Аутентификация в РЕД ОС 7.3 при помощи RSA ключей на Рутокен ЭЦП

Подключите устройств семейства Рутокен ЭЦП к компьютеру.

Перед началом работы, установите следующие пакеты:

```
sudo dnf update
sudo dnf install ccid opensc pam_pkcs11 p11-kit
```

Загрузите модуль librtpkcs11ecp.so и установите:

```
sudo rpm -i librtpkcs11ecp-X.X.X.X-X.x86_64.rpm
```

Создание ключей и сертификатов

Проверьте наличие libpkcs11.so по пути: `/usr/lib64/engines-1.1/`. Если ее нет, то для начала установите libpkcs11.so для того, чтобы OpenSSL смог общаться к Рутокеном.

Способ 1

Для этого соберите библиотеку libp11 из репозитория. Вместе с ней идет libpkcs11.so начиная с версии 0.4.

Способ 2

Скачайте два пакета libp11 и engine_pkcs11 из репозитория Fedora и установите их с помощью команд:

Установка пакетов

```
sudo rpm-i <rpm_name>
```

Вы можете пропустить данный раздел, если у вас уже имеются необходимые RSA ключи.
Если ключей нет, ниже команда для их созданию:

```
pkcs11-tool --module /usr/lib64/librtpkcs11ecp.so --keypairgen --key-type rsa:2048 -l --id 45
```

Параметр id задает идентификатор ключевой пары.

Теперь нужно получить сертификат:

- создайте самоподписанный сертификат:

```
openssl

OpenSSL> engine dynamic -pre SO_PATH:/usr/lib64/engines-1.1/libpkcs11.so -pre ID:pkcs11 -pre LIST_ADD:1 -pre
LOAD -pre MODULE_PATH:/usr/lib64/librtpkcs11ecp.so

OpenSSL> req -engine pkcs11 -new -key 0:45 -keyform engine -x509 -out cert.crt -outform DER
```

- или создайте запрос на сертификат для передачи его в УЦ:

```
openssl

OpenSSL> engine dynamic -pre SO_PATH:/usr/lib64/engines-1.1/libpkcs11.so -pre ID:pkcs11 -pre LIST_ADD:1 -pre
LOAD -pre MODULE_PATH:/usr/lib64/librtpkcs11ecp.so

OpenSSL> req -engine pkcs11 -new -key 0:45 -keyform engine -out request.req
```

Сохраните сертификат на токене:

```
pkcs11-tool --module /usr/lib64/librtpkcs11ecp.so -l -y cert -w cert.crt --id 45
```

Проверьте, что токен подключен и на нем сохранены сертификаты и ключи.

Добавление сертификата в список доверенных

Создайте базу данных доверенных сертификатов

```
sudo mkdir /etc/pam_pkcs11/nssdb

sudo chmod 777 /etc/pam_pkcs11/nssdb

sudo certutil -d /etc/pam_pkcs11/nssdb -N #

sudo modutil -dbdir /etc/pam_pkcs11/nssdb/ -add p11-kit-trust -libfile /usr/lib64/pkcs11/p11-kit-trust.so
```

Выгрузите ваш сертификат с токена (если вы пользовались для получения сертификата вышеописанной инструкцией , то ID = 45):

```
pkcs11-tool --module=/usr/lib64/librtpkcs11ecp.so -l -r -y cert -d <ID> -o cert.crt
```

Добавьте сертификат в доверенные:

```
sudo cp cert.crt /etc/pki/ca-trust/source/anchors/ #    ,

sudo update-ca-trust force-enable

sudo update-ca-trust extract #
```

Настройка pam_pkcs11

Создайте (например, на рабочем столе) текстовый файл pam_pkcs11.conf со следующим содержимым:

```
pam_pkcs11 {
  nullok = false;
  debug = false;
  use_first_pass = false;
  use_authtok = false;
  card_only = false;
  wait_for_card = false;
  use_pkcs11_module = rutokenecp;

  # Aktiv Rutoken ECP
  pkcs11_module rutokenecp {
    module = /usr/lib64/librtpkcs11ecp.so;
    slot_num = 0;
    support_thread = true;
    ca_dir = /etc/pam_pkcs11/cacerts;
    crl_dir = /etc/pam_pkcs11/crls;
    cert_policy = signature;
  }

  use_mappers = digest;

  mapper_search_path = /usr/lib64/pam_pkcs11;

  mapper digest {
   debug = false;
   module = internal;
   algorithm = "sha1";
   mapfile = file:///etc/pam_pkcs11/digest_mapping;
  }

}
```

Поместите файл в каталог /etc/pam_pkcs11/:

```
cd /etc/pam_pkcs11/

sudo mv pam_pkcs11.conf pam_pkcs11.conf.default #

sudo mkdir cacerts crls

sudo cp /path/to/your/pam_pkcs11.conf /etc/pam_pkcs11/
```

Регистрация модуля PAM PKCS11 для аутентификации в системе

Подключите модуль к системе авторизации PAM:

```
sudo vim /etc/pam.d/system-auth
#
sudo vim /etc/pam.d/password-auth
```

Перед первым использованием модуля pam_unix добавьте туда строку со следующим содержимым:

```
auth   sufficient             pam_pkcs11.so pkcs11_module=/usr/lib64/librtpkcs11ecp.so
```

Сохраните файл и узнайте поля вашего сертификата с помощью следующей команды:

```
sudo pkcs11_inspect
```

В результате отобразится сообщение:

```
[user@redos ~]$ sudo pkcs11_inspect
PIN for token:
Printing data for mapper digest:
CB:13:CA:34:AC:04:CD:BF:A6:17:29:2F:C8:00:6A:D5:54:B8:0B:BB
```

Скопируйте строчку с описанием сертификата в файл /etc/pam_pkcs11/digest_mapping в формате:

```
<  pkcs11_inspect> -> <_>
```

Пример заполнения файла:

```
[user@redos ~]$ sudo cat /etc/pam_pkcs11/digest_mapping
CB:13:CA:34:AC:04:CD:BF:A6:17:29:2F:C8:00:6A:D5:54:B8:0B:BB -> user
```

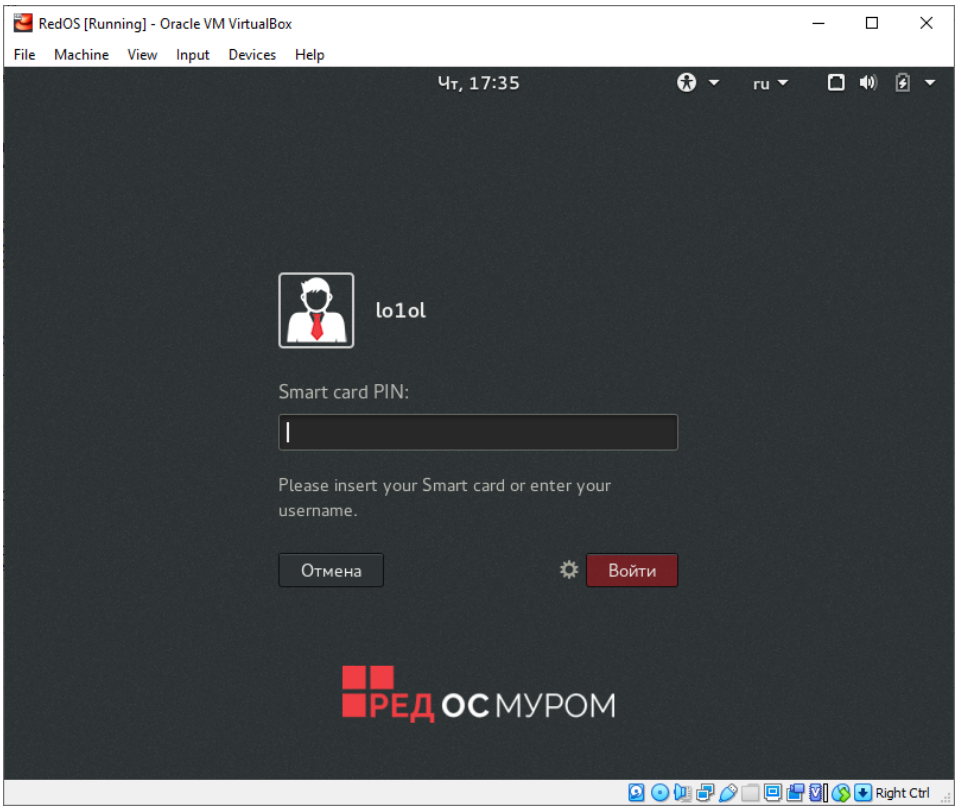
Попробуйте аутентифицироваться:

```
su <username>
```

Терминал должен запросить PIN код рутокена:

```
[user@redos ~]$ su user
Smart card found.
  Rutoken ECP <no label>!
Smart card PIN:
verifying certificate
Checking signature
[user@redos ~]$
```

В окне экрана приветствия аналогично:



Настройка автоблокировки

В состав пакета libpam-pkcs11 входит утилита pkcs11_eventmgr, которая позволяет выполнять различные действия при возникновении событий PKCS#11.

Для настройки pkcs11_eventmgr служит файл конфигурации - /etc/pam_pkcs11/pkcs11_eventmgr.conf

Пример файла конфигурации представлен ниже:

```
pkcs11_eventmgr
{
    #
    daemon = true;

    #
    debug = false;

    #
    polling_time = 1;

    #  -
    # - 0
    expire_time = 0;

    #  pkcs11
    pkcs11_module = /usr/lib64/librtpkcs11ecp.so;

    #
    #  :
    event card_insert {
        #     (  )
        on_error = ignore ;

        action = "/bin/false";
    }

    #
    event card_remove {
        on_error = ignore;

        #
        action = "cinnamon-screensaver-command --lock";
    }

    #
    event expire_time {
        #     (  )
        on_error = ignore;

        action = "/bin/false";
    }
}
```

После этого добавьте приложение pkcs11_eventmgr в автозагрузку и перезагрузите компьютер.

Для этого создайте файл /etc/xdg/autostart/smartcard-screensaver.desktop

```
[Desktop Entry]
Type=Application
Name=Smart Card Screensaver
Comment=Application to lock screen on smart card removal.
Exec=/usr/bin/pkcs11_eventmgr daemon
```