Вход в Linux по сертификату на JaCarta

Версия ПО: GNU/Linux, ID Protect для linux, OpenSSL, libpam-pkcs11, XCA

Токены: Любые

Проблема:

Настроить вход на рабочую станцию, используя сертификат с закрытым ключом, сгенерированным на стороне смарт-карты.

Решение:

Настройку входа в систему осуществляем при помощи **OpenSSL** и его графического интерфейса **XCA**. Работа на стенде с Debian testing.

Устанавливаем пакеты **pcscd, libccid, openssl, xca, libengine-pkcs11-openssl, libengine-pkcs11-openssl1.1, libpam-pkcs11**

*Замечание: пакет **libengine-pkcs11-openssl** в репозиториях представлен в двух версиях. Устанавливались оба.*

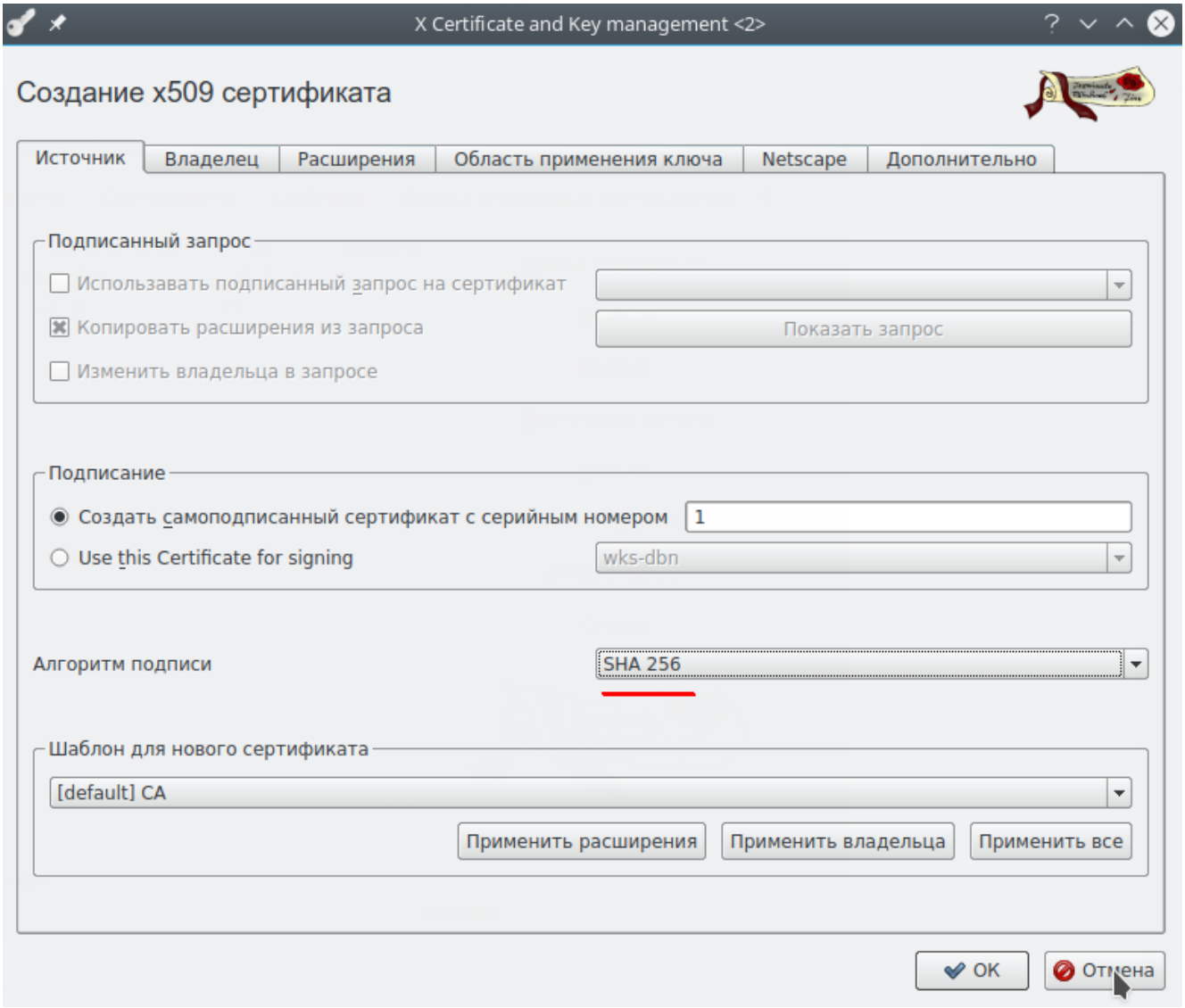
Драйвер для JaCarta
PKI: https://www.aladdin-rd.ru/support/downloads/989e9d58-8600-4d97-a039-e209c7c8fa8e

1. Настройка ЦС.

Запускаем xca от имени root.

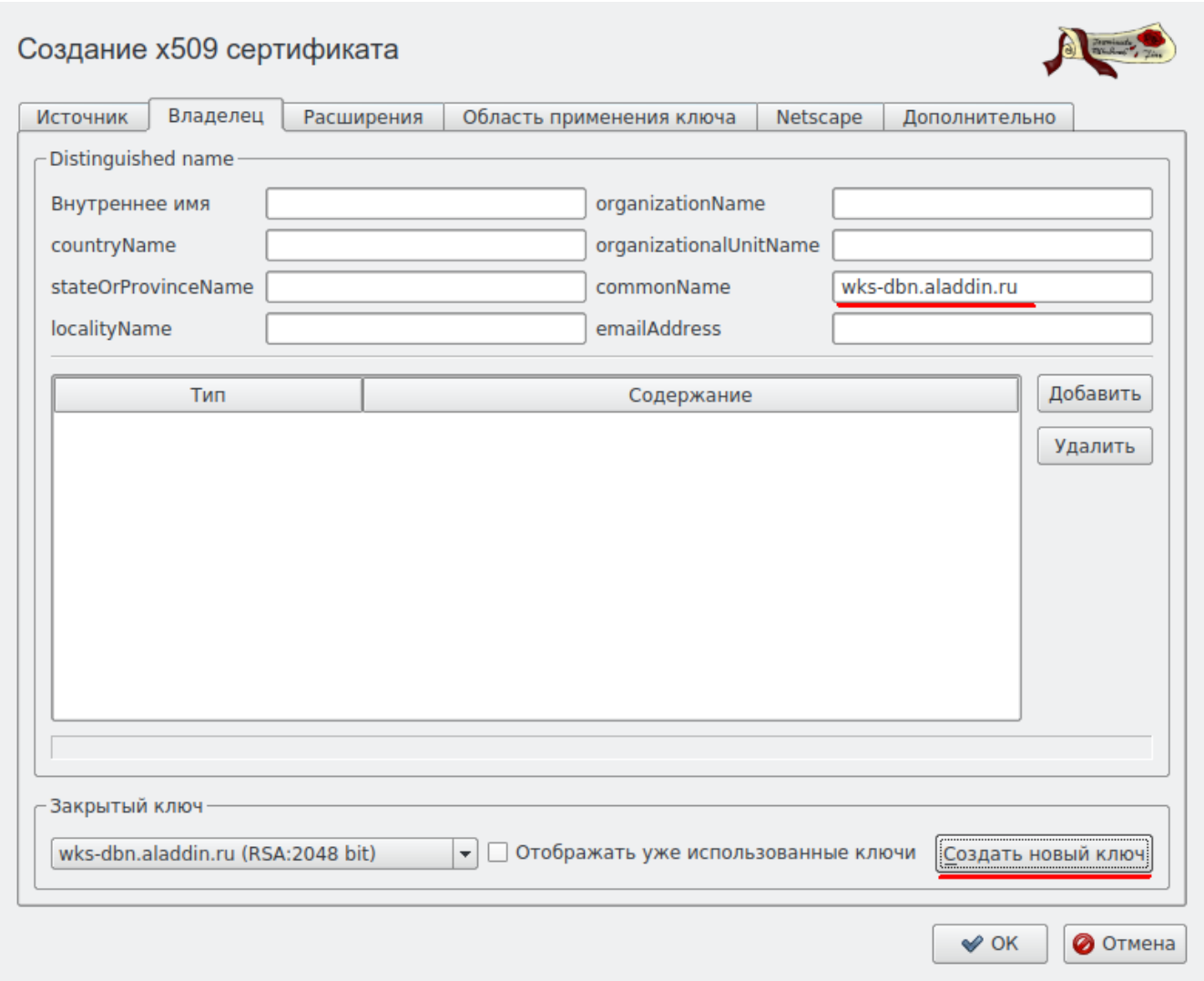
Замечание: вероятно, можно это делать и без привилегий суперпользователя. Но стенд настраивался именно так.

Переходим на вкладку сертификаты, создаем новый для сервера CA.



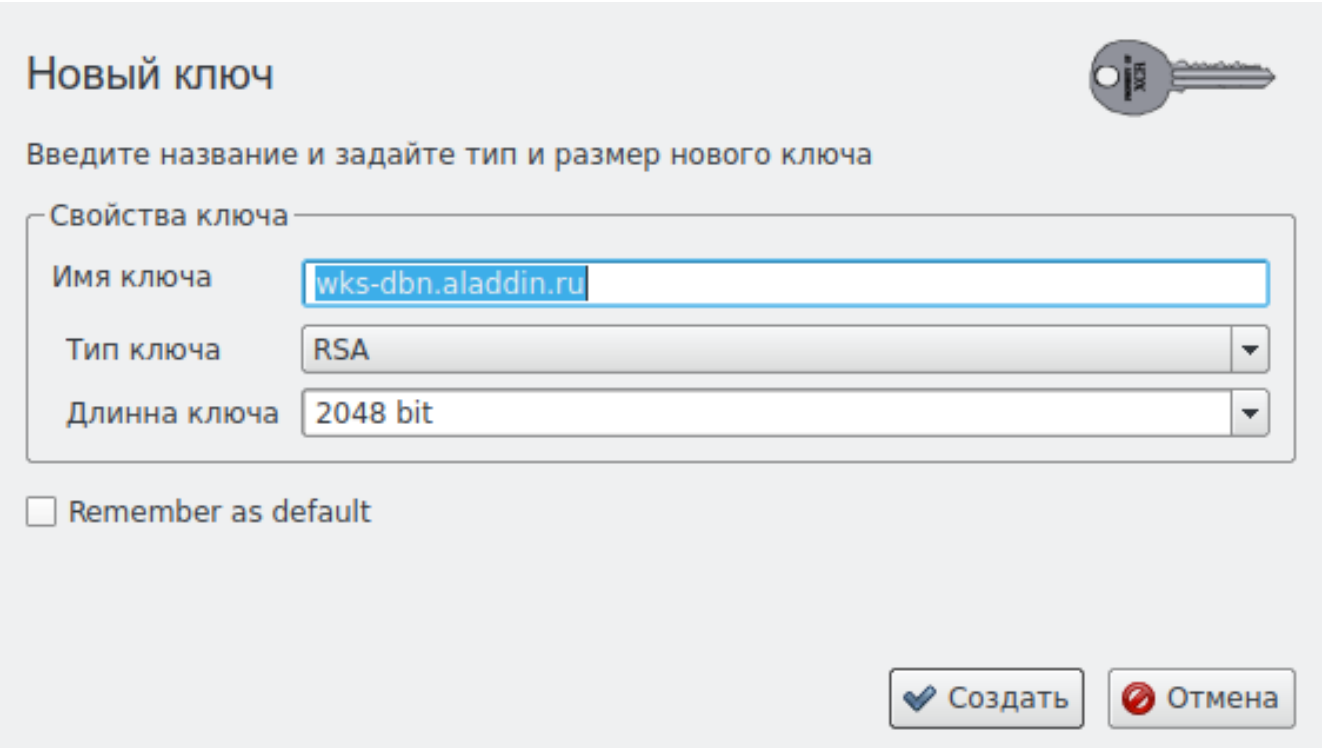
На вкладке "Источник" выбираем самоподписанный сертификат, алгоритм подписи SHA 256.

Далее идём на вкладку "Владелец":



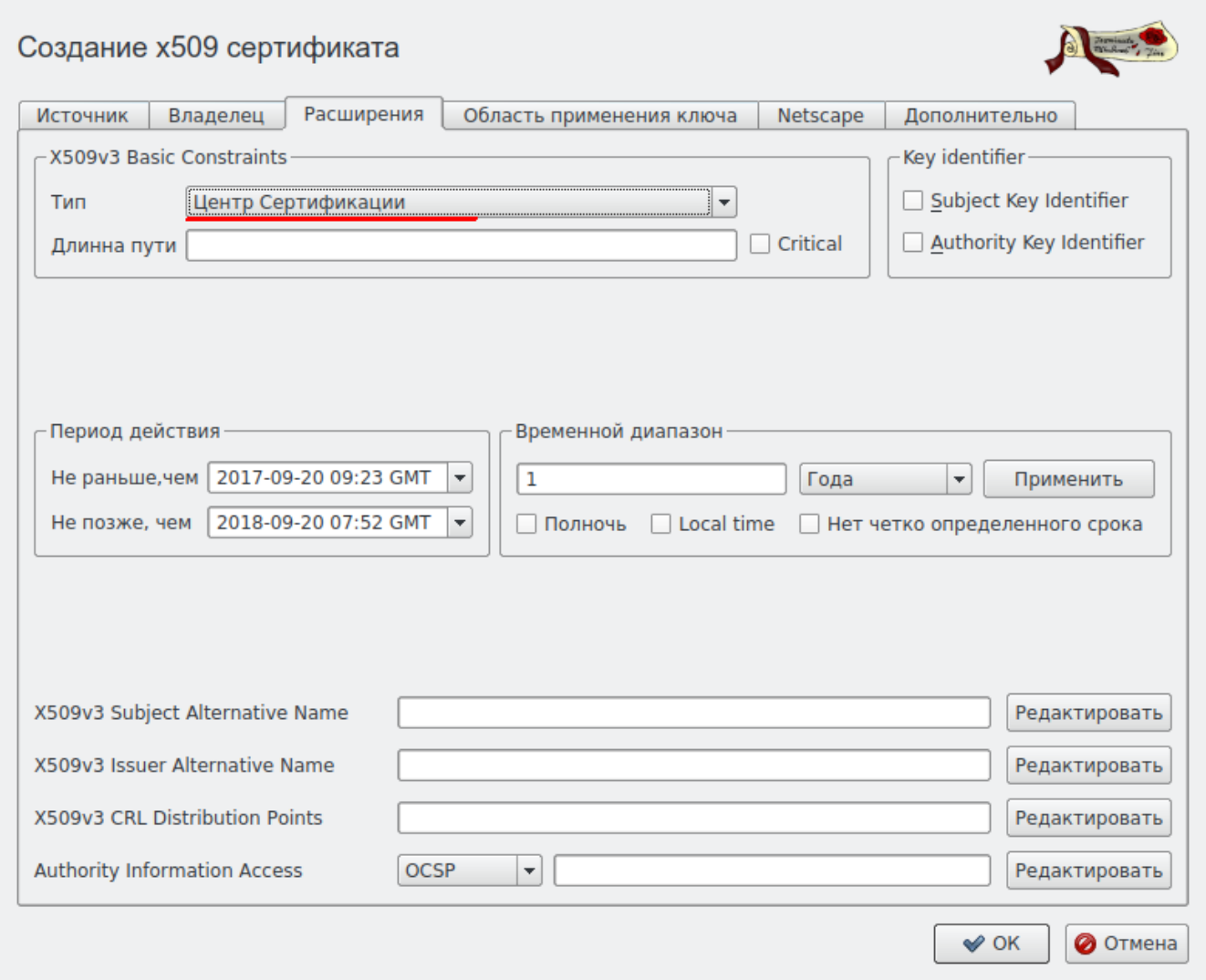
Заполняем необходимые поля, в поле **commonName** указываем имя хоста.

Здесь же нажимаем "Создать новый ключ".



Нажимаем "Создать".

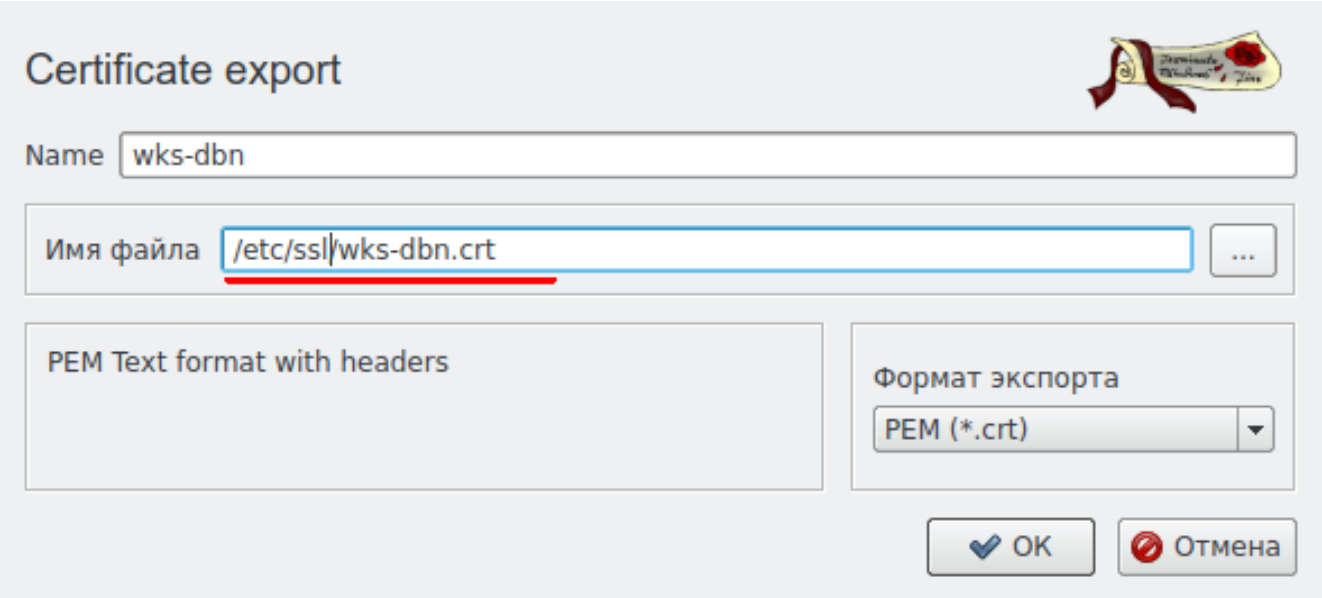
Далее переходим на вкладку "Расширения".



В поле "Тип" указываем "Центр сертификации". При необходимости изменяем другие параметры.

Далее нажимаем кнопку "OK" и убеждаемся, что новый сертификат появился в списке.

Устанавливаем указатель на этот сертификат, нажимаем "Экспорт"

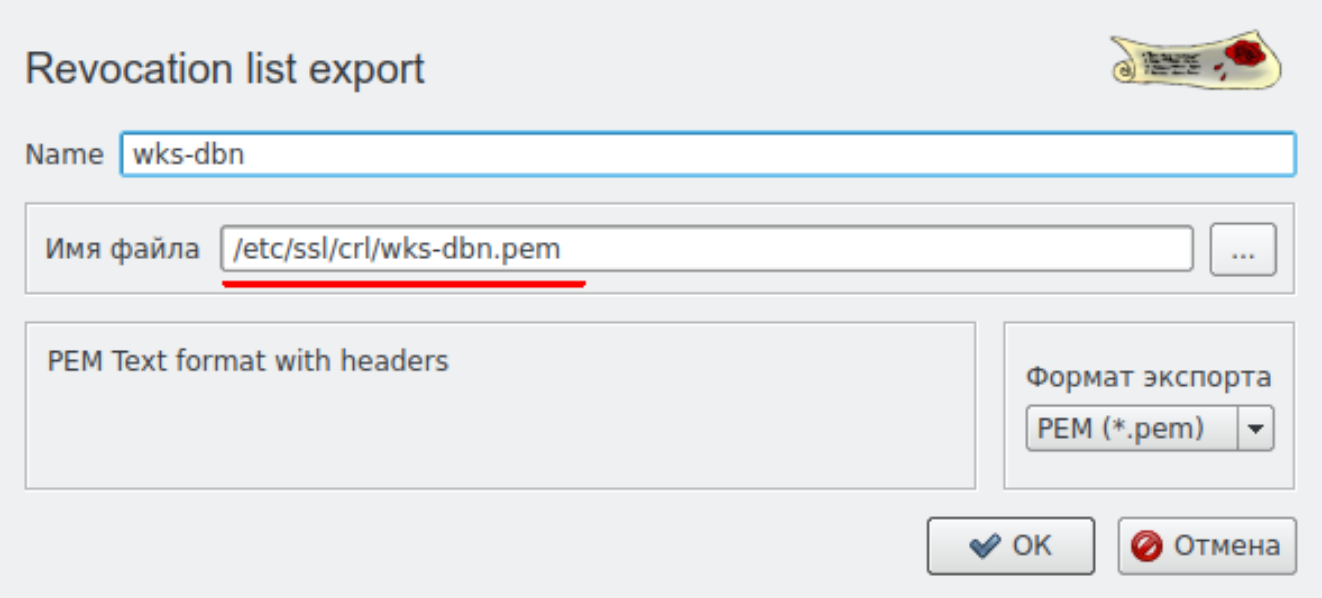


Указываем каталог **/etc/ssl**.

Далее ПКМ на сертификате: ЦС -> Создать CRL.

Переходим на вкладку "Списки отозванных сертификатов".

Выделяем появившийся список, нажимаем "Экспорт"



Каталог для экспорта **/etc/ssl/crl**.

Далее создаем хэши для наших данных.

Для этого понадобится скрипт **make_hash_link.sh**. Содержимое можно взять здесь: https://github.com/OpenSC/pam_pkcs11/blob/master/tools/pkcs11_make_hash_link

Создаем файл make_hash_link.sh, делаем его исполняемым:

```
sudo chmod +x make_hash_link.sh
```

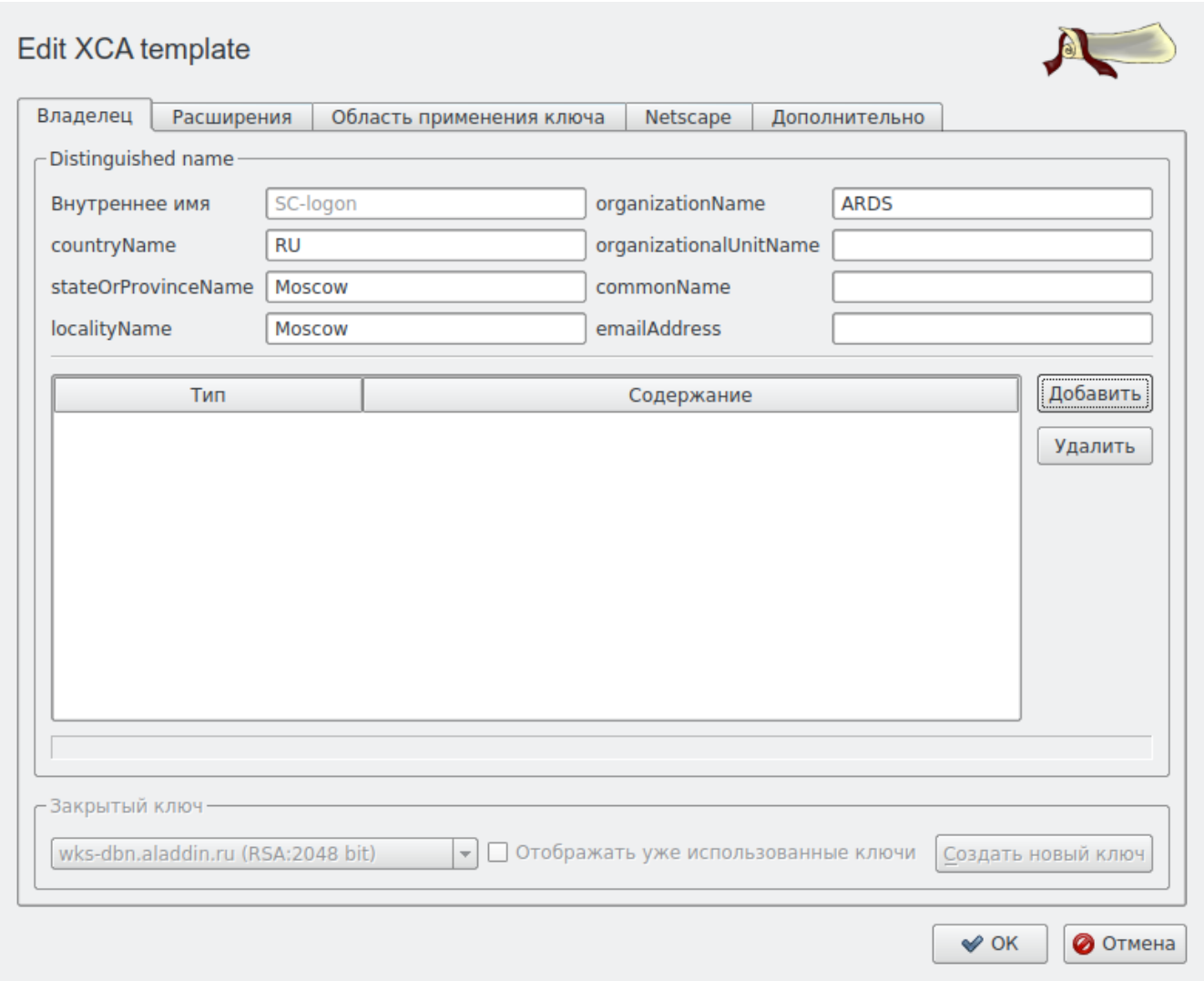
Далее последовательно выполняем:

```
sudo ./make_hash_link.sh /etc/ssl
```

```
sudo ./make_hash_link.sh /etc/ssl/crl
```

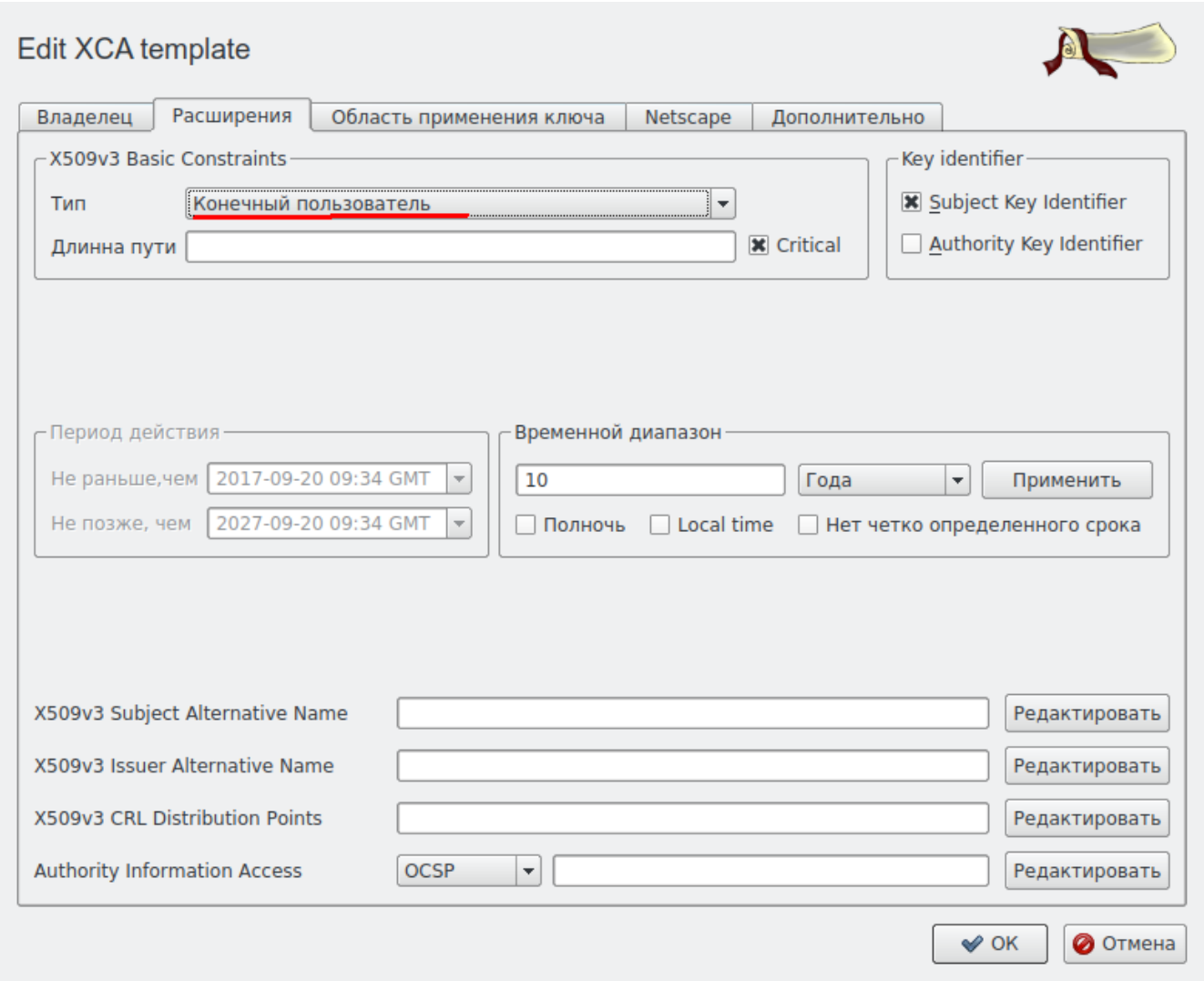
2. Шаблон сертификата.

Возвращаемся в интерфейс XCA, делаем шаблон для своего сертификата. Переходим на вкладку "Шаблоны", нажимаем "Новый шаблон"



На вкладке "Владелец" заполняем необходимые поля.

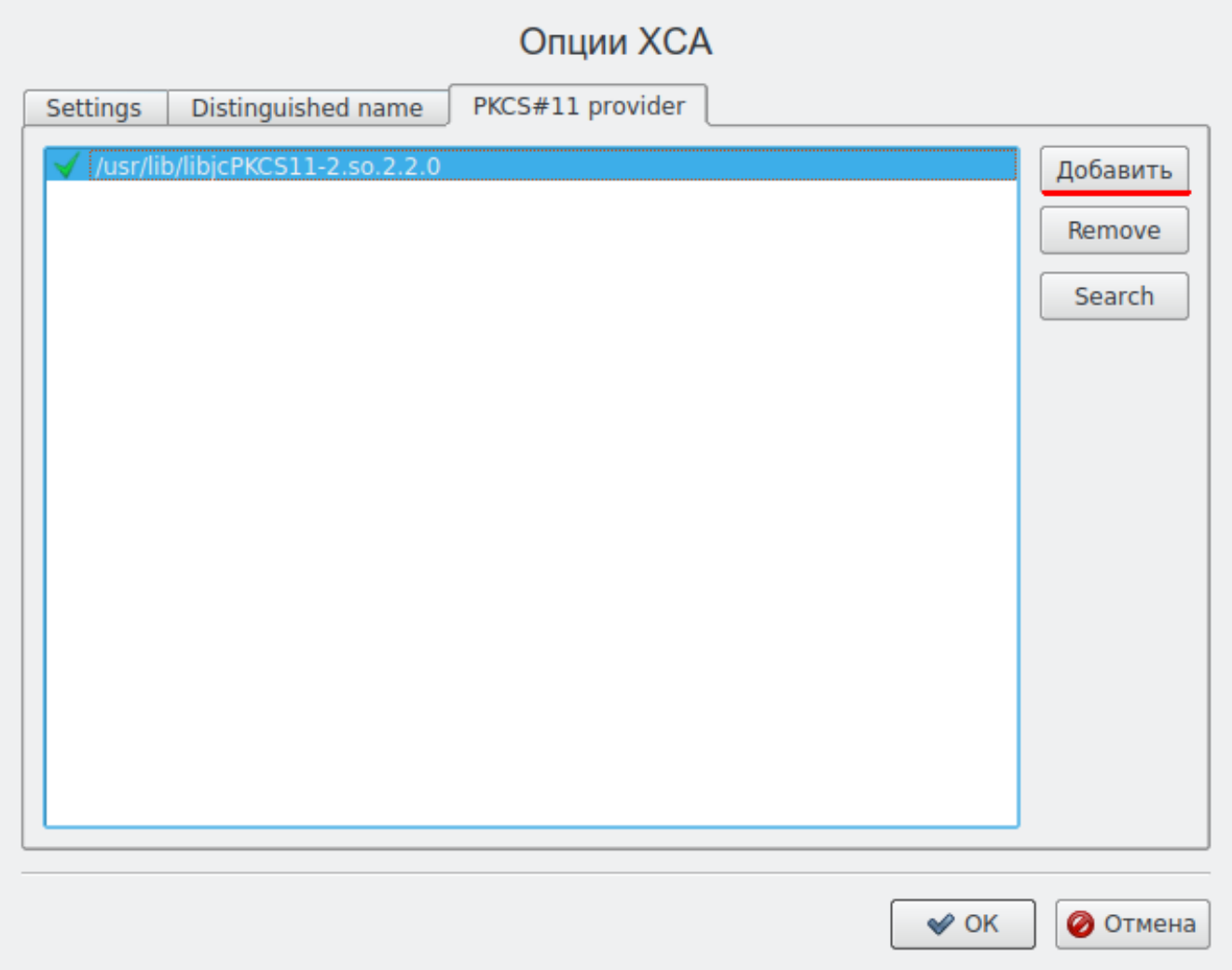
На вкладке "Расширения" указываем "Конечный пользователь"



Нажимаем "OK"

3. Настройка модуля для смарт-карт.

Для того, чтобы можно было выпускать сертификаты с ключевой парой и сертификатами прямо на токене, необходимо добавить модуль **pkcs11** в настройках. В меню Файл -> Опции, вкладка PKCS#11 provider.



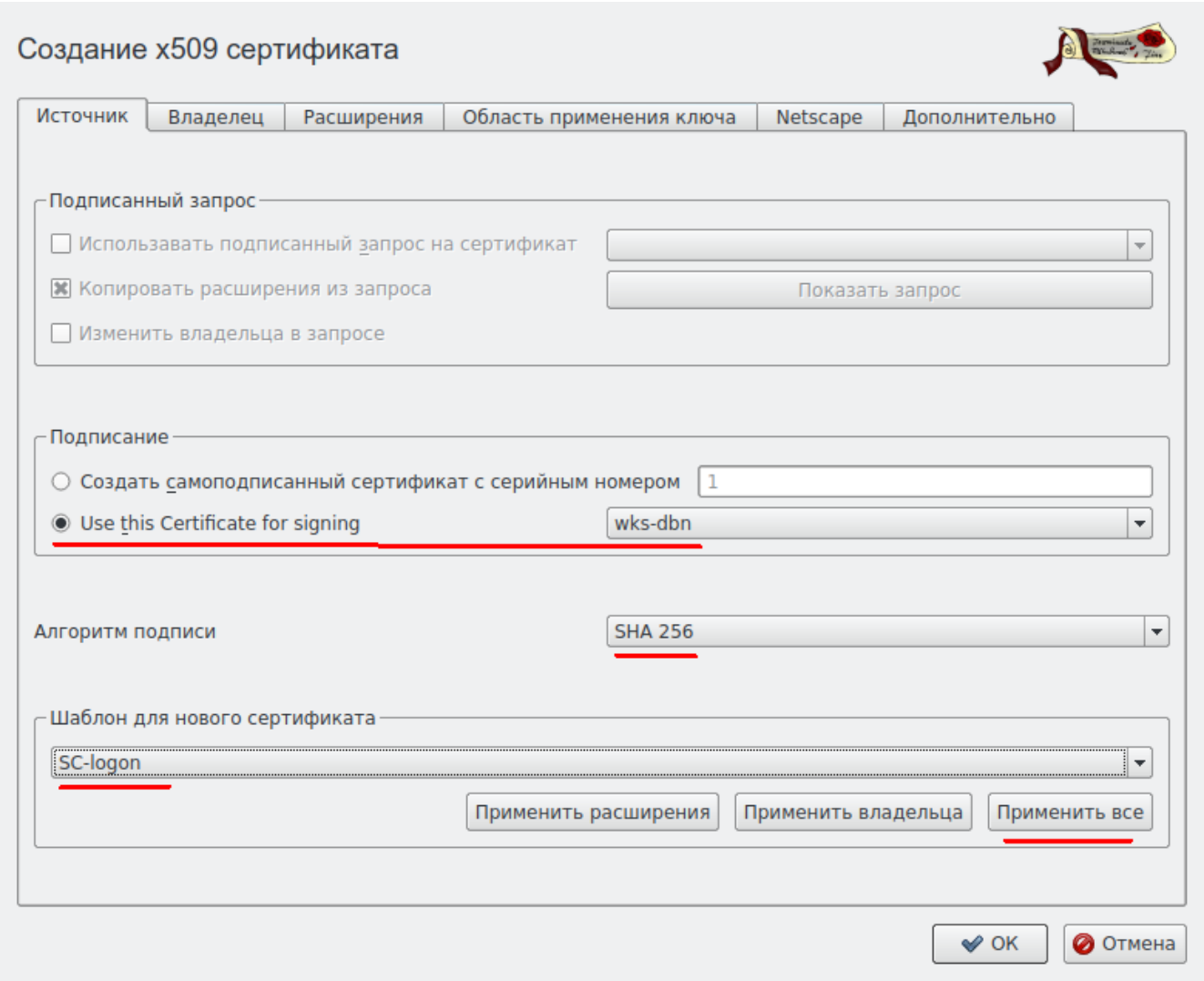
Необходимо добавить библиотеку **/usr/lib/libPKCS11.so.2**.

Замечание: пути могут меняться в зависимости от дистрибутива и версий ПО.

4. Выпуск сертификата.

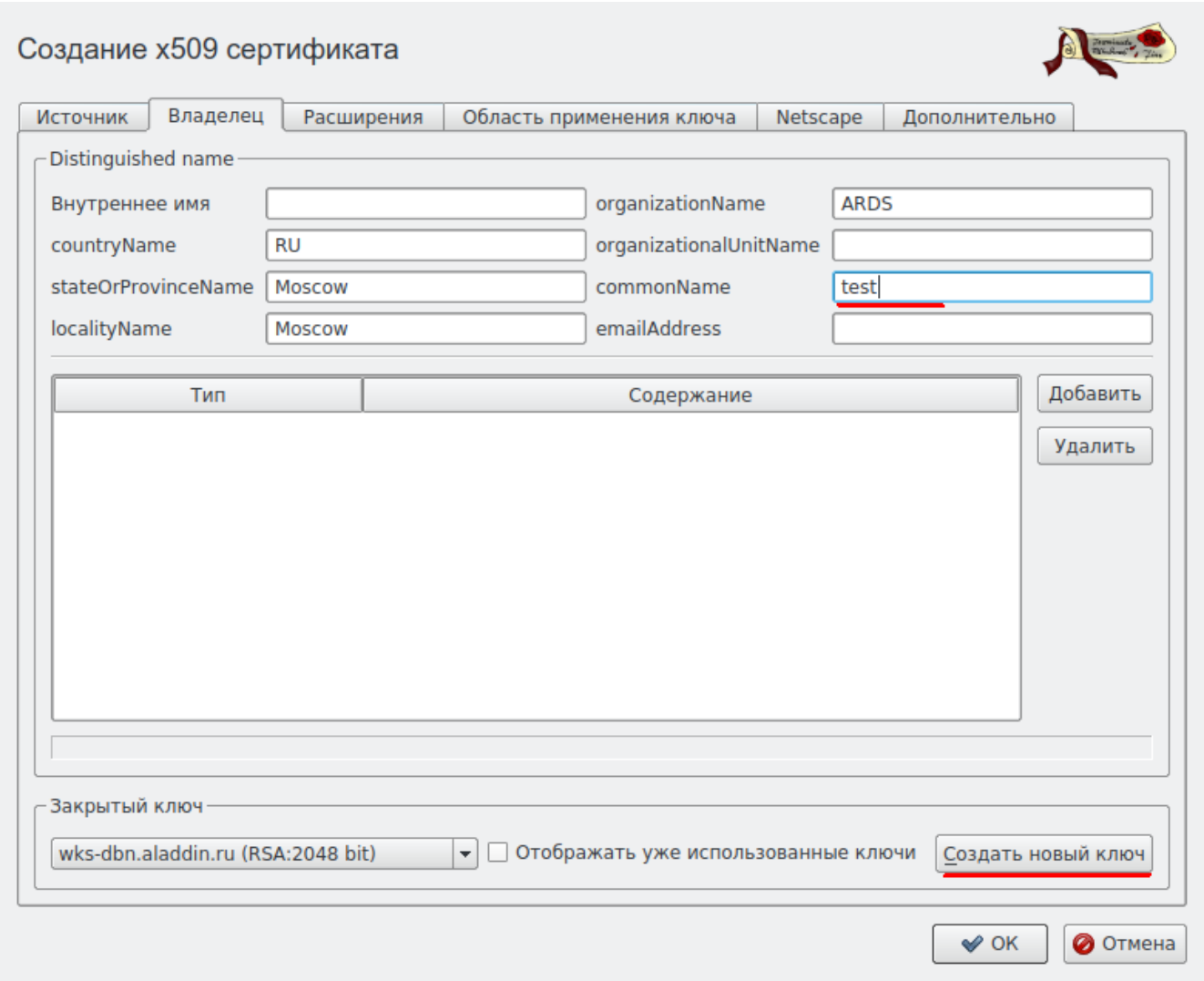
Выпускаем сертификат для пользователя.

Вкладка "Сертификаты", кнопка "Новый сертификат", вкладка "Источник"



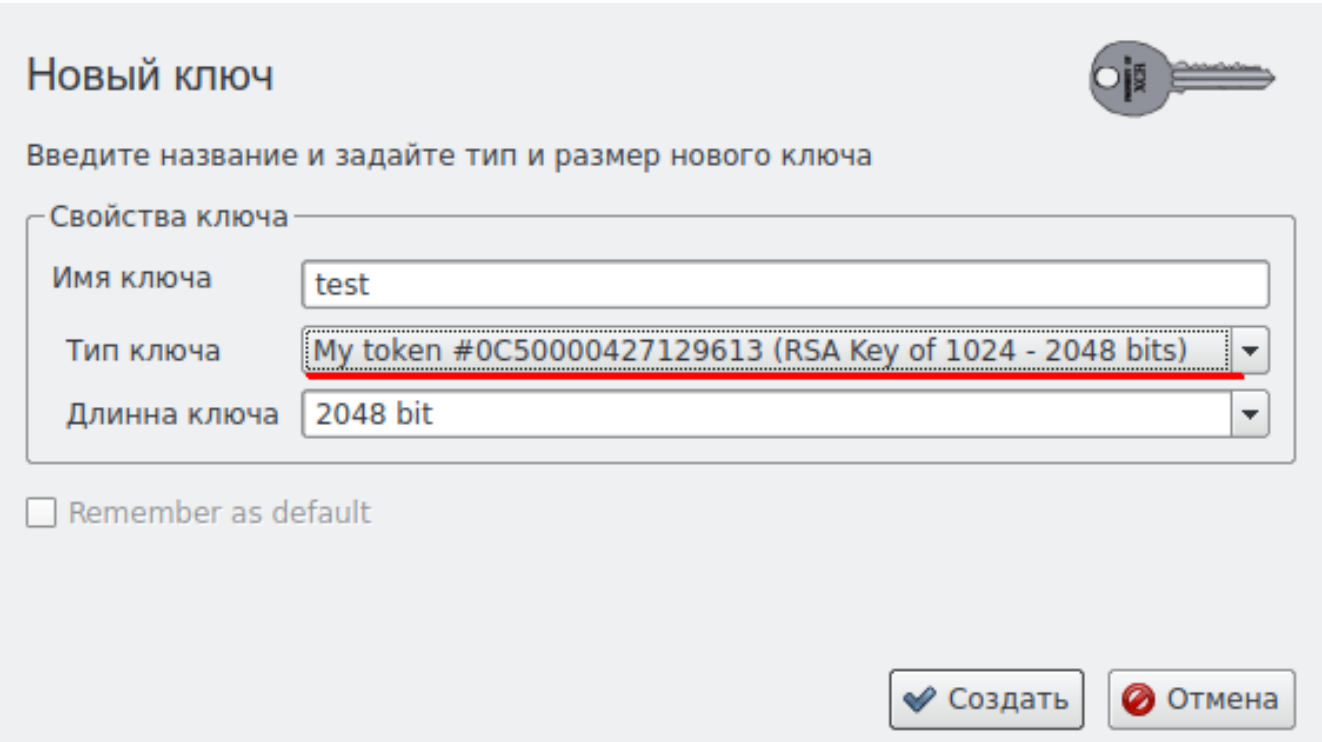
Выбираем "Use this Certificate for sighning", алгоритм подписи SHA 256, выбираем созданный ранее шаблон. Нажимаем "Применить всё".

Идём на вкладку "Владелец":



Заполняем необходимые поля. В поле **commonName** указываем имя нужного пользователя.

Далее нажимаем "Создать новый ключ".



В поле "Тип ключа" выбираем наш токен, алгоритм RSA. Нажимаем "Создать". После ввода пин-кода будет сгенерирована ключевая пара.

Далее нажимаем "ОК", соглашаемся, чтобы сертификат был сохранён на токене.

Теперь этот токен можно использовать для входа в систему.

5. Настройка pam.d и pam_pkcs11.

Для возможности входа по токену в pam.d необходимо добавить модуль pam_pkcs11.

Настраивается модуль в файле /etc/pam_pkcs11/pam_pkcs11.conf. Если файла нет, пример его можно взять по
ссылке: https://github.com/OpenSC/pam_pkcs11/blob/master/etc/pam_pkcs11.conf.example.in

В секцию **pam_pkcs11** добавляем модуль JaCarta:

```
pkcs11_module JaCarta {
        module = /usr/lib/libjcPKCS11-2.so;
        description = "JaCarta PKCS#11 module";
        slot_num = 0;
        support_threads = true;
        ca_dir = /etc/ssl;
        crl_dir = /etc/ssl/crl;
        cert_policy = ca,signature; }
```

Замечание: пути могут меняться в зависимости от дистрибутива и версий ПО.

Меняем параметр:

```
use_pkcs11_module = JaCarta;
```

Меняем файл pam.d. Находим в каталоге /etc/pam.d файл вашего менеджера, например lightdm. В начало файла вставим строку:

```
auth sufficient pam_pkcs11.so config_file=/etc/pam_pkcs11/pam_pkcs11.conf
```

*Замечание: если мы хотим исключить другие методы аутентификации, меняем директиву **sufficient** на **required**. Не рекомендуется экспериментировать на файлах типа login, common-auth и т.д.*

Далее при подключенном токене при входе в систему ввести пин-код (некоторые DM выдают явное сообщение, что нужно авторизоваться именно на токене) и войти в систему.

ID статьи: 227

Последнее обновление: 18 Oct, 2017

Ревизия: 1

JaCarta -> Вход в Linux по сертификату на JaCarta

https://kbp-6.aladdin-rd.ru/index.php?View=entry&EntryID=227