

# Настройка двухфакторной аутентификации в macOS Catalina и Big Sur

## Обеспечение поддержки входа в macOS по токену или смарт-карте Рутокен

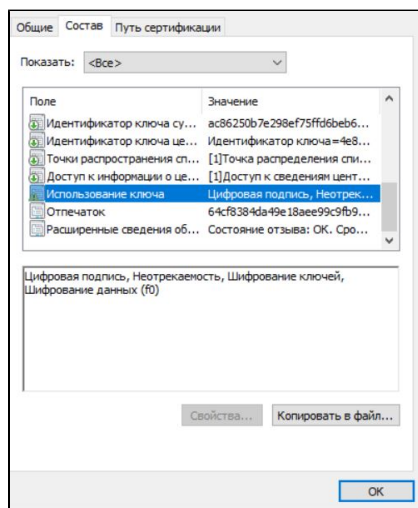
В качестве примера настраивается аутентификация для учетной записи *user* сертификатом *Шавровой Ксении*.

Инструкция применима к macOS версии 10.15 Catalina

### Предварительная подготовка.

Для аутентификации в системе используется сертификат RSA (1024 бит или больше), выписанный на Рутокен ЭЦП 2.0.

Обратите внимание, что в области применения сертификата должно быть также указано Цифровая подпись, Шифрование данных и Шифрование ключей.



Также вам понадобится корневой сертификат (в виде .cer файла). Для аутентификации в macOS может использоваться любое из следующих устройств:

- Рутокен ЭЦП 2.0
- Рутокен ЭЦП РК1
- Рутокен ЭЦП 2.0 2100
- Рутокен ЭЦП 2.0 3000
- Смарт-карты Рутокен ЭЦП SC
- Смарт-карты Рутокен ЭЦП 2.0
- Рутокен ЭЦП 2.0 Bluetooth (при подключении по USB-кабелю)

Начиная с macOS 10.15 Catalina был компанией Apple был изменен механизм работы с токенами. Теперь сертификаты с токенов в "Связке ключей (KeyChain)" больше не видны.

### Настройка аутентификации

1. Установите на компьютер под управлением macOS программу "[Рутокен для macOS](#)".

После установки этой программы сертификаты с устройства Рутокен смогут использоваться внутри приложений, которые обновили механизм работы с токенами для macOS Catalina.

2. Перезагрузите компьютер

3. Подключите Рутокен с сертификатом

4. Операции по настройке аутентификации будут производиться через "Терминал" (Terminal), его можно найти в "Программы"/"Утилиты" (Applications/Utilities).

5. Первоначально необходимо активировать возможность аутентификации по смарт-картам с помощью команды:

```
sudo security authorizationdb smartcard enable
```

```
user — -zsh — 80x24
Last login: Mon Mar 30 15:05:38 on ttys000
user@MacBook-Pro-user ~ % sudo security authorizationdb smartcard enable
Password:
YES (0)
user@MacBook-Pro-user ~ % █
```

В случае успешного выполнения, команда возвращает YES (0)

6. Далее нужно определить хэш сертификата:

```
sc_auth identities
```

Будет выведен список сертификатов и соответствующих им хешей, в нашем случае он один:

```
user — -zsh — 80x24
Last login: Mon Mar 30 15:09:26 on ttys000
user@MacBook-Pro-user ~ % sc_auth identities
SmartCard: ru.rutoken.RutokenApp.RutokenCTK:Rutoken ECP <no label>(955285620)
Unpaired identities:
14AAFC35DEE22C475C9750A2E180140D755F8B92          Шаврова Ксения
user@MacBook-Pro-user ~ % █
```

7. Свяжем пользователя с приватным ключом на токене по хешу сертификата, выполнив

```
sudo sc_auth pair -u user -h hash
```

Где в качестве *hash* используется хеш сертификата, находящегося на токене. Его можно скопировать любым удобным способом из выдачи предыдущей команды. В качестве *user* нужно использовать имя вашего пользователя (в данном случае - *user*):

```
user — -zsh — 80x24
user@MacBook-Pro-user ~ % sc_auth identities
SmartCard: ru.rutoken.RutokenApp.RutokenCTK:Rutoken ECP <no label>(955285620)
Unpaired identities:
14AAFC35DEE22C475C9750A2E180140D755F8B92          Шаврова Ксения
user@MacBook-Pro-user ~ % sudo sc_auth pair -u user -h 14AAFC35DEE22C475C9750A2E180140D755F8B92
Password:
Failed to store Login keychain unlock key. На выбранной смарт-карте не найден подходящий ключ.
User was successfully paired (public key hash: 14AAFC35DEE22C475C9750A2E180140D755F8B92) but user password will be required after next SmartCard login to unlock Login keychain.
```

Проверить корректность привязки можно с помощью команды:

```
sc_auth list -u user
```

Команда должна вернуть указанный нами хеш.

8. Также необходимо сделать установку для пользователя `_securityagent`

```
cd /Applications/\ \ macOS.app/Contents/PlugIns
sudo launchctl asuser _securityagent pluginkit -a "RutokenCTK.appex"
```

9. В "Связке ключей" (Keychain) необходимо разместить корневой сертификат в связке "Система" (System). Сделать это можно перетаскиванием сертификата в окошко связки "Система".

**Связки ключей**

**Вход**

Локальные объекты

Система

Центры...ртификации

**Rutoken TEST CA (P) RSA**  
 Самоподписанный корневой сертификат  
 Истекает: понедельник, 17 сентября 2029 г. в 17:29:54 Москва, стандартное время  
 ⚠ Сертификат не был проверен третьей стороной

| Имя                     | Тип                 | Дата изменения             | Срок действия              | Связка кл |
|-------------------------|---------------------|----------------------------|----------------------------|-----------|
| 1                       | Пароль сети AirPort | 27 марта 2020 г., 15:55:11 | --                         | Система   |
| 1                       | Пароль сети AirPort | 27 марта 2020 г., 18:40:55 | --                         | Система   |
| com.apple.kerberos.kdc  | Сертификат          | --                         | 22 марта 2040 г., 15:56:22 | Система   |
| com.apple.kerberos.kdc  | Открытый ключ       | --                         | --                         | Система   |
| com.apple.kerberos.kdc  | Закрытый ключ       | --                         | --                         | Система   |
| com.apple.systemdefault | Сертификат          | --                         | 22 марта 2040 г., 15:56:21 | Система   |
| com.apple.systemdefault | Открытый ключ       | --                         | --                         | Система   |
| com.apple.systemdefault | Закрытый ключ       | --                         | --                         | Система   |
| Rutoken TEST CA (P) RSA | Сертификат          | --                         | 17 сент. 2029 г., 17:29:54 | Система   |

Категория

Все объекты

Пароли

Секретные заметки

Мои сертификаты

Ключи

Сертификаты

10. Для проверки корректности работы аутентификации извлечем токен Рутокен и перезагрузим систему.

Первоначально мы увидим стандартный запрос пароля, однако с подключением токена вместо "пароль" в окошке ввода появится "PIN".

**Чтобы открыть экран, необходимо ввести пароль.**

Имя пользователя:

PIN:

\*Если аутентификация по смарт-карте невозможна после включения/перезагрузки, ознакомьтесь с [инструкцией](#).