**Limit overrun race conditions**

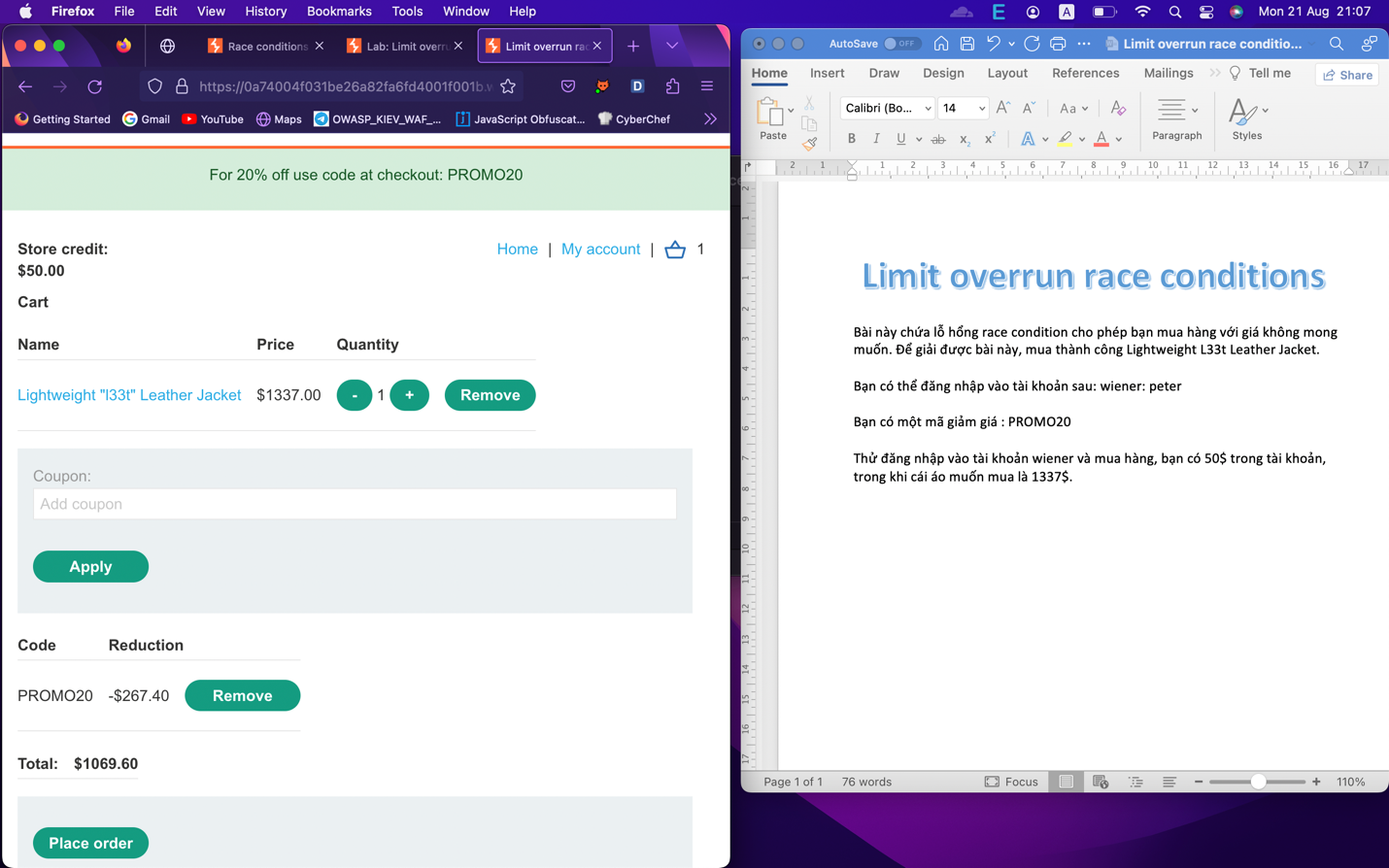
Bài này chứa lỗ hổng race condition cho phép bạn mua hàng với giá không mong muốn. Để giải được bài này, mua thành công Lightweight L33t Leather Jacket.

Bạn có thể đăng nhập vào tài khoản sau: wiener: peter

Bạn có một mã giảm giá : PROMO20

Thử đăng nhập vào tài khoản wiener và mua hàng, bạn có 50$ trong tài khoản, trong khi cái áo muốn mua là 1337$.

Apply mã giảm giá lần 1:



Nếu bạn áp thử lại mã giảm giá nó sẽ báo lỗi: Coupon already applied

Làm sao để biết ứng dụng bị lỗi race conditions?

Mình sẽ áp mã nhiều lần trước khi cơ sở dữ liệu cập nhật lại trạng thái sử dụng của mã code.

1. Chắc chắn mã giảm giá chưa được áp dụng cho giỏ hàng của bạn

2. Gửi yêu cầu để áp mã giảm giá POST /cart/coupon đến repeater 20 lần (có thể làm nhanh bằng cmd+r)

3. Trong repeater, thêm tất cả 20 tabs đó vào 1 nhóm.

4. Gửi 1 nhóm các yêu cầu đó theo thứ tự bằng separate connections để giảm sự can thiệp từ bên ngoài

5. Quan sát phản hồi đầu tiên áp dụng mã thành công nhưng các phản hồi còn lại từ chối mã đó với phản hồi giống nhau: Coupon already applied

Tiếp theo, mình sẽ test race condition nè:

1. Bỏ mã giảm giá ra khỏi giỏ hàng

2. Trong repeater, gửi 1 nhóm các yêu cầu 1 lần nữa nhưng lần này ở chế độ parallel.

3. Nghiên cứu các phản hồi và thấy nhiều yêu cầu áp dụng mã thành công 🡪 race conditions

Refresh giỏ hàng và kiểm tra đơn đặt hàng:

Nếu áo bạn mua vẫn chưa đủ tiền thì bỏ mã giảm giá và lặp lại tấn công lần nữa

**Bypassing rate limits via [race conditions](https://portswigger.net/web-security/race-conditions)**

Cơ chế đăng nhập của bài này sử dụng rate limiting để chống lại brute-force. Tuy nhiên, điều này có thể bỏ qua bằng race condition.

Để giải được bài này:

1. Tìm ra cách khai thác race condition để bỏ qua rate limit

2. Brute force mật khẩu của carlos thành công

3. Đăng nhập và truy cập admin panel

4. Xóa người dùng carlos

Bài này quá rõ ràng rồi, form đăng nhập sẽ dính lỗi race condition. Tập trung khai thác nó thôi.

Quan sát thấy nếu bạn nhập sai mật khẩu hơn 3 lần sẽ bị khóa tạm thời. Thử dùng 1 username khác và bạn vẫn thấy Invalid username or pass 🡪 ứng dụng rate limit dựa trên username chứ không phải là session 🡪 số lần đăng nhập thất bại của 1 username sẽ được lưu trữ ở phía máy chủ.

Mình sẽ đăng nhập thật nhiều trước khi cơ sở dữ liệu kịp cập nhật và chặn mình.

Gửi POST /login đến repeater 20 lần, tạo 1 group.

Gửi group đó theo mode separate connections, quan sát thấy sau hơn hai lần đăng nhập thất bại, bạn vẫn bị chặn tạm thời.

Gửi group đó ở parallel, bạn sẽ thấy mình đăng nhập được nhiều lần hơn trước khi bị nó chặn 🡪 race condition

Để có thể tấn công dồn dập cùng 1 lúc, mình phải dùng turbo intruder.

* Vẫn ở repeater, highlight giá trị của mật khẩu trong yêu cầu POST /login
* Chuột phải và chọn Extensions > Turbo Intruder > Send to turbo intruder
* Trong turbo intruder, giá trị của mk được tự động đánh dấu là vị trí đặt payload với %s
* Thay đổi username là carlos
* Chọn mẫu examples/race-single-packet-attack.py
* Chỉnh sửa template như sau:

def queueRequests(target, wordlists):

# as the target supports HTTP/2, use engine=Engine.BURP2 and concurrentConnections=1 for a single-packet attack

engine = RequestEngine(endpoint=target.endpoint,

concurrentConnections=1,

engine=Engine.BURP2

)

# assign the list of candidate passwords from your clipboard

passwords = wordlists.clipboard

# queue a login request using each password from the wordlist

# the 'gate' argument withholds the final part of each request until engine.openGate() is invoked

for password in passwords:

engine.queue(target.req, password, gate='1')

# once every request has been queued

# invoke engine.openGate() to send all requests in the given gate simultaneously

engine.openGate('1')

def handleResponse(req, interesting):

table.add(req)

* Lưu ý chúng ta gán danh sách mật khẩu từ clipboard bằng cách tham chiếu wordlists.clipboard. Sao chép danh sách mật khẩu đến clipboard.
* Thực hiện tấn công
* Nghiên cứu các phản hồi:
  + Nếu bạn không đăng nhập thành công, đợi thời gian khóa tài khoản được reset và thực hiện lại tấn công. Bạn có thể muốn bỏ một số mật khẩu mà bạn biết là nó sai.
  + Nếu bạn nhận phản hồi 302, việc đăng nhập này đã thành công
  + Đợi khi khóa tài khoản được reset, sau đó đăng nhập tài khoản carlos, truy cập admin panel và xóa người dùng carlos