
Яндекс

Яндекс

Авторизация
и не только

Роман Парадеев

Идентификация, аутентификация, авторизация

Теория

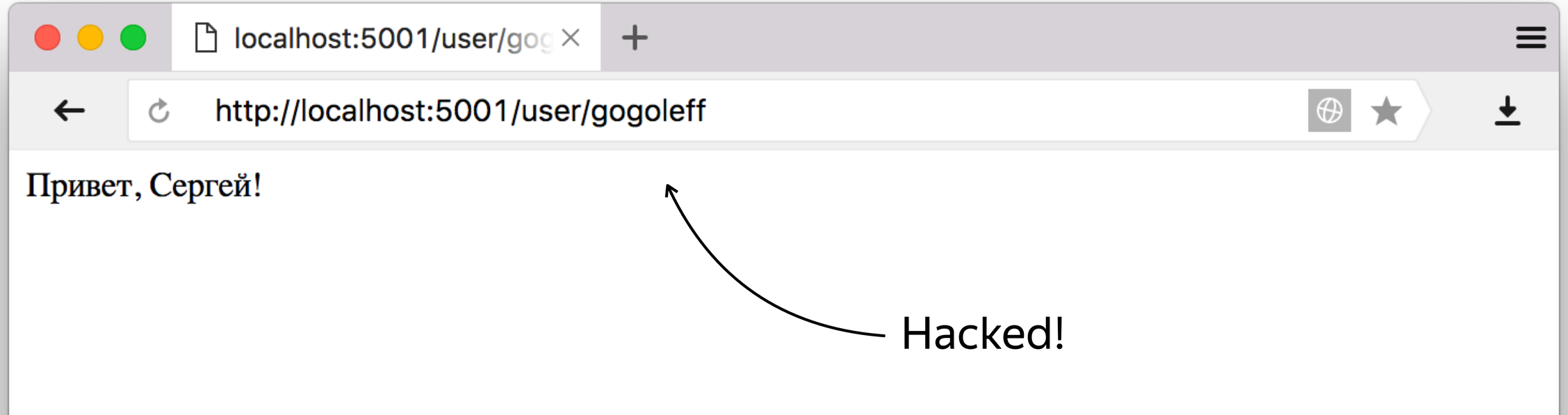
Идентификация

Процедура нахождения пользователя
по уникальному идентификатору

Идентификация

Процедура нахождения пользователя
по уникальному идентификатору

```
const users = {   grumpy: { name:  'Роман' },
                  gogoleff: { name: 'Сергей' }}

app.get('/user/:login', (req, res) ⇒ {
    const user = users[req.params.login]
    res.send(`Привет, ${user.name}!`)
})
```


localhost:5001/user/gog
http://localhost:5001/user/gogoleff
Привет, Сергей!
Hacked!

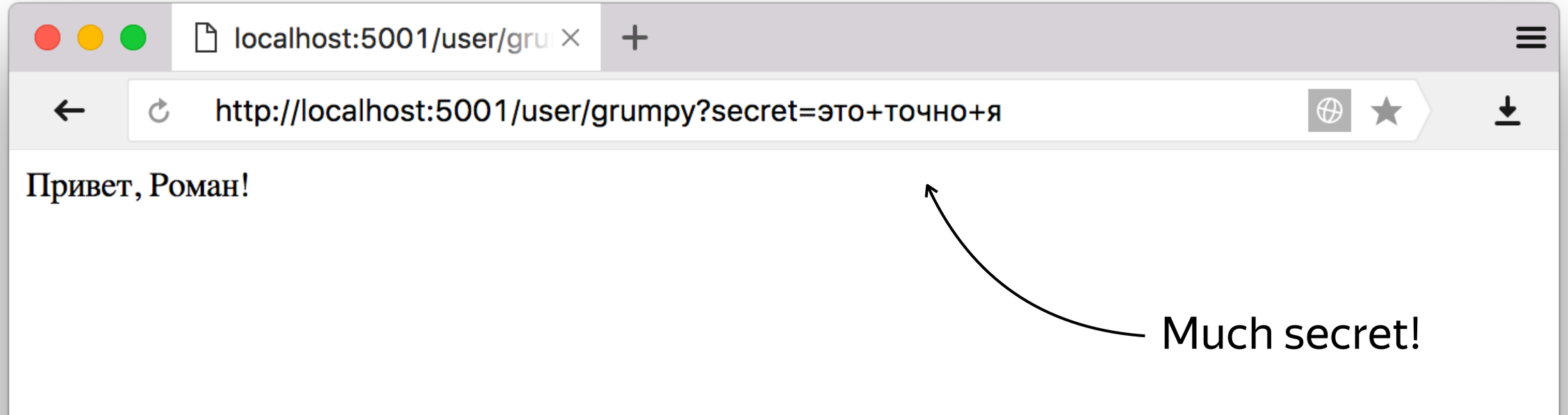
Аутентификация

Процедура проверки подлинности пользователя
например, путём сравнения введённого им пароля
с паролем, сохранённым в базе данных

Аутентификация

Процедура проверки подлинности пользователя

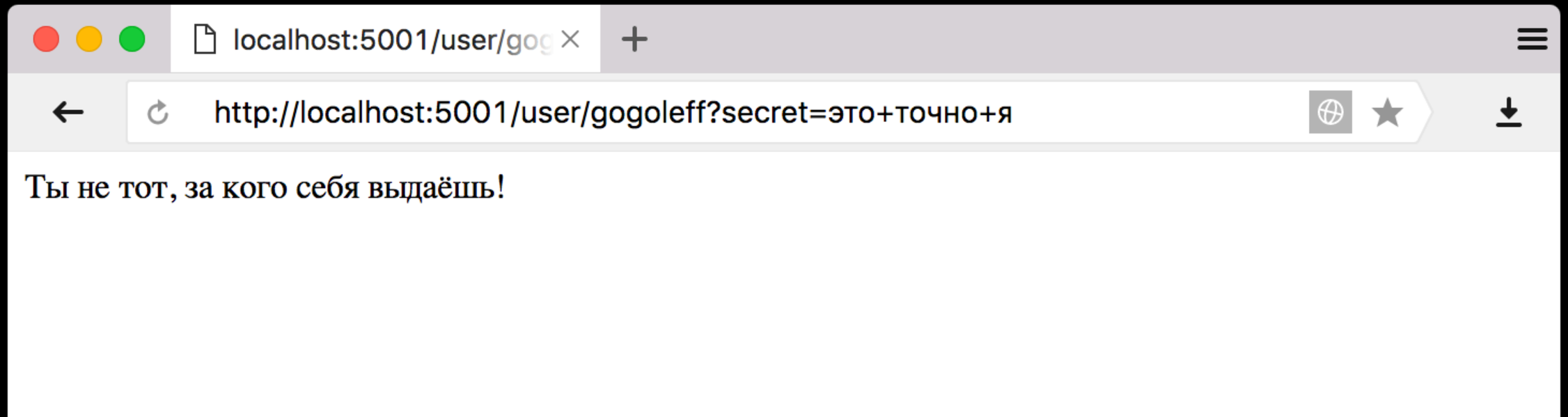
например, путём сравнения введённого им пароля
с паролем, сохранённым в базе данных



Аутентификация

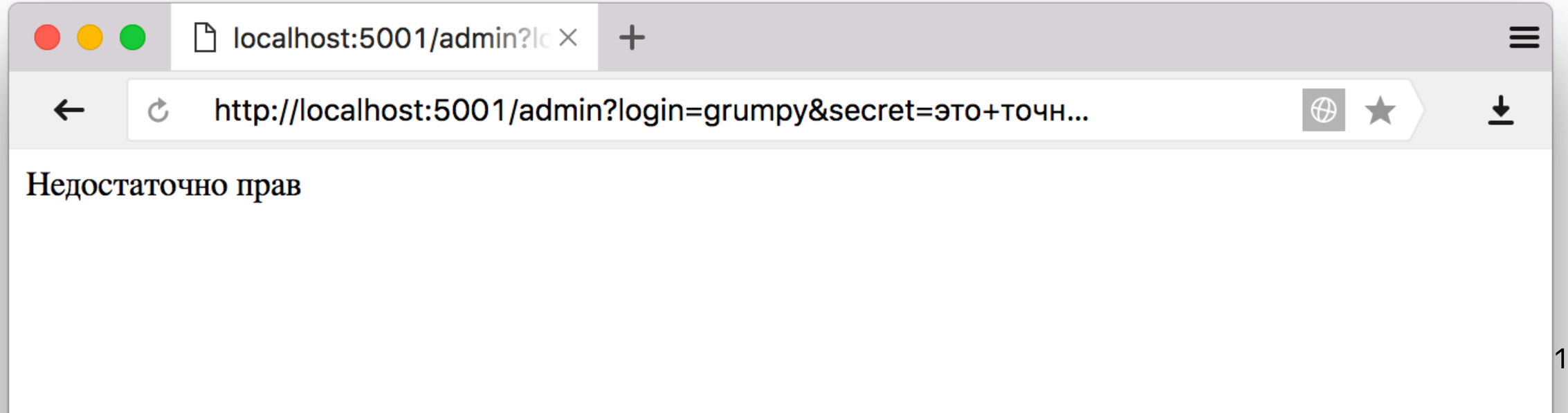
```
const users = {   grumpy: { name:  'Роман', secret: 'это точно я' },
                  gogoleff: { name: 'Сергей', secret: 'это не он'   }}

function findUser(login, secret) {
    if (!secret)
        throw new Error('Это точно ты?')
    if (!users[login])
        throw new Error('Я тебя не знаю!')
    if (users[login].secret !== secret)
        throw new Error('Ты не тот, за кого себя выдаёшь!')
    return users[login] }
```

```
app.get('/user/:login', (req, res) ⇒ {
    try {
        const user = findUser(req.params.login, req.query.secret)
        res.send(`Привет, ${user.name}!`)
    } catch (e) {
        res.send(403, e.message)
    }
})
```



Авторизация

Процесс подтверждения прав пользователя на выполнение определённых действий



```
const permissions = {   grumpy: [/* none */],
                      gogoleff: ['/admin'] }

function authorized(req, res, next) {
    const user = findUser(req.params.login, req.query.secret)
    if (!permissions[login].includes(req.route.path)) {
        res.send(403, 'Недостаточно прав')
    }
    next()
}

app.get('/admin', authorized, (req, res) ⇒ {
    res.send('Админка')
})
```

One More Time With Feeling

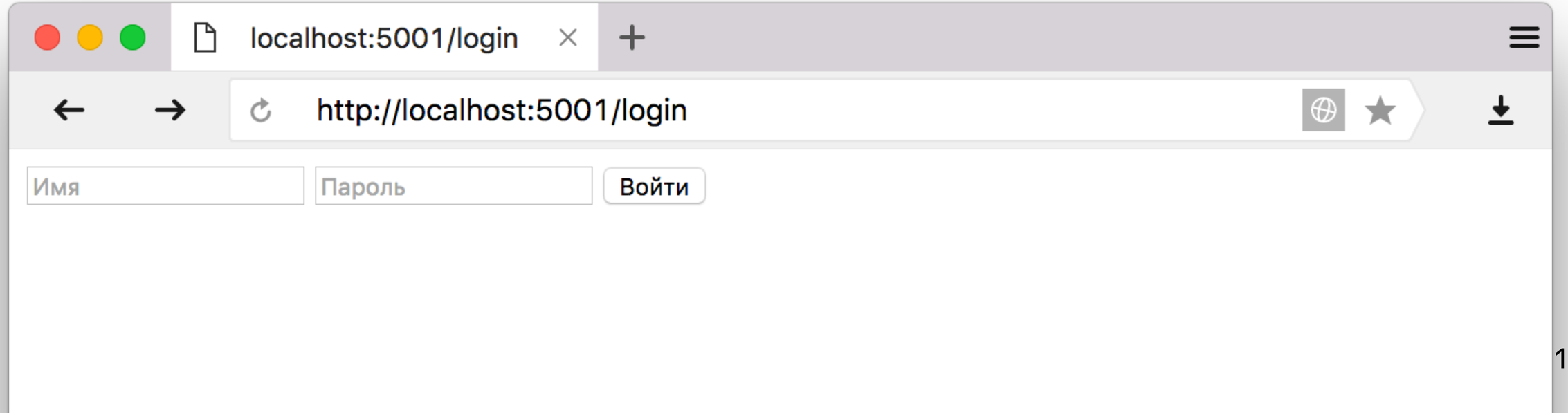
- Идентификация wiki
  - Кто ты?
- Аутентификация wiki
  - Ты точно тот, за кого себя выдаёшь?
- Авторизация wiki
  - Есть ли у тебя право на это действие?

Сессии, регистрация, passport

Практика

Сессия

Механизм, позволяющий хранить на сервере данные, связанные с сеансом пользователя.

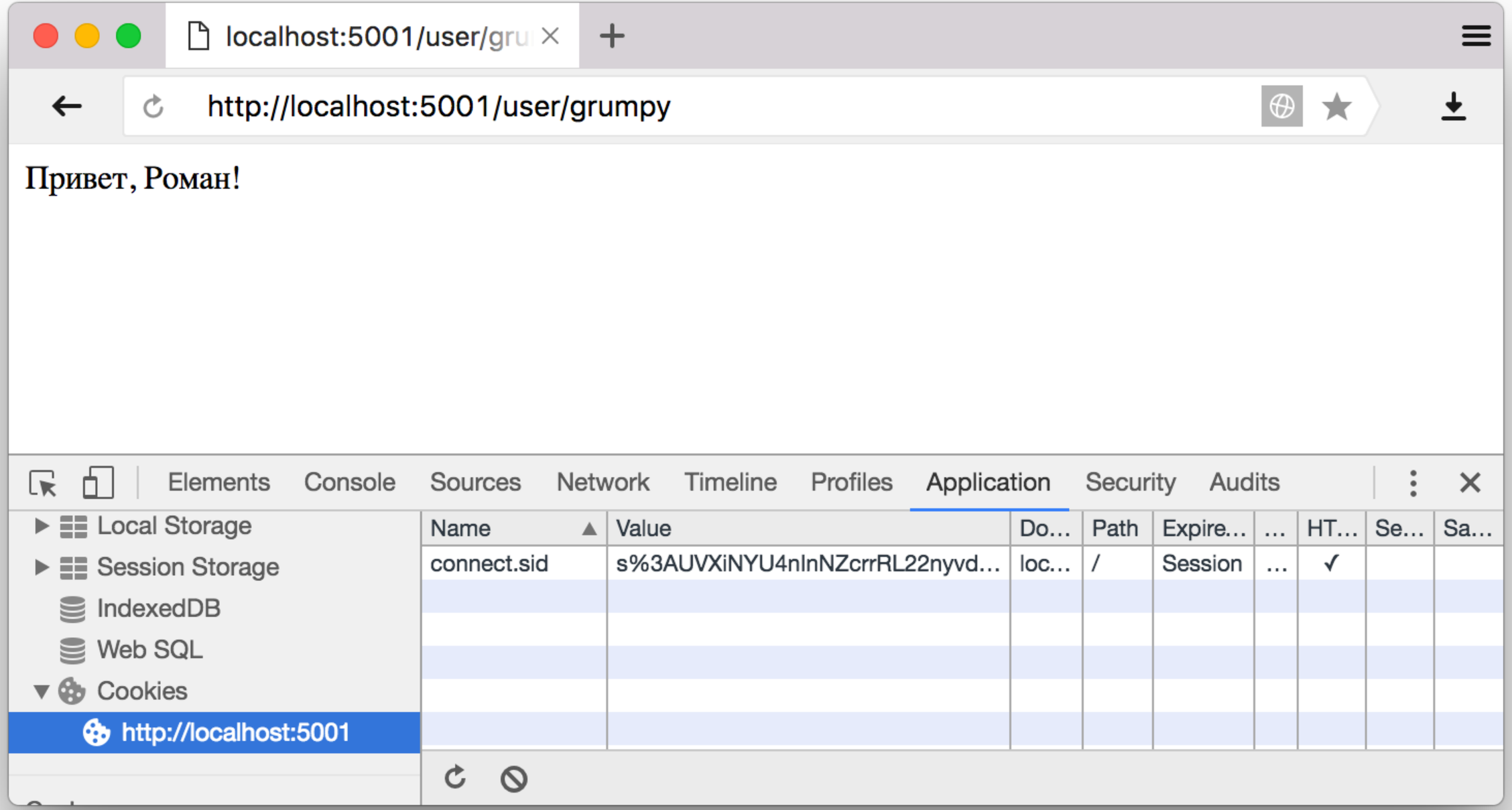


Сессия

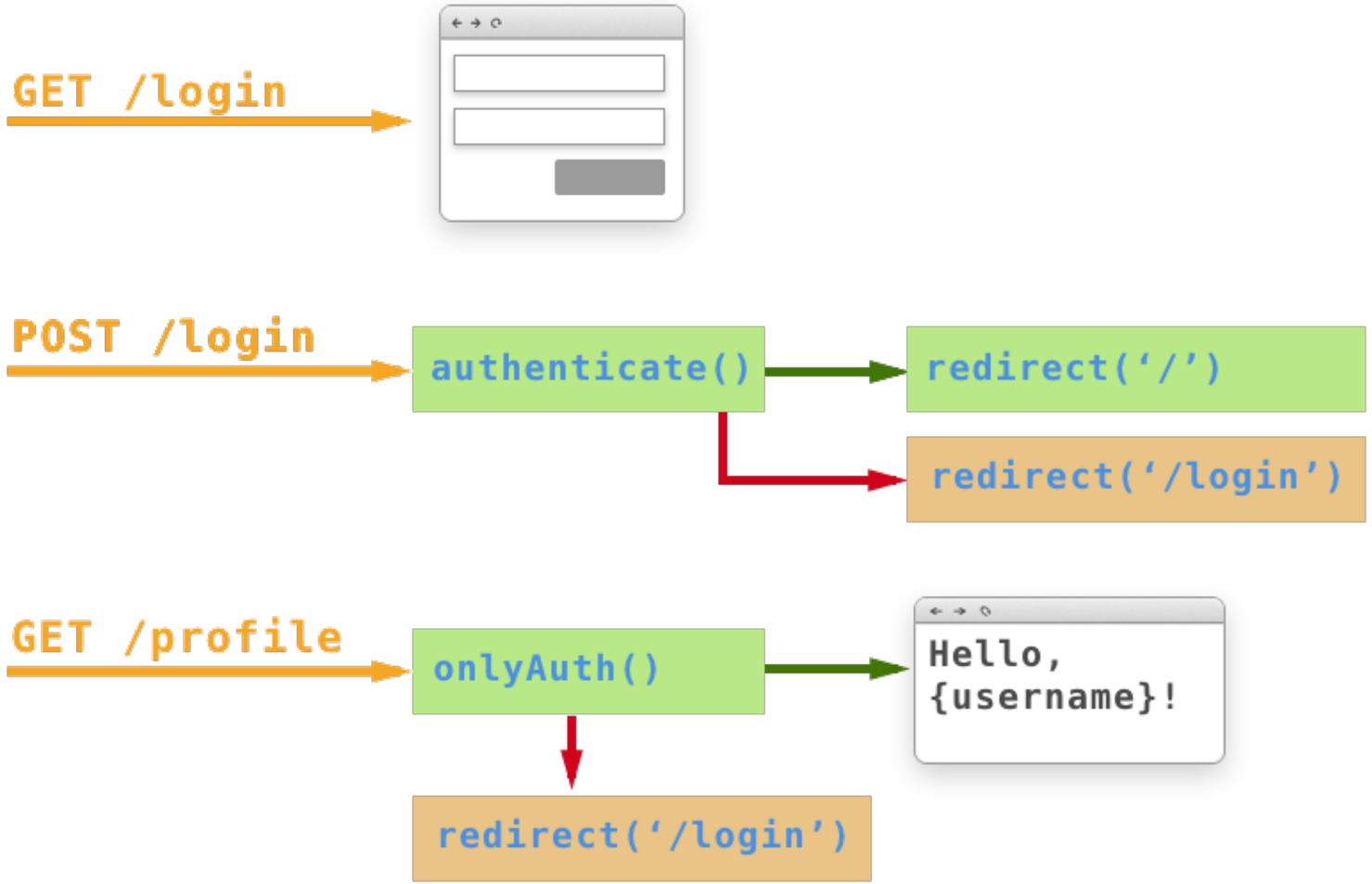
```
const session = require('express-session')
const { urlencoded } = require('body-parser')

app.use(urlencoded({ extended: false }))
app.use(session({
  secret: 'secret',
  resave: false,
  saveUninitialized: false,
}))
```

```
app.get('/login', (req, res) ⇒ {
    res.send(`<form method="POST">
                  <input name="login">
                  <input name="password" type="password">
                  <button>Войти</button>
              </form>`) })

app.post('/login', (req, res) ⇒ {
   try {
        const user = findUser(req.body.login, req.body.password)
        req.session.user = req.body.login
        res.redirect(`/user/${req.body.login}`)
    } catch (e) {
        res.redirect('/login')
    }})
```

```
function authenticated(req, res, next) {
    if (!req.session.user) {
        res.redirect('/login')
    }
    req.user = users[req.session.user]
    next()
}

app.get('/user/:login', authenticated, (req, res) ⇒ {
   try {
       res.send(`Привет, ${req.user.name}!`)
   } catch (e) {
       res.send(403, e.message)
   }
})
```


localhost:5001/user/gru
http://localhost:5001/user/grumpy
Привет, Роман!
Elements
Console
Sources
Network
Timeline
Profiles
Application
Security
Audits
Local Storage
Session Storage
IndexedDB
Web SQL
Cookies
http://localhost:5001
Name
Value
Do…
Path
Expire…
…
HT…
Se…
Sa…
connect.sid
s%3AUVXiNYU4nInNZcrrRL22nyvd…
loc…
/
Session
…
✓

Приложение «Заметки»



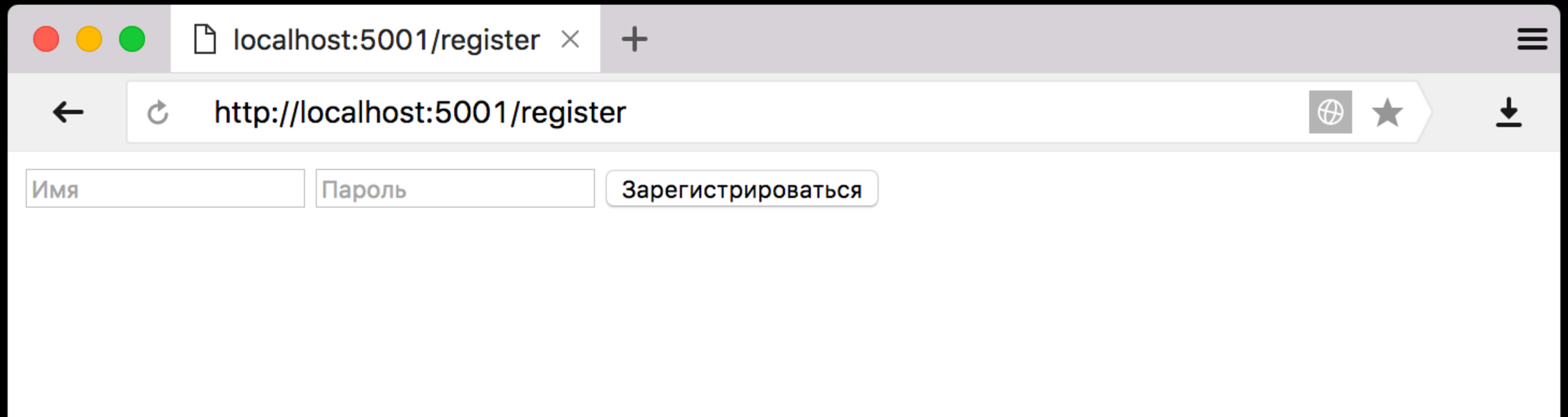
passport.js

- хэлперы для распространённых сценариев
- обработка ошибок
- работа с сессиями
- поддержка различных стратегий аутентификации

```
app.post('/login',
  passport.authenticate('local', { successRedirect: '/',
                                   failureRedirect: '/login' }))
```

Регистрация нового пользователя

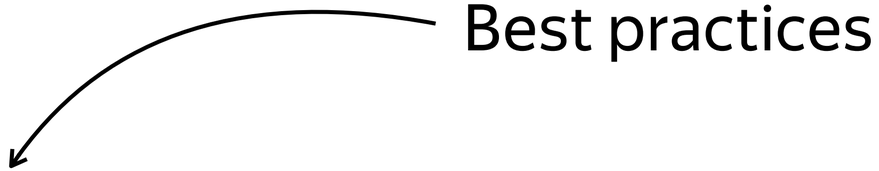
```
GET /register    // отдаёт форму регистрации
POST /register   // создаёт нового пользователя
```



Пароль нельзя хранить в открытом виде!

Хранение паролей

- хэширование (например, sha256; md5 – не подходит!)
- соль (случайная, генерируется для каждой записи)
- несколько итераций (PBKDF2, bcrypt)



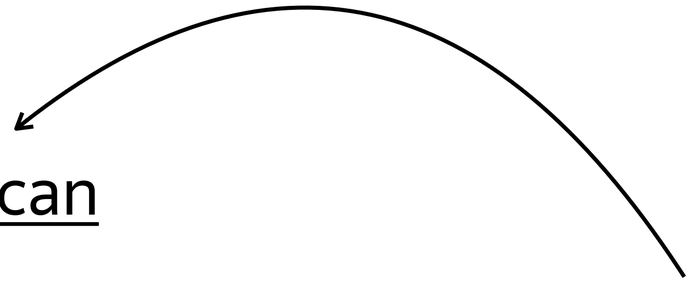
How To Store Your User's Passwords Safely

Хранение паролей

```
const bkfd2Password = require('pbkdf2-password')
const hasher = bkfd2Password({
    iterations: 10000,
    digest: 'sha512'
})

hasher({ password }, (err, pass, salt, hash) ⇒ {
    user.salt = salt
    user.hash = hash
})
```

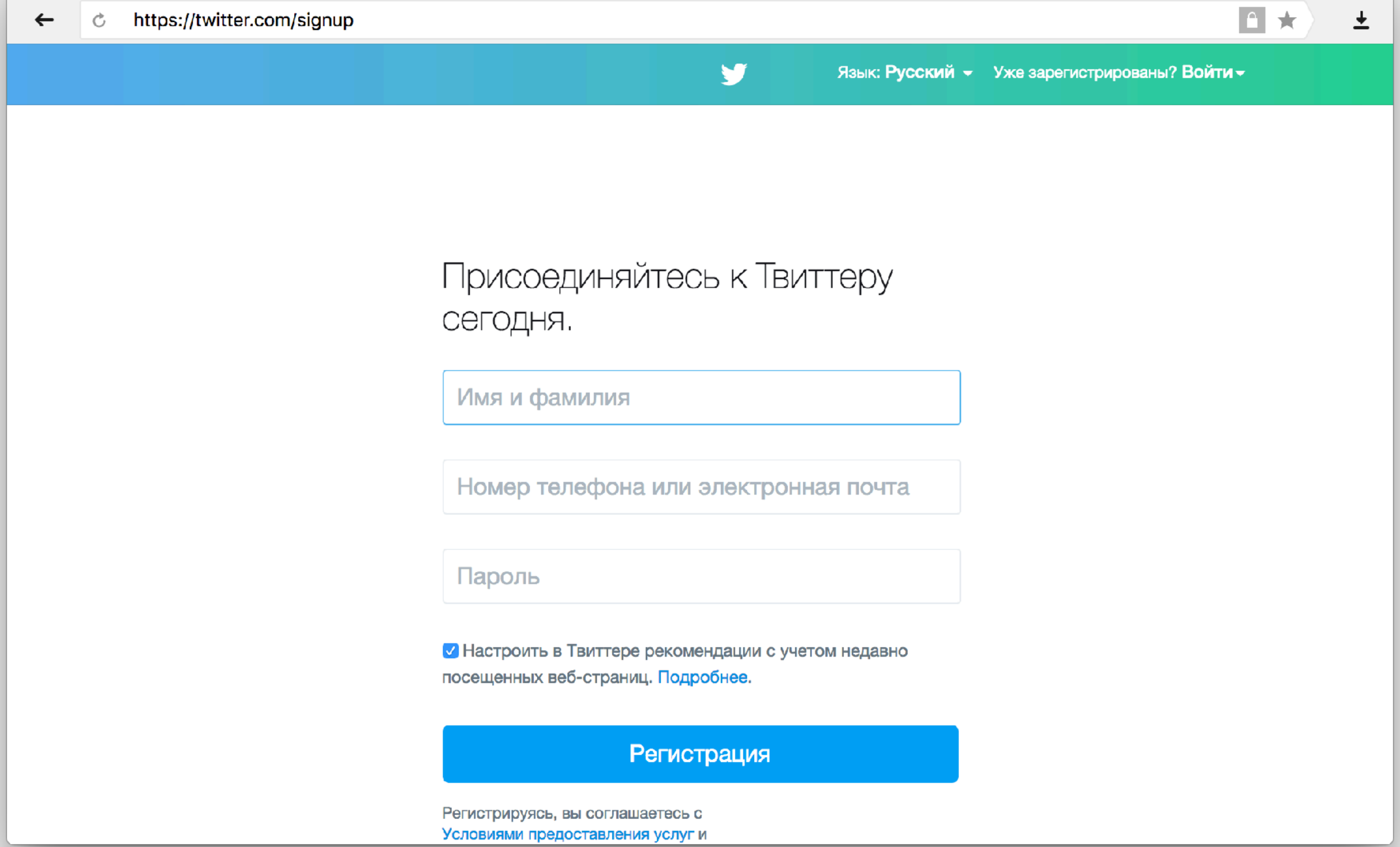
Ссылки

- Приложение "Заметки"
- Node.js Authentication using Passport.js
- express/examples/route-middleware
- express/examples/auth
- github.com/vadimdemedes/cancan



Библиотека для авторизации

UX


https://twitter.com/signup
Язык: Русский
Уже зарегистрированы? Войти
Присоединяйтесь к Твиттеру сегодня.
Имя и фамилия
Номер телефона или электронная почта
Пароль
Настроить в Твиттере рекомендации с учетом недавно посещенных веб-страниц. Подробнее.
Регистрация
Регистрируясь, вы соглашаетесь с
Условиями предоставления услуг и

https://slack.com/

slack

Product Pricing Support Create a new team Find your team Sign in

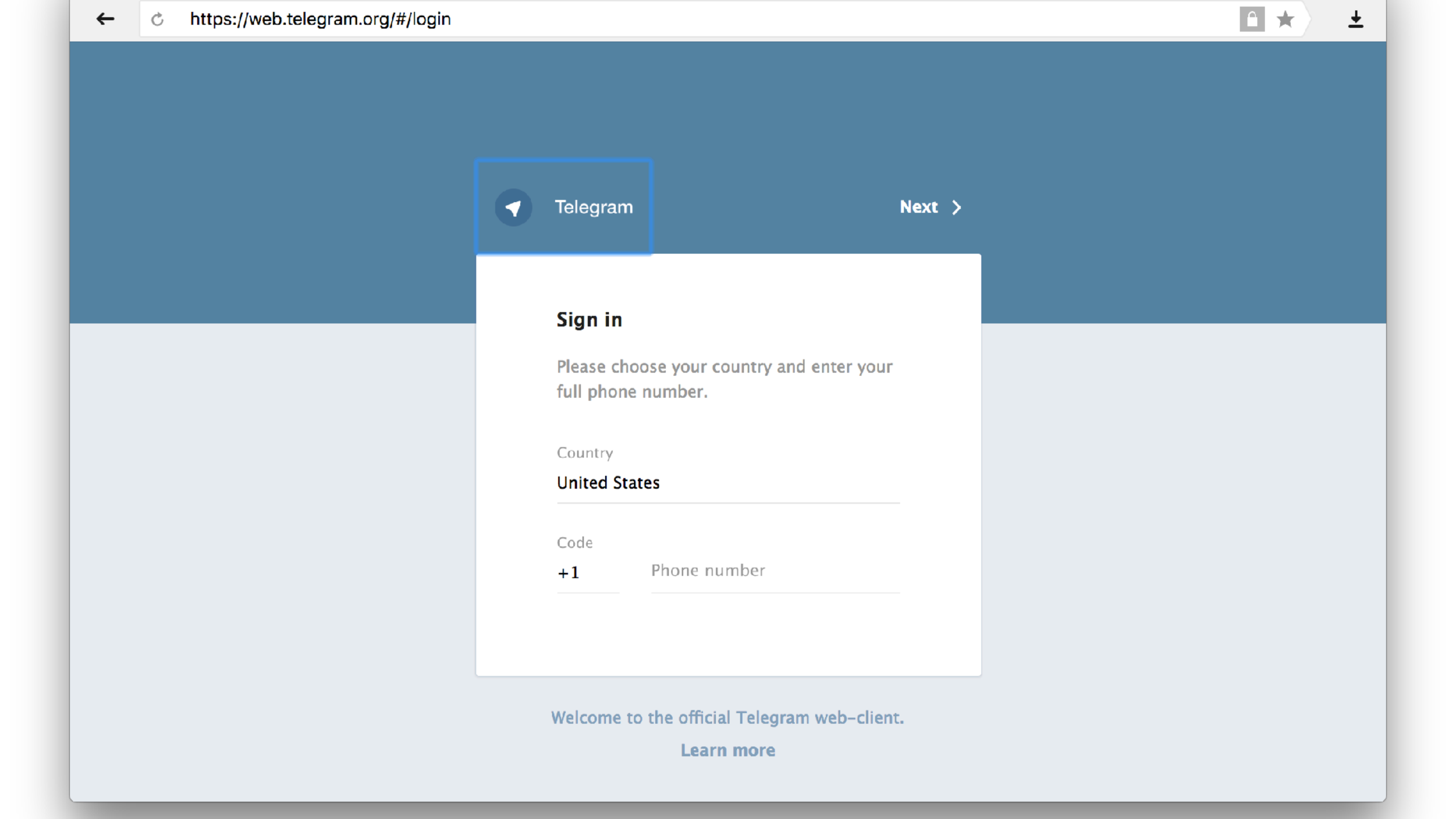
Where work* happens.

* Whatever work means for you, Slack brings all the pieces and people you need together so you can actually get things done.

Email address

Get Started for Free

Already joined a Slack team? Sign in


https://web.telegram.org/#/login
Telegram
Next
Sign in
Please choose your country and enter your full phone number.
Country
United States
Code
+1
Phone number
Welcome to the official Telegram web-client.
Learn more

https://trello.com/signup

кофиденциальности

Создать новый аккаунт

У вас есть аккаунт Google?

Зарегистрироваться с помощью Google

Уже есть аккаунт? Войти.

© 2017, Trello, Inc. Все права защищены.

https://www.ozon.ru/context/cart/#cartType=ozon

ozon.ru выбирайте

1 613 сейчас выбирают

Москва
419 пунктов выдачи заказов
Условия доставки

Помощь, доставка, оплата
Круглосуточно
+7 495 730 67 67

Моя карта
OZON.ru от БИНБАНКа

ozonstatus

Каталог товаров | Выбирайте... | Все разделы | Мой OZON | Корзина 1

- Сравнение
- Личный кабинет
- Кодовые слова и сертификаты
- Вход

Корзина

Доставим завтра в г. Москва

Tchibo Espresso Milano Style кофе в зернах, 500 г 510 г

\- 1 +

638,00 ₽

1 товар · 510 г

Итого без доставки: 638 ₽

Кодовое слово или сертификат | Активировать

Перейти к оформлению

60 дней на возврат товара

Мы гарантируем надёжную защиту данных вашей банковской карты при оплате онлайн

Всё, что нужно, по выгодным ценам:

Спросить сейчас

Конец