

# Создание системы автоматизированной подписи rpm пакетов

## Подготовка в os windows (дока для технарей)

- Скачиваем Vagrant
- Скачиваем Oracle Virtualbox (скрипт писался на 6.0.12)
- Стартуем vm > vagrant up
- Логинимся на vm > vagrant ssh
- sudo su

Для подписания пакетов есть готовый vagrant файл на <https://gitlab.tools.russianpost.ru/private-cloud/gpg-sign> и скрипты для запуска служб, конфиги служб, скрипт для подписи пакетов.

Сами пакеты для подписи нужно расположить в поддиректории `./rpm`, относительно `vagrantfile`, подписанные пакеты будут перемещены в поддиректорию `./signed`

Если запуск происходит в первый раз необходимо запустить `./gen_opengpg.sh` и импортировать ключ.

### - gen\_key

На текущий момент для импорта, необходимо заполнить поля `name` и `comment`, поле `email` берется с ключа (из поля `object` сертификата `x.509`)

Скрипт импортирует ключ в `gpg` базу, затем экспортирует его в `rpm` базу и создаст на его основе `~/.rpmmacros`.

В `~/.rpmmacros` заполняется поле `~/gpg_name`, где указывается идентификатор ключа (достаточно указать `email`)

### - sign\_rpm

Подпишет пакеты `rpm` ключем который указал в `~/.rpmmacros`

Команда может запросить ввести PIN.

После выполнения скрипта должны увидеть на подобие такого

```
/mnt/12.rpm
Header V4 RSA/SHA512 Signature, key ID b6ea8c0d: OK
/mnt/librtpkcs11ecp-1.9.15.0-1.x86_64.rpm
Header V4 RSA/SHA512 Signature, key ID b6ea8c0d: OK
```

Экспорт ключей на Token выполняется скриптом `./export_key.sh` относительно `vagrantfile`.

В поддиректории `./keys` расположить два файла:

```
keys.der - private rsa ключ в формате DER
cert.der - x.509 сертификат в формате DER
```

Если у нас есть уже готовый `rsa private` ключ и `x.509`, то можно их загрузить на Token. пункт 1.3

Важный момент, перед тем как копировать файлы необходимо инициализировать ключ.

- `pkcs11-tool --module /usr/lib/librtpkcs11ecp.so --init-token--label mytoken`
- `pkcs11-tool --module /usr/lib/librtpkcs11ecp.so --init-pin --login`

- `rsa x.509 Token.`  
1 :

```
openssl genrsa -out keys.pem 2048
```

1.1 Создаем самоподписанный сертификат:

```
$ openssl req -new -key keys.pem -out cert.csr
$ openssl x509 -req -days 700 -incert.csr -signkey keys.pem -out cert.cert
```

1.2 Конвертируем ключи и сертификат в DER-формат:

```
$ openssl rsa -inform PEM -inkeys.pem -out keys.der -outform DER
$ openssl x509 -incert.cert -out cert.der -outform der
```

### 1.3 Импортируем ключ и сертификат в DER-формате на Рутокен:

```
$ pkcs11-tool --module /usr/lib/librtpkcs11ecp.so -l -y privkey -w keys.der --id10 --label Rutoken1
$ pkcs11-tool --module /usr/lib/librtpkcs11ecp.so -l -y cert -w cert.der --id10 --label Rutoken1
```

Источник <https://dev.rutoken.ru/pages/viewpage.action?pageId=3440675>

P.S. основа для разработки

- <https://craftware.xyz/securitybricks/2017/07/17/using-tokens-in-Ubuntu-with-pgp.html>