также устраняются сертификаты, не привязанные к пользователю и дублирующие друг друга.

Сертификаты пользователя отображаются списком (рис. 5.12), в котором также отображается их статус.

▲ Рабочий стол

Список личных сертификатов пользователя (2, vib) Петров Иван ☆

[Вернуться к списку](#)
[+ Добавить сертификат](#)
⚙

<input type="checkbox"/>	ID	Владелец	Инд. валидности	Действителен с	Действителен по	Акт.	Издатель
<input type="checkbox"/>	199	TEST3 - Vladimir Baykov (Владимир Байков)	●	14.05.2012 17:00:55	04.10.2014 11:09:41	Y	Test Center CRYPTO-PRO
<input type="checkbox"/>	203	vbl_RSA_01	●	13.06.2012 11:05:07	13.06.2013 11:15:07	Y	CT RSA Test CA 2

☐ Для всех

< 1 >
На странице: 20
Список личных сертификатов пользователя 1 – 2 из 2

Рис. 5.12. Список сертификатов, привязанных к пользователю

Каждый из сертификатов аутентификации пользователей имеет определенные свойства. Эти свойство можно посмотреть и отредактировать вручную (рис. 5.13).

Рабочий стол

Редактирование информации о сертификате пользователя (2, vib) Петров Иван ☆

[Вернуться к списку сертификатов пользователя](#)

Редактирование данных о сертификате

Редактирование данных о сертификате

Имя (владелец сертификата):

Активность элемента: ☒

Начало активности:

Окончание активности:

Издатель сертификата:

Хэш сертификата:

Серийный номер сертификата:

Назначение сертификата:

Тело сертификата:

-----BEGIN CERTIFICATE-----
MIIFWTCCBEGgAwIBAgIKYRTJC
QAAAAAAjANBgkqhkiG9w0BAQ
UFADBmMR4wHAYJ
KoZlhvdNAQkBFg9pbmZvQHRyd
XN0ZWQucnUxOzA1BjNVBAYTA
IJVMRwwGgYDVQQKQ
ExNDaWZyb3ZpZSB1ZWhub2xvZ
2lpMRkwFwYDVQQDExB0VCBSU
0EgVGVzdCBDQSAy

Сертификат действителен с:

Сертификат действителен по:

[Сохранить](#) [Применить](#) [Отменить](#)

*Поля, обязательные для заполнения.

Рис. 5.13. Форма просмотра/редактирования свойств сертификата

Усиленная авторизация по сертификату

В целях повышения безопасности в процессе аутентификации каждому пользователю в отдельности можно отменить стандартную аутентификацию по логину/паролю, предоставив только право аутентификации по сертификату. Это действие производится через панель (рабочий стол) администратора в разделе **Настройки → Пользователи → Список пользователей**, где необходимо выбрать из списка требуемого пользователя и вызвать его форму редактирования (рис. 5.14).

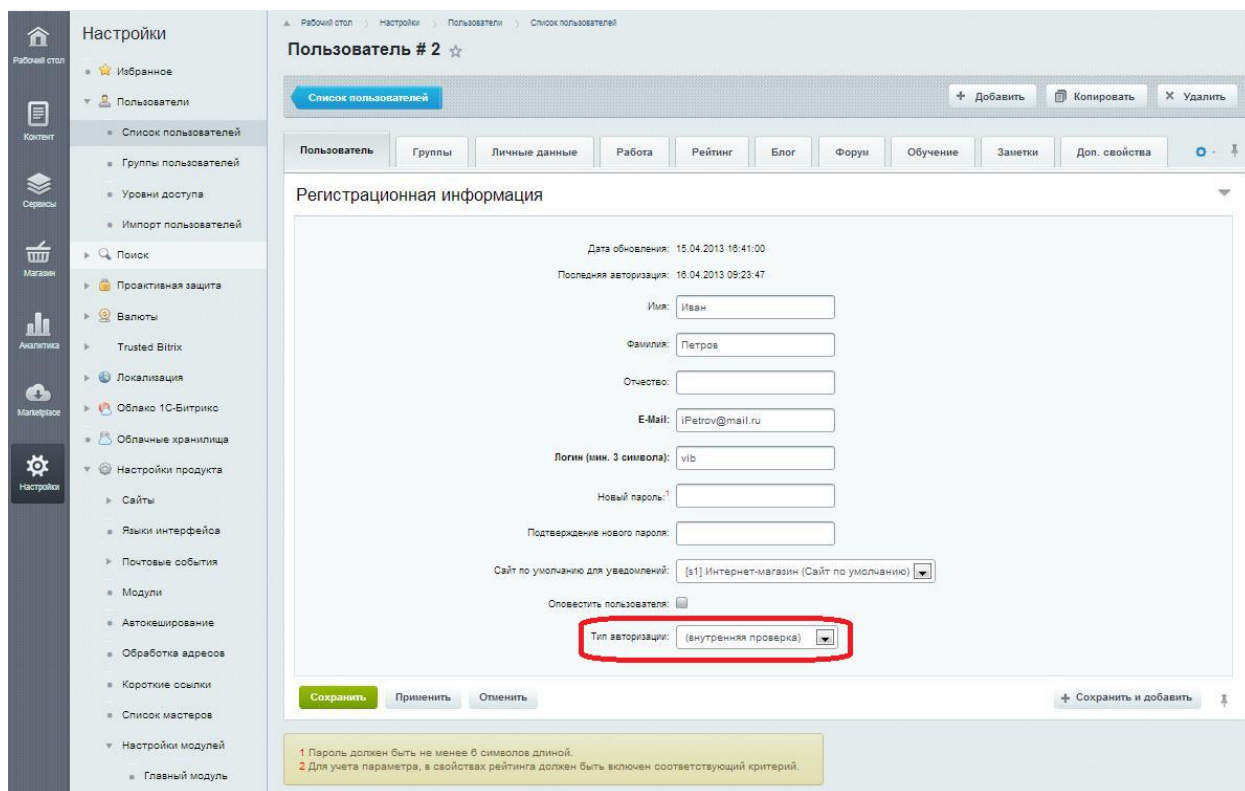


Рис 5.14. Изменение информации о пользователе.

В поле «Тип авторизации» необходимо сменить значение «(внутренняя проверка)» на значение «По сертификату (trusted bitrix)».

После сохранения измененных данных этот пользователь не сможет авторизоваться только по сертификату.

Чтобы поле «Тип авторизации» было доступно, необходимо в настройке формы добавить его отображение на вкладке «Пользователь» (рис. 5.15).

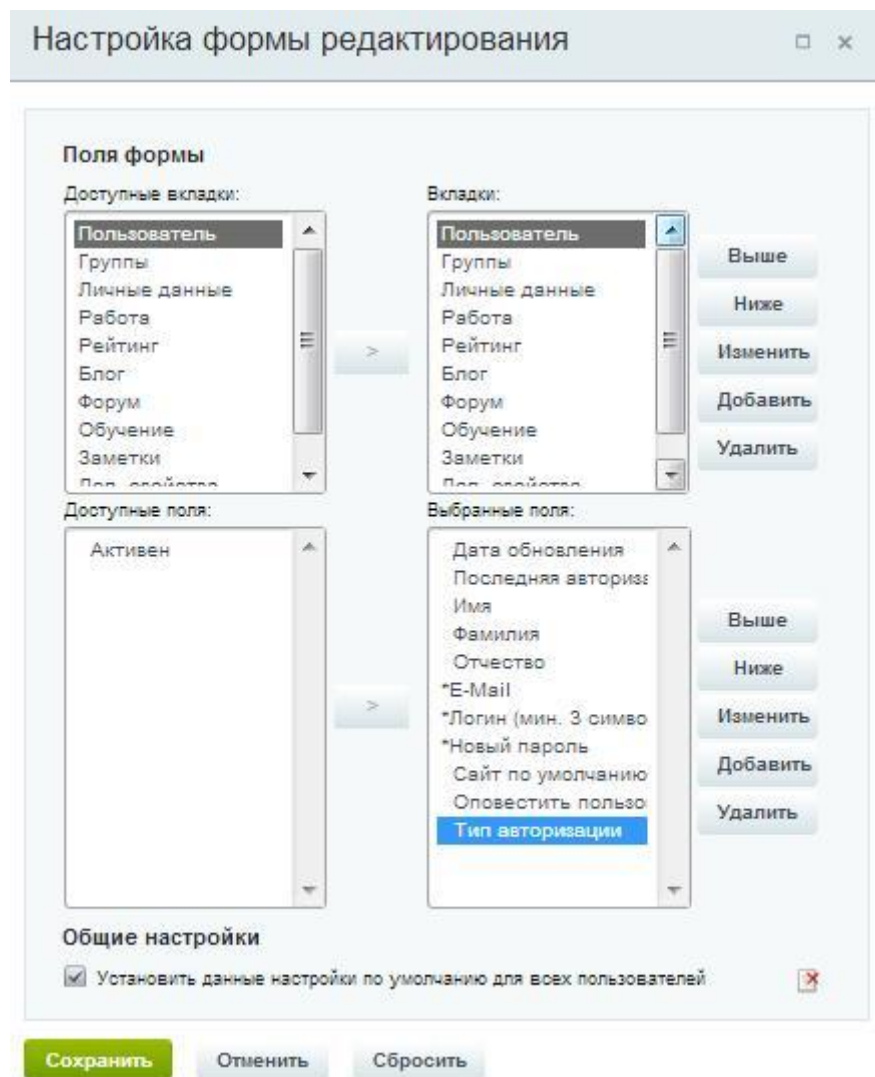


Рис. 5.15. Настройка формы редактирования пользователя.

Удаление модуля

Удаление модуля производится через панель (рабочий стол) администратора в разделе **Marketplace** → **Установленные решения** (рис. 5.16).

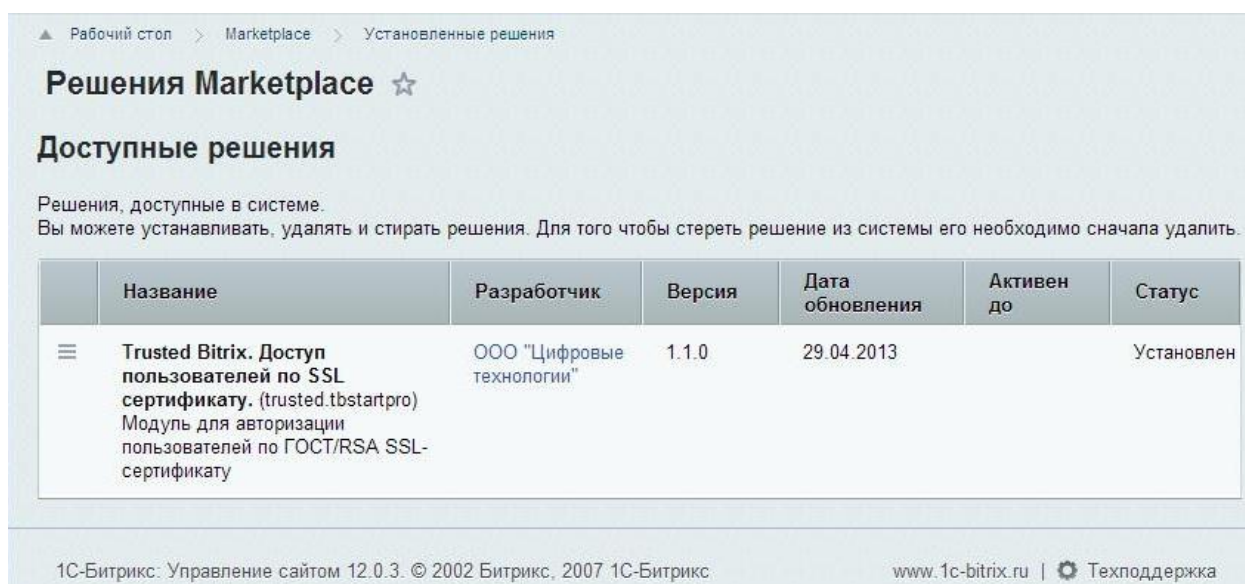


Рис. 5.16. Список установленных решений

Из контекстного меню элемента списка нужно выбрать команду **Удалить**, после чего запустится мастер удаления. На первом шаге мастера можно выбрать: удалить или оставить инфоблоки с данными о сертификатах и пользователях (рис. 5.17).



Рис. 5.17. Мастер удаления модуля

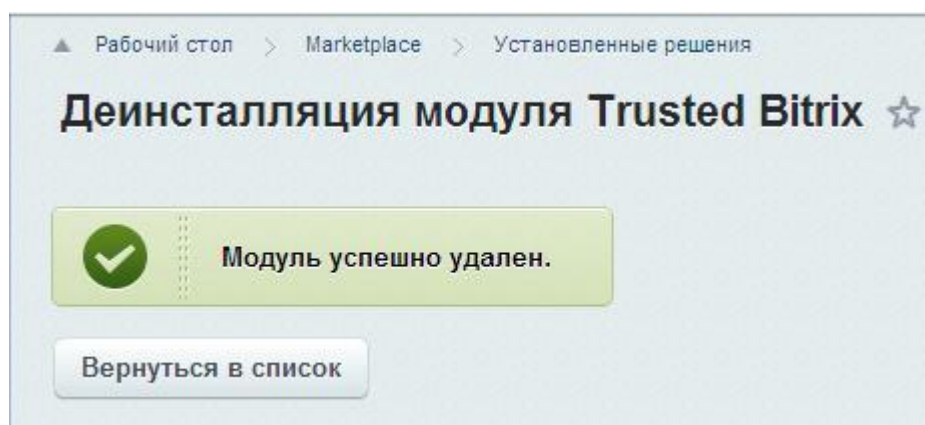


Рис. 5.18. Завершение работы мастера удаления модуля

Раздел 6. Настройка TLS-соединений

Для обеспечения в приложениях 1С-Битрикс авторизации клиентов по сертификату необходимо настроить передачу этих сертификатов от прокси-сервера до веб-окружения 1С-Битрикс (до Apache http-сервера) или при отсутствии прокси-серверов получение их непосредственно от главного Apache http-сервера. Далее даются рекомендации по настройке прокси-серверов на примере [Apache](#) http-сервера и [Nginx](#). Также описывается настройка TLS для Apache-сервера при взаимодействии с клиентом без схемы проксирования запросов.

Генерация сертификатов через OpenSSL

С подробным описанием генерации ГОСТовых сертификатов для использования их в TLS можно ознакомиться в [документации](#) к продукту [Trusted TLS](#). Ниже приводится краткое описание для создания сертификатов на базе RSA-ключей с использованием утилиты openssl.

После установки OpenSSL на машину, где будут генерироваться сертификаты для клиентского рабочего места и сервера, необходимо внести в файл конфигурации (/etc/ssl/openssl.cnf) следующие изменения:

[ca]

default_ca = CA_default

[CA_default]

dir = . # Это каталог для работы с ssl
certs = \$dir/ssl.crt # Каталог сохранения сертификатов
crl_dir = \$dir/ssl.crl # Каталог листов "отзыва подписей"
database = \$dir/index.txt # index file для индексирования запросов на подпись
new_certs_dir = \$dir/ssl.crt # Каталог записи новых сертификатов
certificate = \$dir/company-ca.pem # Корневой сертификат
serial = \$dir/serial # текущий серийный номер запроса
crl = \$dir/ssl.crl/company-ca-crl.pem # Текущий лист отзывов подписей
private_key = \$dir/ssl.key/company-ca.key # Секретный ключ для основного сертификата
RANDFILE = \$dir/ssl.key/.rand #
default_days = 365 # Количество дней действия сертификата
default_md = default

x509_extensions = usr_cert # Расширения, добавляемые в сертификат
copy_extensions = copy # Копирование расширений из запроса в сертификат

[policy_anything]

countryName = optional
stateOrProvinceName = optional
localityName = optional
organizationName = optional
organizationalUnitName = optional

commonName = supplied
emailAddress = optional

[req]
Секция основных опций
default_bits = 1024
distinguished_name = req_distinguished_name
prompt = yes
x509_extensions = v3_ca # Расширения, добавляемые в самоподписываемый сертификат
Отключаем v3_req. Включаем управление подключениями секций в запросе
#req_extensions = v3_req # Расширения, добавляемые в запрос на сертификат

[req_distinguished_name]
В данной секции вносим свои данные,
которые будут использоваться по умолчанию
при генерации запроса на создание сертификат.
countryName = Country Name (2 letter code)
countryName_default = RU
countryName_min = 2
countryName_max = 2
stateOrProvinceName = State or Province Name (full name)
stateOrProvinceName_default = Russian
localityName = Locality Name (eg, city)
localityName_default = NEW_CITY
0.organizationName = Organization Name (eg, company)
0.organizationName_default = MY_COMPANY
organizationalUnitName = Organizational Unit Name (eg, section)
organizationalUnitName_default = ADMIN

[usr_cert]
Расширения, добавляемые при подписании запроса на сертификат
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer

basicConstraints=CA:FALSE
nsComment = "OpenSSL Generated Certificate "
#nsCertType = client, email, server

Типичное использование для клиентского сертификата
keyUsage = nonRepudiation, digitalSignature, keyEncipherment

[v3_req_server]
keyUsage = nonRepudiation, digitalSignature, keyEncipherment, dataEncipherment
extendedKeyUsage = serverAuth # заменяет extendedKeyUsage = 1.3.6.1.5.5.7.3.1

[v3_req_client]
keyUsage = nonRepudiation, digitalSignature, keyEncipherment, dataEncipherment

extendedKeyUsage = clientAuth # заменяет extendedKeyUsage = 1.3.6.1.5.5.7.3.2

[v3_ca]

subjectKeyIdentifier=hash

authorityKeyIdentifier=keyid:always,issuer:always

basicConstraints = CA:true

#nsCertType = sslCA, emailCA

Далее необходимо создать корневой сертификат. Для удобства, можно перейти в каталог с конфигурацией Apache, где располагаются подкаталоги с искомыми сертификатами:

cd /usr/local/etc/apache

Корневой сертификат (сертификат УЦ) является корнем дерева подписей. Секретный ключ (нужен для того, чтобы можно было воспользоваться вашим корневым сертификатом для подписи остальных) и сертификат создаются одной командой:

openssl req -config /etc/ssl/openssl.cnf -new -x509 -keyout ssl.key/company-ca.pem -out company-ca.pem -days 3650

Для удобства генерации прочих сертификатов можно снять пароль на приватный ключ:

openssl rsa -in ssl.key/company-ca.pem -out ssl.key/company-ca.key

Ссылка на файл company-ca.key используется в конфигурационном файле openssl.cnf.

При генерации будет запрошен пароль - введите и запомните его. Все остальные поля можно заполнить по своему усмотрению. В результате проделанных действий должен быть создан самоподписанный корневой сертификат.

Следует создать два файла с некоторой индексной информацией. Создайте индексный файл (ключевое слово database из openssl.cnf):

touch index.txt

Создайте файл серийных номеров (ключевое слово serial из openssl.cnf):

echo '01' > serial

Этот файл должен содержать две цифры (обязательно). Если вы ранее не создавали никаких сертификатов кроме корневого, файл должен содержать 01.

Далее требуется создать сертификат сервера. Создание сертификатов сервера состоит из процедуры создания запроса на подпись, а затем подписи этого запроса сертификатом УЦ. Создайте запрос на подпись нового сертификата и секретный ключ к нему:

```
# openssl req -config /etc/ssl/openssl.cnf -new -reqexts v3_req_server -keyout  
ssl.key/server-rsa.pem -out ssl.csr/server-rsa.pem
```

Обратите внимание на подключение секции **v3_req_server** при генерации серверного ключа. В указанной секции указывается назначение сертификата – «Подтверждение подлинности сервера». Вводя данные, учтите, что поле Common Name должно содержать определённое доменное имя (FQDN) того сайта, где вы будете использовать https-протокол, чтобы браузеры не выдавали предупреждения о несоответствии имени. Если необходимо, можно снять пароль с ключа:

```
# openssl rsa -in ssl.key/server-rsa.pem -out ssl.key/server-rsa.key
```

Подпишите запрос (подписание запроса и есть создание нового сертификата) ранее сформированным корневым сертификатом:

```
# openssl ca -config /etc/ssl/openssl.cnf -policy policy_anything -out ssl.crt/server-  
rsa.pem -infiles ssl.csr/server-rsa.pem
```

Подготовьте сертификат к использованию:

```
# openssl x509 -in ssl.crt/server-rsa.pem -out ssl.crt/server-rsa.crt
```

Создание клиентского сертификата производится аналогично с учетом того, что при создании запроса производится подключение секции **v3_req_client**, которая указывает назначение сертификата – «Подтверждение подлинности клиента»:

```
# openssl req -config /etc/ssl/openssl.cnf -new -reqexts v3_req_client -keyout  
ssl.key/client-rsa.pem -out ssl.csr/client-rsa.pem
```

```
# openssl rsa -in ssl.key/client-rsa.pem -out ssl.key/client-rsa.key
```

```
# openssl ca -config /etc/ssl/openssl.cnf -policy policy_anything -out ssl.crt/client-  
rsa.pem -infiles ssl.csr/client-rsa.pem
```

```
# openssl x509 -in ssl.crt/client-rsa.pem -out ssl.crt/client-rsa.crt
```

Для установки сертификатов/ключей на клиентские рабочие места необходимо экспортировать их в формат PKCS#12. Для этого необходимо выполнить следующие операции. Во-первых, связать вместе сертификат УЦ (необязательно), клиентский сертификат и приватный ключ:

Для Unix -

```
# cat ./company-ca.pem ./ssl.crt/client-rsa.pem ./ssl.key/client-rsa.pem >  
./ssl.pkcs12/client-rsa.pem
```

Для Windows -

```
> type .\company-ca.pem .\ssl.crt\client-rsa.pem .\ssl.key\client-rsa.pem >  
.\ssl.pkcs12\client-rsa.pem
```

И, окончательно,

```
# openssl pkcs12 -export -in .\ssl.pkcs12\client-rsa.pem -out .\ssl.pkcs12\client-rsa.p12 -  
name "Client Certificate - client-rsa"
```

Файл `.\ssl.pkcs12\client-rsa.p12` можно передать клиенту для установки его на рабочем месте.

Настройка TLS на сервере Apache

В данном разделе рассматривается вопрос конфигурирования TLS-соединения напрямую до приложения 1С-Битрикс (в рамках Apache веб-сервера, входящего в веб-окружение).

Ниже приводится пример настройки виртуального сервера, функционирующего на Apache http-сервере. В файле конфигурации (обычно `httpd.conf`) необходимо указать директивы загрузки следующих модулей:

модуля для стандартной поддержки SSL

```
LoadModule ssl_module modules/mod_ssl.so
```

или модуля для добавления в SSL поддержки ГОСТ-криптографии с использованием СКЗИ КриптоПРО CSP 3.6 (из продукта «Trusted TLS»)

```
LoadModule ssl_module modules/mod_digt_tls.so
```

и также необходимо подключить файл настройки SSL, например,

```
Include conf/extra/httpd-ssl.conf
```

В подключаемом файле конфигурации (`conf/extra/httpd-ssl.conf`) настраиваем виртуальный хост на прием SSL-соединений:

```
Listen 4434  
<VirtualHost *:4434>  
DocumentRoot "E:\bitrix-db\SC_12.0.3"  
  
ServerName vme-1cbitrix:4434  
ServerAdmin admin@example.com  
ErrorLog logs/sc1203-error.log  
TransferLog logs/sc1203-access.log  
  
SSLEngine on  
SSLCipherSuite ALL:+HIGH  
  
SSLCertificateFile "conf/server-gost.cer"  
SSLCertificateKeyFile "conf/server-gost.cer"  
  
SSLCertificateFile conf/server-rsa.pem
```

```

SSLCertificateKeyFile conf/server-rsa.pem

#SSLCACertificatePath "conf/ssl.crt"
SSLCACertificateFile "conf/ssl.crt/ca-bundle.crt"

#SSLCARevocationPath "conf/ssl.crl"
#SSLCARevocationFile " conf/ssl.crl/ca-bundle.crl"

<Location />
    SSLVerifyClient none
#    SSLVerifyDepth 3
</Location>

<Location /bitrix/admin>
    SSLVerifyClient optional
</Location>
<Location /auth>
    SSLVerifyClient optional
</Location>

SSLOptions +FakeBasicAuth +ExportCertData +StrictRequire
<FilesMatch "\.(cgi|shtml|phtml|php)$">
    SSLOptions +StdEnvVars
</FilesMatch>

BrowserMatch ".*MSIE.*" \
    nokeepalive ssl-unclean-shutdown \
    downgrade-1.0 force-response-1.0

CustomLog logs/ssl_request.log \
    "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"

</VirtualHost>

```

В приведенном примере конфигурации включается прослушивание на порту 4434 (директива – Listen 4434) защищенного соединения (директива - SSLEngine on) на двух ключах/сертификатах: ГОСТ-ключе (директивы - SSLCertificateFile "conf/server-gost.cer" и SSLCertificateKeyFile "conf/server-gost.key") и RSA-ключе (директивы - SSLCertificateFile conf. server-rsa.pem и SSLCertificateKeyFile conf/server-rsa.pem). В случае использования модуля **mod_digt_tls.so** файл conf/server-gost.key должен быть идентичен файлу conf/server-gost.cer.

Список сертификатов УЦ, сужающий список доступных к ресурсам веб-сервера сертификатов клиентов, задается в одном файле (директива - SSLCACertificateFile "conf/ssl.crt/ca-bundle.crt"). При необходимости можно подключить список отозванных сертификатов (СОС) для проверки статуса клиентских сертификатов, используя вариант хранения его одним файлом (директива - SSLCARevocationFile " conf/ssl.crl/ca-bundle.crl").

В приведенном примере конфигурации директива **SSLVerifyClient** устанавливает для трех ресурсов различные требования на предъявление сертификата клиентом. В первом случае (директива - SSLVerifyClient none) не требуется предъявление сертификата, что распространяется на все нижележащие ресурсы за исключением ресурсов **/bitrix/admin** и **/auth**, на которых запрашивается ввод сертификата клиентом. На этих двух ресурсах предполагается использование сертификата клиента и в случае его предъявления авторизации клиента. На этих двух ресурсах может располагаться стандартный скрипт авторизации из конфигурации 1С-Битрикс, на который могут быть переадресованы запросы от расширенных скриптов авторизации для проведения внутренней авторизации.

При использовании модуля **mod_digt_tls.so** из программного продукта «Trusted TLS 2.2» необходимо произвести следующие действия с файлами Apache httpd-сервера:

- На платформе Windows требуется заменить в каталоге веб-окружения apache2\bin\ файлы httpd.exe, libhttpd.dll, а также выложить в каталог веб-окружения apache2\modules файл mod_digt_tls.so.
- На платформе Linux требуется выложить в каталог modules Apache веб-сервера файл mod_digt_tls.so.

Настройка TLS на прокси-сервере Nginx

Ниже приводится пример настройки виртуального сервера на Nginx

```
server {
    listen          4431;
    ssl             on;

    keepalive_timeout 70;
    keepalive_requests 150;

    #ssl_protocols    SSLv3 TLSv1;
    #ssl_ciphers      AES128-SHA:AES256-SHA:RC4-SHA:DES-CBC3-
    SHA:RC4-MD5;
    ssl_certificate   /etc/nginx/ssl/cert.pem;
    ssl_certificate_key /etc/nginx/ssl/cert.pem;
    ssl_session_cache shared:SSL:10m;
    ssl_session_timeout 10m;

    server_name_in_redirect off;

#    proxy_set_header    HTTPS          YES;
#    proxy_set_header    X-Real-IP      $remote_addr;
#    proxy_set_header    X-Forwarded-For $proxy_add_x_forwarded_for;
#    proxy_set_header    X-Forwarded-Host $host:4431;

    ssl_verify_client    optional;
    proxy_set_header     X-Forwarded-Server $host;
    proxy_set_header     Forwarded-SSL-Client-Cert $ssl_client_cert;
    proxy_set_header     Client_Verified_on_Proxy $ssl_client_verify;
```

```

ssl_client_certificate /etc/nginx/ssl/ca_bundle.crt;


location / {
    proxy_pass http://vme-1cbitrix:6451;
}

# include bx/conf/bitrix.conf;
# include bx/server_monitor.conf;
}

```

В примере конфигурации виртуальный сервер настраивается на порт 4431 (директива - listen 4431;) для приема https-запросов (директива – *ssl on*;). Поднятие защищенного канала осуществляется на ключе и сертификате сервера, размещенных в файле /etc/nginx/ssl/cert.pem (директивы - *ssl_certificate* и *ssl_certificate_key*).

Требование необязательного предъявления сертификата клиентом описывается директивой **ssl_verify_client optional**.

 **Важно!** В отличие от Apache http-сервера при настройке Nginx данная директива может быть установлена только от корня виртуального сервера.

Передача сертификата клиента настраивается через заголовок **Forwarded-SSL-Client-Cert** (директива - proxy_set_header Forwarded-SSL-Client-Cert \$ssl_client_cert;).

Передача статуса проверки сертификата настраивается через заголовок **Client_Verified_on_Proxy** (директива - proxy_set_header Client_Verified_on_Proxy \$ssl_client_verify;).

Передача DNS-имени прокси-сервера настраивается через заголовок **X-Forwarded-Server** (директива - proxy_set_header X-Forwarded-Server \$host;).

Перечисленные имена формируемых заголовков используются при [настройке параметров модуля](#) TrustedBitrix.

И, окончательно, само проксирование https-запросов в данном примере осуществляется через использование директивы **proxy_pass**.

Настройка TLS на прокси-сервере Apache

В данном разделе рассматривается вопрос конфигурирования TLS-соединения с использованием схемы проксирования запросов до приложения 1С-Битрикс.

Ниже приводится пример настройки виртуального сервера, функционирующего на Apache http-сервере. В файле конфигурации (обычно **httpd.conf**) необходимо указать директивы загрузки следующих модулей:

```

LoadModule headers_module modules/mod_headers.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so

```

модуля для стандартной поддержки SSL

```
LoadModule ssl_module modules/mod_ssl.so
```

или модуля для добавления в SSL поддержки ГОСТ-криптографии (при использовании продукта «Trusted TLS»)

```
LoadModule ssl_module modules/mod_digt_tls.so
```

и также необходимо подключить файл настройки SSL

```
Include conf/extra/httpd-ssl.conf
```

В подключаемом файле конфигурации (например, conf/extra/**httpd-ssl.conf**) настраиваем виртуальный хост на прием SSL-соединений:

```
Listen 4433
```

```
<VirtualHost *:4433>
```

```
ServerName vme-1cbitrix:4433
```

```
ServerAdmin admin@example.com
```

```
ErrorLog logs/sc1203-error.log
```

```
TransferLog logs/sc1203-access.log
```

```
SSLEngine on
```

```
SSLCipherSuite ALL:+HIGH
```

```
SSLCertificateFile "conf/server-gost.cer"
```

```
SSLCertificateKeyFile "conf/server-gost.key"
```

```
SSLCertificateFile conf/server-rsa.pem
```

```
SSLCertificateKeyFile conf/server-rsa.pem
```

```
#SSLCACertificatePath "conf/ssl.crt"
```

```
SSLCACertificateFile "conf/ssl.crt/ca-bundle.crt"
```

```
#SSLCARevocationPath "conf/ssl.crl"
```

```
#SSLCARevocationFile " conf/ssl.crl/ca-bundle.crl"
```

```
<Location />
```

```
    SSLVerifyClient none
```

```
#    SSLVerifyDepth 3
```

```
</Location>
```

```
<Location /bitrix/admin>
```

```
    SSLVerifyClient optional
```

```
</Location>
```

```
<Location /auth>
```

```
    SSLVerifyClient optional
```

</Location>

```
BrowserMatch ".*MSIE.*" \
    nokeepalive ssl-unclean-shutdown \
    downgrade-1.0 force-response-1.0
```

```
CustomLog logs/ssl_request.log \
    "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
```

```
<IfModule mod_proxy.c>
    ProxyPass / http://localhost:6451/
    ProxyPassReverse / http://localhost:6451/
</IfModule>
```

```
<IfModule mod_headers.c>
    RequestHeader set Forwarded-SSL-CLIENT-CERT "%{SSL_CLIENT_CERT}s"
    RequestHeader set Client_Verified_on_Proxy "%{SSL_CLIENT_VERIFY}s"
</IfModule>
```

</VirtualHost>

Описание используемых директив аналогично рассмотренному в разделе [настройки TLS на сервере Apache](#). Отличие состоит только во включении механизма проксирования запросов и определения необходимых заголовков запросов.

Директива

```
RequestHeader set Forwarded-SSL-CLIENT-CERT "%{SSL_CLIENT_CERT}s"
```

формирует заголовок запроса **Forwarded-SSL-Client-Cert** для передачи в нем сертификата клиента.

Директива

```
RequestHeader set Client_Verified_on_Proxy "%{SSL_CLIENT_VERIFY}s"
```

формирует заголовок запроса **Client_Verified_on_Proxy** для передачи в нем статуса проверки сертификата. Передача DNS-имени прокси-сервера осуществляется директивой **ProxyPass**, которая автоматически формирует заголовок **X-Forwarded-Server**.

Перечисленные имена формируемых заголовков используются при [настройке параметров модуля](#) TrustedBitrix.

Проксирование https-запросов осуществляется через простое использование директивы **ProxyPass** (совместно с **ProxyPassReverse**) на сервер **localhost** на порт 6451, на котором развернуто веб-окружение 1С-Битрикс и один из его продуктов.

Раздел 7. Настройки браузеров

⚠ Важно! В данный момент разработчиками в решении по аутентификации и защите информационного канала поддерживаются браузеры – Microsoft Internet Explorer, Srware Iron и Google Chrome. Если на клиентском рабочем месте установлены эти браузеры и вход на портал осуществляется по сертификату, то решение гарантирует корректную работу TLS по ГОСТ алгоритмам, предоставляемым СКЗИ КриптоПро CSP. Обычный вход на портал (по логину/паролю) пользователь может осуществлять, используя любой из доступных браузеров.

Настройка браузера Microsoft Internet Explorer

Для настройки браузера Microsoft Internet Explorer выполните следующее (инструкции приведены для 8 версии браузера):

- Найдите в строке меню закладку **Сервис** и выберите пункт **Свойства обозревателя**. На экране должно появиться окно *Свойства обозревателя*.
- Перейдите в окне *Свойства обозревателя* на закладку *Дополнительно* и убедитесь, что в списке параметров браузера отмечен пункт **TLS 1.0** (рис. 7.1):

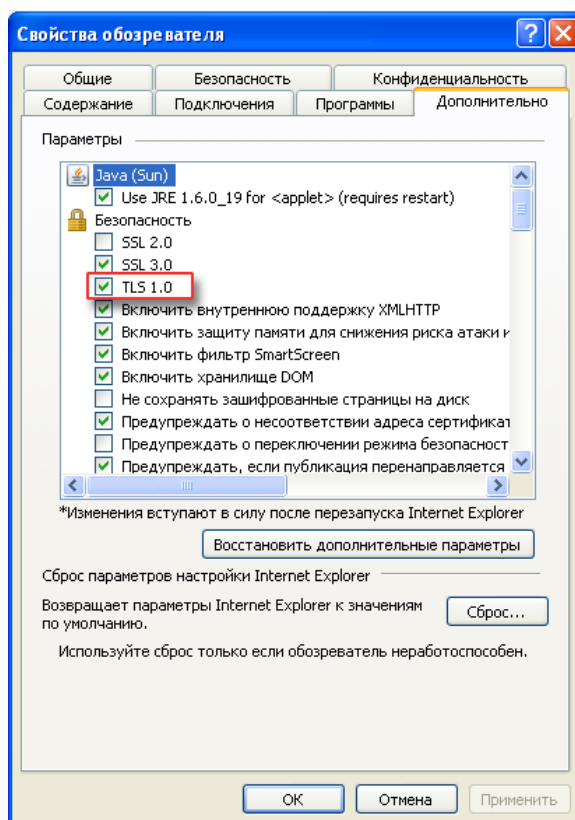



Рис. 7.1 Включение поддержки TLS в браузере

- Перейдите на закладку **Безопасность** и выберите раздел **Надежные узлы** . Нажмите на кнопку **Узлы** (рис. 7.2) и в появившемся диалоге *Надежные узлы* добавьте адрес того веб-ресурса на который выполняется вход по сертификату (рис. 7.3).

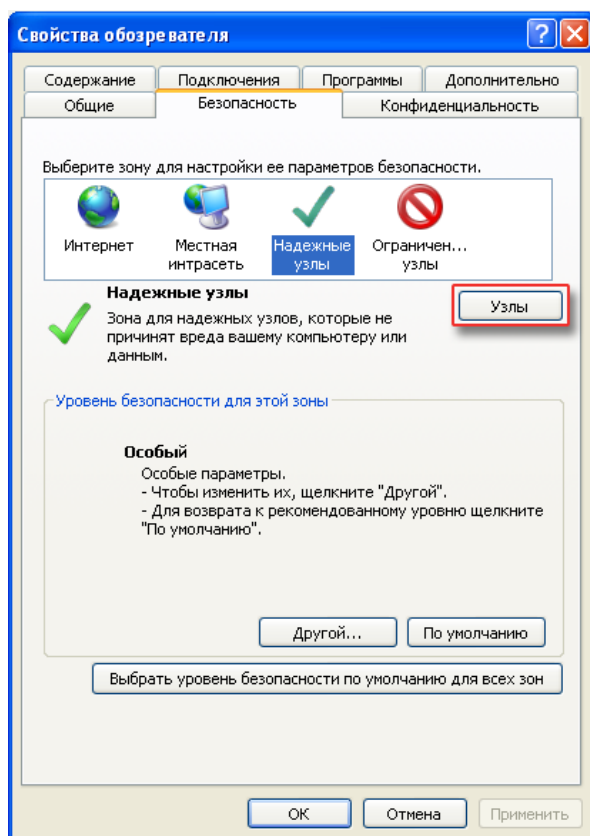


Рис. 7.2 Доступ к диалогу установки доверенных узлов

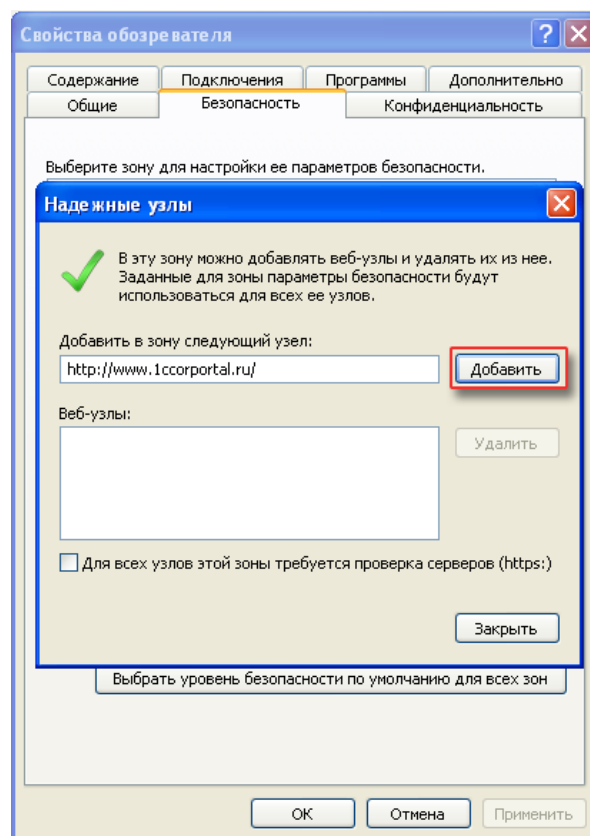


Рис. 7.3 Добавление ссылки на портал в доверенные узлы

Настройка браузера Google Chrome

Для настройки браузера Google Chrome выполните следующее (инструкции приведены для 5 версии браузера):

- Откройте меню **Настройка** и выберите пункт **Параметры** (рис. 7.4)

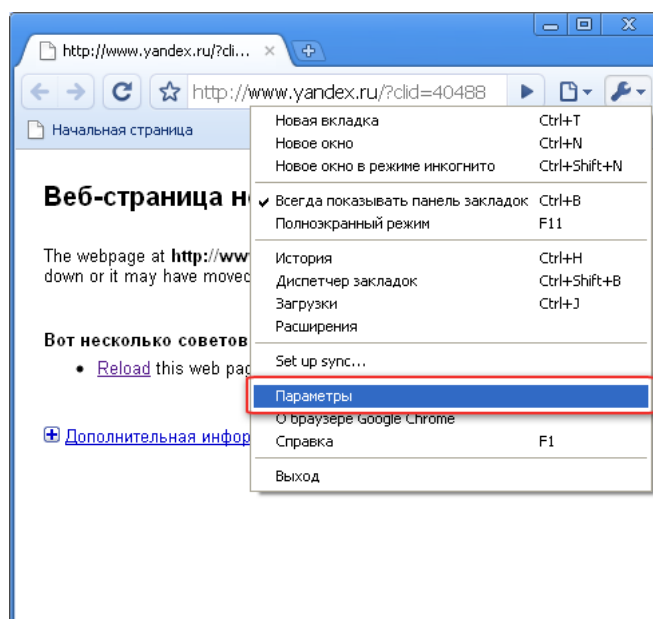


Рис. 7.4 Включение поддержки TLS в браузере

- В открывшемся окне *Параметры Google Chrome* на вкладке **Расширенные** включить режим **Использовать SSL 2.0** (Рис. 7.5):

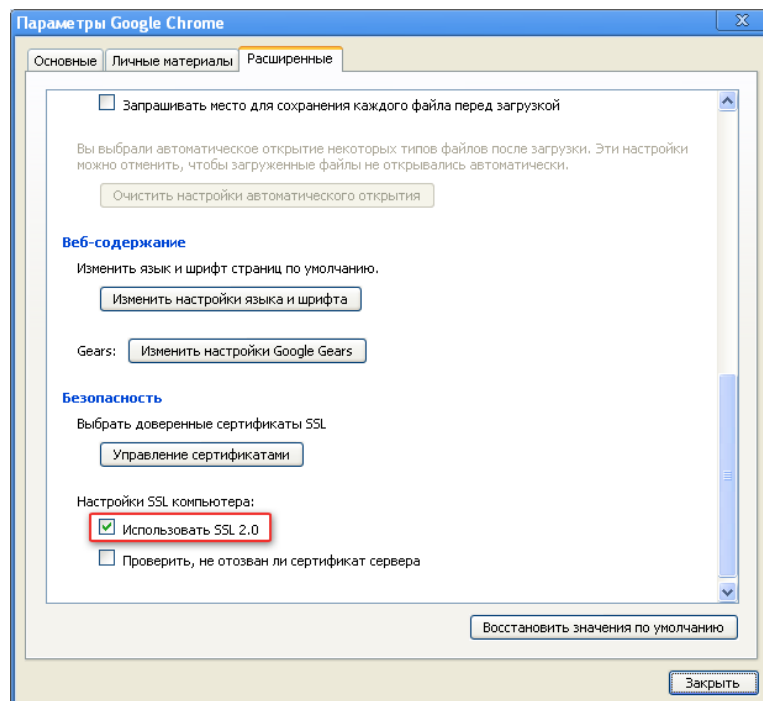


Рис. 7.5 Включение поддержки TLS в браузере

Для настройки браузера Google Chrome более новой версии (например, для версии 25.x.x.x) и для поддержки ГОСТ-алгоритмов достаточно в строке запуска браузера указать опцию `--use-system-ssl`.

⚠ Важно! В браузере Google Chrome версии 26.0.1410.43 опция `--use-system-ssl` перестала работать.

Для экспорта ключа и сертификатов в настройках браузера (в.15.0) необходимо выбрать пункт «Управление сертификатами» (рис. 7.6).

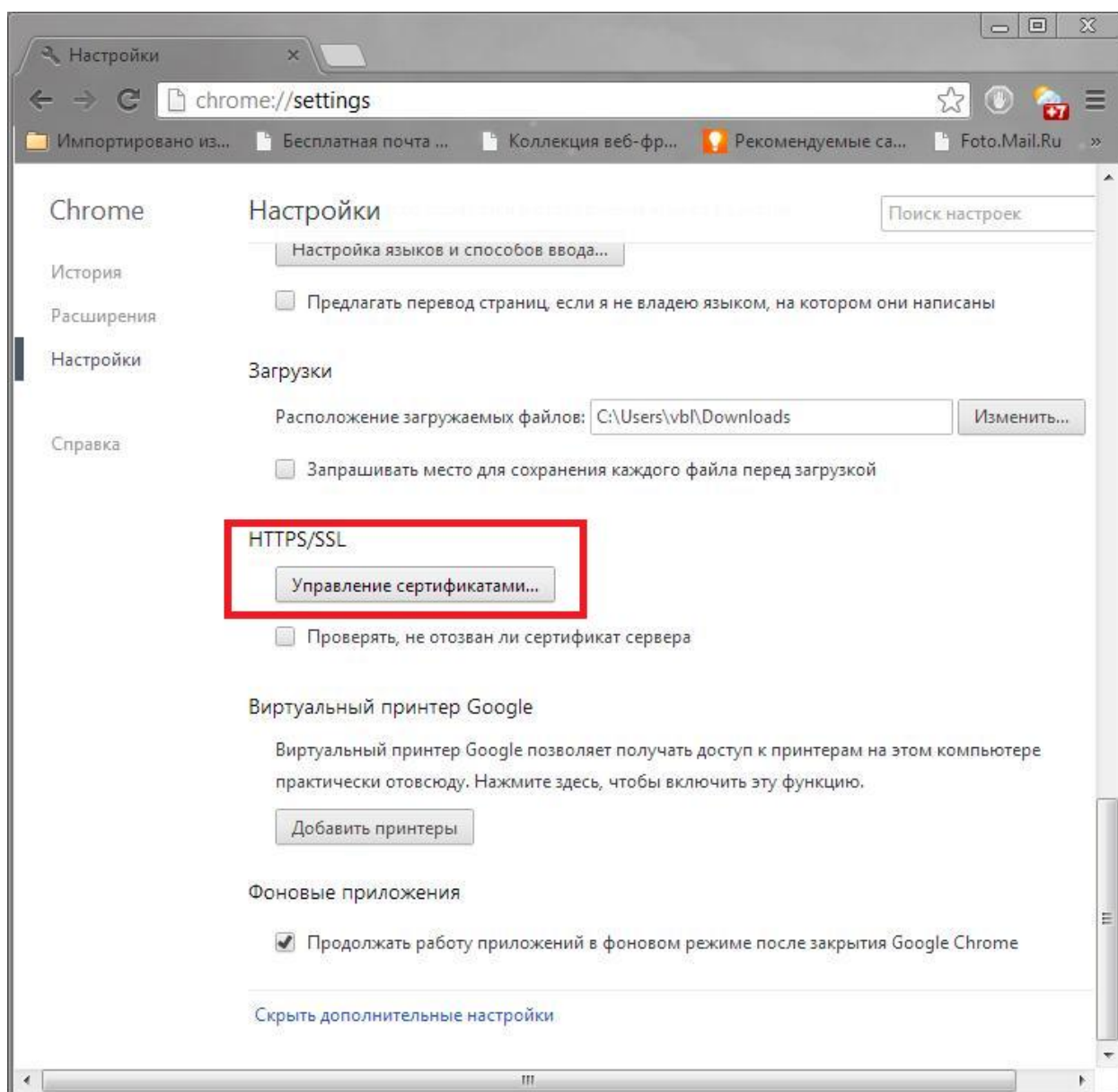


Рис. 7.6. Настройки браузера Google Chrome.

В появившемся окне на закладке «Личные» нужно выбрать кнопку «Импорт...» (рис. 7.7).

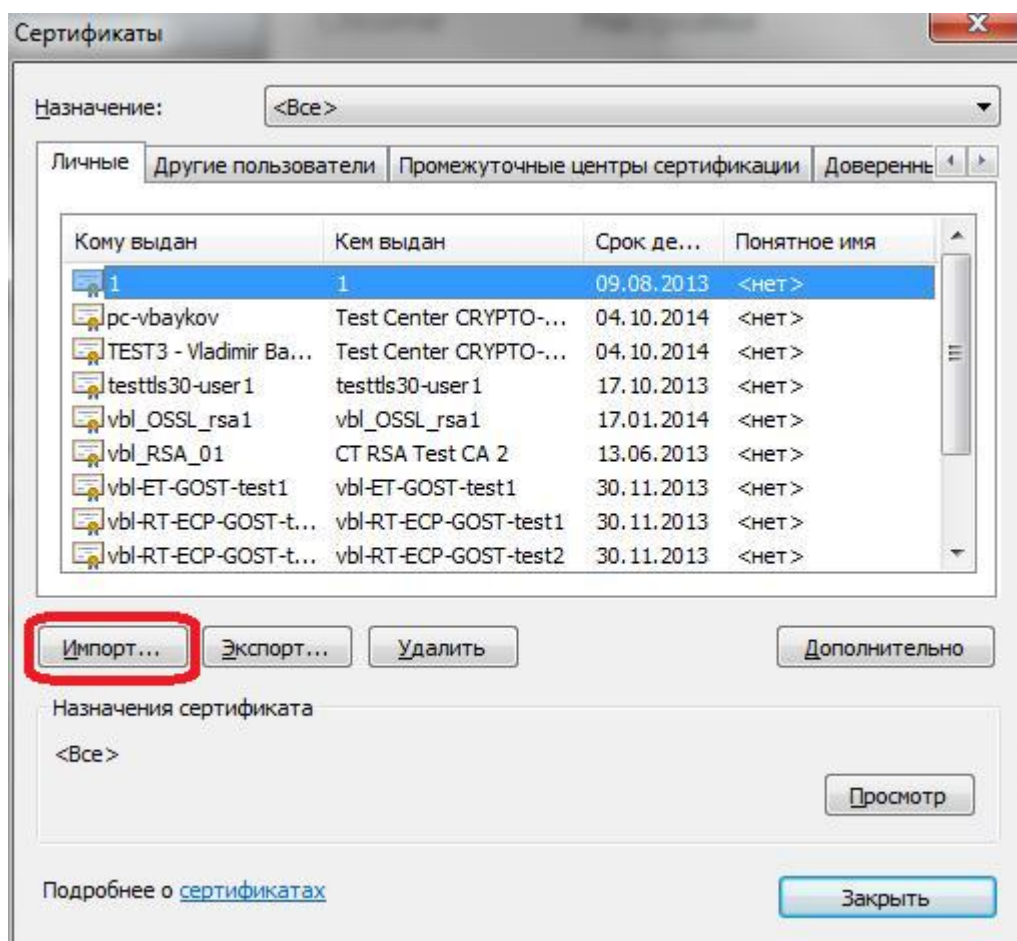


Рис. 7.7. Импорт из PKCS#12 файла

Пройти по всем шагам мастера, указав расположение PKCS#12 файла ключа (рис. 7.8).

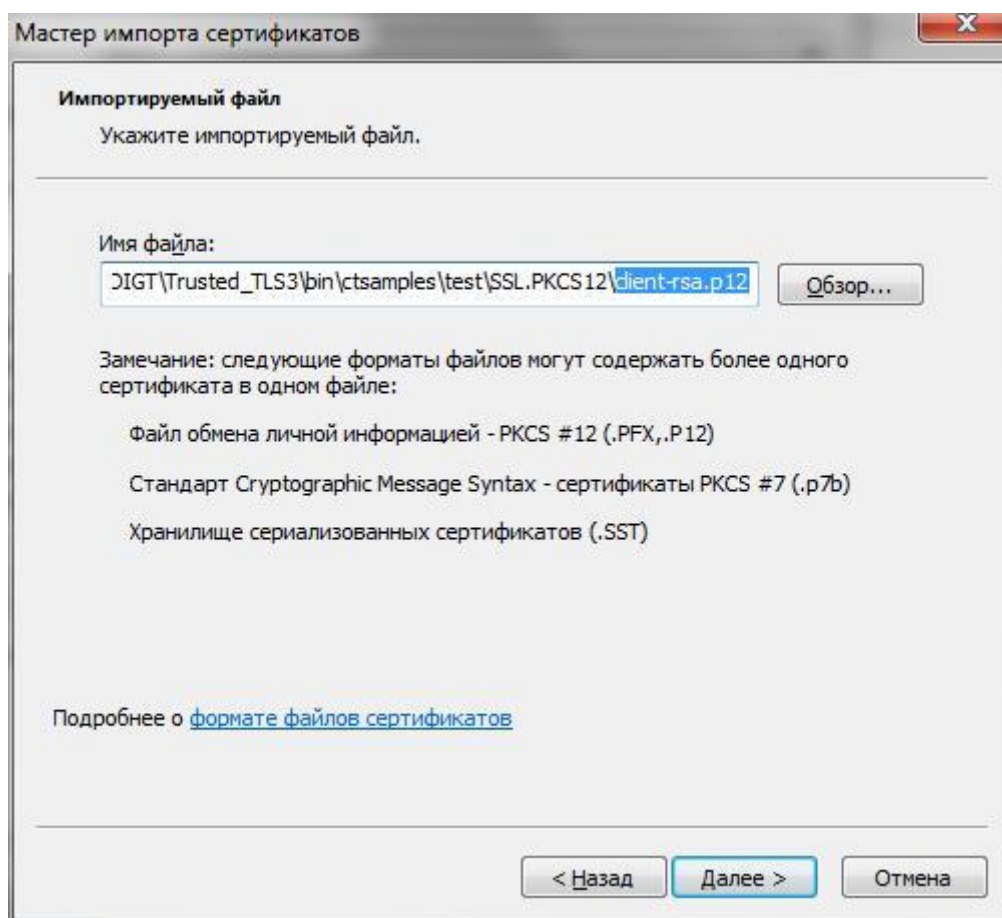


Рис. 7.8. Выбор импортируемого файла

После завершения установки данный ключ можно будет использовать дополнительно в браузерах IE, Srware Iron. Для использования в Firefox необходимо произвести аналогичную установочную процедуру.

Настройка браузера Srware Iron

Для настройки браузера Srware Iron и для поддержки ГОСТ-алгоритмов достаточно в строке запуска браузера указывать опцию --use-system-ssl.

Раздел 8. Работа с токенами ruToken (модели S,CSP,ECP)

В данном разделе описываются примеры генерации сертификатов и ключей на токенах ruToken, подключения их на клиентском месте для организации доступа к ним при установлении TLS-соединений.

Подготовка клиентского места

На клиентском месте под управлением одной из ОС семейства Windows необходимо установить драйвера для токена ruToken со [страницы загрузки](#). Также понадобится установить программу КристоАРМ с [сайта](#) разработчика (рекомендуется/протестировано обновление 7.0 (32 или 64 bit) (сборка 8055 от 26.10.2012 г.)).

Установка драйверов для токенов ruToken производится согласно официальной [документации](#). Информацию по установке, настройке и использованию программы КристоАРМ можно найти в соответствующей [инструкции](#).

Для поддержки ГОСТ-криптографии на клиентском месте предлагается установка СКЗИ КристоПРО CSP или КристоПРО Рутокен CSP. В этом случае предполагается, что закрытые ключи хранятся на токенах в формате, поддерживаемом данным криптопровайдером.

Генерация на токене закрытого ключа и запроса на сертификат

Если генерация закрытого ключа и выпуск сертификата производились удостоверяющим центром, выдавшим токен клиенту, или клиент получил готовый токен каким-либо иным способом, то данный пункт можно пропустить, перейдя к разделу «[Подключение отчуждаемого носителя](#)».

Для генерации закрытого ключа в режиме «Пользователь» программы КристоАРМ необходимо выбрать раздел «Сертификаты».

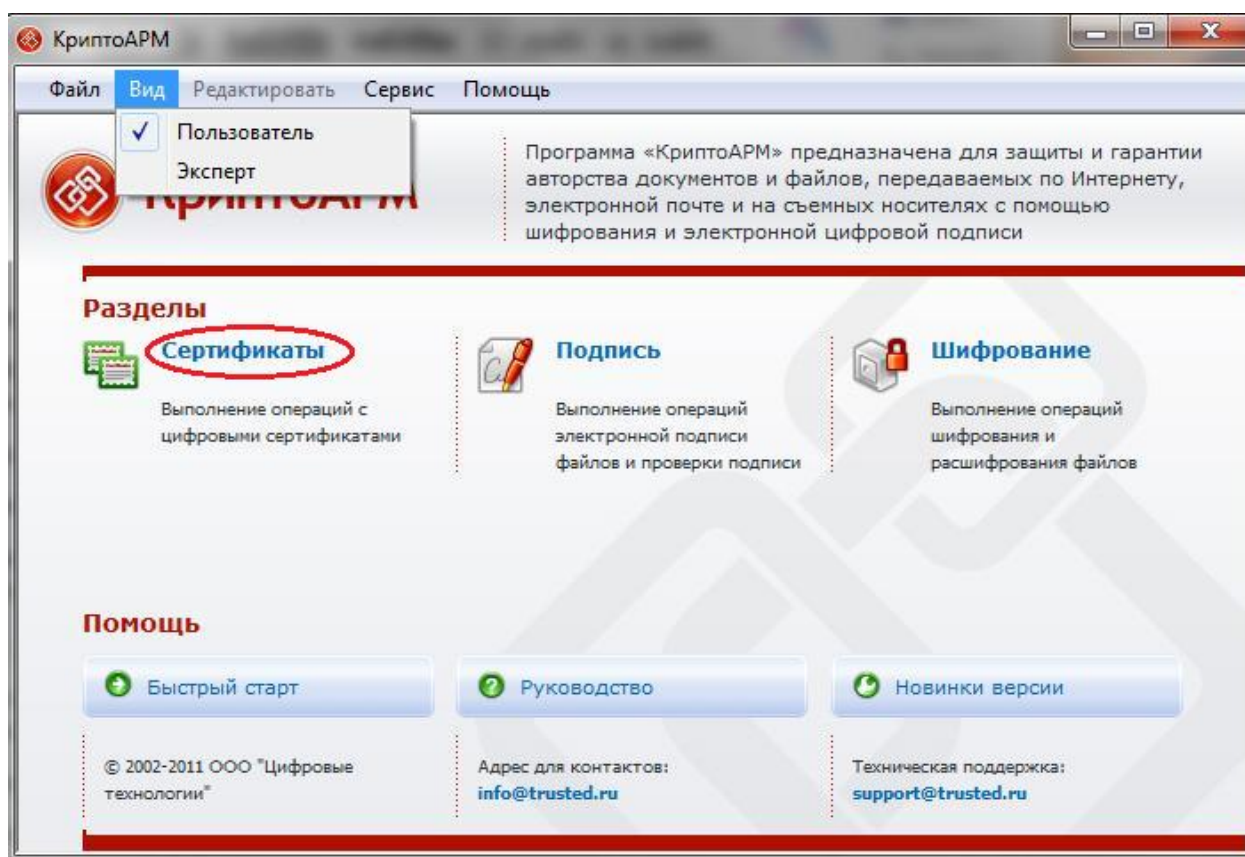


Рис. 8.1. Начало генерации закрытого ключа.

Далее выбираем пункт «Создать запрос на сертификат». Если не предполагается использование услуг удостоверяющего центра, то можно выбрать пункт «Создать самоподписанный сертификат».

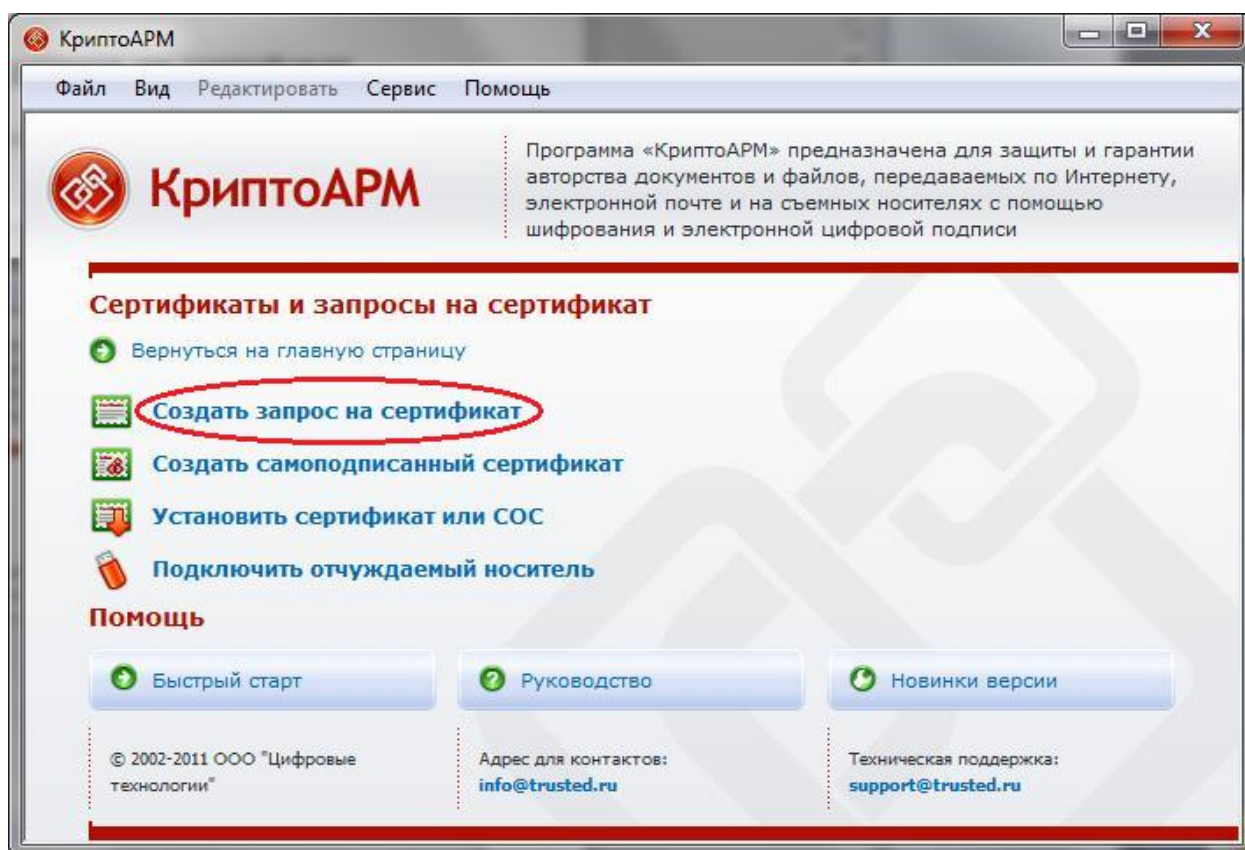


Рис. 8.2. Выбор пункта «Создать запрос на сертификат».

Далее активируется мастер создания запроса, два первых экрана которого можно пропустить, выбрав кнопку «Далее». На следующем экране нужно заполнить необходимые данные для формирования поля «Subject» сертификата.

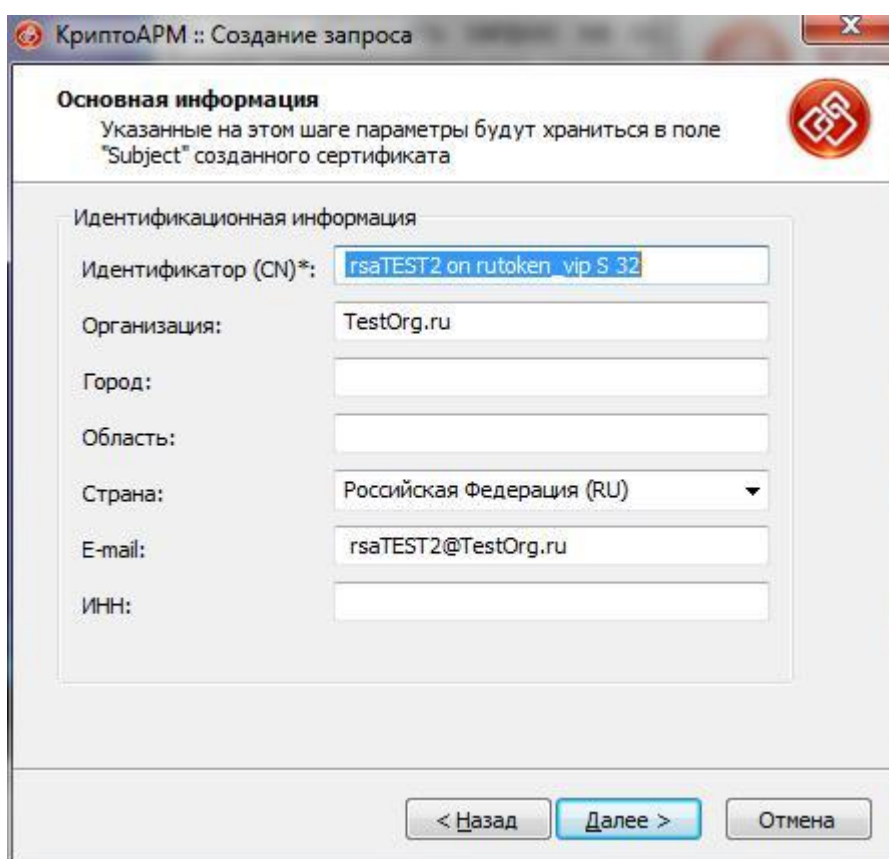


Рис. 8.3. Формирование поля «Subject» сертификата.

В следующем окне для записи ключевого набора на токен ruToken в качестве

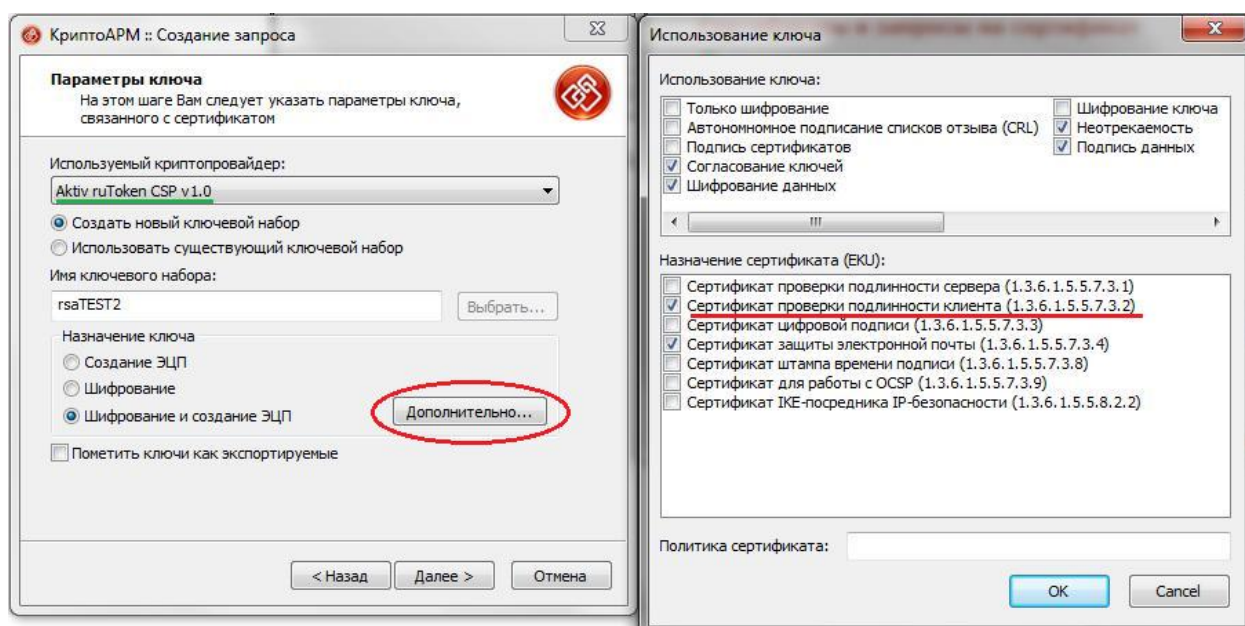


Рис.8.4. Выбор параметров закрытого ключа.

криптопровайдера нужно выбрать «Aktiv ruToken CSP v1.0» (Для использования ГОСТ-криптографии можно использовать «Crypto-Pro GOST R 34.10-2001 Cryptographic Service

Provider» или «GOST R 34.10-2001 Rutoken CSP»). В поле «Имя ключевого набора» для удобства можно задать удобочитаемое (уникальное для токена) имя. Активировав окно «Использование ключа», нажав на кнопку «Дополнительно...» необходимо для назначения сертификата проверить/выставить галочку на «Сертификат проверки подлинности клиента 1.3.6.1.5.5.7.3.2».

Далее указывается расположение файла запроса на сертификат после его сохранения.

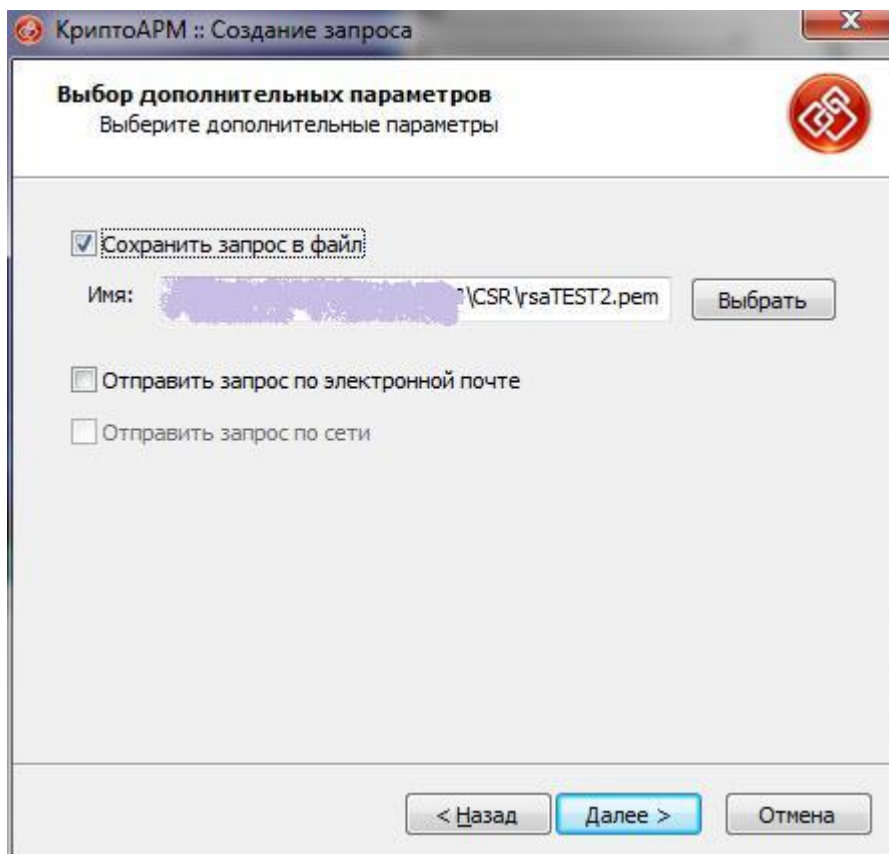


Рис. 8.5. Сохранение запроса в файл.

На следующем завершающем экране формирования запроса для подтверждения нужно нажать пункт «Готово». После этого появится диалог ввода PIN-кода доступа к токenu.

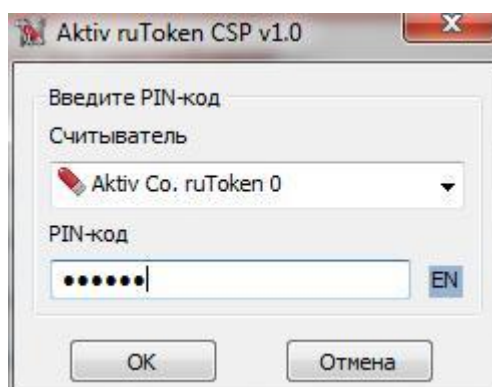


Рис.8.6. Ввод PIN-кода.

При удачной генерации ключевого набора выдается соответствующее сообщение. Файл запроса на сертификат теперь можно передать на формирование сертификата в УЦ. (При использовании СКЗИ КристоПРО CSP перед вводом PIN-кода в отдельном окне производится выбор устройства, в которое вставлен токен, а затем производится набор произвольных нажатий на клавиатуре/мышке при генерации ключа.)

Согласно описанной [выше](#) процедуре издать клиентский сертификат можно с помощью утилиты OpenSSL, выполнив подписание запроса на сертификат. В этом случае в файле запроса придется добавить в начале строку «-----BEGIN CERTIFICATE REQUEST-----» и в конце завершить строкой – «-----END CERTIFICATE REQUEST-----», что является требованием библиотеки OpenSSL.

Установка сертификата на токен

Если генерация закрытого ключа и выпуск сертификата производились удостоверяющим центром, выдавшим токен клиенту, или клиент получил готовый токен каким-либо иным способом, то данный пункт можно пропустить, перейдя к разделу [«Подключение отчуждаемого носителя»](#).

После получения выпущенного сертификата его нужно установить на токен. Для этого в режиме «Пользователь» программы КристоАРМ необходимо выбрать раздел «Установить сертификат или СОС».

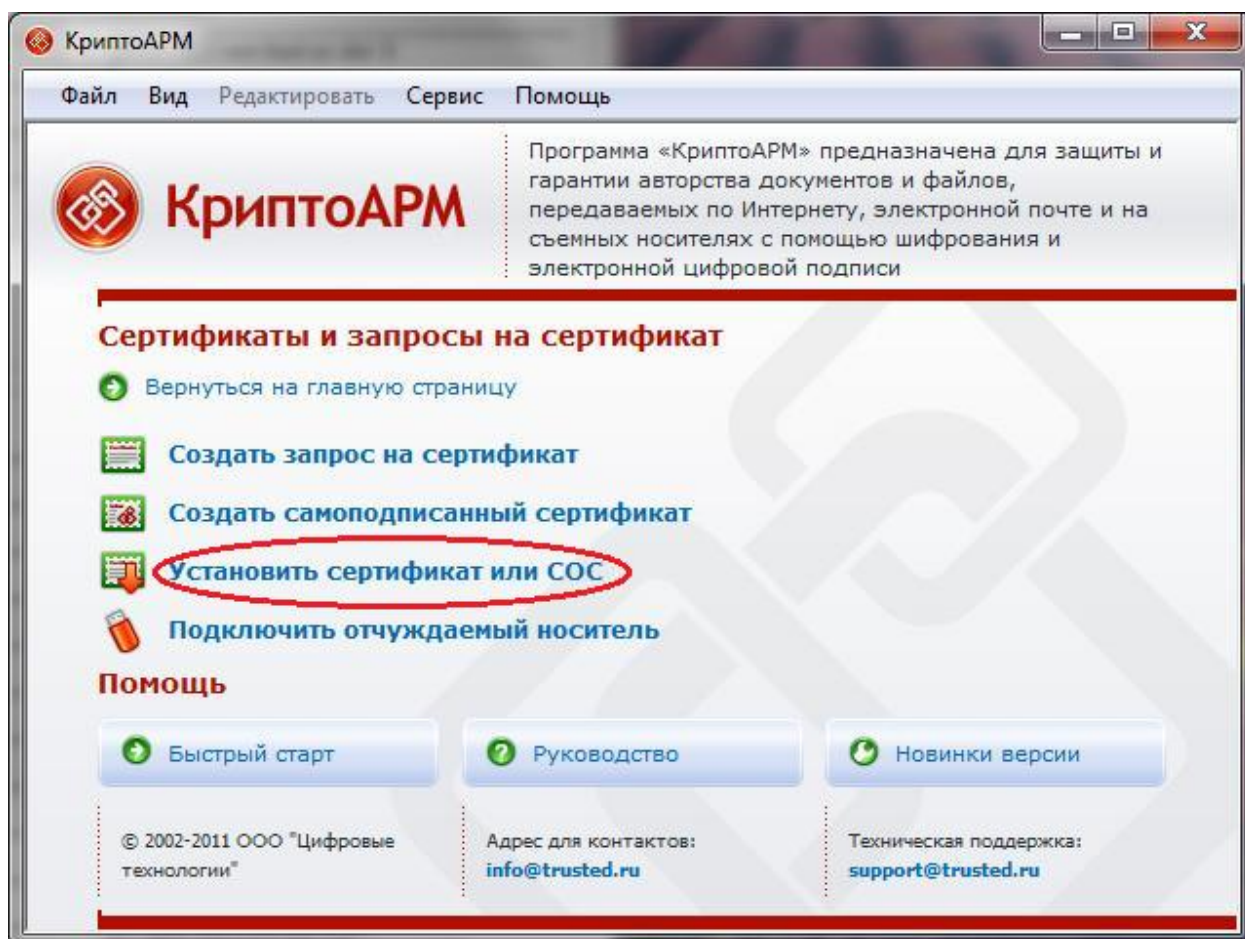


Рис.8.6. Начало установки сертификата.

На появившемся стартовом экране мастера установки сертификата необходимо выставить галочки на пунктах «Установить личный сертификат» и «Поместить сертификат в контейнер».

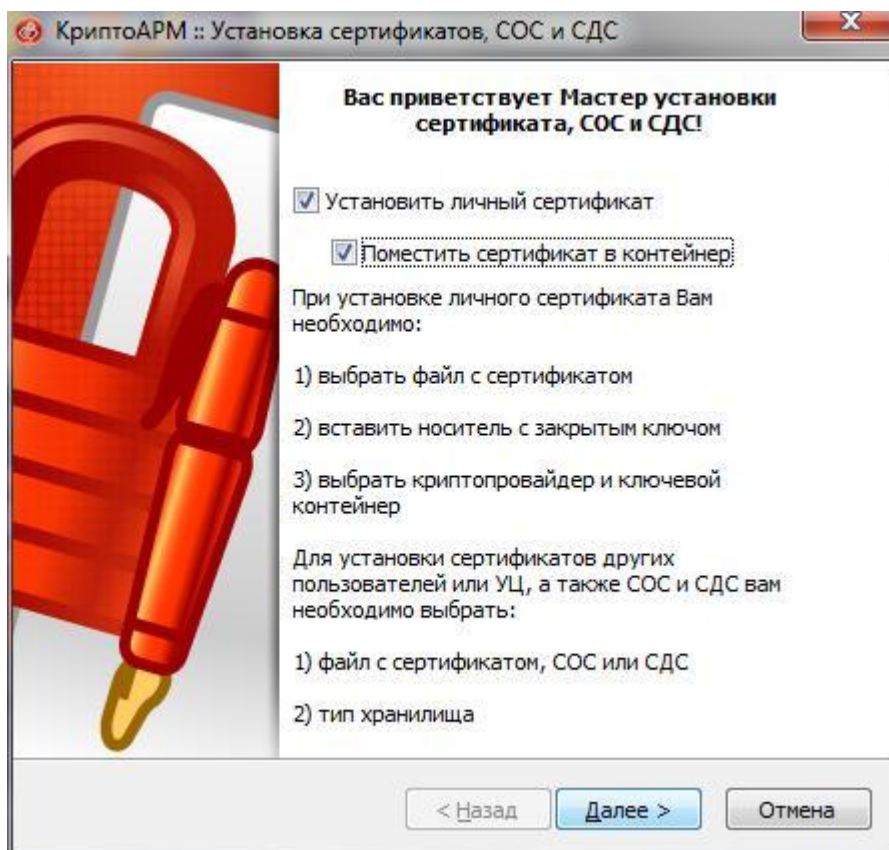


Рис.8.7. Параметры установки сертификата.

На следующем экране выбирается расположение файла сертификата.

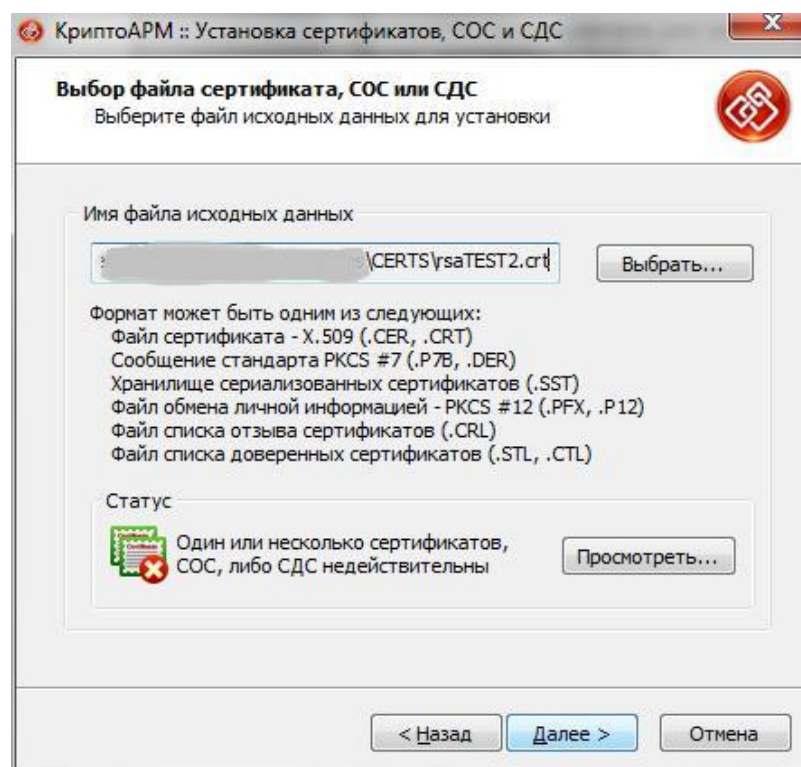


Рис. 8.8. Выбор файла сертификата.

На следующих двух экранах предоставляется информация по выбранным параметрам мастера установки сертификата, на последнем из которых требуется выбрать пункт «Готово» для запуска процесса установки. Если запрос на сертификат каким-то образом пропал (был удален) с рабочего места, то далее производится выбор криптопровайдера и контейнера ключа. Затем производится запрос PIN-кода доступа к токenu и при успешной установке сертификата выдается соответствующее завершающее окно.

Подключение отчуждаемого носителя

Данный раздел описывает действия по подключению токена в случае, когда генерация закрытого ключа и выпуск сертификата производились удостоверяющим центром, выдавшим токен клиенту, или когда клиент получил готовый токен каким-либо иным способом. На данном этапе сертификат клиента с токена устанавливается в личном хранилище пользователя.

Для активации мастера в режиме «Пользователь» программы КриптоАРМ необходимо выбрать раздел «Подключить отчуждаемый носитель». После приветствия мастера на следующем экране необходимо выбрать в качестве криптопровайдера «Aktiv ruToken CSP v1.0» (для использования ГОСТ-криптографии использовать «Crypto-Pro GOST R 34.10-2001 Cryptographic Service Provider») и в качестве носителя «Смарт-карта/USB-Токен».

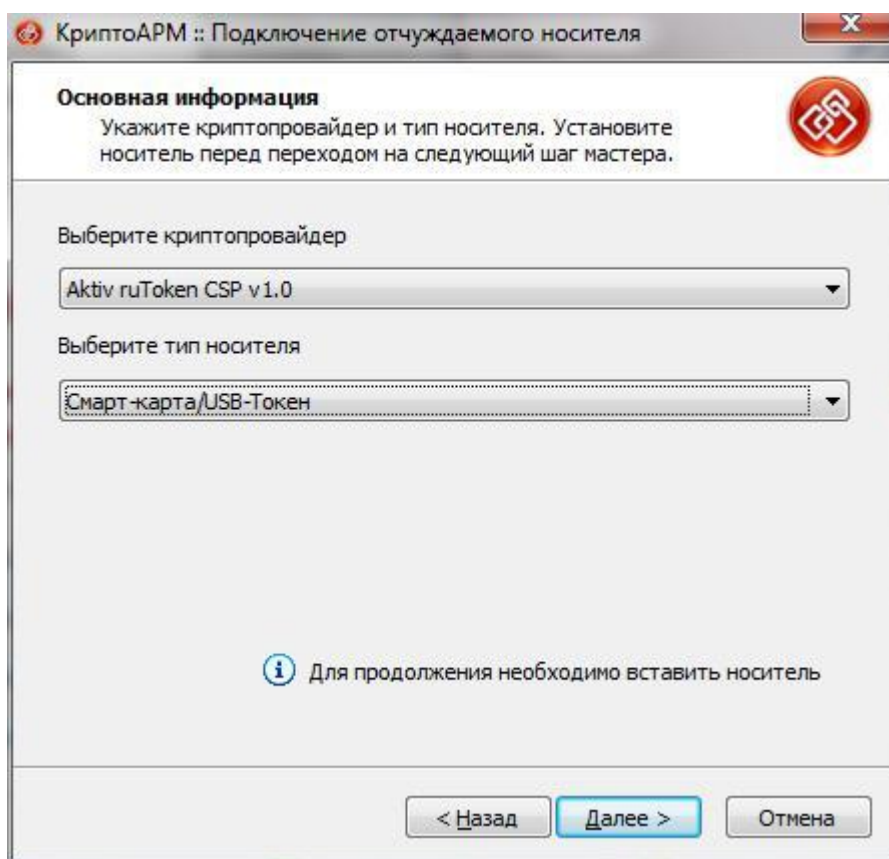


Рис. 8.9. Выбор подключения носителя.

На следующем экране выбирается контейнер (установка «галочки» в имени) с установленным в него сертификатом. Проверить наличие сертификата в контейнере можно по нажатию на кнопку «Просмотреть сертификат в контейнере».

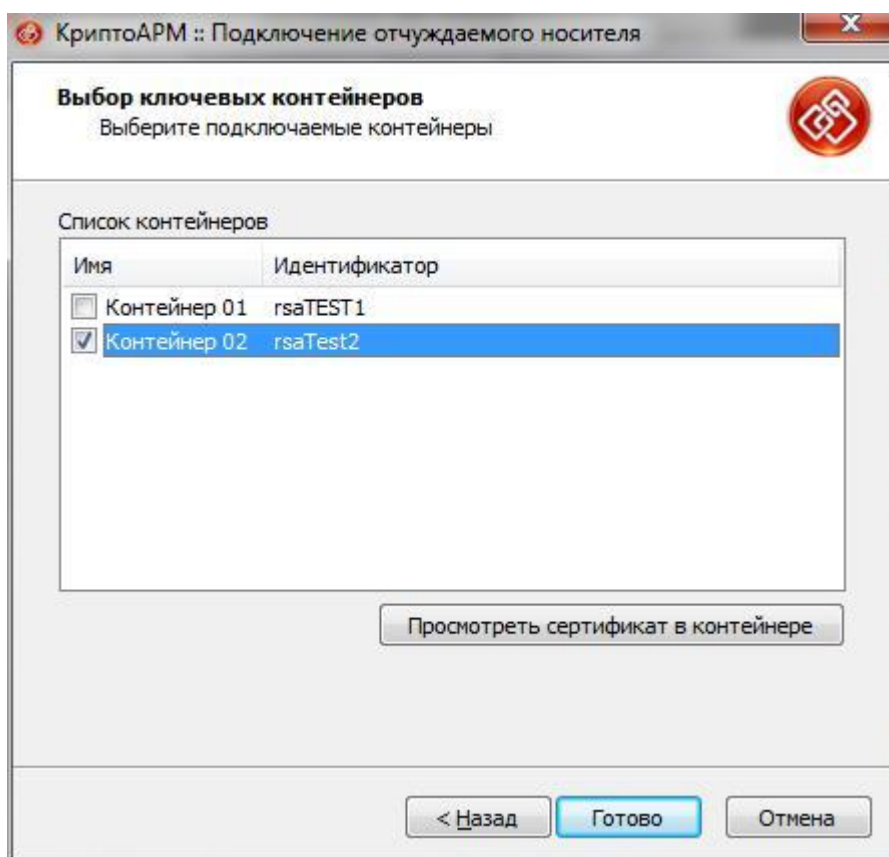


Рис. 8.10. Выбор контейнера ключа и сертификата.

После нажатия кнопки «Готово» можно посмотреть результат подключения токена.

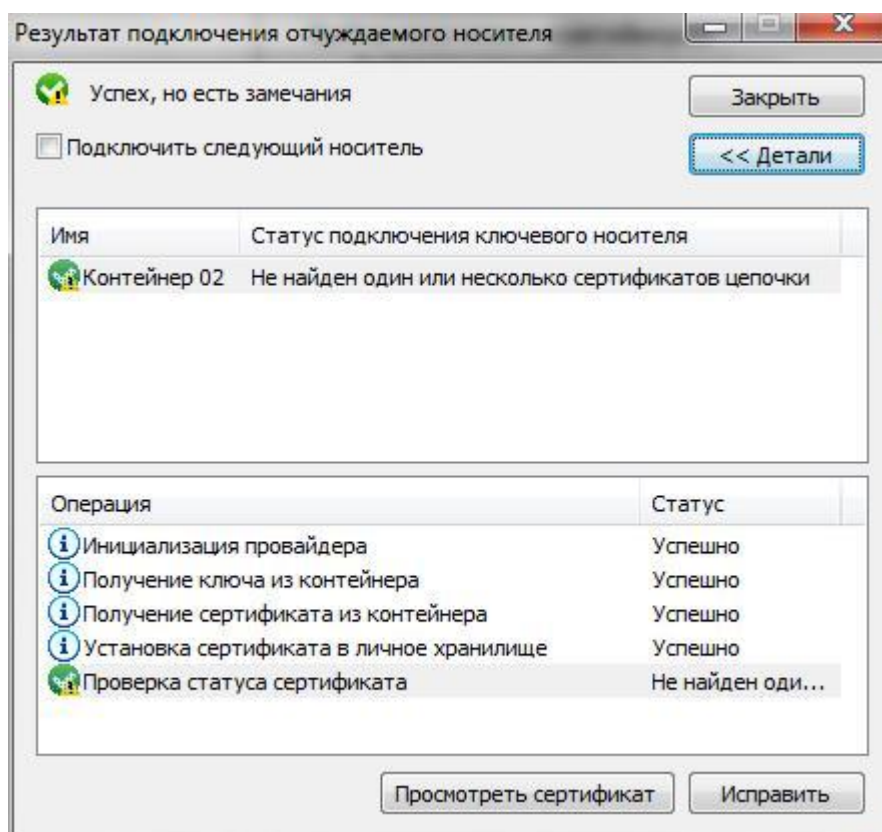


Рис. 8.11. Результат подключения токена.

Установку промежуточных и корневых сертификатов в систему при необходимости можно провести согласно инструкции на КристоАРМ.

Настройка браузеров для работы с токенами

При выполнении на клиентском месте действий из пункта «[Установка сертификата на токен](#)» или из пункта «[Подключение отчуждаемого носителя](#)» автоматически появляется возможность использовать токен из-под браузеров IE, Chromium-браузеров - Google Chrome, Srware Iron.

Если эти действия не были проделаны ввиду того, что в настоящее время для их активации требуется покупка лицензии на программу КристоАРМ, то при использовании токенов Rutoken можно воспользоваться панелью управления Рутокен, поставляемой в составе драйверов последней версии.

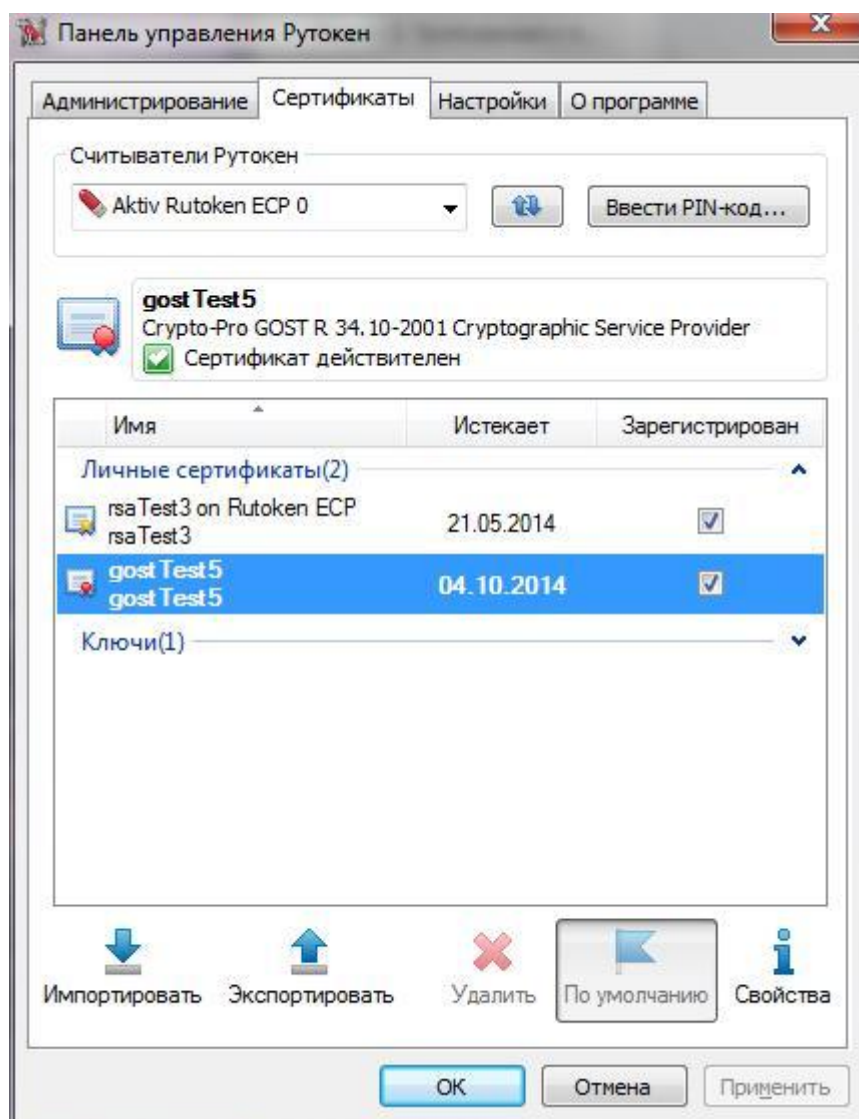


Рис. 8.12. Установка сертификата через панель управления Рутокен.

В закладке «Сертификаты», в которой отображается таблица сертификатов, сохраненных на токене и привязанных к контейнерам закрытых ключей, установку сертификатов в личное хранилище можно провести при активации флажка в столбце «Зарегистрирован» таблицы сертификатов.

Для поддержки токенов в FireFox необходимо проделать следующие действия.

В окне дополнительных настроек FireFox в закладке «Шифрование» необходимо выбрать кнопку «Устройства защиты».

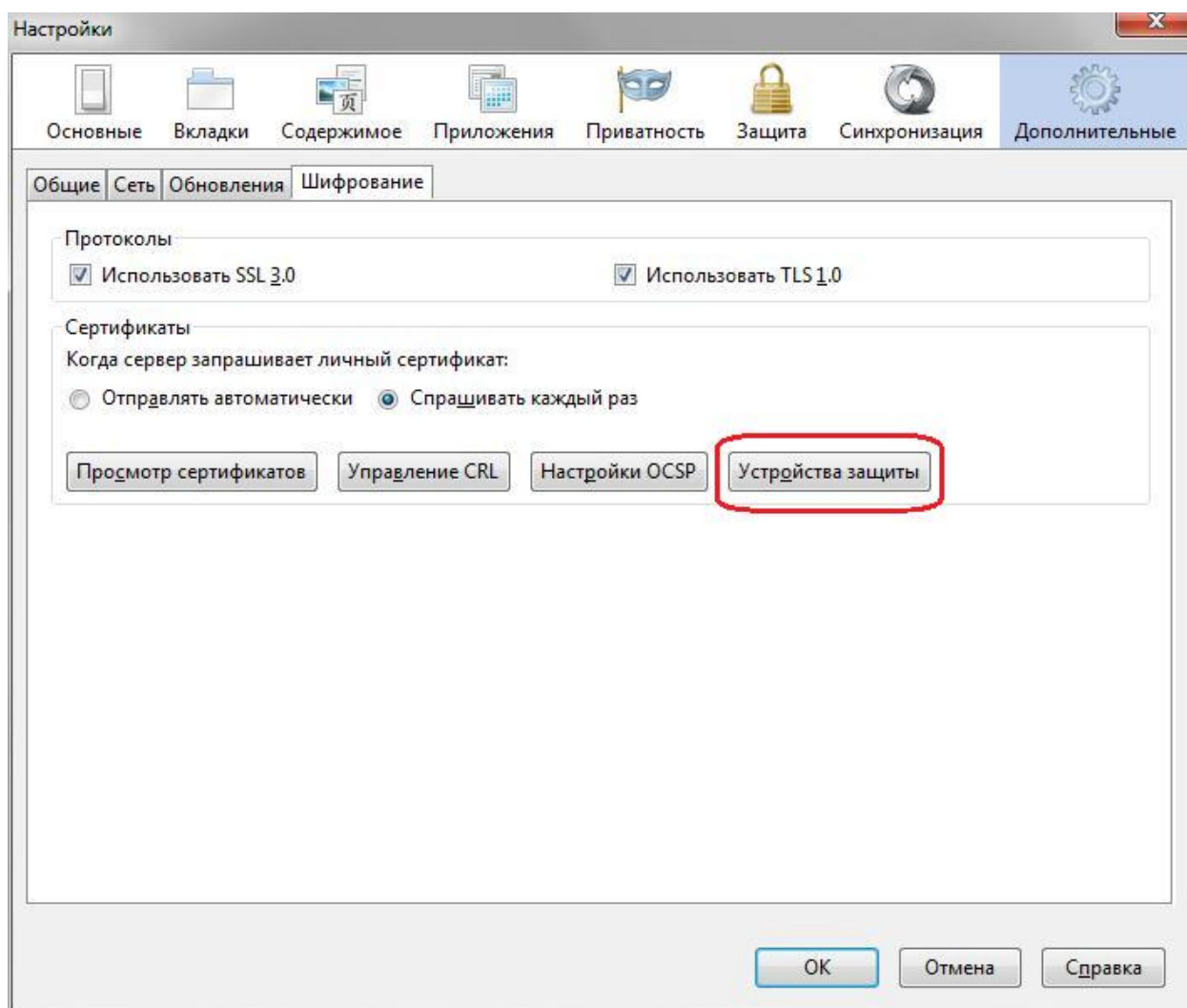


Рис. 8.12. Выбор управления устройствами защиты.

В открывшемся окне списка поддерживаемых устройств должна присутствовать запись, соответствующая подгружаемому файлу `rtPKCS11.dll` модуля, например, `ruToken`.

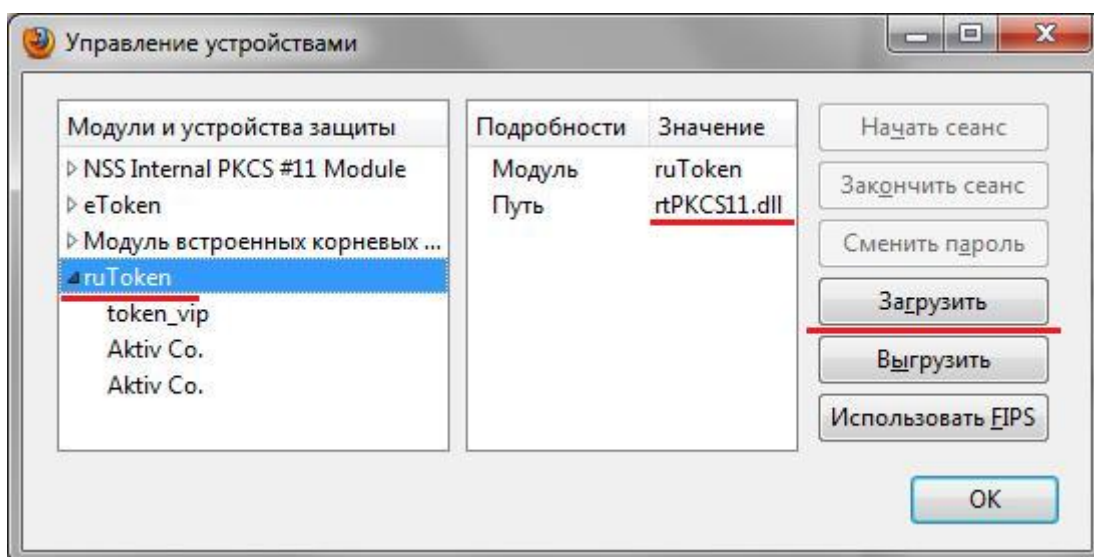


Рис. 8.13. Список устройств защиты.

Если таково устройства нет, то необходимо выбрать кнопку «Загрузить» и ввести имя модуля и путь до указанного файла модуля.

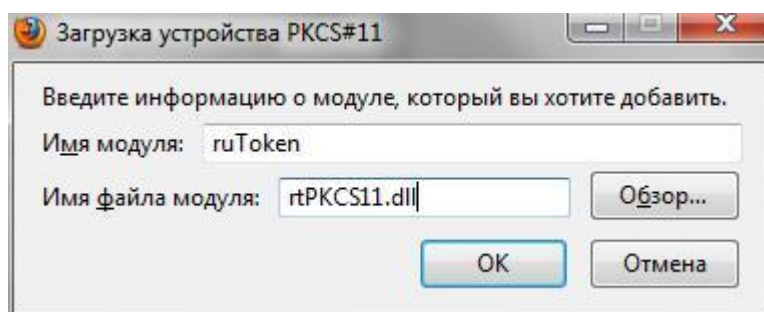


Рис. 8.13. Добавление устройства защиты.

При наличии указанного устройства в списке устройств можно использовать сохраненные на токенах сертификаты (ключи) для организации TLS-защищенных соединений с авторизацией клиента только по RSA-сертификатам.

Раздел 9. Коды ошибок модуля

В таблице приводятся коды ошибок и сообщения, появляющиеся при возникновении критических ошибок и блокирующих дальнейшую работу модуля.

№	Код ошибки	Текст сообщения
1	Error #1	В системе не установлен необходимый модуль IBlock
2	Error #2	Ошибка удаления инфоблока или типа инфоблока MAPPING_STORE
3	Error #3	Ошибка удаления инфоблока или типа инфоблока CERT_STORE
4	Error #4	Ошибка создания типа инфоблока MAPPING_STORE
5	Error #5	Ошибка создания типа инфоблока CERT_STORE
6	Error #6	Ошибка создания инфоблока MAPPING_STORE
7	Error #7	Ошибка создания инфоблока CERT_STORE
8	Error #8	Ошибка создания свойств инфоблока MAPPING_STORE
9	Error #9	Ошибка создания свойств инфоблока CERT_STORE
10	Error #10	Не завершена установка модуля trusted.tbstart
11	Error #11	Инфоблок хранилища не обнаружен
12	Error #12	Не удалось изменить свойства инфоблока
13	Error #13	Не удалось удалить элемент инфоблока
14	Error #14	Ошибка добавления сертификата в инфоблок CERT_STORE
15	Error #15	Ошибка добавления привязки в инфоблоке MAPPING_STORE
16	Error #16	Ошибка удаления сертификата из хранилища CERT_STORE
17	Error #17	Ошибка активации сертификата
18	Error #18	Ошибка деактивации сертификата
19	Error #19	Ошибка проверки корректности сертификата
20	Error #20	Ошибка получения опций, установленных по умолчанию