Аутентификация в CentOS 7 и Goslinux при помощи RSA ключей на Рутокен ЭЦП

Подключите устройство семейства Рутокен ЭЦП к компьютеру

Проверка работы Рутокен ЭЦП

Подключите Рутокен ЭЦП к компьютеру.

Убедитесь в том, что на USB-токене или считывателе для смарт-карт светится индикатор.

Откройте Terminal.

Для проверки корректности работы Рутокен ЭЦП 2.0 введите команду:

```
$ pcsc_scan
```

Если Рутокен ЭЦП 2.0 не работает, то в окне терминала отобразится сообщение об этом.

Если Рутокен ЭЦП 2.0 работает, то в окне терминала отобразится сообщение об этом.

Для остановки сервиса pcscd введите команду:

```
$ sudo service pcscd stop
```

Настройка системы

Перед началом работы, установите следующие пакеты:

```
sudo yum install ccid opensc pam_pkcs11 gdm-plugin-smartcard p11-kit

sudo yum remove coolkey
```

Загрузите модуль librtpkcs11ecp.so и установите

```
sudo rpm -i librtpkcs11ecp_1.9.15.0-1_x86_64.rpm
```

Создание ключей и сертификатов

Для начала установите engine_pkcs11.so для того, чтобы OpenSSL смог общаться с токеном. Для этого соберите библиотеку libp11 из репозитори
. Вместе с ней идет engine_pkcs11.so начиная с версии 0.4

Вы можете пропустить данный раздел, если у вас уже имеются необходимые RSA ключи

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

Теперь создайте самоподписанный сертификат:

```
openssl

OpenSSL> engine dynamic -pre SO_PATH:/usr/lib/x86_64-linux-gnu/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
```

Поместите его на токен

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

Проверьте, что токен подключен и сертификаты с ключами на нем имеются.

Добавление сертификата в список доверенных

```
pkcs11-tool --module /usr/lib64/librtpkcs11ecp.so -O -l
```

Создайте базу данных доверенных сертификатов

```
sudo mkdir /etc/pam_pkcs11/nssdb

sudo chmod 0644 /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 вышеописанной инструкцией для получения сертификата, 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 = true;

  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 = subject;


  mapper_search_path = /usr/lib64/pam_pkcs11;


  mapper subject {

        debug = true;

        module = internal;

        ignorecase = false;

        mapfile = file:///etc/pam_pkcs11/subject_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 /home/<_>/Desktop/pam_pkcs11.conf /etc/pam_pkcs11/
```

Регистрация модуля для аутентификации в системе

Подключите модуль к системе авторизации PAM:

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

Добавьте туда строку со следующим содержимым:

```
auth   sufficient pam_pkcs11.so

pkcs11_module=/usr/lib64/librtpkcs11ecp.so debug
```

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

```
sudo pkcs11_inspect
```

На выходе вы увидите сообщение:

```
[root@dc1 oleg]# pkcs11_inspect
PIN for token:
DEBUG:subject_mapper.c:116: Subject mapper started. debug: 1, mapfile: file:///etc/pam_pkcs11/subjec
t_mapping, icase: 0
Printing data for mapper subject:
E=o.mihailov@rosalinux.ru,CN=Mikhaylov Oleg Andreevich,OU=Programming,O=NTCIT ROSA,L=Moscow,ST=Mosco
w,C=RU
[root@dc1 oleg]#
```

Скопируйте строчку с описанием сертификата в файл /etc/pam_pkcs11/subject_mapping в формате

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

```
[oleg@dc1 ~]$ cat /etc/pam_pkcs11/subject_mapping
E=o.mihailov@rosalinux.ru,CN=Mikhaylov Oleg Andreevich,OU=Programming,O=NTCIT ROS
A,L=Moscow,ST=Moscow,C=RU -> oleg
[oleg@dc1 ~]$
```

Попробуйте аутентифицироваться

```
su <username>
```

Вывод будет примерно следующим:

```
 oleg@dc1 ~]$ su oleg
DEBUG:pam_config.c:238: Using config file /etc/pam_pkcs11/pam_pkcs11.conf
DEBUG:pkcs11_lib.c:182: Initializing NSS ...
DEBUG:pkcs11_lib.c:192: Initializing NSS ... database=/etc/pam_pkcs11/nssdb
DEBUG:pkcs11_lib.c:212: ...  NSS Complete
Please insert your Smart card or enter your username.
DEBUG:pam_pkcs11.c:304: username = [oleg]
DEBUG:pam_pkcs11.c:315: loading pkcs #11 module...
DEBUG:pkcs11_lib.c:237: Looking up module in list
DEBUG:pkcs11_lib.c:240: modList = 0x34757850 next = 0x34766460

DEBUG:pkcs11_lib.c:241: dllName= <null>

DEBUG:pkcs11_lib.c:240: modList = 0x34766460 next = 0x0

DEBUG:pkcs11_lib.c:241: dllName= p11-kit-trust.so

DEBUG:pkcs11_lib.c:287: loading Module explictly, moduleSpec=<library="/usr/lib64/librtpkcs11ecp.so" name="SmartCard"> module=
/usr/lib64/librtpkcs11ecp.so
DEBUG:pkcs11_lib.c:301: load module complete
DEBUG:pam_pkcs11.c:324: initialising pkcs #11 module...
Found the Smart card.
Добро пожаловать Rutoken ECP <no label>!
Smart card PIN:
DEBUG:pkcs11_lib.c:761: cert 0: found ((null)), "E=o.mihailov@rosalinux.ru,CN=Mikhaylov Oleg Andreevich,OU=Programming,O=NTCIT
 ROSA,L=Moscow,ST=Moscow,C=RU"
DEBUG:mapper_mgr.c:172: Retrieveing mapper module list
DEBUG:mapper_mgr.c:73: Loading static module for mapper 'subject'
DEBUG:subject_mapper.c:116: Subject mapper started. debug: 1, mapfile: file:///etc/pam_pkcs11/subject_mapping, icase: 0
DEBUG:mapper_mgr.c:197: Inserting mapper [subject] into list
DEBUG:pam_pkcs11.c:490: verifing the certificate #1
DEBUG:cert_vfy.c:34: Verifying Cert: (null) (E=o.mihailov@rosalinux.ru,CN=Mikhaylov Oleg Andreevich,OU=Programming,O=NTCIT ROS
A,L=Moscow,ST=Moscow,C=RU)
DEBUG:mapper.c:157: Using mapping file: 'file:///etc/pam_pkcs11/subject_mapping' to search 'E=o.mihailov@rosalinux.ru,CN=Mikha
ylov Oleg Andreevich,OU=Programming,O=NTCIT ROSA,L=Moscow,ST=Moscow,C=RU'
DEBUG:uri.c:588: parsing uri:
DEBUG:uri.c:252: protocol = [file]
DEBUG:uri.c:253: user = [(null)]
DEBUG:uri.c:254: password = [(null)]
DEBUG:uri.c:255: host = []
```

Такой подробный вывод можно отключить, убрав опцию debug для pam модуля в файле конфигурации /etc/pam.d/system-auth