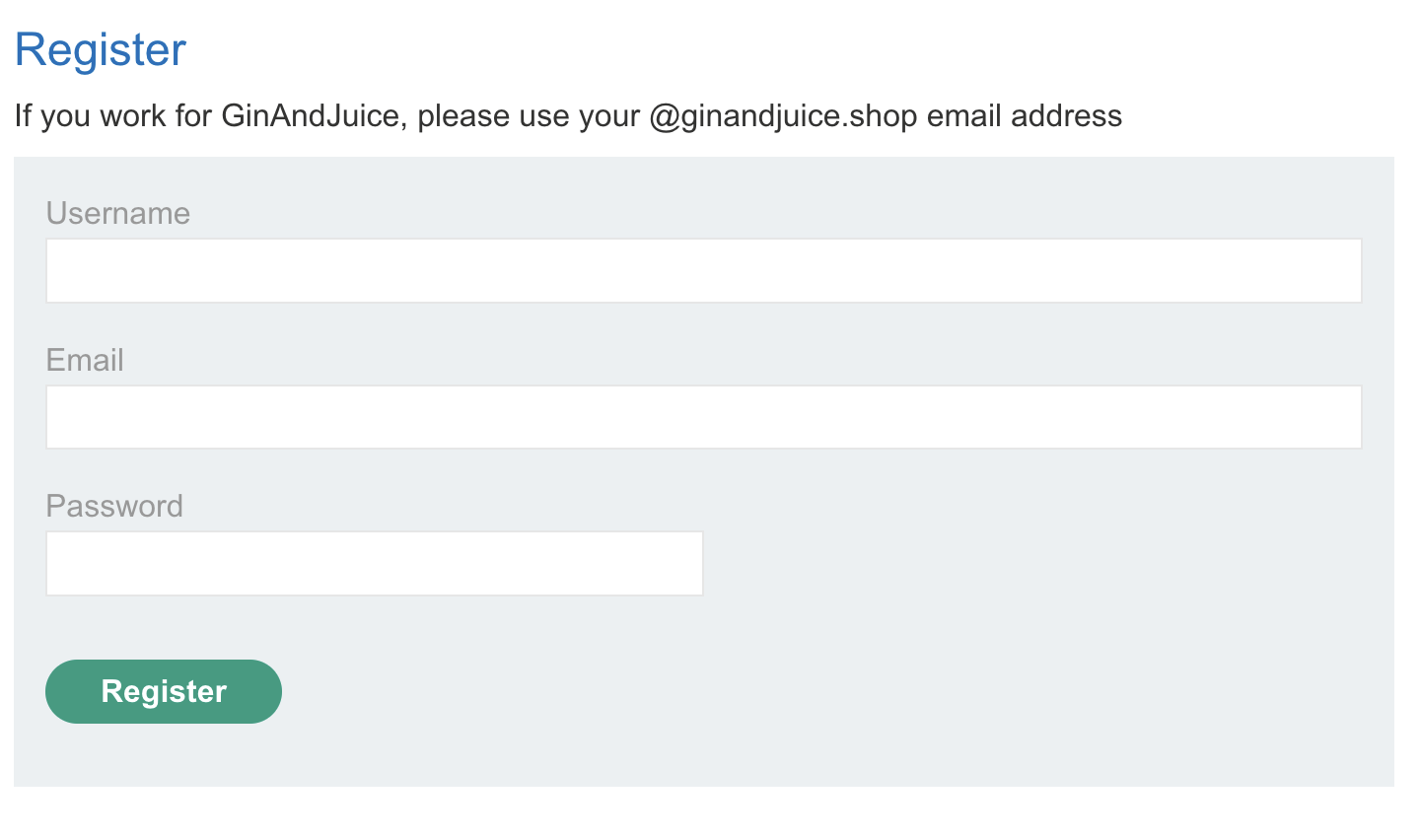
**Partial construction race conditions**

Bài này có chức năng đăng kí tài khoản. Race condtion cho phép bạn vượt qua được xác thực email và đăng kí 1 địa chỉ email bất kỳ mà không phải của bạn.

Để giải được bài này, khai thác lỗi race condition để tạo 1 tài khoản sau đó đăng nhập và xóa người dùng carlos.

Đầu tiên, chúng ta phải tìm hiểu cơ chế đăng kí tài khoản



Nếu chúng ta có được địa chỉ email là @ginandjuice.shop thì có thể truy cập giao diện admin.

Nhưng trước hết, chúng ta phải đặt địa chỉ email của chúng ta để hiểu cơ chế của nó trước đã.

Email client: @exploit-0ac000ed039ca480829c5a7b010200f6.exploit-server.net

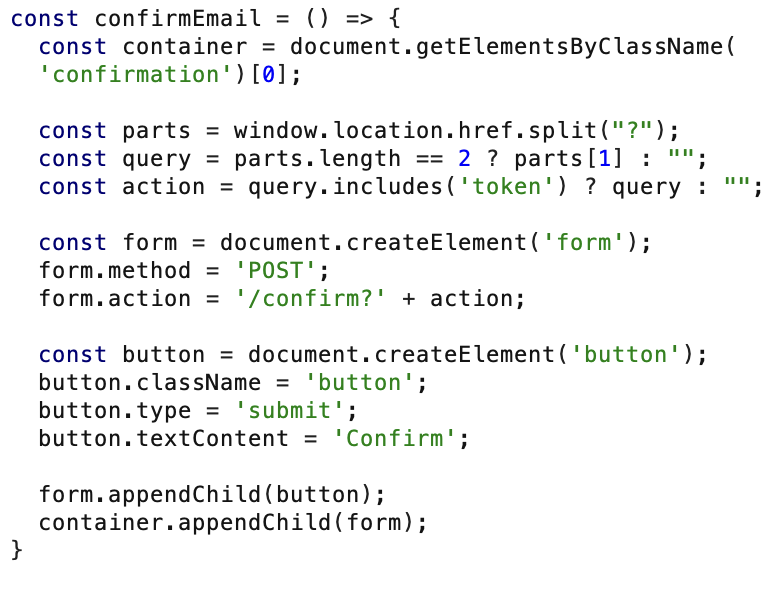
Nhưng trang web cứ báo lỗi: địa chỉ email không hợp lệ

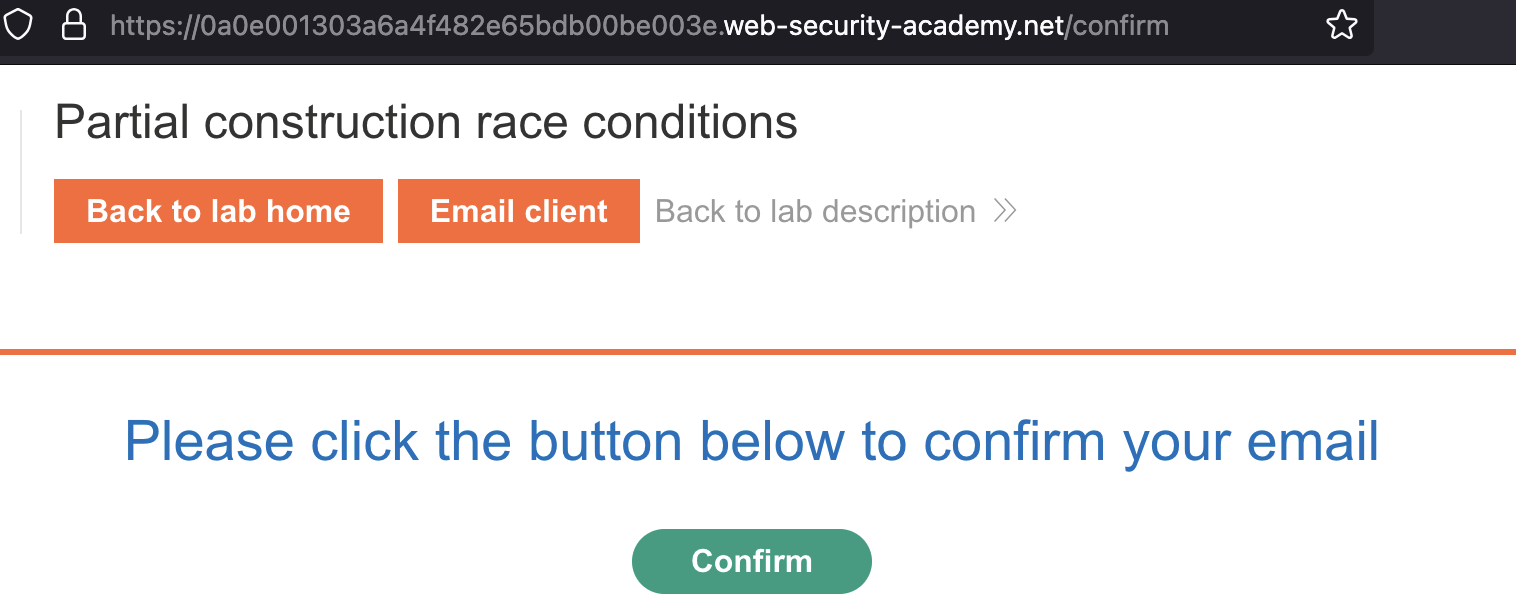
Bài này chỉ cho phép tạo tài khoản với email @ginandjuice.shop

Khi đăng kí thành viên mới, nó sẽ gửi token đến email. Trong email có đường dẫn xác nhận.

Trong Burp, từ lịch sử của proxy, để ý thấy có 1 yêu cầu fetch /resources/static/users.js

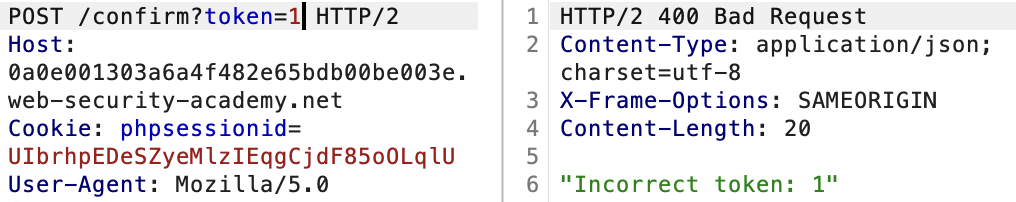
Nó có thêm chức năng kiểm tra email:





Đọc code Javascript và để ý thấy hàm tạo 1 form cho trang /confirm, có lẽ được liên kết từ email xác nhận. Điều này rò rỉ 1 thực tế là xác nhận cuối cùng sẽ được gửi bằng yêu cầu POST đến /confirm với token được cung cấp trong chuỗi truy vấn.

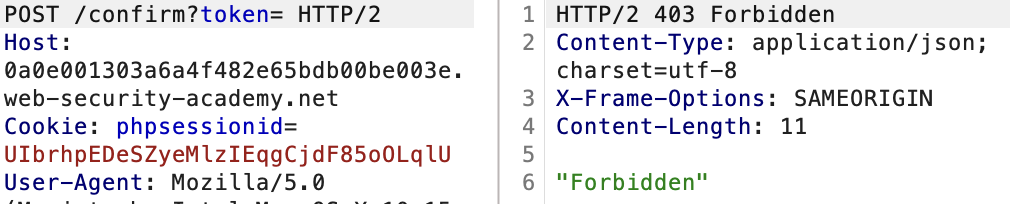
Trong Repeater, hãy tạo 1 yêu cầu tương tự với những gì mà trình duyệt của bạn có thể gửi khi bấm chọn đường dẫn liên kết. Ví dụ:



Thử nghiệm với tham số token. Quan sát thấy:

* Nếu bạn gửi 1 token bất kỳ, bạn nhận được phản hồi "Incorrect token: token"
* Nếu bạn xóa tham số đi, bạn nhận được phản hồi "Missing parameter: token"
* Nếu bạn gửi token rỗng, bạn nhận được phản hồi "Forbidden"

Xem xét phản hồi Forbidden, lập trình viên đã vá 1 lỗ hổng có thể được khai thác bằng cách gửi 1 token rỗng

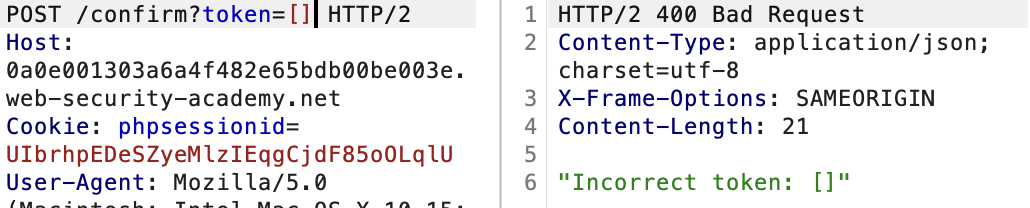


Mình đoán có 1 race window giữa:

* Khi bạn gửi 1 yêu cầu để đăng kí người dùng
* Khi token đăng kí được tạo mới thường lưu trữ trong cơ sở dữ liệu

Nếu vậy, có 1 trạng thái trung gian mà null là 1 token hợp lệ để xác nhận đăng kí

Thử nghiệm với nhiều cách khác nhau bằng cách gửi tham số token với 1 giá trị tương đương với null. Ví dụ, một số thư viện cho phép bạn truyền 1 mảng rỗng như sau:



Thay vì phản hồi Forbidden, bạn nhận được thông báo "Invalid token: Array". Điều này cho thấy ứng dụng chấp nhận việc bạn truyền 1 mảng rỗng, có thể phù hợp với 1 token khi chưa khởi tạo.

Thêm hai yêu cầu POST /register và POST /confirm?token=1 vào 1 nhóm

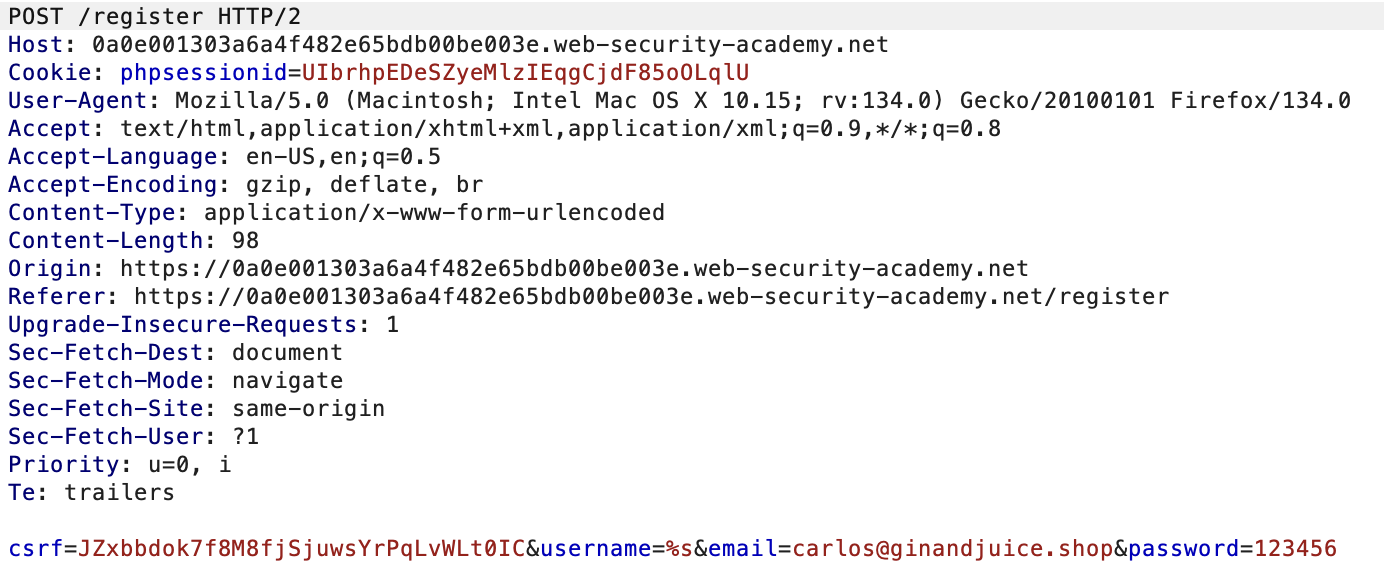
Gửi nhóm yêu cầu trên theo tuần tự và song song. Đổi tên người dùng mỗi lần để tránh gặp thông báo "Account already exists with this name". Để ý thấy phản hồi xác nhận lúc nào cũng đến nhanh hơn phản hồi đăng kí.

Trong yêu cầu POST /register, tô đậm giá trị của username, sau đó chuột phải và chọn Extensions🡪Turbo Intruder🡪 Send to turbo intruder

Trong turbo intruder, để ý thấy tham số username được đánh dấu %s

Đảm bảo tham số email đặt là carlos@ginandjuice.shop (email chưa được đăng kí trên trang web)

Nhớ ghi lại giá trị của tham số password để tiện đăng nhập sau này



Chọn template examples/race-single-packet-attack.py

Chỉnh sửa tập lệnh như sau:

def queueRequests(target, wordlists):

engine = RequestEngine(endpoint=target.endpoint,

concurrentConnections=1,

engine=Engine.BURP2

)

confirmationReq = '''POST /confirm?token[]= HTTP/2

Host: YOUR-LAB-ID.web-security-academy.net

Cookie: phpsessionid=YOUR-SESSION-TOKEN

Content-Length: 0

'''

for attempt in range(20):

currentAttempt = str(attempt)

username = 'User' + currentAttempt

# queue a single registration request

engine.queue(target.req, username, gate=currentAttempt)

# queue 50 confirmation requests - note that this will probably sent in two separate packets

for i in range(50):

engine.queue(confirmationReq, gate=currentAttempt)

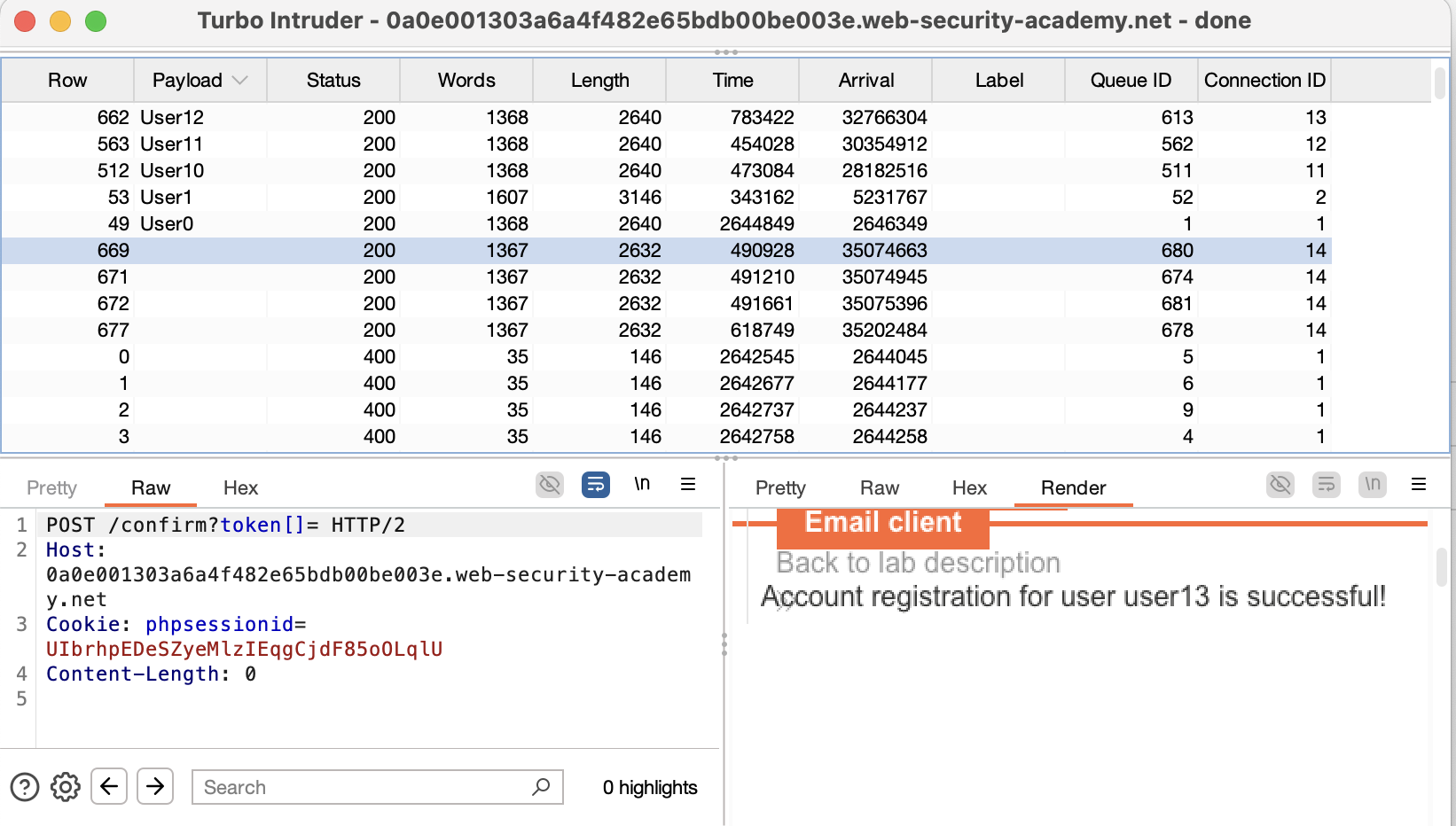
# send all the queued requests for this attempt

engine.openGate(currentAttempt)

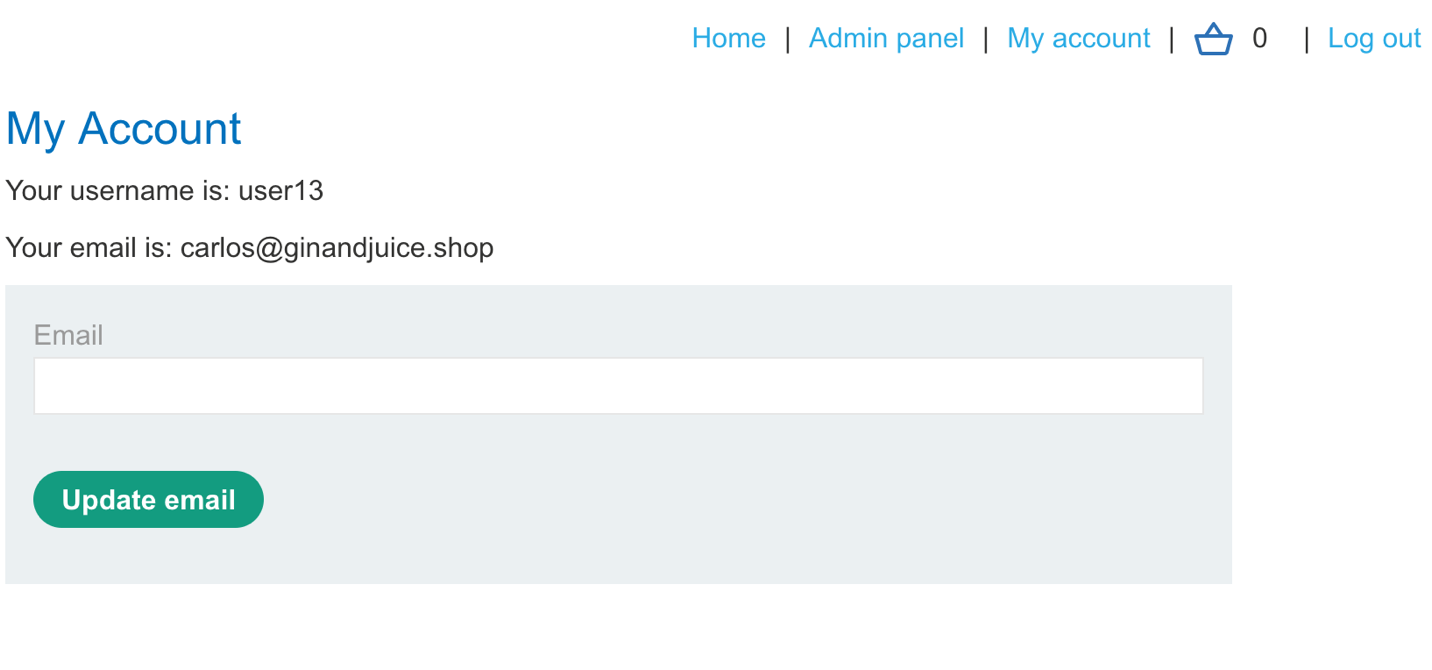
def handleResponse(req, interesting):

table.add(req)

Tập lệnh đăng kí 20 người dùng mới, mỗi người dùng sẽ gửi yêu cầu xác nhận 50 lần. Gửi tham số token rỗng trước khi token được khởi tạo trong cơ sở dữ liệu.



Đăng kí thành công cho user13



Xóa người dùng carlos