Локальная аутентификация по Рутокен ЭЦП в Ubuntu/Debian

Проверка модели устройства

1. Подключите USB-токен к компьютеру.
2. Для определения названия модели USB-токена откройте **Терминал** и введите команду:

```
lsusb
```

В результате в окне Терминала отобразится название модели USB-токена:

```
[dmitrieva@localhost ~]$ lsusb
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 004: ID 0a89:0030 Aktiv Rutoken ECP
Bus 002 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 002 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
```

Убедитесь в том, что используете: **Aktiv Rutoken ECP**.

Введение

Pluggable Authentication Modules (PAM, подключаемые модули аутентификации) — это набор разделяемых библиотек, которые позволяют интегрировать различные низкоуровневые методы аутентификации в виде единого высокоуровневого API. Это позволяет предоставить единые механизмы для управления, встраивания прикладных программ в процесс аутентификации.

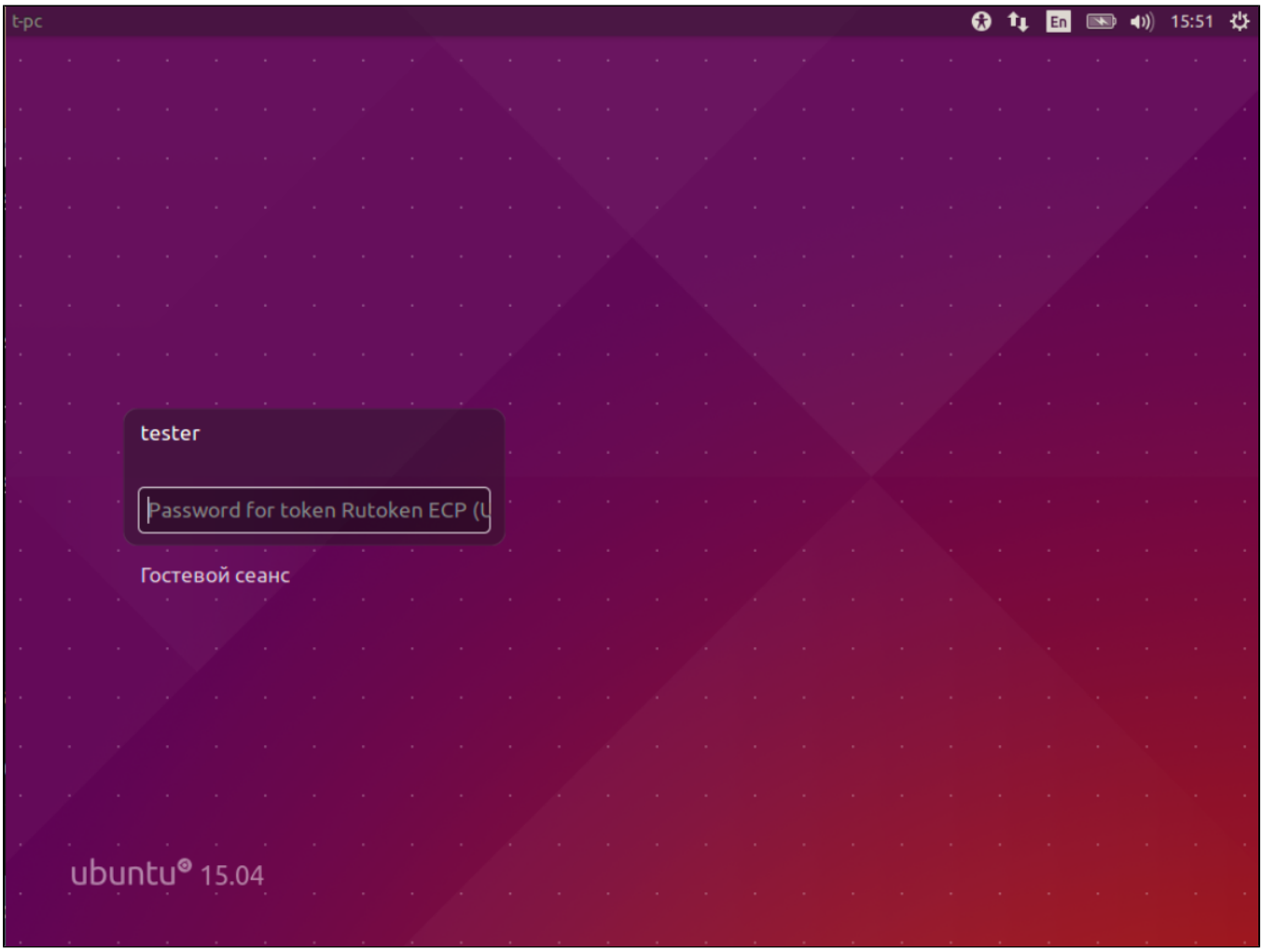
Для PAM существует проект pam_p11, развивающийся как часть OpenSC, позволяющий внедрить аутентификацию по токенам. В старых версиях (ниже 0.2.0) модуль аутентификации разделен на два:

1. `pam_p11_openssh`: позволяет аутентифицировать пользователя по открытым ключам ssh в `~/.ssh/authorized_keys`
2. `pam_p11_opensc`: аутентификация по сертификатам из файла `~/.eid/authorized_certificates`

Проект OpenSC также предоставляет пакет pam_pkcs11, который представляет собой более гибкий набор модулей аутентификации.
Мы рассмотрим конфигурацию pam_p11, как наиболее простое решения. Однако, pam_p11 не имеет понятия о цепочках сертификатов, списках отзыва и OCSP. При необходимости, pam_pkcs11 можно настроить руководствуясь инструкцией (на английском, ее источник). Также имеется инструкция для Gentoo (источник).
Общий порядок действий для настройки PAM следующий:

1. Сгенерировать на токене ключевую пару RSA.
2. Если требуется сертификат, то с помощью OpenSSL или другого ПО сгенерировать сертификат и записать его на токен.
3. Записать открытый ключ или сертификат в необходимый каталог.

В итоге выглядит это так:



Предварительная подготовка

Демонстрация работы проводится на Ubuntu 18.04. Описанная последовательность действий актуальна также для других версий Ubuntu и систем, основанных на Debian.

Для конфигурации модуля PAM необходимо установить пакеты:

```
sudo apt-get install pcscd opensc openssl libpam-p11 libengine-pkcs11-openssl
```

Общий порядок действий

Настройка pam_p11

До начала работы с токеном стоит настроить модуль pam_p11:

1. Создать файл `/usr/share/pam-configs/p11` со следующим содержанием:
    - Для версий libpam-p11 ниже 0.2.0

```
Name: Pam_p11
Default: yes
Priority: 800
Auth-Type: Primary
Auth: sufficient pam_p11_opensc.so /usr/lib/librtpkcs11ecp.so
```

- Для версий libpam-p11 выше или равных 0.2.0

```
Name: Pam_p11
Default: yes
Priority: 800
Auth-Type: Primary
Auth: sufficient pam_p11.so /usr/lib/librtpkcs11ecp.so
```

2. Обновить конфигурацию PAM:

```
sudo pam-auth-update
```

3. В появившемся диалоге необходимо удостовериться, что выбран pam_p11. Если вы хотите отключить аутентификацию по паролям, то можно отключить Unix authentication.

Создание ключей на токене

1. Создаем ключевую пару RSA длины 2048 бит с ID "45" (id стоит запомнить, он понадобится при создании сертификата). Аутентификация на токене происходит под сущностью пользователя.

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

2. Проверим сгенерированный ключ:

```
$ pkcs11-tool --module /usr/lib/librtpkcs11ecp.so -O
```

Создание сертификата и импорт его на токен

3. Запускаем `openssl`

```
$ openssl
```

4. Формируем самоподписанный сертификат или заявку на сертификат:

```
OpenSSL> engine dynamic -pre SO_PATH:/usr/lib/x86_64-linux-gnu/engines-1.1/pkcs11.so -pre ID:pkcs11 -pre
LIST_ADD:1 -pre LOAD -pre MODULE_PATH:librtpkcs11ecp.so
...
OpenSSL> req -engine pkcs11 -x509 -new -key 45 -keyform engine -out client.req -subj "/C=RU/ST=Moscow
/L=Moscow/O=Aktiv/OU=dev/CN=testuser/emailAddress=testuser@mail.com"
```

Если вы используете не Ubuntu 18.04, вам необходимо проверить местонахождение pkcs11.so. Он может располагаться, например, в /usr/lib/openssl/engines/. Если его найти не удается воспользуйтесь командой find.

Примечание

Если при создании сертификата в OpenSSL убрать ключ `-x509,` то на выходе получим заявку на сертификат.

5. Сохраняем сертификат на токен:

```
$ pkcs11-tool --module /usr/lib/librtpkcs11ecp.so -l -y cert -w ./client.pem --id 45
```

Занесение сертификата в список доверенных

6. Теперь нам необходимо записать его в файл доверенных сертификатов:

```
mkdir ~/.eid
chmod 0755 ~/.eid
cat client.pem >> ~/.eid/authorized_certificates
chmod 0644 ~/.eid/authorized_certificates
```

Теперь при загрузке Ubuntu мы можем использовать токен для аутентификации.

Примечание

На стадии выбора пользователя информация о подключенном токене может не обновляться динамически. Если вы подключили токен и не видите поля ввода PIN-кода, вам может понадобиться перенести фокус на "гостевой сеанс" и обратно на вашего пользователя.