PHẦN II

XÂY DỰNG PHÒNG THỬ NGHIỆM API

4

HỆ THỐNG HACK API CỦA BẠN

Chương này sẽ hướng dẫn bạn cách thiết lập bộ công cụ hack API của bạn. Chúng tôi sẽ đề cập đến ba công cụ đặc biệt hữu ích dành cho tin tặc API: Chrome DevTools, Burp Suite và Postman.

Ngoài việc khám phá các tính năng có trong phiên bản Burp Suite Pro trả phí, tôi sẽ cung cấp danh sách các công cụ có thể bù đắp cho các tính năng còn thiếu trong Burp Suite Community Edition miễn phí, cũng như một số công cụ hữu ích khác để khám phá và khai thác API lỗ hổng. Ở cuối chương này, chúng ta sẽ xem qua một phòng thí nghiệm, trong đó bạn sẽ học cách sử dụng một số công cụ này để tương tác với các API đầu tiên của chúng tôi.

KaliLinux

Xuyên suốt cuốn sách này, chúng ta sẽ chạy các công cụ và phòng thí nghiệm bằng Kali, một bản phân phối Linux dựa trên Debian mã nguồn mở. Kali được xây dựng để thử nghiệm thâm nhập và đi kèm với nhiều công cụ hữu ích đã được cài đặt sẵn. Bạn có thể tải xuống Kali tại https://www.kali.org/downloads. Rất nhiều hướng dẫn có thể hướng dẫn bạn cách thiết lập trình ảo hóa mà bạn chọn và cài đặt Kali trên đó. Tôi khuyên bạn nên sử dụng "Cách bắt đầu với Kali Linux" của Null Byte hoặc hướng dẫn tại https://www.kali.org/docs/installation.

Sau khi phiên bản Kali của bạn được thiết lập, hãy mở một thiết bị đầu cuối và thực hiện cập nhật và nâng cấp:

```
$ cập nhật apt sudo
$ sudo apt nâng cấp đầy đủ -y
```

Tiếp theo, cài đặt Git, Python 3 và Golang (Go), bạn sẽ cần sử dụng một số công cụ khác trong hộp hack của mình:

```
$ sudo apt-get cài đặt git python3 golang
```

Với những thông tin cơ bản này đã được cài đặt, bạn nên sẵn sàng thiết lập phần còn lại của các công cụ hack API.

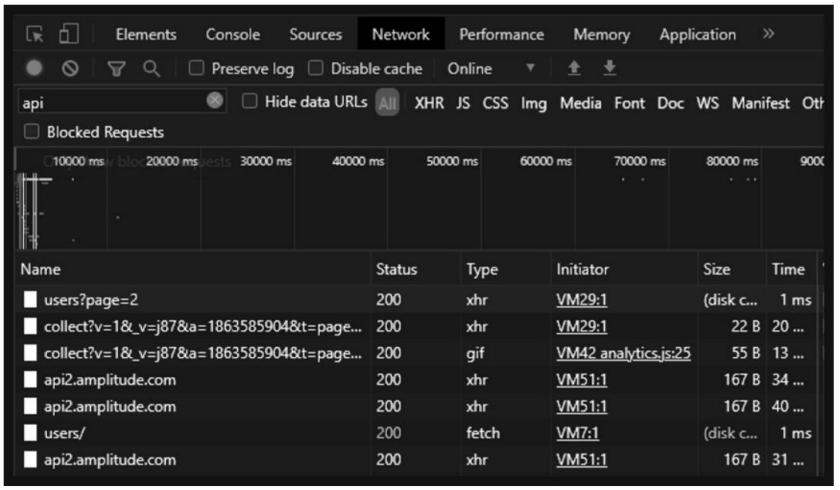
Phân tích ứng dụng web với DevTools

DevTools của Chrome là bộ công cụ dành cho nhà phát triển được tích hợp trong trình duyệt Chrome cho phép bạn xem trình duyệt web của mình đang chạy gì từ góc độ của nhà phát triển web. DevTools là một tài nguyên thường bị đánh giá thấp, nhưng nó có thể rất hữu ích cho các hacker API. Chúng tôi sẽ sử dụng nó cho các tương tác đầu tiên với các ứng dụng web mục tiêu để khám phá các API; tương tác với các ứng dụng web bằng bảng điều khiển; xem tiêu đề, xem trước và phản hồi; và phân tích các tệp nguồn ứng dụng web.

Để cài đặt Chrome, bao gồm DevTools, hãy chạy như sau lệnh:

```
$ sudo wget https://dl.google.com/linux/direct/google-chrome-stable_current_amd64.deb
$ sudo apt cài đặt ./google-chrome-stable_current_amd64.deb
```

Bạn có thể khởi chạy Chrome thông qua dòng lệnh với google -lệnh chrome . Khi bạn đã chạy Chrome, hãy điều hướng đến URL mà bạn muốn điều tra và khởi chạy DevTools bằng cách sử dụng CTRL-SHIFT-I hoặc F12 hoặc điều hướng đến Settings4More Tools và chọn menu Công cụ dành cho nhà phát triển . Tiếp theo, hãy làm mới trang hiện tại của bạn để cập nhật thông tin trong bảng DevTools. Bạn có thể làm điều này bằng cách sử dụng phím tắt CTRL-R. Trong bảng điều khiển Mạng, bạn sẽ thấy các tài nguyên khác nhau được yêu cầu từ các API (xem Hình 4-1).



Hình 4-1: Bảng điều khiển Mạng Chrome DevTools

Chuyển bảng bằng cách chọn tab mong muốn ở trên cùng. Bảng điều khiển DevTools liệt kê chức năng của các tùy chọn bảng khác nhau. Tôi đã tóm tắt những điều này trong Bảng 4-1.

Bảng 4-1: Bảng DevTools

bảng điều khiển	Chức năng
yếu tố	Cho phép bạn xem CSS và Mô hình đối tượng tài liệu (DOM) của trang hiện tại, cho phép bạn kiểm tra HTML cấu trúc trang web.
Bảng điều khiển	Cung cấp cho bạn các cảnh báo và cho phép bạn tương tác với trình gỡ lỗi JavaScript để thay đổi trang web hiện tại.
nguồn	Chứa các thư mục tạo nên ứng dụng web và nội dung của các tệp nguồn.
Mạng	Liệt kê tất cả các yêu cầu tệp nguồn tạo nên phối cảnh ứng dụng web của ứng dụng khách.
Hiệu suất	Cung cấp một cách để ghi lại và phân tích tất cả các sự kiện diễn ra khi tải một trang web.
Ký ức	Cho phép bạn ghi lại và phân tích cách trình duyệt tương tác với bộ nhớ hệ thống của bạn.
Ứng dụng	Cung cấp cho bạn bảng kê khai ứng dụng, các mục lưu trữ (như cookie và thông tin phiên), bộ đệm và các dịch vụ nền.
Bảo vệ	Cung cấp thông tin chuyên sâu về mã hóa chuyển tuyến, nguồn gốc nội dung nguồn và chi tiết chứng chỉ.

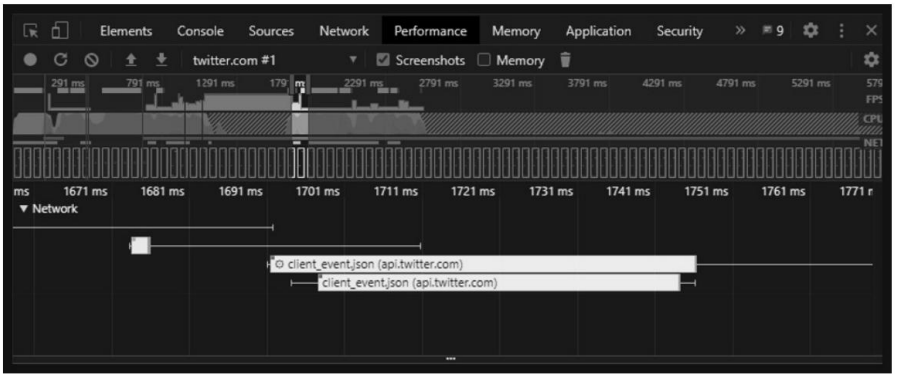
Khi chúng ta bắt đầu tương tác với một ứng dụng web lần đầu tiên, chúng ta thường sẽ bắt đầu với bảng điều khiển Mạng để có cái nhìn tổng quan về các tài nguyên cung cấp năng lượng cho ứng dụng web. Trong Hình 4-1, mỗi mục được liệt kê đại diện cho một yêu cầu đã được thực hiện đối với một tài nguyên cụ thể. Sử dụng bảng điều khiển Mạng, bạn

có thể đi sâu vào từng yêu cầu để xem phương thức yêu cầu đã được sử dụng, mã trạng thái phản hồi, tiêu đề và nội dung phản hồi. Để thực hiện việc này, hãy nhấp vào tên của URL quan tâm một lần bên dưới cột Tên. Điều này sẽ mở ra một bảng điều khiển ở phía bên phải của DevTools. Giờ đây, bạn có thể xem lại yêu cầu đã được thực hiện trong tab Tiêu đề và xem cách máy chủ phản hồi trong tab Phản hồi.

Tìm hiểu sâu hơn về ứng dụng web, bạn có thể sử dụng bảng Nguồn để kiểm tra các tệp nguồn đang được sử dụng trong ứng dụng. Trong các sự kiện nắm bắt cờ (CTF) (và đôi khi trong thực tế), bạn có thể tìm thấy các khóa API hoặc các bí mật được mã hóa cứng khác tại đây. Bảng Nguồn được trang bị chức năng tìm kiếm mạnh mẽ sẽ giúp bạn dễ dàng khám phá hoạt động bên trong của ứng dụng.

Bảng điều khiển rất hữu ích để chạy và gỡ lỗi JavaScript của trang web. Bạn có thể sử dụng nó để phát hiện lỗi, xem cảnh báo và thực hiện lệnh. Bạn sẽ có cơ hội sử dụng bảng Console trong phòng thí nghiệm ở Chương 6.

Chúng tôi sẽ dành phần lớn thời gian của mình trong bảng điều khiển, Nguồn và Mạng. Tuy nhiên, các bảng khác cũng có thể hữu ích. Ví dụ: bảng Hiệu suất chủ yếu được sử dụng để cải thiện tốc độ của trang web, nhưng chúng ta cũng có thể sử dụng bảng này để quan sát ứng dụng web tương tác với API ở điểm nào, như trong Hình 4-2.



Hình 4-2: Tab Performance của DevTool hiển thị thời điểm chính xác ứng dụng Twitter tương tác với Twitter API

Trong Hình 4-2, chúng ta thấy rằng, sau 1.700 mili giây, một sự kiện máy khách được kích hoạt ứng dụng Twitter để tương tác với API. Với tư cách là khách hàng, sau đó chúng tôi có thể liên kết sự kiện đó với một hành động mà chúng tôi đã thực hiện trên trang, chẳng hạn như xác thực với ứng dụng web, để biết ứng dụng web đang sử dụng API để làm gì. Chúng ta càng thu thập được nhiều thông tin trước khi tấn công một API, khả năng tìm kiếm và khai thác lỗ hổng của chúng ta càng cao.

Để biết thêm thông tin về DevTools, hãy xem Google Developers tài liệu tại https://developers.google.com/web/tools/chrome-devtools.

Nắm bắt và sửa đổi yêu cầu với Burp Suite

Burp Suite là một bộ công cụ kiểm tra ứng dụng web tuyệt vời được PortSwigger phát triển và liên tục cải tiến. Tất cả các chuyên gia an ninh mạng ứng dụng web, thợ săn tiền thưởng lỗi và tin tặc API nên học cách sử dụng Burp, cho phép bạn nắm bắt các yêu cầu API, ứng dụng mạng nhện, API fuzz, v.v.

Thu thập thông tin trên web, hoặc thu thập thông tin trên web, là phương pháp mà bot sử dụng để tự động phát hiện đường dẫn URL và tài nguyên của máy chủ lưu trữ. Thông thường, spidering được thực hiện bằng cách quét HTML của các trang web để tìm các siêu liên kết. Quét mạng là một cách hay để có ý tưởng cơ bản về nội dung của một trang web, nhưng nó sẽ không thể tìm thấy các đường dẫn ẩn hoặc những đường dẫn không tìm thấy liên kết trong các trang web. Để tìm các đường dẫn ẩn, chúng ta sẽ cần sử dụng một công cụ như Kiterunner để thực hiện các cuộc tấn công brute-force thư mục một cách hiệu quả. Trong một cuộc tấn công như vậy, một ứng dụng sẽ yêu cầu nhiều đường dẫn URL có thể có khác nhau và xác thực xem chúng có thực sự tồn tại hay không dựa trên phản hồi của máy chủ.

Theo mô tả của trang cộng đồng OWASP về chủ đề này, fuzzing là "nghệ thuật tìm lỗi tự động". Sử dụng kỹ thuật tấn công này, chúng tôi sẽ gửi nhiều loại đầu vào khác nhau trong yêu cầu HTTP, cố gắng tìm đầu vào hoặc tải trọng khiến ứng dụng phản hồi theo những cách không mong muốn và để lộ lỗ hổng. Ví dụ: nếu bạn đang tấn công một API và phát hiện ra rằng bạn có thể đăng dữ liệu lên nhà cung cấp API, thì bạn có thể thử gửi các lệnh SQL khác nhau cho nó. Nếu nhà cung cấp không làm sạch đầu vào này, thì có khả năng bạn sẽ nhận được phản hồi cho biết cơ sở dữ liệu SQL đang được sử dụng.

Burp Suite Pro, phiên bản trả phí của Burp, cung cấp tất cả các tính năng mà không bị hạn chế, nhưng nếu sử dụng Burp Suite Community Edition (CE) miễn phí là tùy chọn duy nhất của bạn, thì bạn có thể làm cho nó hoạt động. Tuy nhiên, khi bạn đã nhận được phần thưởng tiền thưởng lỗi hoặc ngay sau khi bạn có thể thuyết phục được nhà tuyển dụng của mình, bạn nên chuyển sang Burp Suite Pro. Chương này bao gồm phần "Công cụ tinh thần dẻo dai" sẽ giúp thay thế chức năng bị thiếu trong Burp Suite CE.

Burp Suite CE được bao gồm tiêu chuẩn với phiên bản Kali mới nhất. Nếu vì bất kỳ lý do gì mà nó không được cài đặt, hãy chạy như sau:

```
$ sudo apt-get cài đặt burpsuite
```

LƯU Ý

p Suite cung cấp phiên bản dùng thử 30 ngày đầy đủ tính năng của Burp Suite Pro tại https://portwigger.net/requestfreetrial/pro. Để biết thêm hướng dẫn về cách sử dụng Burp Suite, hãy truy cập https://portswigger.net/burp/communitydownload.

Trong các phần tiếp theo, chúng ta sẽ chuẩn bị giàn hack API để sử dụng Burp Suite, xem tổng quan về các mô-đun Burp khác nhau, tìm hiểu cách chặn các yêu cầu HTTP, tìm hiểu sâu hơn về mô-đun Intruder và xem qua một số tiện ích mở rộng thú vị mà bạn có. có thể sử dụng để nâng cao Burp Suite Pro.

Thiết lập FoxyProxy

Một trong những tính năng chính của Burp Suite là khả năng chặn các yêu cầu HTTP. Nói cách khác, Burp Suite nhận yêu cầu của bạn trước khi chuyển tiếp đến máy chủ và sau đó nhận phản hồi của máy chủ trước khi gửi đến trình duyệt, cho phép bạn xem và tương tác với các yêu cầu và phản hồi đó. Để tính năng này hoạt động, chúng tôi cần thường xuyên gửi yêu cầu từ trình duyệt đến Burp Suite. Điều này được thực hiện bằng cách sử dụng proxy web.

Proxy là một cách để chúng tôi định tuyến lại lưu lượng truy cập của trình duyệt web tới Burp trước khi nó được gửi tới nhà cung cấp API. Để đơn giản hóa quy trình này, chúng tôi sẽ thêm một công cụ có tên FoxyProxy vào trình duyệt của mình để giúp chúng tôi lưu lượng truy cập proxy chỉ bằng một lần bấm nút.

Các trình duyệt web có chức năng proxy được tích hợp sẵn nhưng việc thay đổi và cập nhật các cài đặt này mỗi khi bạn muốn sử dụng Burp sẽ rất tốn thời gian. Thay vào đó, chúng tôi sẽ sử dụng tiện ích bổ sung của trình duyệt có tên là FoxyProxy cho phép bạn bật và tắt proxy của mình chỉ bằng một nút bấm đơn giản. FoxyProxy có sẵn cho cả Chrome và Firefox.

Làm theo các bước sau để cài đặt FoxyProxy:

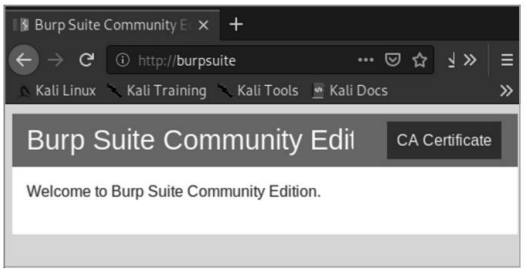
1. Điều hướng đến cửa hàng tiện ích bổ sung hoặc trình cắm của trình duyệt của bạn và tìm kiếm FoxyProxy.
2. Cài đặt FoxyProxy Standard và thêm nó vào trình duyệt của bạn.
3. Nhấp vào biểu tượng con cáo ở góc trên bên phải trình duyệt của bạn (bên cạnh URL) và chọn Tùy chọn.
4. Chọn Proxies4Add New Proxy4Manual Proxy Configuration.
5. Thêm 127.0.0.1 làm địa chỉ IP máy chủ.
6. Cập nhật cổng thành 8080 (Cài đặt proxy mặc định của Burp Suite).
7. Trong tab Chung, đổi tên proxy thành Hackz (tôi sẽ tham khảo cái này cài đặt proxy trong toàn bộ phòng thí nghiệm).

Bây giờ bạn chỉ cần nhấp vào tiện ích bổ sung của trình duyệt và chọn proxy bạn muốn sử dụng để gửi lưu lượng truy cập của mình đến Burp. Khi bạn đã hoàn thành việc chặn các yêu cầu, bạn có thể tắt proxy bằng cách chọn tùy chọn Tắt FoxyProxy.

Thêm chứng chỉ Burp Suite

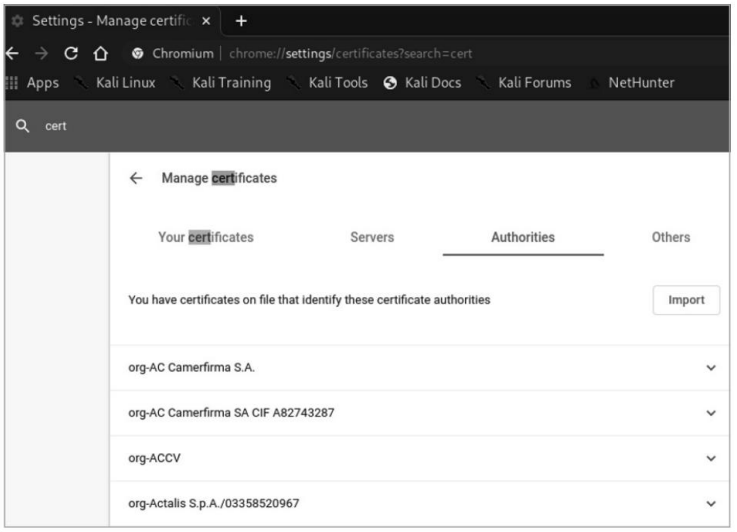
HTTP Strict Transport Security (HSTS) là một chính sách bảo mật ứng dụng web phổ biến ngăn không cho Burp Suite có thể chặn các yêu cầu.
Cho dù sử dụng Burp Suite CE hay Burp Suite Pro, bạn sẽ cần cài đặt chứng chỉ của tổ chức phát hành chứng chỉ (CA) của Burp Suite. Để thêm chứng chỉ này, hãy làm theo các bước sau:

1. Bắt đầu Burp Suite.
2. Mở trình duyệt bạn chọn.
3. Sử dụng FoxyProxy, chọn Hackz proxy. Điều hướng đến http://burpsuite, như trong Hình 4-3 và nhấp vào Chứng chỉ CA. Thao tác này sẽ bắt đầu quá trình tải xuống chứng chỉ Burp Suite CA.



Hình 4-3: Trang đích bạn sẽ thấy khi tải xuống chứng chỉ CA của Burp Suite

4. Lưu chứng chỉ ở nơi bạn có thể tìm thấy.
5. Mở trình duyệt của bạn và nhập chứng chỉ. Trong Firefox, mở Tùy chọn và sử dụng thanh tìm kiếm để tra cứu chứng chỉ. Nhập chứng chỉ.
6. Trong Chrome, mở Cài đặt, sử dụng thanh tìm kiếm để tra cứu chứng chỉ, chọn More4Manage Certificates4Authorities và nhập chứng chỉ (xem Hình 4-4). Nếu không thấy chứng chỉ, bạn có thể cần mở rộng tùy chọn loại tệp thành "DER" hoặc "Tất cả tệp".



Hình 4-4: Trình quản lý chứng chỉ Chrome với tab Cơ quan được chọn

Bây giờ bạn đã thêm chứng chỉ PortSwigger CA vào trình duyệt của mình, bạn sẽ có thể chặn lưu lượng truy cập mà không gặp sự cố.

Điều hướng Burp Suite

Như bạn có thể thấy trong Hình 4-5, ở trên cùng của Burp là 13 mô-đun.

Hình 4-5: Các mô-đun Burp Suite

Trang tổng quan cung cấp cho bạn tổng quan về nhật ký sự kiện và các bản quét mà bạn có chạy chống lại mục tiêu của bạn. Trang tổng quan hữu ích hơn trong Burp Suite Pro so với trong CE vì nó cũng sẽ hiển thị bất kỳ sự cố nào được phát hiện trong quá trình thử nghiệm.

Tab Proxy là nơi chúng tôi sẽ bắt đầu thu thập các yêu cầu và phản hồi từ trình duyệt web và Người đưa thư của bạn . Proxy chúng tôi thiết lập sẽ gửi bất kỳ lưu lượng truy cập web nào dành cho trình duyệt của bạn tại đây. Thông thường, chúng tôi sẽ chọn chuyển tiếp hoặc loại bỏ lưu lượng truy cập đã nắm bắt cho đến khi chúng tôi tìm thấy trang web được nhắm mục tiêu mà chúng tôi muốn tương tác. Từ Proxy, chúng tôi sẽ chuyển tiếp yêu cầu hoặc phản hồi tới các mô-đun khác để tương tác và giả mạo.

Trong tab Mục tiêu , chúng tôi có thể xem bản đồ của trang web và quản lý các mục tiêu mà chúng tôi định tấn công. Bạn cũng có thể sử dụng tab này để định cấu hình phạm vi thử nghiệm của mình bằng cách chọn tab Phạm vi và bao gồm hoặc loại trừ URL. Việc bao gồm các URL trong phạm vi sẽ giới hạn các URL bị tấn công chỉ ở những URL mà bạn có quyền tấn công.

Trong khi sử dụng tab Mục tiêu, bạn sẽ có thể định vị Sơ đồ trang web, nơi bạn có thể xem tất cả các URL mà Burp Suite đã phát hiện trong phiên Burp Suite hiện tại của bạn. Khi bạn thực hiện quét, thu thập dữ liệu và lưu lượng proxy, Burp Suite sẽ bắt đầu biên dịch danh sách các ứng dụng web mục tiêu và các thư mục được phát hiện. Đây là một nơi khác mà bạn có thể thêm hoặc xóa URL khỏi phạm vi.

Tab Intruder là nơi chúng ta sẽ thực hiện các cuộc tấn công fuzzing và brute-force đối với các ứng dụng web . Khi bạn đã nắm bắt được một yêu cầu HTTP, bạn có thể chuyển tiếp yêu cầu đó tới Intruder, nơi bạn có thể chọn chính xác các phần của yêu cầu mà bạn muốn thay thế bằng tải trọng bạn chọn trước khi gửi đến máy chủ.

Repeater là một mô-đun cho phép bạn thực hiện các điều chỉnh thực tế đối với các yêu cầu HTTP , gửi chúng đến máy chủ web được nhắm mục tiêu và phân tích nội dung của phản hồi HTTP.

Công cụ Sequencer sẽ tự động gửi hàng trăm yêu cầu và sau đó thực hiện phân tích entropy để xác định mức độ ngẫu nhiên của một chuỗi đã cho. Chúng tôi sẽ chủ yếu sử dụng công cụ này để phân tích xem cookie, mã thông báo, khóa và các tham số khác có thực sự ngẫu nhiên hay không.

Bộ giải mã là một cách nhanh chóng để mã hóa và giải mã HTML, base64, ASCII hex, thập lục phân, bát phân, nhị phân và Gzip.

Bộ so sánh có thể được sử dụng để so sánh các yêu cầu khác nhau. Thông thường, bạn sẽ muốn so sánh hai yêu cầu tương tự và tìm các phần của yêu cầu đã bị xóa, thêm và sửa đổi.

Nếu Burp Suite quá sáng đối với mắt hacker của bạn, hãy điều hướng đến Tùy chọn người dùng4Display và thay đổi Giao diện thành Darcula. Trong tab Tùy chọn người dùng, bạn cũng có thể tìm thấy các cấu hình kết nối bổ sung, cài đặt TLS và các tùy chọn khác để tìm hiểu các phím tắt hoặc định cấu hình các phím nóng của riêng bạn. Sau đó, bạn có thể lưu các cài đặt ưa thích của mình bằng Tùy chọn dự án, cho phép bạn lưu và tải các cấu hình cụ thể mà bạn muốn sử dụng cho mỗi dự án.

Tìm hiểu là một bộ tài nguyên tuyệt vời để giúp bạn tìm hiểu cách sử dụng Burp Thượng hạng. Tab này chứa các video hướng dẫn, Trung tâm hỗ trợ Burp Suite, chuyến tham quan có hướng dẫn về các tính năng của Burp và liên kết đến Học viện bảo mật web PortSwigger. Chắc chắn kiểm tra các tài nguyên này nếu bạn chưa quen với Burp!

Trong Bảng điều khiển, bạn có thể tìm thấy Burp Suite Pro Scanner. Máy quét là trình quét lỗ hổng ứng dụng web của Burp Suite Pro. Nó cho phép bạn tự động thu thập dữ liệu các ứng dụng web và quét các điểm yếu.

Extender là nơi chúng tôi sẽ lấy và sử dụng các tiện ích mở rộng của Burp Suite . Burp có một cửa hàng ứng dụng cho phép bạn tìm các tiện ích bổ sung để đơn giản hóa quá trình thử nghiệm ứng dụng web. Nhiều tiện ích mở rộng yêu cầu Burp Suite Pro, nhưng chúng tôi sẽ tận dụng tối đa các tiện ích mở rộng miễn phí để biến Burp thành một cường quốc hack API.

Chặn giao thông

Phiên Burp Suite thường sẽ bắt đầu bằng việc chặn lưu lượng truy cập. Nếu bạn đã thiết lập FoxyProxy và chứng chỉ Burp Suite đúng cách, quy trình sau sẽ hoạt động trơn tru. Bạn có thể sử dụng các hướng dẫn này để chặn mọi lưu lượng HTTP bằng Burp Suite:

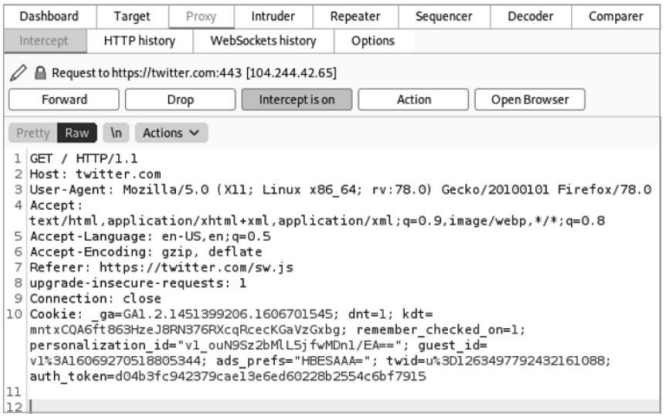
1. Khởi động Burp Suite và thay đổi tùy chọn Đánh chặn thành Đánh chặn đang bật (xem Hình 4-6).

Hình 4-6: Chặn được bật trong Burp Suite.

2. Trong trình duyệt của bạn, chọn proxy Hackz bằng FoxyProxy và duyệt đến mục tiêu của bạn, chẳng hạn như https://twitter.com (xem Hình 4-7). Trang web này sẽ không tải trong trình duyệt vì nó chưa bao giờ được gửi đến máy chủ; thay vào đó, yêu cầu sẽ đợi bạn trong Burp Suite.

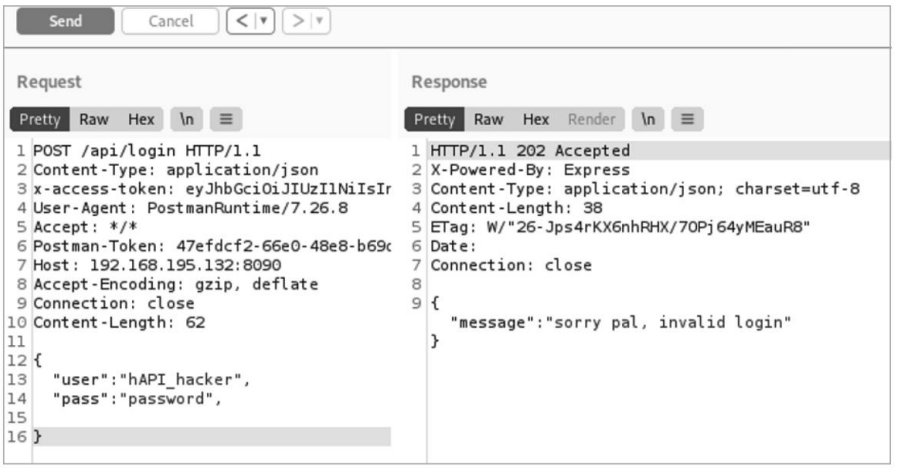
Hình 4-7: Yêu cầu tới Twitter được gửi tới Burp Suite thông qua proxy Hackz.

3. Trong Burp Suite, bạn sẽ thấy một cái gì đó giống như Hình 4-8. Điều này sẽ cho bạn biết rằng bạn đã chặn thành công một yêu cầu HTTP.



Hình 4-8: Một yêu cầu HTTP tới Twitter bị chặn bởi Burp Suite

Khi bạn đã nắm bắt được một yêu cầu, bạn có thể chọn một hành động để thực hiện với nó, chẳng hạn như chuyển tiếp yêu cầu bị chặn tới các mô-đun Burp Suite khác nhau. Bạn thực hiện các hành động bằng cách nhấp vào nút Hành động phía trên ngăn yêu cầu hoặc bằng cách nhấp chuột phải vào cửa sổ yêu cầu. Sau đó, bạn sẽ có tùy chọn chuyển tiếp yêu cầu tới một trong các mô-đun khác, chẳng hạn như Bộ lặp (xem Hình 4-9).



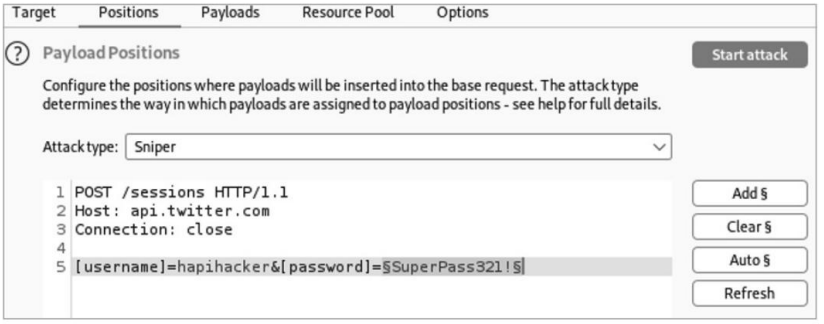
Hình 4-9: Bộ lặp Burp Suite

Mô-đun Bộ lặp là cách tốt nhất để xem cách máy chủ web phản hồi một yêu cầu. Điều này hữu ích để xem loại phản hồi nào bạn có thể mong đợi nhận được từ API trước khi bắt đầu một cuộc tấn công. Nó cũng hữu ích khi bạn cần thực hiện những thay đổi nhỏ đối với yêu cầu và muốn xem máy chủ phản hồi như thế nào.

Thay đổi yêu cầu với Intruder

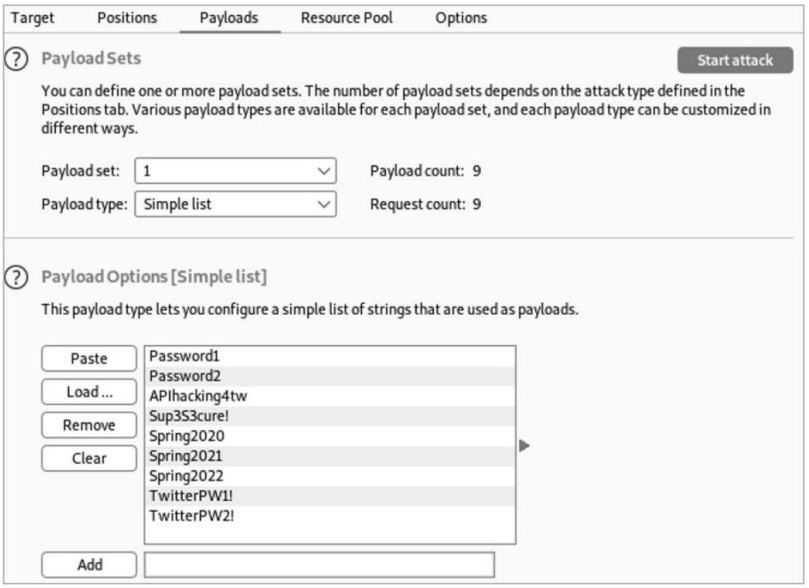
Chúng tôi đã đề cập rằng Intruder là một công cụ quét và làm mờ ứng dụng web. Nó hoạt động bằng cách cho phép bạn tạo các biến trong yêu cầu HTTP bị chặn, thay thế các biến đó bằng các bộ tải trọng khác nhau và gửi một loạt yêu cầu tới nhà cung cấp API.

Bất kỳ phần nào của yêu cầu HTTP đã nắm bắt đều có thể được chuyển đổi thành một vị trí có thể thay đổi hoặc tấn công bằng cách bao quanh nó bằng các ký hiệu § . Tải trọng có thể là bất kỳ thứ gì, từ danh sách từ đến tập hợp số, ký hiệu và bất kỳ loại đầu vào nào khác sẽ giúp bạn kiểm tra cách nhà cung cấp API sẽ phản hồi. Ví dụ, trong Hình 4-10, chúng tôi đã chọn mật khẩu làm vị trí tấn công, như được biểu thị bằng các ký hiệu §.



Hình 4-10: Một cuộc tấn công của Intruder vào api.twitter.com

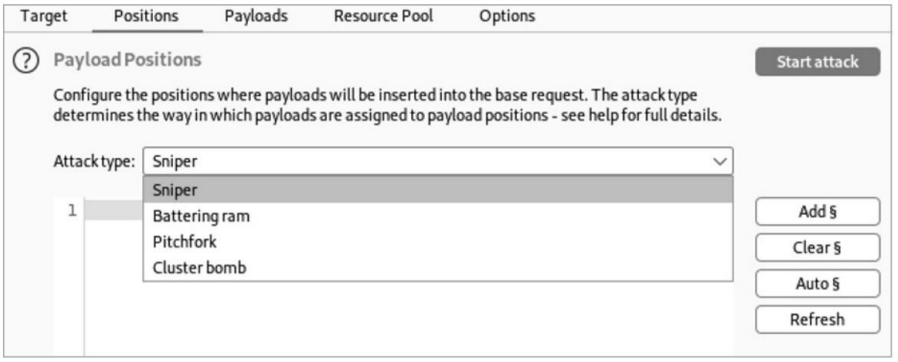
Điều này có nghĩa là SuperPass321! sẽ được thay thế bằng các giá trị từ danh sách các chuỗi được tìm thấy trong Tải trọng. Điều hướng đến tab Tải trọng để xem các chuỗi này, được hiển thị trong Hình 4-11.



Hình 4-11: Intruder Payloads với một danh sách các mật khẩu

Dựa trên danh sách tải trọng được hiển thị ở đây, Kẻ xâm nhập sẽ thực hiện một yêu cầu cho mỗi tải trọng được liệt kê trong tổng số chín yêu cầu. Khi một cuộc tấn công bắt đầu, mỗi chuỗi trong Tùy chọn tải trọng sẽ thay thế SuperPass123! lần lượt và tạo yêu cầu tới nhà cung cấp API.

Các kiểu tấn công của Kẻ xâm nhập xác định cách xử lý tải trọng. Như bạn có thể thấy trong Hình 4-12, có bốn kiểu tấn công khác nhau: bắn tỉa, đập phá, chĩa ba và bom chùm.



Hình 4-12: Các kiểu tấn công của Intruder

Bắn tỉa là kiểu tấn công đơn giản nhất; nó thay thế vị trí tấn công được thêm vào bằng một chuỗi được cung cấp từ một bộ tải trọng duy nhất. Một cuộc tấn công bắn tỉa được giới hạn trong việc sử dụng một trọng tải duy nhất, nhưng nó có thể có một số vị trí tấn công. Một cuộc tấn công bắn tỉa sẽ thay thế một vị trí tấn công theo yêu cầu, luân chuyển qua các vị trí tấn công khác nhau trong mỗi yêu cầu. Nếu bạn đang tấn công ba biến khác nhau với một tải trọng duy nhất, thì nó sẽ giống như thế này:

```
§Biến1§, §biến2§, §biến3§
Yêu cầu 1:     Tải trọng1, biến2, biến3
Yêu cầu 2:     Biến1, tải trọng1, biến3
Yêu cầu 3:     Biến1, biến2, tải trọng1
```

Đập ram giống như cuộc tấn công bắn tỉa ở chỗ nó cũng sử dụng một trọng tải, nhưng nó sẽ sử dụng trọng tải đó trên tất cả các vị trí tấn công trong một yêu cầu. Nếu bạn đang thử nghiệm SQL injection trên một số vị trí đầu vào trong một yêu cầu, bạn có thể làm mờ tất cả chúng đồng thời với ram đập.

Pitchfork được sử dụng để kiểm tra nhiều kết hợp tải trọng cùng một lúc thời gian. Ví dụ: nếu bạn có một danh sách kết hợp tên người dùng và mật khẩu bị rò rỉ, bạn có thể sử dụng hai tải trọng cùng nhau để kiểm tra xem có bất kỳ thông tin xác thực nào được sử dụng với ứng dụng đang được kiểm tra hay không. Tuy nhiên, cuộc tấn công này không thử các kết hợp tải trọng khác nhau; nó sẽ chỉ quay vòng qua các bộ tải trọng như sau: user1:pass1, user2:pass2, user3:pass3.

Bom chùm sẽ quay vòng qua tất cả các kết hợp có thể có của các trọng tải được cung cấp. Nếu bạn cung cấp hai tên người dùng và ba mật khẩu, tải trọng sẽ được sử dụng theo các cặp sau: user1:pass1, user1:pass2, user1:pass3, user2:pass1, user2:pass2, user2:pass3.

Loại tấn công để sử dụng phụ thuộc vào tình huống của bạn. Nếu bạn đang làm mờ một vị trí tấn công đơn lẻ, hãy sử dụng xạ thủ. Nếu bạn đang làm mờ nhiều vị trí tấn công cùng một lúc, hãy sử dụng ram đập. Khi bạn cần kiểm tra các tổ hợp tải trả phí đã đặt, hãy sử dụng pitchfork. Đối với nỗ lực rải mật khẩu, hãy sử dụng bom chùm.

Kẻ xâm nhập sẽ giúp bạn tìm ra các lỗ hổng API như ủy quyền cấp đối tượng bị hỏng, lộ dữ liệu quá mức, xác thực bị hỏng, ủy quyền cấp chức năng bị hỏng, gán hàng loạt, tiêm và quản lý tài sản không phù hợp. Intruder về cơ bản là một công cụ fuzzing thông minh cung cấp một danh sách các kết quả chứa các yêu cầu và phản hồi riêng lẻ. Bạn có thể tương tác với yêu cầu bạn muốn fuzz và thay thế vị trí tấn công bằng đầu vào bạn chọn. Các lỗ hổng API này thường được phát hiện bằng cách gửi đúng trọng tải đến đúng vị trí.

Ví dụ: nếu một API dễ bị tấn công ủy quyền như BOLA, chúng tôi có thể thay thế các ID tài nguyên được yêu cầu bằng một tải trọng chứa danh sách các ID tài nguyên có thể có. Sau đó, chúng tôi có thể bắt đầu cuộc tấn công bằng Intruder, thứ sẽ thực hiện tất cả các yêu cầu và cung cấp cho chúng tôi danh sách kết quả để xem xét. Tôi sẽ đề cập đến API fuzzing trong Chương 9 và tấn công ủy quyền API trong Chương 10.

MỞ RỘNG SỨC MẠNH CỦA BURP SUITE

Một trong những lợi ích chính của Burp Suite là bạn có thể cài đặt các tiện ích mở rộng tùy chỉnh. Các tiện ích mở rộng này có thể giúp bạn định hình Burp Suite thành công cụ hack API tối ưu. Để cài đặt tiện ích mở rộng, hãy sử dụng thanh tìm kiếm để tìm tiện ích bạn đang tìm rồi nhấp vào nút Cài đặt . Một số tiện ích mở rộng yêu cầu tài nguyên bổ sung và có các yêu cầu cài đặt phức tạp hơn. Đảm bảo bạn làm theo hướng dẫn cài đặt cho từng tiện ích mở rộng. Tôi khuyên bạn nên thêm những cái sau.

TỰ ĐỘNG HÓA

Autorize là một tiện ích mở rộng giúp tự động kiểm tra ủy quyền, đặc biệt đối với các lỗ hổng BOLA. Bạn có thể thêm mã thông báo của tài khoản UserA và UserB, sau đó thực hiện một loạt hành động để tạo và tương tác với các tài nguyên với tư cách là UserA. Ngoài ra, Autorize có thể tự động cố gắng tương tác với tài nguyên của UserA bằng tài khoản UserB. Autorize sẽ đánh dấu bất kỳ yêu cầu thú vị nào có thể dễ bị BOLA tấn công.

JSON WEB TOKENS

Tiện ích mở rộng JSON Web Tokens giúp bạn phân tích và tấn công JSON Web Tokens. Chúng ta sẽ sử dụng phần mở rộng này để thực hiện các cuộc tấn công ủy quyền sau trong Chương 8.

MÁY QUÉT InQL

InQL là một tiện ích mở rộng sẽ hỗ trợ chúng tôi trong các cuộc tấn công chống lại API GraphQL. Chúng ta sẽ tận dụng tối đa phần mở rộng này trong Chương 14.

(còn tiếp)

XOAY IP

IP Rotate cho phép bạn thay đổi địa chỉ IP mà bạn đang tấn công để chỉ ra các máy chủ đám mây khác nhau ở các khu vực khác nhau. Điều này cực kỳ hữu ích đối với các nhà cung cấp API chỉ chặn các cuộc tấn công dựa trên địa chỉ IP.

BỎ QUA WAF

Tiện ích mở rộng WAF Bypass thêm một số tiêu đề cơ bản vào các yêu cầu của bạn để vượt qua một số tường lửa ứng dụng web (WAF). Một số WAF có thể bị đánh lừa bằng cách đưa vào một số tiêu đề IP nhất định trong yêu cầu. WAF Bypass giúp bạn không phải thêm các tiêu đề theo cách thủ công như X-Originating-IP, X-Forwarded-For, X-Remote-IP và X-Remote-Addr. Các tiêu đề này thường bao gồm một địa chỉ IP và bạn có thể chỉ định một địa chỉ mà bạn cho là được phép, chẳng hạn như địa chỉ IP bên ngoài của mục tiêu (127.0.0.1) hoặc một địa chỉ mà bạn nghi ngờ là đáng tin cậy.

Trong phần thực hành ở cuối chương này, tôi sẽ hướng dẫn bạn cách tương tác với một API, nắm bắt lưu lượng truy cập bằng Burp Suite và sử dụng Intruder để khám phá danh sách các tài khoản người dùng hiện có. Để tìm hiểu thêm về Burp Suite, hãy truy cập PortSwigger WebSecurity Academy tại https://portswigger.net/web-security hoặc tham khảo tài liệu về Burp Suite tại https://portswigger.net/burp/documentation.

Tạo các yêu cầu API trong Postman, Trình duyệt API

Chúng tôi sẽ sử dụng Postman để giúp chúng tôi tạo các yêu cầu API và trực quan hóa các phản hồi. Bạn có thể coi Postman như một trình duyệt web được xây dựng để tương tác với các API. Ban đầu được thiết kế như một ứng dụng API REST, giờ đây nó có tất cả các loại khả năng để tương tác với REST, SOAP và GraphQL. Ứng dụng này được tích hợp nhiều tính năng để tạo yêu cầu HTTP, nhận phản hồi, tạo tập lệnh, kết nối các yêu cầu với nhau, tạo thử nghiệm tự động và quản lý tài liệu API.

Chúng tôi sẽ sử dụng Postman làm trình duyệt được chọn để gửi yêu cầu API tới máy chủ, thay vì mặc định cho Firefox hoặc Chrome. Phần này bao gồm các tính năng quan trọng nhất của Postman và bao gồm các hướng dẫn sử dụng trình tạo yêu cầu Postman, tổng quan về cách làm việc với các bộ sưu tập và một số điều cơ bản xung quanh việc xây dựng các bài kiểm tra yêu cầu. Ở phần sau của chương này, chúng ta sẽ định cấu hình Postman để hoạt động trơn tru với Burp Suite.

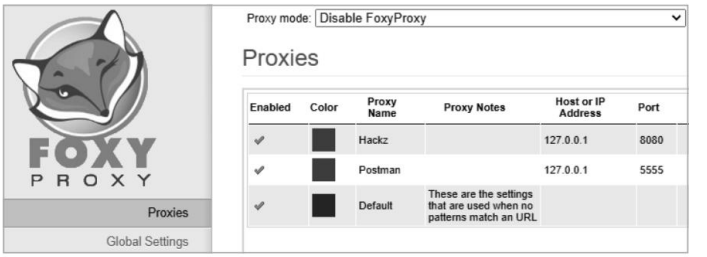
Để thiết lập Postman trên Kali, hãy mở thiết bị đầu cuối của bạn và nhập thông tin sau lệnh:

```
$ sudo wget https://dl.pstmn.io/download/latest/linux64 -O postman-linux-x64.tar.gz
$ sudo tar -xvzf postman-linux-x64.tar.gz -C /opt
$ sudo ln -s /opt/Người đưa thư/Người đưa thư /usr/bin/người đưa thư
```

Nếu mọi thứ diễn ra theo đúng kế hoạch, bạn sẽ có thể khởi chạy Postman bằng cách nhập postman vào thiết bị đầu cuối của mình. Đăng ký tài khoản miễn phí bằng địa chỉ email, tên người dùng và mật khẩu. Người đưa thư sử dụng tài khoản

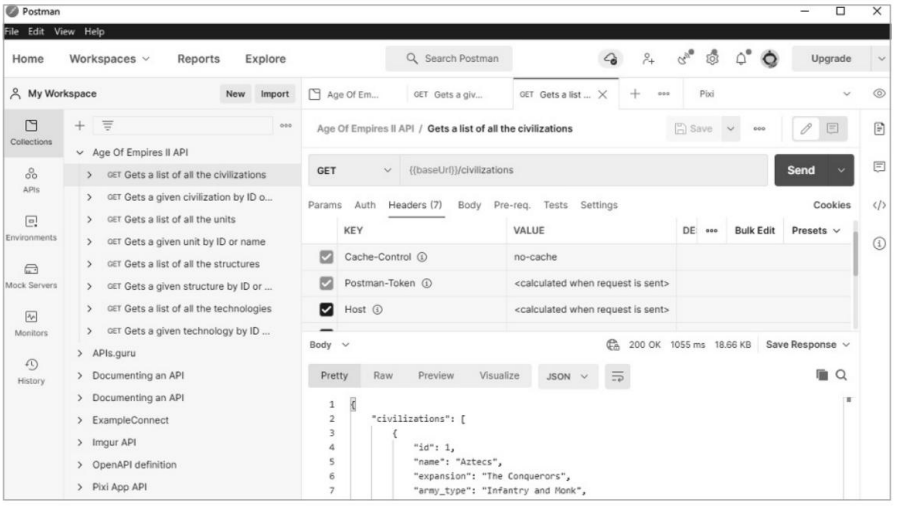
để cộng tác và đồng bộ hóa thông tin trên các thiết bị. Ngoài ra, bạn có thể bỏ qua màn hình đăng nhập bằng cách nhấp vào nút Bỏ qua đăng nhập và đưa tôi thẳng đến nút ứng dụng.

Tiếp theo, bạn sẽ cần thực hiện quy trình thiết lập FoxyProxy lần thứ hai (tham khảo phần “Thiết lập FoxyProxy” trước đó trong chương này) để Postman có thể chặn các yêu cầu. Quay lại bước 4 và thêm proxy mới. Thêm cùng một địa chỉ IP máy chủ, 127.0.0.1 và đặt cổng thành 5555, cổng mặc định cho proxy của Postman. Cập nhật tên của proxy trong tab Chung thành Người đưa thư và lưu lại. Tab FoxyProxy của bạn bây giờ sẽ giống như Hình 4-13.



Hình 4-13: FoxyProxy với các proxy Hackz và Postman được thiết lập

Từ launchpad, hãy mở một tab mới giống như bạn làm trong bất kỳ trình duyệt nào khác bằng cách nhấp vào nút tab mới (+) hoặc sử dụng phím tắt CTRL-T. Như bạn có thể thấy trong Hình 4-14, giao diện của Postman có thể hơi khó hiểu nếu bạn chưa quen với nó.

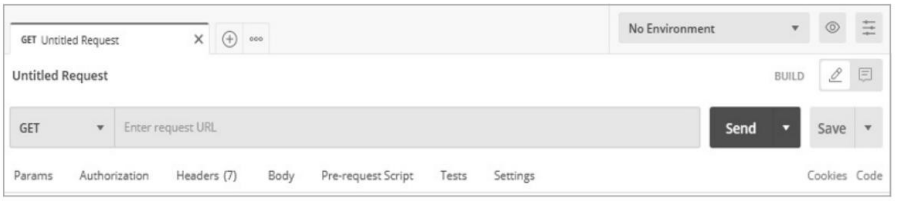


Hình 4-14: Trang đích chính của Postman với phản hồi từ bộ sưu tập API

Hãy bắt đầu bằng cách thảo luận về trình tạo yêu cầu mà bạn sẽ thấy khi mở một tab mới.

Trình tạo yêu cầu

Trình tạo yêu cầu, như trong Hình 4-15, là nơi bạn có thể tạo từng yêu cầu bằng cách thêm tham số, tiêu đề ủy quyền, v.v.



Hình 4-15: Trình tạo yêu cầu Postman

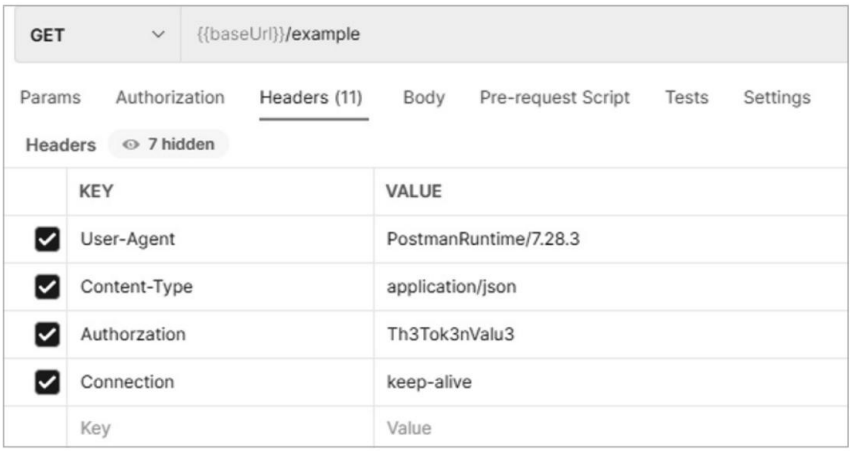
Trình tạo yêu cầu chứa một số tab hữu ích để xây dựng chính xác các tham số, tiêu đề và nội dung của yêu cầu. Tab Thông số là nơi bạn có thể thêm các tham số truy vấn và đường dẫn vào một yêu cầu. Về cơ bản, điều này cho phép bạn nhập các cặp khóa/giá trị khác nhau cùng với mô tả về các tham số đó. Một tính năng tuyệt vời của Postman là bạn có thể tận dụng sức mạnh của các biến khi tạo các yêu cầu của mình. Nếu bạn nhập một API và nó chứa một biến như :company trong http://example.com/:company/profile, Postman sẽ tự động phát hiện điều này và cho phép bạn cập nhật biến đó thành một giá trị khác, chẳng hạn như giá trị thực Tên công ty. Chúng ta sẽ thảo luận về các bộ sưu tập và môi trường sau trong phần này.

Tab Ủy quyền bao gồm nhiều hình thức ủy quyền tiêu chuẩn tiêu đề để bạn đưa vào yêu cầu của mình. Nếu đã lưu mã thông báo trong một môi trường, bạn có thể chọn loại mã thông báo và sử dụng tên của biến để đưa vào. Bằng cách di chuột qua tên biến, bạn có thể thấy thông tin xác thực được liên kết. Một số tùy chọn ủy quyền có sẵn trong trường Loại sẽ giúp bạn tự động định dạng tiêu đề ủy quyền. Các loại ủy quyền bao gồm một số tùy chọn dự kiến, chẳng hạn như không có xác thực, khóa API, Mã thông báo mang và Xác thực cơ bản. Ngoài ra, bạn có thể sử dụng quyền được đặt cho toàn bộ bộ sưu tập bằng cách chọn xác thực kế thừa từ cấp độ gốc.

Tab Tiêu đề bao gồm các cặp khóa và giá trị cần thiết cho các yêu cầu HTTP nhất định. Postman có một số chức năng tích hợp sẵn để tự động tạo các tiêu đề cần thiết và đề xuất các tiêu đề phổ biến với các tùy chọn đặt trước.

Trong Postman, có thể thêm các giá trị cho tham số, tiêu đề và các phần của công việc cơ thể bằng cách nhập thông tin trong cột Khóa và cột Giá trị tương ứng (xem Hình 4-16). Một số tiêu đề sẽ được tạo tự động, nhưng bạn có thể thêm tiêu đề của riêng mình khi cần thiết.

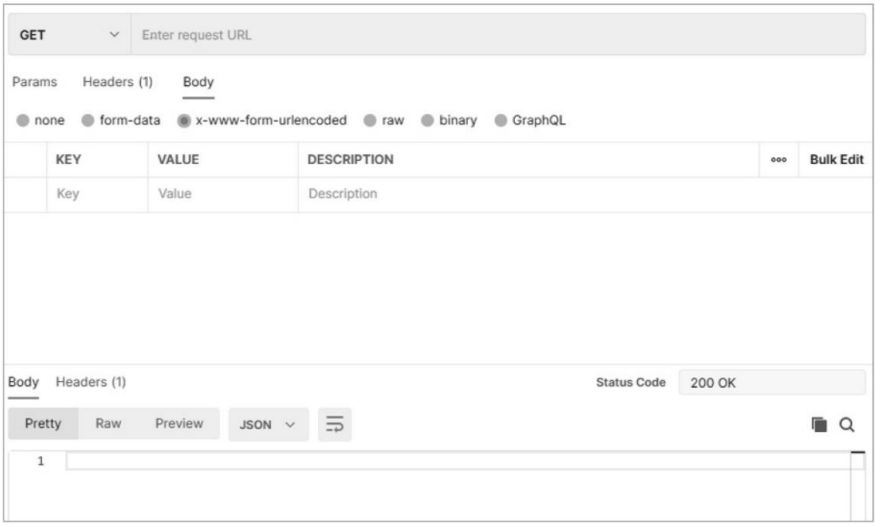
Trong các khóa và giá trị, bạn cũng có khả năng sử dụng các biến bộ sưu tập và biến môi trường. (Chúng ta sẽ đề cập đến các bộ sưu tập sau trong phần này.) Ví dụ: chúng ta đã biểu thị giá trị cho khóa mật khẩu bằng cách sử dụng tên biến {admin_creds}.



Hình 4-16: Tiêu đề giá trị và khóa của người đưa thư

Trình tạo yêu cầu cũng có thể chạy các tập lệnh yêu cầu trước, có thể xâu chuỗi cùng nhau các yêu cầu khác nhau phụ thuộc vào nhau. Ví dụ: nếu yêu cầu 1 đưa ra một giá trị tài nguyên cần thiết cho yêu cầu sau, bạn có thể tạo tập lệnh để tự động thêm giá trị tài nguyên đó vào yêu cầu 2.

Trong trình tạo yêu cầu của Postman, bạn có thể sử dụng một số bảng để tạo các yêu cầu API phù hợp và xem xét phản hồi. Khi bạn đã gửi yêu cầu, phản hồi sẽ hiển thị trong bảng phản hồi (xem Hình 4-17).



Hình 4-17: Bảng yêu cầu và phản hồi của Postman

Bạn có thể đặt bảng phản hồi ở bên phải hoặc bên dưới bảng yêu cầu. Bằng cách nhấn CTRL-ALT-V, bạn có thể chuyển bảng yêu cầu và phản hồi giữa chế độ xem một ngăn và nhiều ngăn.

Trong Bảng 4-2, tôi đã tách các mục thành bảng yêu cầu và bảng phản hồi.

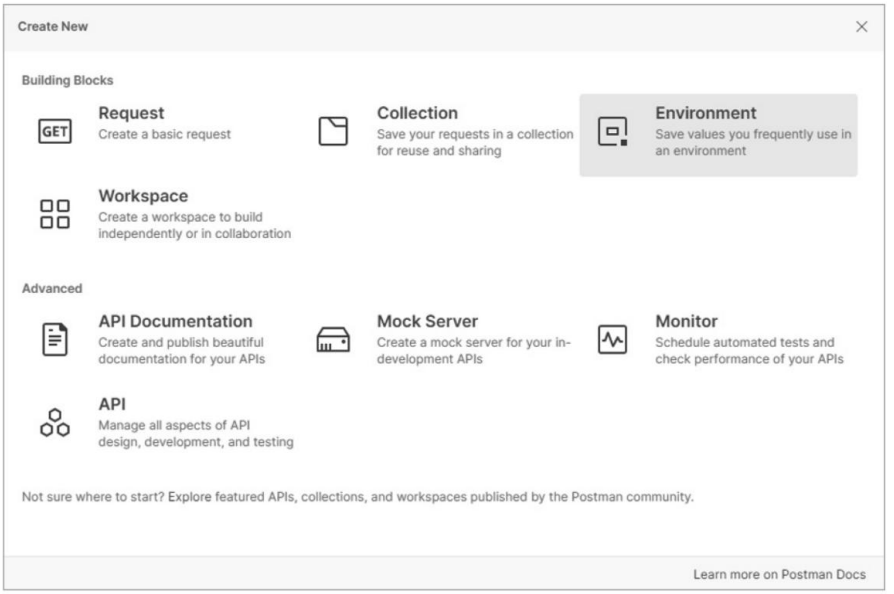
Bảng 4-2: Bảng trình tạo yêu cầu

bảng điều khiển	Mục đích
Lời yêu cầu	
Phương thức yêu cầu HTTP	Phương thức yêu cầu được tìm thấy ở bên trái của thanh URL yêu cầu (ở trên cùng bên trái của Hình 4-17, nơi có menu thả xuống cho GET). Các tùy chọn bao gồm tất cả các yêu cầu tiêu chuẩn: GET, POST, PUT, PATCH, DELETE, HEAD và OPTIONS. Nó cũng bao gồm một số phương thức yêu cầu khác như COPY, LINK, UNLINK, PURGE, LOCK, UNLOCK, PROPFIND và VIEW.
Thân hình	Trong Hình 4-17, đây là tab thứ ba trong ngăn yêu cầu. Điều này cho phép thêm dữ liệu nội dung vào yêu cầu, được sử dụng chủ yếu để thêm hoặc cập nhật dữ liệu khi sử dụng PUT, POST hoặc PATCH.
tùy chọn cơ thể	Tùy chọn nội dung là định dạng của phản hồi. Chúng được tìm thấy bên dưới tab Nội dung khi nó được chọn. Các tùy chọn hiện bao gồm none, form-data, x-www-formurlencoded, raw, binary và GraphQL. Các tùy chọn này cho phép bạn xem dữ liệu phản hồi ở nhiều dạng khác nhau.
Kịch bản yêu cầu trước	Tập lệnh dựa trên JavaScript có thể được thêm và thực thi trước khi gửi yêu cầu. Điều này có thể được sử dụng để tạo các biến, giúp khắc phục lỗi và thay đổi các tham số yêu cầu.
Bài kiểm tra	Không gian này cho phép viết các bài kiểm tra dựa trên JavaScript được sử dụng để phân tích và kiểm tra phản hồi API. Điều này được sử dụng để đảm bảo các phản hồi API đang hoạt động như mong đợi.
Cài đặt	Các cài đặt khác nhau về cách Người đưa thư sẽ xử lý các yêu cầu.
Phản ứng	
nội dung phản hồi	Phần thân của phản hồi HTTP. Nếu Postman là một trình duyệt web điển hình, thì đây sẽ là cửa sổ chính để xem thông tin được yêu cầu.
Bánh quy	Điều này hiển thị tất cả các cookie, nếu có, được bao gồm trong phản hồi HTTP. Tab này sẽ bao gồm thông tin về loại cookie, giá trị cookie, đường dẫn, ngày hết hạn và cờ bảo mật cookie.
tiêu đề	Đây là nơi đặt tất cả các tiêu đề phản hồi HTTP.
Kết quả kiểm tra	Nếu bạn đã tạo bất kỳ thử nghiệm nào cho yêu cầu của mình, thì đây là nơi bạn có thể xem kết quả của những thử nghiệm đó.

môi trường

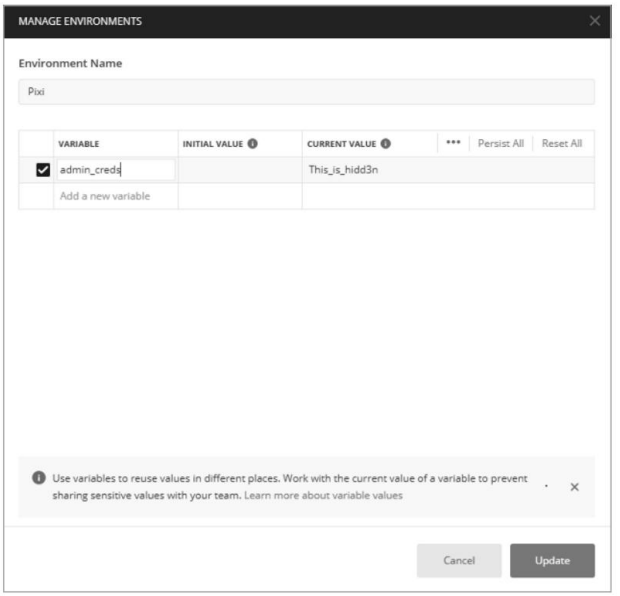
Một môi trường cung cấp một cách để lưu trữ và sử dụng các biến giống nhau trên các API. Biến môi trường là một giá trị sẽ thay thế một biến trong một môi trường. Ví dụ: giả sử bạn đang tấn công API sản xuất nhưng cũng phát hiện ra phiên bản thử nghiệm của API sản xuất; bạn có thể muốn sử dụng một môi trường để chia sẻ giá trị giữa các yêu cầu của mình với hai API. Rốt cuộc, có khả năng các API sản xuất và thử nghiệm chia sẻ các giá trị như mã thông báo API, đường dẫn URL và ID tài nguyên.

Để tạo các biến môi trường, hãy tìm Môi trường ở trên cùng bên phải của trình tạo yêu cầu (menu thả xuống có nội dung "Không có môi trường" theo mặc định) rồi nhấn CTRL-N để hiển thị bảng Tạo mới và chọn Môi trường, như minh họa trong Hình 4-18.



Hình 4-18: Bảng Tạo mới trong Postman

Bạn có thể cung cấp cho một biến môi trường cả giá trị ban đầu và giá trị thuê hiện tại (xem Hình 4-19). Giá trị ban đầu sẽ được chia sẻ nếu bạn chia sẻ môi trường Postman của mình với một nhóm, trong khi giá trị hiện tại không được chia sẻ và chỉ được lưu trữ cục bộ. Ví dụ: nếu bạn có khóa riêng, bạn có thể lưu trữ khóa riêng dưới dạng giá trị hiện tại. Sau đó, bạn sẽ có thể sử dụng biến ở những nơi mà bạn sẽ phải dán khóa riêng tư.



Hình 4-19: Cửa sổ Manage Environments trong Postman hiển thị biến admin_creds với giá trị hiện tại là This_is_hidd3n

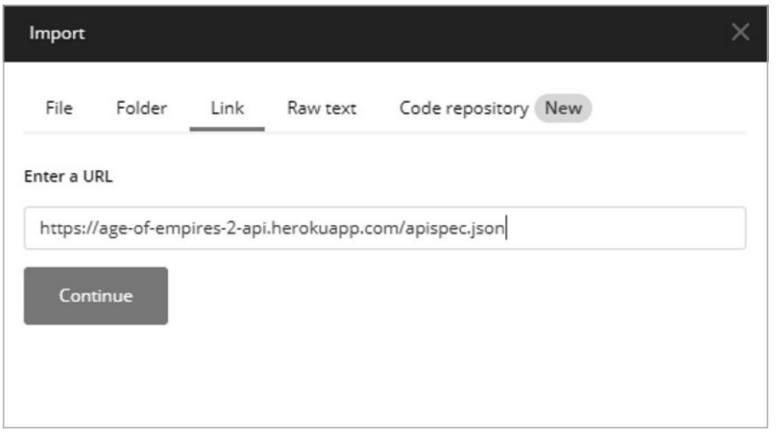
bộ sưu tập

Bộ sưu tập là các nhóm yêu cầu API có thể được nhập vào Postman. Nếu một nhà cung cấp API cung cấp một bộ sưu tập, bạn sẽ không phải nhập từng yêu cầu một cách vật lý. Thay vào đó, bạn chỉ có thể nhập bộ sưu tập của nó. Cách tốt nhất để hiểu chức năng này là tải xuống bộ sưu tập API công khai cho Người đưa thư của bạn từ https://www.postman.com/explore/collections. Ví dụ xuyên suốt phần này, tôi sẽ tham khảo bộ sưu tập Age of Empires II.

Nút Nhập cho phép bạn nhập các bộ sưu tập, môi trường và thông số API. Hiện tại, Postman hỗ trợ OpenAPI 3.0, RAML 0.8, RAML 1.0, GraphQL, cURL, WADL, Swagger 1.2, Swagger 2.0, Runscope và DHC. Bạn có thể thực hiện thử nghiệm của mình dễ dàng hơn một chút nếu bạn có thể nhập đặc tả API mục tiêu của mình. Làm điều này sẽ giúp bạn tiết kiệm thời gian phải tạo thủ công tất cả các yêu cầu API.

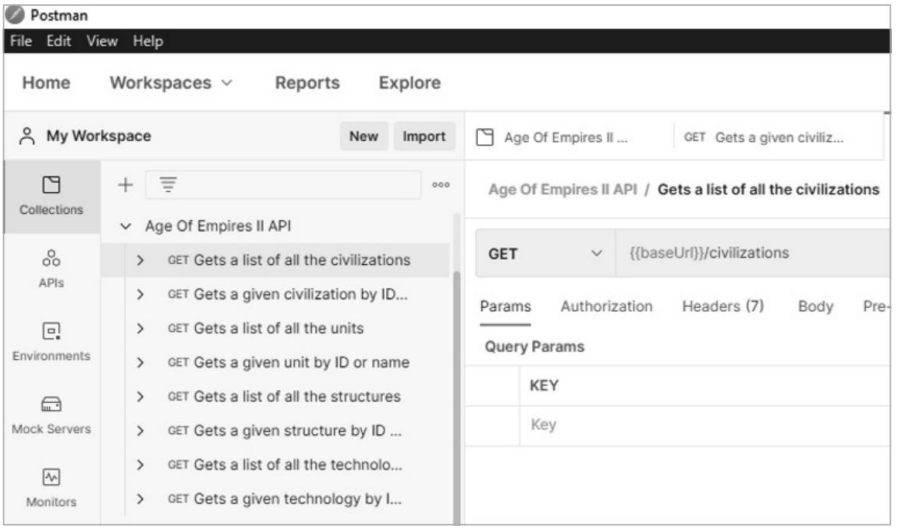
Tất cả các bộ sưu tập, môi trường và thông số kỹ thuật đều có thể được nhập dưới dạng tệp, thư mục, liên kết hoặc thử nghiệm thô hoặc thông qua liên kết tài khoản GitHub của bạn. Ví dụ: bạn có thể nhập API cho trò chơi PC cổ điển Age of Empires II từ https://age-of-empires-2-api.herokuapp.com/apispec.json như sau:

1. Nhấp vào nút Nhập ở trên cùng bên trái của Postman.
2. Chọn tab Liên kết (xem Hình 4-20).
3. Dán URL vào đặc tả API và nhấp vào Tiếp tục.
4. Trên màn hình Xác nhận nhập của bạn, hãy nhấp vào Nhập.



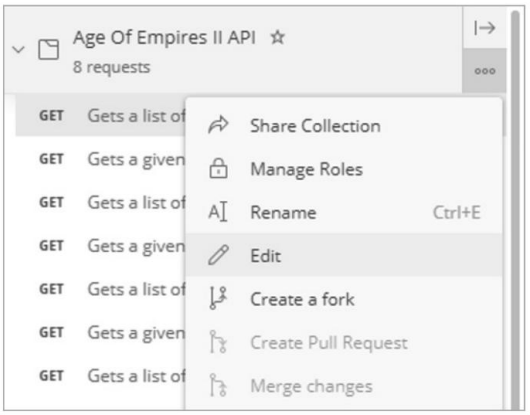
Hình 4-20: Nhập đặc tả API trong Postman bằng cách sử dụng tab Liên kết trong Nhập bảng điều khiển

Khi quá trình này hoàn tất, bạn sẽ có bộ sưu tập Age of Empires II được lưu trong Postman. Bây giờ kiểm tra nó ra. Chọn một trong các yêu cầu trong bộ sưu tập được hiển thị trong Hình 4-21 và nhấp vào Gửi.



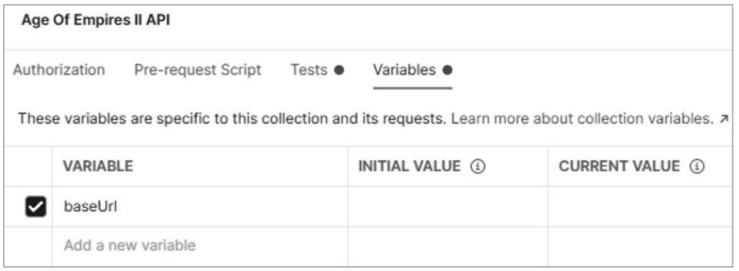
Hình 4-21: Thanh bên Bộ sưu tập với các yêu cầu GET API Age of Empires II đã nhập

Để yêu cầu hoạt động, trước tiên bạn có thể phải kiểm tra các biến của bộ sưu tập để đảm bảo chúng được đặt thành giá trị chính xác. Để xem các biến của bộ sưu tập, bạn cần điều hướng đến cửa sổ Chỉnh sửa Bộ sưu tập bằng cách chọn ing Chỉnh sửa trong nút Xem Thêm Tác vụ (được biểu thị bằng ba vòng tròn, như trong Hình 4-22).



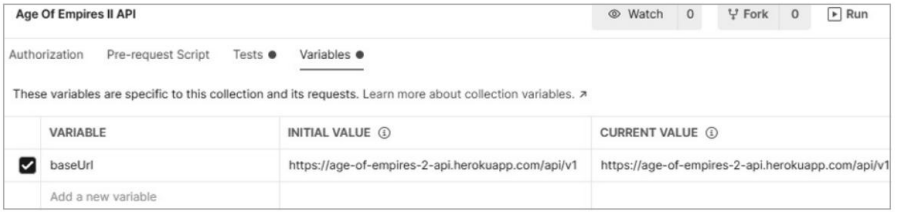
Hình 4-22: Chỉnh sửa bộ sưu tập trong Postman

Khi bạn đang ở trong cửa sổ Chỉnh sửa Bộ sưu tập, hãy chọn Biến, như thể hiện trong Hình 4-23.



Hình 4-23: Các biến bộ sưu tập API của Age of Empires II

Ví dụ: bộ sưu tập API Age of Empires II sử dụng biến {{baseUrl}}. Vấn đề với {{baseUrl}} hiện tại là không có giá trị nào. Chúng tôi cần cập nhật biến này thành URL đầy đủ của API công khai, https://age-of-empires-2-api.herokuapp.com/api/v1. Thêm URL đầy đủ và nhấp vào Lưu để cập nhật các thay đổi của bạn (xem Hình 4-24).



Hình 4-24: Biến baseURL được cập nhật

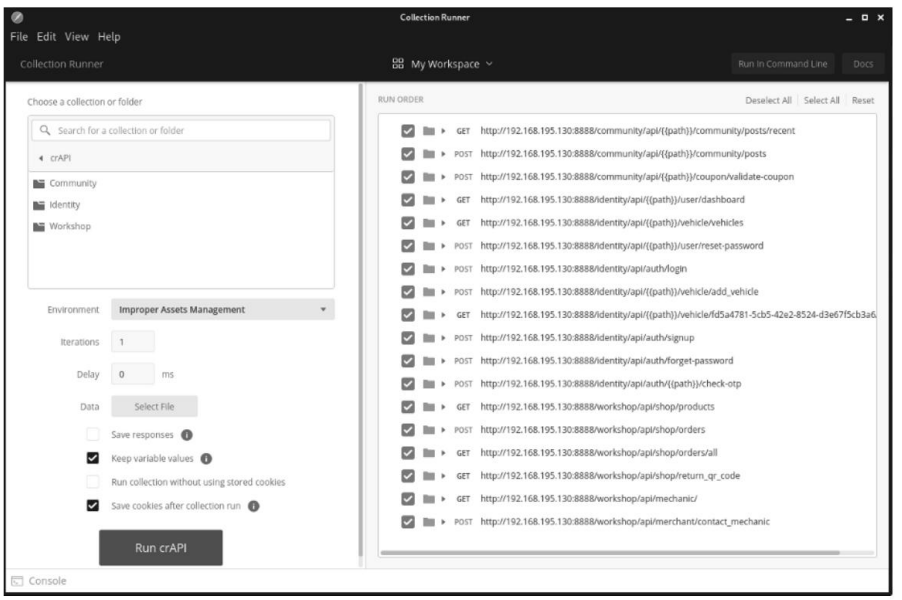
Bây giờ biến đã được cập nhật, bạn có thể chọn một trong các yêu cầu và nhấp vào Gửi. Nếu thành công, bạn sẽ nhận được phản hồi tương tự như trong Hình 4-25.

Hình 4-25: Sử dụng thành công bộ sưu tập API Age of Empires II trong Postman

Bất cứ khi nào bạn nhập một bộ sưu tập và gặp lỗi, bạn có thể sử dụng quy trình này để khắc phục sự cố các biến của bộ sưu tập. Ngoài ra, hãy đảm bảo kiểm tra xem bạn có bỏ qua bất kỳ yêu cầu ủy quyền nào không.

Người chạy bộ sưu tập

Collection Runner cho phép bạn chạy tất cả các yêu cầu đã lưu trong một bộ sưu tập (xem Hình 4-26). Bạn có thể chọn bộ sưu tập bạn muốn chạy, môi trường bạn muốn ghép nối với nó, số lần bạn muốn chạy bộ sưu tập và độ trễ trong trường hợp có các yêu cầu giới hạn tốc độ.



Hình 4-26: Người chạy bộ sưu tập Postman

Các yêu cầu cũng có thể được đưa vào một thứ tự cụ thể. Khi Trình chạy bộ sưu tập đã chạy, bạn có thể xem lại Tóm tắt lần chạy để xem từng yêu cầu đã được xử lý như thế nào. Ví dụ: nếu tôi mở Trình chạy bộ sưu tập, chọn Twitter API v2 và chạy Trình chạy bộ sưu tập, tôi có thể xem tổng quan về tất cả các yêu cầu API trong bộ sưu tập đó.

Đoạn mã

Ngoài các bảng điều khiển, bạn cũng nên biết về tính năng đoạn mã. Ở trên cùng bên phải của ngăn yêu cầu, bạn sẽ thấy nút Mã. Nút này có thể được sử dụng để dịch yêu cầu đã tạo thành nhiều mat khác nhau, bao gồm cURL, Go, HTTP, JavaScript, NodeJS, PHP và Python.
Đây là một tính năng hữu ích khi chúng tôi tạo một yêu cầu với Postman và sau đó cần chuyển sang một công cụ khác. Bạn có thể tạo một yêu cầu API phức tạp trong Postman, tạo một yêu cầu cURL, sau đó sử dụng yêu cầu đó với các công cụ dòng lệnh khác.

Bảng kiểm tra

Bảng Kiểm tra cho phép bạn tạo các tập lệnh sẽ được chạy dựa trên các phản hồi đối với yêu cầu của bạn. Nếu bạn không phải là lập trình viên, bạn sẽ đánh giá cao việc Postman đã tạo sẵn các đoạn mã dựng sẵn ở phía bên phải của bảng Kiểm tra. Bạn có thể dễ dàng tạo thử nghiệm bằng cách tìm một đoạn mã dựng sẵn, nhấp vào đoạn mã đó và điều chỉnh thử nghiệm để phù hợp với nhu cầu thử nghiệm của bạn. Tôi khuyên bạn nên kiểm tra các đoạn sau:

- Mã trạng thái: Mã là 200
- Thời gian đáp ứng nhỏ hơn 200ms
- Nội dung phản hồi: chứa chuỗi

Các đoạn mã JavaScript này khá đơn giản. Ví dụ,
bài kiểm tra Mã trạng thái: Mã là 200 như sau:

```
pm.test("Mã trạng thái là 200", function() {
    pm.response.to.have.status(200);
});
```

Bạn có thể thấy rằng tên của bài kiểm tra sẽ được hiển thị trong kết quả kiểm tra là "Mã trạng thái là 200." Hàm đang kiểm tra để đảm bảo phản hồi của Người đưa thư có trạng thái 200. Chúng tôi có thể dễ dàng điều chỉnh JavaScript để kiểm tra bất kỳ mã trạng thái nào bằng cách cập nhật (200) thành mã trạng thái mong muốn và thay đổi tên kiểm tra cho phù hợp. Ví dụ: nếu chúng tôi muốn kiểm tra mã trạng thái 400, chúng tôi có thể thay đổi mã như sau:

```
pm.test("Mã trạng thái là 400", function() {
    pm.response.to.have.status(400);
});
```

Nó đơn giản như vậy! Bạn thực sự không cần phải là một lập trình viên để hiểu những đoạn mã JavaScript này.

Hình 4-27 cho thấy một loạt các thử nghiệm được bao gồm trong yêu cầu API tới API công khai AOE2. Các bài kiểm tra bao gồm kiểm tra mã trạng thái 200, độ trễ dưới 200 ms và "Người Ba Tư" trong chuỗi phản hồi.


```
1  pm.test("Status code is 200", function () {
2      pm.response.to.have.status(200);
3  });
4  pm.test("Response time is less than 200ms", function () {
5      pm.expect(pm.response.responseTime).to.be.below(200);
6  });
7  pm.test("Body matches string", function () {
8      pm.expect(pm.response.text()).to.include("Persians");
9  });
```

Hình 4-27: Kiểm tra API công khai AOE2

Sau khi các bài kiểm tra của bạn được định cấu hình, bạn có thể kiểm tra tab Kết quả kiểm tra của phản hồi để xem các bài kiểm tra thành công hay thất bại. Một thực hành tốt với việc tạo ra các bài kiểm tra là đảm bảo rằng các bài kiểm tra không thành công. Các bài kiểm tra chỉ có hiệu quả nếu chúng vượt qua và thất bại khi chúng được cho là như vậy. Do đó, hãy gửi một yêu cầu có thể tạo ra các điều kiện mà bạn cho rằng sẽ vượt qua hoặc không đạt bài kiểm tra để đảm bảo rằng nó hoạt động bình thường. Để biết thêm thông tin về cách tạo tập lệnh thử nghiệm, hãy xem tài liệu của Postman (https://learning.postman.com/docs/viết-kịch bản/kiểm tra-kịch bản).

Bây giờ bạn có nhiều tùy chọn khác để khám phá trong Postman. Giống như Burp Suite, Postman có Trung tâm học tập (https://learning.postman.com) cho các tài nguyên trực tuyến dành cho những người muốn phát triển sự hiểu biết sâu hơn về phần mềm. Ngoài ra, nếu bạn muốn xem lại tài liệu về Postman, bạn có thể tìm thấy nó tại https://learning.postman.com/docs/getting-started/introduction.

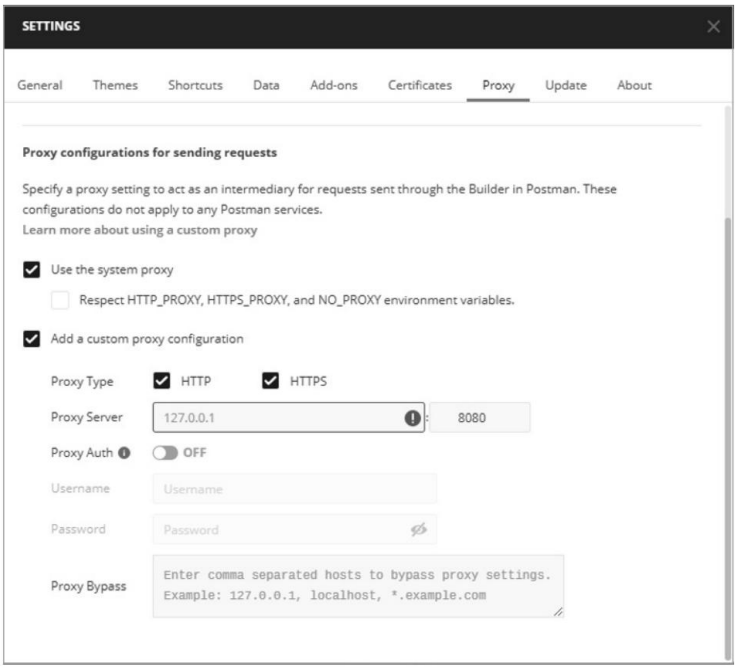
Cấu hình Postman để làm việc với Burp Suite

Postman rất hữu ích để tương tác với các API và Burp Suite là một công cụ mạnh mẽ để thử nghiệm ứng dụng web. Nếu bạn kết hợp các ứng dụng này, bạn có thể định cấu hình và kiểm tra API trong Postman, sau đó ủy quyền lưu lượng truy cập qua Burp Suite tới các thư mục brute-force, can thiệp vào các tham số và làm mờ tất cả mọi thứ.

Như khi bạn thiết lập FoxyProxy, bạn sẽ cần định cấu hình Postman proxy để gửi lưu lượng truy cập đến Burp Suite bằng các bước sau (xem Hình 4-28):

1. Mở cài đặt Postman bằng cách nhấn CTRL-, (dấu phẩy) hoặc điều hướng đến File4Settings.
2. Nhấp vào tab Ủy quyền .
3. Nhấp vào hộp kiểm để thêm cấu hình proxy tùy chỉnh.
4. Đảm bảo đặt máy chủ proxy thành 127.0.0.1.
5. Đặt cổng máy chủ proxy thành 8080.
6. Chọn tab Chung và Tắt xác minh chứng chỉ SSL .

7. Trong Burp Suite, chọn tab Proxy .
8. Nhấp vào nút để Bật Chặn .



Hình 4-28: Cấu hình cài đặt proxy của Postman để tương tác với Burp Suite

Thử gửi yêu cầu bằng Postman; nếu nó bị chặn bởi Burp Suite, thì bạn đã định cấu hình mọi thứ đúng cách. Giờ đây, bạn có thể bật proxy và bật chức năng “bật chặn chặn” của Burp Suite khi bạn muốn nắm bắt các yêu cầu và phản hồi.

Công cụ bổ sung

Phần này nhằm cung cấp các tùy chọn bổ sung và hỗ trợ những người bị giới hạn bởi các tính năng có sẵn trong Burp Suite CE. Các công cụ sau đây rất xuất sắc với những gì chúng làm, mã nguồn mở và miễn phí. Cụ thể, các công cụ quét API được đề cập ở đây phục vụ một số mục đích khi bạn đang tích cực kiểm tra mục tiêu của mình. Các công cụ như Nikto và OWASP ZAP có thể giúp bạn chủ động khám phá các điểm cuối API, cấu hình sai bảo mật và các đường dẫn thú vị, đồng thời chúng cung cấp một số thử nghiệm cấp độ bề mặt của API. Nói cách khác, chúng hữu ích khi bạn bắt đầu tương tác tích cực với mục tiêu, trong khi các công cụ như Wfuzz và Arjun sẽ hữu ích hơn khi bạn đã phát hiện ra API và muốn thu hẹp trọng tâm thử nghiệm của mình. Sử dụng các công cụ này để tích cực kiểm tra API nhằm khám phá các đường dẫn, thông số, tệp và chức năng độc đáo. Mỗi công cụ này đều có trọng tâm và mục đích riêng sẽ bổ sung cho chức năng còn thiếu trong Phiên bản Cộng đồng Burp Suite miễn phí.

Thực hiện trinh sát với OWASP Amass

OWASP Amass là một công cụ thu thập thông tin nguồn mở có thể được sử dụng để do thám chủ động và thụ động. Công cụ này được tạo ra như một phần của dự án OWASP Amass, do Jeff Foley đứng đầu. Chúng tôi sẽ sử dụng Amass để khám phá bề mặt tấn công của các tổ chức mục tiêu của chúng tôi. Chỉ cần một tên miền của mục tiêu, bạn có thể sử dụng Amass để quét qua nhiều nguồn internet cho các miền và miền phụ được liên kết của mục tiêu để có được danh sách các API và URL mục tiêu tiềm năng.

Nếu OWASP Amass chưa được cài đặt, hãy sử dụng lệnh sau:

```
$ sudo apt-get cài đặt tích lũy
```

Amass khá hiệu quả mà không cần thiết lập nhiều. Tuy nhiên, bạn có thể thực hiện nó thành một cường quốc thu thập thông tin bằng cách thiết lập nó bằng các khóa API từ nhiều nguồn khác nhau. Tôi khuyên bạn ít nhất nên thiết lập tài khoản với GitHub, Twitter và Censys. Khi bạn đã thiết lập các tài khoản này, bạn có thể tạo khóa API cho các dịch vụ này và cắm chúng vào Amass bằng cách thêm chúng vào tệp cấu hình của Amass, config.ini. Kho lưu trữ Amass GitHub có tệp config.ini mẫu mà bạn có thể sử dụng tại https://github.com/OWASP/Amass/blob/master/examples/config.ini.

Trên Kali, Amass sẽ cố gắng tự động tìm tệp config.ini tại vị trí sau:

```
$ HOME/.config/amass/config.ini
```

Để tải xuống nội dung của tệp config.ini mẫu và lưu nó vào vị trí tệp cấu hình Amass mặc định, hãy chạy lệnh sau từ thiết bị đầu cuối:

```
$ mkdir $HOME/.config/amass
$ curl https://raw.githubusercontent.com/OWASP/Amass/master/examples/config.ini >$HOME/.config/
tích lũy/config.ini
```

Khi bạn đã tải xuống tệp đó, bạn có thể chỉnh sửa tệp đó và thêm API các phím bạn muốn bao gồm. Nó sẽ trông giống như thế này:

```
# https://umbrella.cisco.com (Doanh nghiệp trả phí)
# Khóa apikey phải là mã thông báo truy cập API được tạo thông qua giao diện người dùng quản lý Điều tra
#[data_sources.Umbrella]
#apikey =

#https://urlscan.io (Miễn phí)
Có thể sử dụng #URLScan mà không cần khóa API
#apikey =

# https://virustotal.com (Miễn phí)
#[data_sources.URLScan]
#apikey =
```

Như bạn có thể thấy, bạn có thể xóa nhận xét (#) và chỉ cần dán khóa API cho bất kỳ dịch vụ nào bạn muốn sử dụng. Tệp config.ini thậm chí còn cho biết khóa nào miễn phí. Bạn có thể tìm thấy danh sách các nguồn có API mà bạn có thể sử dụng để nâng cao Amass tại https://github.com/OWASP/Amass. Mặc dù sẽ hơi tốn thời gian, nhưng tôi khuyên bạn nên tận dụng ít nhất tất cả các nguồn miễn phí được liệt kê trong phần API.

Khám phá các điểm cuối API với Kiterunner

Người thả diều (https://github.com/assetnote/kiterunner) là một công cụ khám phá nội dung được thiết kế đặc biệt để tìm tài nguyên API. Kiterunner được xây dựng với Go và mặc dù nó có thể quét với tốc độ 30.000 yêu cầu mỗi giây, nhưng nó tính đến thực tế là bộ cân bằng tải và tường lửa ứng dụng web có thể sẽ thực thi giới hạn tốc độ.

Khi nói đến API, các kỹ thuật tìm kiếm của Kiterunner vượt trội so với các công cụ khám phá nội dung khác như dirbuster, dirb, Gobuster và dirsearch vì công cụ này được xây dựng với nhận thức về API. Danh sách từ, phương thức yêu cầu, tham số, tiêu đề và cấu trúc đường dẫn của nó đều tập trung vào việc tìm kiếm các điểm cuối và tài nguyên API. Đáng chú ý, công cụ này bao gồm dữ liệu từ 67.500 tệp Swagger. Kiterunner cũng đã được thiết kế để phát hiện chữ ký của các API khác nhau, bao gồm Django, Express, FastAPI, Flask, Nginx, Spring và Tomcat (chỉ nêu tên một số).

Một trong những khả năng hữu ích nhất của công cụ, mà chúng ta sẽ tận dụng trong Chương 6, là tính năng phát lại yêu cầu. Nếu Kiterunner phát hiện điểm cuối khi quét, nó sẽ hiển thị kết quả này trên dòng lệnh. Sau đó, bạn có thể đi sâu hơn vào kết quả bằng cách khám phá yêu cầu chính xác đã kích hoạt kết quả.

Để cài đặt Kiterunner, hãy chạy các lệnh sau:

```
$ git clone https://github.com/assetnote/kiterunner.git
$ cd diều hâu
$ thực hiện xây dựng
$ sudo ln -s $(pwd)/dist/kr /usr/local/bin/kr
```

Sau đó, bạn có thể sử dụng Kiterunner từ dòng lệnh bằng cách nhập thông tin sau:

```
$ kr
Kite là một trình quét web dựa trên ngữ cảnh sử dụng các đường dẫn api phổ biến để khám
phá nội dung của các đường dẫn api của ứng dụng.

Cách sử dụng:
  diều [lệnh]

Các lệnh có sẵn:
  brute brute một hoặc nhiều máy chủ với danh sách từ được cung cấp
  giúp trợ giúp về bất kỳ lệnh nào
  kb thao tác lược đồ kitebuilder
  quét quét một hoặc nhiều máy chủ với danh sách từ được cung cấp
  danh sách phiên bản nhị phân bạn đang chạy
  từ phiên bản xem danh sách từ được lưu trong bộ nhớ cache của bạn và danh sách từ từ xa
```

```
Cờ:
       --config string        tệp cấu hình (mặc định là $HOME/.kiterunner.yaml)
   -h, --help                 giúp thả diều
   -o, --output string        định dạng đầu ra. có thể là json, text, pretty (mặc định
"pretty")
   -q, --quiet chế độ im lặng. sẽ tắt tiếng văn bản đẹp không cần thiết
   -v, --verbose mức độ chi tiết của chuỗi ghi nhật ký. có thể là lỗi,
thông tin, gỡ lỗi, dấu vết ("thông tin" mặc định)

Sử dụng "kite [lệnh] --help" để biết thêm thông tin về lệnh.
```

Bạn có thể cung cấp cho Kiterunner nhiều danh sách từ khác nhau, sau đó nó sẽ sử dụng danh sách này làm trọng tải cho một loạt yêu cầu. Những yêu cầu này sẽ giúp bạn khám phá các điểm cuối API thú vị. Kiterunner cho phép bạn sử dụng tệp JSON của Swagger, tệp .kites của Assetnote và danh sách từ .txt . Hiện tại, Assetnote phát hành danh sách từ của mình, chứa các cụm từ tìm kiếm được thu thập từ các lần quét trên toàn internet, hàng tháng. Tất cả các danh sách từ được lưu trữ tại https://wordlists.assetnote.io.
Tạo một thư mục danh sách từ API như sau:

```
$ mkdir -p ~/api/danh sách từ
```

Sau đó, bạn có thể chọn danh sách từ mong muốn và tải chúng xuống thư mục /api/wordlists :

```
$ curl https://wordlists-cdn.assetnote.io/data/automated/httparchive_apiroutes_2021_06_28.txt > latest_api_wordlist.txt

  % Tổng số % Đã nhận % Xferd Tốc độ trung bình Thời gian Dload  Thời gian  Thời gian hiện tại
                                 Tải lên Tổng chi tiêu                 Tốc độ bên trái
100 6651k 100 6651k    0     0 16.1M      0 --:--:-- --:--:-- -:--:-- 16,1M
```

Bạn có thể thay thế httparchive_apiroutes_2021_06_028.txt bằng bất kỳ danh sách từ nào phù hợp với bạn nhất. Ngoài ra, hãy tải xuống tất cả danh sách từ của Assetnote cùng một lúc:

```
$ wget -r --no-parent -R "index.html*" https://wordlists-cdn.assetnote.io/data/ -nH
```

Được cảnh báo rằng việc tải xuống tất cả các danh sách từ của Assetnote sẽ chiếm khoảng 2,2 GB dung lượng, nhưng việc lưu trữ chúng chắc chắn là xứng đáng.

Quét tìm lỗ hổng với Nikto

Nikto là một trình quét lỗ hổng ứng dụng web dòng lệnh khá hiệu quả trong việc thu thập thông tin. Tôi sử dụng Nikto ngay sau khi phát hiện ra sự tồn tại của một ứng dụng web, vì nó có thể chỉ cho tôi những khía cạnh thú vị của ứng dụng. Nikto sẽ cung cấp cho bạn thông tin về máy chủ web mục tiêu, cấu hình sai bảo mật và các lỗ hổng ứng dụng web khác. Vì Nikto được bao gồm trong Kali nên nó không yêu cầu bất kỳ thiết lập đặc biệt nào.

Để quét một miền, hãy sử dụng lệnh sau:

```
$ nikto -h https://example.com
```

Để xem các tùy chọn bổ sung của Nikto, hãy nhập nikto -Help trên dòng lệnh. Một vài tùy chọn mà bạn có thể thấy hữu ích bao gồm -tên tệp đầu ra để lưu kết quả Nikto vào một tệp được chỉ định và -maxtime #ofseconds để giới hạn thời gian quét Nikto.

Kết quả từ quá trình quét Nikto sẽ bao gồm các phương thức HTTP được phép của ứng dụng, thông tin tiêu đề thú vị, điểm cuối API tiềm năng và các thư mục khác có thể đáng để kiểm tra. Để biết thêm thông tin về Nikto, hãy xem lại tài liệu có tại https://cirt.net/
tài liệu nikto2.

Quét các lỗ hổng với OWASP ZAP

OWASP đã phát triển ZAP, một trình quét ứng dụng web mã nguồn mở và đây là một công cụ kiểm tra bảo mật ứng dụng web thiết yếu khác. OWASP ZAP nên được đưa vào Kali, nhưng nếu không, bạn có thể sao chép nó từ GitHub tại https://github.com/zaproxy/zaproxy.

ZAP có hai thành phần: quét tự động và khám phá thủ công. ZAP quét tự động thực hiện thu thập dữ liệu web, phát hiện các lỗ hổng và kiểm tra các phản hồi của ứng dụng web bằng cách thay đổi các tham số yêu cầu. Quét tự động rất tốt để phát hiện các thư mục bề mặt của ứng dụng web, bao gồm khám phá các điểm cuối API. Để chạy nó, hãy nhập URL mục tiêu vào giao diện ZAP và nhấp vào nút để bắt đầu cuộc tấn công. Khi quá trình quét đã hoàn tất, bạn sẽ nhận được một danh sách các cảnh báo được phân loại theo mức độ nghiêm trọng của phát hiện. Vấn đề với tính năng quét tự động của ZAP là nó có thể bị nhầm lẫn với các thông tin xác thực sai, vì vậy điều quan trọng là phải kiểm tra và xác thực các cảnh báo. Thử nghiệm cũng được giới hạn ở bề mặt của một ứng dụng web. Trừ khi có những thư mục vô tình bị lộ, ZAP sẽ không thể xâm nhập vượt quá yêu cầu xác thực. Đây là nơi tùy chọn khám phá thủ công ZAP có ích.

Khám phá thủ công ZAP đặc biệt hữu ích để khám phá ngoài mặt của ứng dụng web. Còn được gọi là ZAP Heads Up Display (ZAP HUD), khám phá thủ công proxy lưu lượng truy cập trình duyệt web của bạn thông qua ZAP trong khi bạn duyệt. Để khởi chạy nó, hãy nhập URL để khám phá và mở trình duyệt bạn chọn. Khi trình duyệt khởi chạy, có vẻ như bạn đang duyệt trang web như bình thường; tuy nhiên, các cảnh báo và chức năng của ZAP sẽ phủ lên trang web. Điều này cho phép bạn có nhiều quyền kiểm soát hơn đối với thời điểm bắt đầu thu thập thông tin, thời điểm chạy quét tích cực và thời điểm bật "chế độ tấn công". Ví dụ: bạn có thể thực hiện quy trình tạo tài khoản người dùng và quy trình xác thực/ủy quyền với trình quét ZAP đang chạy để tự động phát hiện lỗi trong các quy trình này. Bất kỳ lỗ hổng nào bạn phát hiện sẽ bật lên như thành tích chơi trò chơi. Chúng tôi sẽ sử dụng ZAP HUD để khám phá các API.

Fuzzing với Wfuzz

Wfuzz là một khung làm mờ ứng dụng web dựa trên Python mã nguồn mở. Wfuzz nên đi kèm với phiên bản Kali mới nhất, nhưng bạn có thể cài đặt nó từ GitHub tại https://github.com/xmendez/wfuzz.

Bạn có thể sử dụng Wfuzz để thêm tải trọng trong yêu cầu HTTP bằng cách thay thế các lần xuất hiện của từ FUZZ bằng các từ trong danh sách từ; Wfuzz sau đó sẽ nhanh chóng thực hiện nhiều yêu cầu (khoảng 900 yêu cầu mỗi phút) với tải trọng được chỉ định. Vì phần lớn sự thành công của fuzzing phụ thuộc vào việc sử dụng một danh sách từ tốt, nên chúng ta sẽ dành nhiều thời gian để thảo luận về danh sách từ trong Chương 6.

Đây là định dạng yêu cầu cơ bản của Wfuzz:

```
$ tùy chọn wfuzz -z tải trọng, url tham số
```

Để chạy Wfuzz, sử dụng lệnh sau:

```
$ wfuzz -z tệp,/usr/share/wordlists/list.txt http://targetname.com/FUZZ
```

Lệnh này thay thế FUZZ trong URL http://targetname.com/FUZZ với các từ từ /usr/share/wordlists/list.txt. Tùy chọn -z chỉ định một loại tải trọng theo sau là tải trọng thực tế. Trong ví dụ này, chúng tôi đã chỉ định rằng tải trọng là một tệp và sau đó cung cấp đường dẫn tệp của danh sách từ. Chúng tôi cũng có thể sử dụng -z với danh sách hoặc phạm vi. Sử dụng tùy chọn danh sách có nghĩa là chúng tôi sẽ chỉ định tải trọng trong yêu cầu, trong khi phạm vi đề cập đến một phạm vi số. Ví dụ: bạn có thể sử dụng tùy chọn danh sách để kiểm tra điểm cuối cho danh sách các động từ HTTP:

```
$ wfuzz -X POST -z list,admin-dashboard-docs-api-test http://targetname.com/FUZZ
```

Tùy chọn -X chỉ định phương thức yêu cầu HTTP. Trong ví dụ trước, Wfuzz sẽ thực hiện yêu cầu POST với danh sách từ được sử dụng làm đường dẫn thay cho trình giữ chỗ FUZZ .

Bạn có thể sử dụng tùy chọn phạm vi để dễ dàng quét một dãy số:

```
$ wfuzz -z phạm vi,500-1000 http://targetname.com/account?user_id=FUZZ
```

Điều này sẽ tự động làm mờ tất cả các số từ 500 đến 1000. Điều này sẽ có ích khi chúng tôi kiểm tra các lỗ hổng BOLA.

Để chỉ định nhiều vị trí tấn công, bạn có thể liệt kê một số cờ -z và sau đó đánh số các phần giữ chỗ FUZZ tương ứng , chẳng hạn như FUZZ, FUZ1, FUZ2, FUZ3, v.v., như vậy:

```
$ danh sách wfuzz -z, phạm vi ABC -z, 1-3 http://tên mục tiêu.com/FUZZ/user_id=FUZZ2
```

Chạy Wfuzz với mục tiêu có thể tạo ra rất nhiều kết quả, điều này có thể làm cho nó khó khăn để tìm thấy bất cứ điều gì thú vị. Do đó, bạn nên làm quen với các tùy chọn bộ lọc Wfuzz. Các bộ lọc sau chỉ hiển thị một số kết quả nhất định:

--sc Chỉ hiển thị phản hồi với mã phản hồi HTTP cụ thể

--sl Chỉ hiển thị câu trả lời với một số dòng nhất định

--sw Chỉ hiển thị câu trả lời với một số từ nhất định

--sh Chỉ hiển thị câu trả lời với một số ký tự nhất định

Trong ví dụ sau, Wfuzz sẽ quét mục tiêu và chỉ hiển thị kết quả bao gồm mã trạng thái 200:

```
$ wfuzz -z tệp,/usr/share/wordlists/list.txt –sc 200 http://targetname.com/FUZZ
```

Các bộ lọc sau ẩn các kết quả nhất định:

--hc Ẩn phản hồi với mã trạng thái HTTP cụ thể

--hl Ẩn câu trả lời với một số dòng được chỉ định

--hw Ẩn câu trả lời với một số từ cụ thể

--hh Ẩn câu trả lời với số lượng ký tự được chỉ định

Trong ví dụ sau, Wfuzz sẽ quét mục tiêu và ẩn tất cả kết quả có mã trạng thái là 404 và ẩn kết quả có 950 ký tự:

```
$ wfuzz -z tệp,/usr/share/wordlists/list.txt –sc 404 –sh 950 http://targetname.com/FUZZ
```

Wfuzz là một công cụ fuzzing đa năng mạnh mẽ mà bạn có thể sử dụng để kiểm tra kỹ lưỡng các điểm cuối và tìm ra điểm yếu của chúng. Để biết thêm thông tin về Wfuzz, hãy xem tài liệu tại https://wfuzz.readthedocs.io/en/ muộn nhất.

Khám phá các tham số HTTP với Arjun

Arjun là một fuzzer API dựa trên Python mã nguồn mở khác được phát triển cụ thể để khám phá các tham số ứng dụng web. Chúng tôi sẽ sử dụng Arjun để khám phá chức năng API cơ bản, tìm các tham số ẩn và kiểm tra các điểm cuối API.
Bạn có thể sử dụng nó như một lần quét đầu tiên tuyệt vời cho điểm cuối API trong quá trình thử nghiệm hộp đen hoặc như một cách dễ dàng để xem các thông số được ghi lại của API khớp với kết quả quét như thế nào.

Arjun được cấu hình với một danh sách từ chứa gần 26.000 tham số và không giống như Wfuzz, nó thực hiện một số bộ lọc cho bạn bằng cách sử dụng tính năng phát hiện bất thường được cấu hình sẵn. Để thiết lập Arjun, trước tiên hãy sao chép nó từ GitHub (bạn sẽ cần có tài khoản GitHub để thực hiện việc này):

```
$ cd /opt/
$ sudo git clone https://github.com/s0med3v/Arjun.git
```

Arjun hoạt động bằng cách trước tiên thực hiện một yêu cầu tiêu chuẩn tới điểm cuối API mục tiêu. Nếu mục tiêu phản hồi bằng biểu mẫu HTML, Arjun sẽ thêm tên biểu mẫu vào danh sách tham số trong quá trình quét. Sau đó, Arjun sẽ gửi một yêu cầu với các tham số mà nó dự kiến sẽ trả về phản hồi cho các tài nguyên không tồn tại. Điều này được thực hiện để lưu ý hành vi của một yêu cầu tham số không thành công. Arjun sau đó khởi động 25 yêu cầu chứa tải trọng của gần 26.000 thông số, so sánh các phản hồi của điểm cuối API và bắt đầu quét thêm các điểm bất thường.

Để chạy Arjun, sử dụng lệnh sau:

```
$ python3 /opt/Arjun/arjun.py -u http://target_address.com
```

Nếu bạn muốn có kết quả đầu ra ở một định dạng nhất định, hãy sử dụng tùy chọn -o với loại tệp mong muốn của bạn:

```
$ python3 /opt/Arjun/arjun.py -u http://target_address.com -o arjun_results.json
```

Nếu bạn gặp một mục tiêu có giới hạn tốc độ, Arjun có thể kích hoạt giới hạn tốc độ và khiến kiểm soát bảo mật chặn bạn. Arjun thậm chí còn đưa ra các gợi ý khi mục tiêu không hợp tác. Arjun có thể nhắc bạn với một thông báo lỗi, chẳng hạn như "Target không thể xử lý các yêu cầu, hãy thử --stable switch." Nếu điều này xảy ra, chỉ cần thêm cờ --stable . Đây là một ví dụ:

```
$ python3 /opt/Arjun/arjun.py -u http://target_address.com -o arjun_results.json --stable
```

Cuối cùng, Arjun có thể quét nhiều mục tiêu cùng lúc. Sử dụng cờ -i để chỉ định nếu có danh sách các URL mục tiêu. Nếu bạn đã ủy quyền lưu lượng truy cập bằng Burp Suite, bạn có thể chọn tất cả các URL trong sơ đồ trang web, sử dụng tùy chọn Sao chép các URL đã chọn và dán danh sách đó vào một tệp văn bản. Sau đó chạy Arjun với tất cả Burp Suite tar nhận được đồng thời, như thế này:

```
$ python3 /opt/Arjun/arjun.py -i burp_targets.txt
```

Bản tóm tắt

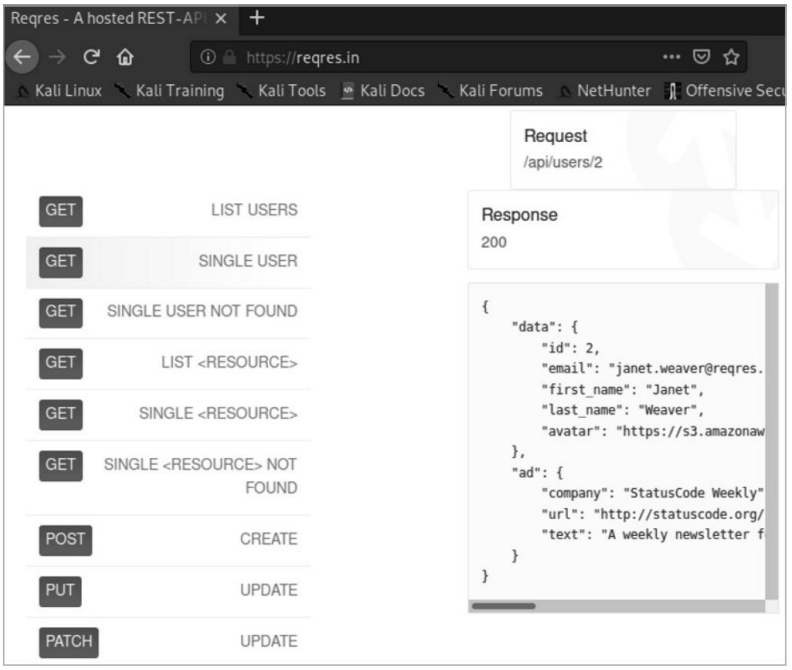
Trong chương này, bạn thiết lập các công cụ khác nhau mà chúng ta sẽ sử dụng để tấn công các API thông qua cuốn sách này. Ngoài ra, chúng tôi đã dành thời gian đào sâu vào các ứng dụng giàu tính năng như DevTools, Burp Suite và Postman. Cảm thấy thoải mái với hộp công cụ hack API sẽ giúp bạn biết khi nào nên sử dụng công cụ nào và khi nào nên xoay trục.

Lab #1: Liệt kê tài khoản người dùng trong API REST

Chào mừng đến với phòng thí nghiệm đầu tiên của bạn.

Trong phòng thí nghiệm này, mục tiêu của chúng tôi rất đơn giản: tìm tổng số tài khoản người dùng trong reqres.in, API REST được thiết kế để thử nghiệm, sử dụng các công cụ được thảo luận trong chương này. Bạn có thể dễ dàng tìm ra điều này bằng cách đoán tổng số tài khoản và sau đó kiểm tra số đó, nhưng chúng tôi sẽ tìm ra câu trả lời nhanh hơn nhiều bằng cách sử dụng sức mạnh của Postman và Burp Suite.
Khi kiểm tra các mục tiêu thực tế, bạn có thể sử dụng quy trình này để khám phá xem liệu có lỗ hổng BOLA cơ bản hay không.

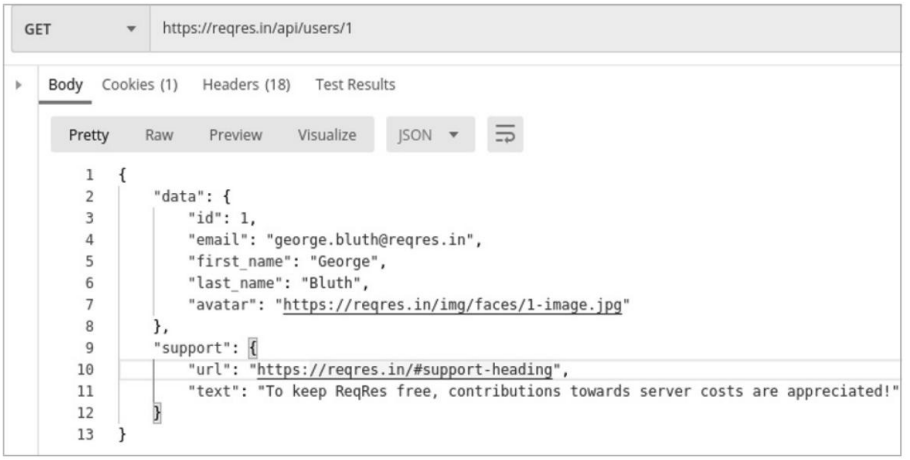
Đầu tiên, điều hướng đến https://reqres.in để xem có tài liệu API hay không. Trên trang đích, chúng tôi tìm thấy tài liệu API tương đương và có thể thấy một yêu cầu mẫu bao gồm việc đưa ra yêu cầu tới /api/users/2 điểm cuối (xem Hình 4-29).



Hình 4-29: Tài liệu API có tại https://reqres.in với hướng dẫn yêu cầu id người dùng:2

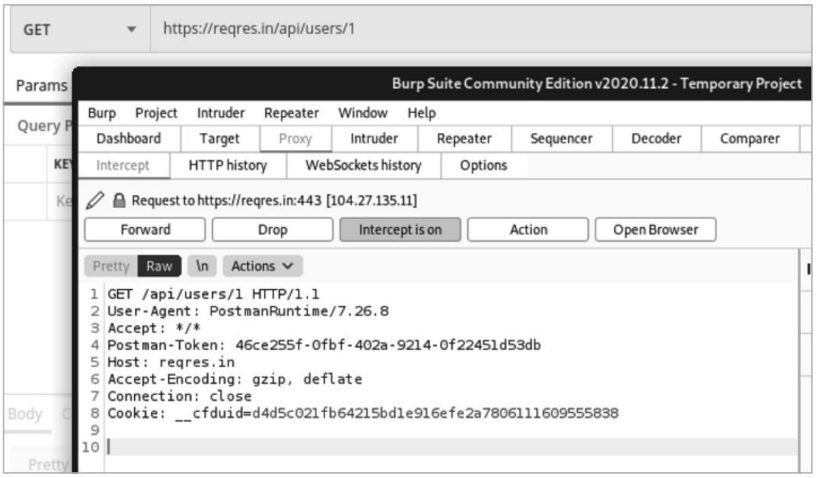
Bạn sẽ nhận thấy điểm cuối Danh sách người dùng; chúng tôi sẽ bỏ qua điều này vì mục đích của phòng thí nghiệm, vì nó sẽ không giúp bạn tìm hiểu các khái niệm dự định. Thay vào đó, chúng tôi sẽ sử dụng điểm cuối Người dùng đơn vì điểm cuối này sẽ giúp bạn xây dựng các kỹ năng cần thiết để khám phá các lỗ hổng như BOLA và BFLA. Yêu cầu API được đề xuất cho Người dùng Đơn lẻ nhằm cung cấp cho người tiêu dùng thông tin tài khoản của người dùng được yêu cầu bằng cách gửi yêu cầu GET tới /api/người dùng/. Chúng ta có thể dễ dàng giả định rằng tài khoản người dùng được tổ chức trong user thư mục theo số id của họ .

Hãy kiểm tra lý thuyết này bằng cách thử gửi yêu cầu tới người dùng có số ID khác. Vì chúng ta sẽ tương tác với một API, hãy thiết lập yêu cầu API bằng Postman. Đặt phương thức thành GET và thêm URL http://reqres.in/api/users/1. Nhấp vào Gửi và đảm bảo bạn nhận được phản hồi. Nếu bạn đã yêu cầu người dùng có ID là 1, phản hồi sẽ tiết lộ thông tin người dùng cho George Bluth, như trong Hình 4-30.



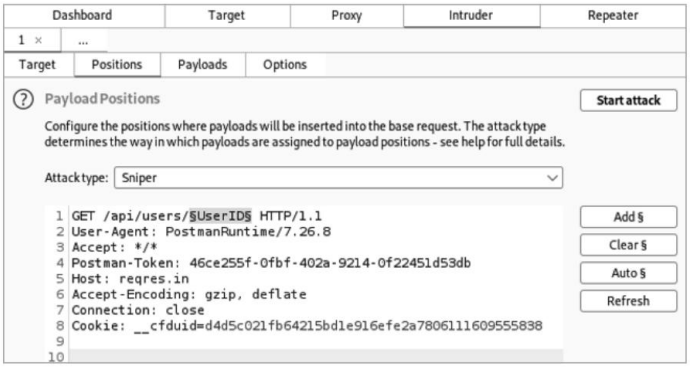
Hình 4-30: Yêu cầu API tiêu chuẩn được thực hiện bằng Postman để truy xuất người dùng 1 từ cơ sở dữ liệu https://reqres.in

Để truy xuất dữ liệu của tất cả người dùng một cách hiệu quả bằng phương pháp này, chúng ta sẽ sử dụng Burp's Intruder. Ủy quyền lưu lượng truy cập từ điểm cuối reqres.in đến Burp Suite và gửi yêu cầu tương tự trong Postman. Di chuyển sang Burp Suite, nơi bạn sẽ thấy lưu lượng truy cập bị chặn trong tab Proxy của Burp Suite (xem Hình 4-31).



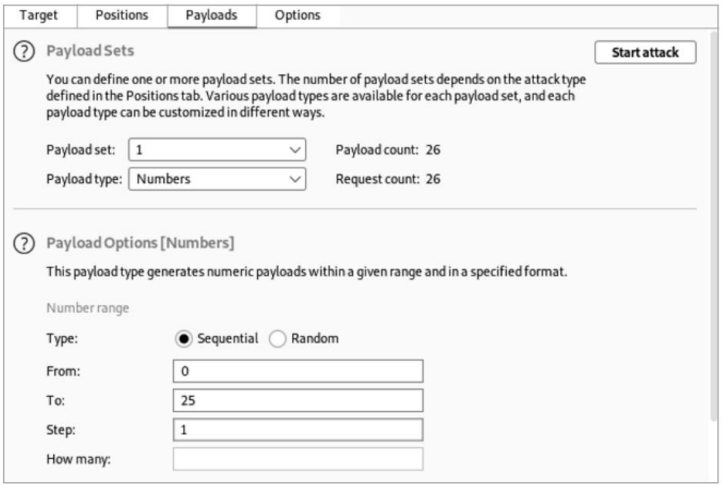
Hình 4-31: Yêu cầu bị chặn được thực hiện bằng Postman để truy xuất người dùng 1

Sử dụng phím tắt CTRL-I hoặc nhấp chuột phải vào yêu cầu bị chặn và chọn Gửi tới Kẻ xâm nhập. Chọn tab Intruder4Positions để chọn các vị trí tải trọng. Đầu tiên, chọn Xóa § để xóa định vị tải trọng tự động. Sau đó chọn số ở cuối URL và nhấp vào nút có nhãn Thêm § (xem Hình 4-32).



Hình 4-32: Kẻ xâm nhập của Burp Suite được định cấu hình với vị trí tấn công được đặt xung quanh phần UserID của đường dẫn

Khi bạn đã chọn vị trí tấn công, hãy chọn tab Payloads (xem Hình 4-33). Vì mục tiêu của chúng tôi là tìm hiểu có bao nhiêu tài khoản người dùng tồn tại, nên chúng tôi muốn thay thế ID người dùng bằng một chuỗi số. Thay đổi loại trọng tải thành Số. Cập nhật phạm vi số để kiểm tra từ 0 đến 25, bước ping bằng 1. Tùy chọn Bước cho biết Burp có bao nhiêu số sẽ tăng với mỗi tải trọng. Bằng cách chọn 1, chúng tôi cho phép Burp thực hiện công việc nặng nhọc là tạo ra tất cả các trọng tải một cách nhanh chóng. Điều này sẽ giúp chúng tôi khám phá tất cả người dùng có ID từ 0 đến 25. Với các cài đặt này, Burp sẽ gửi tổng cộng 26 yêu cầu, mỗi yêu cầu có một số từ 0 đến 25.



Hình 4-33: Tab Tải trọng của kẻ xâm nhập với loại trọng tải được đặt thành số

Cuối cùng, nhấp vào Bắt đầu tấn công để gửi 26 yêu cầu tới reqres.in. Phân tích kết quả sẽ cung cấp cho bạn một dấu hiệu rõ ràng về tất cả người dùng trực tiếp. Nhà cung cấp API phản hồi với trạng thái 200 cho tài khoản người dùng từ 1 đến 12 và trạng thái 404 cho các yêu cầu tiếp theo. Đánh giá theo kết quả, chúng tôi có thể kết luận rằng API này có tổng cộng 12 tài khoản người dùng hợp lệ.

Tất nhiên, đây chỉ là thực hành. Các giá trị bạn thay thế trong tương tác hack API trong tương lai có thể là số ID người dùng, nhưng chúng cũng có thể dễ dàng là số tài khoản ngân hàng, số điện thoại, tên công ty hoặc địa chỉ email. Phòng thí nghiệm này đã chuẩn bị cho bạn tiếp nhận thế giới các lỗ hổng BOLA cơ bản; chúng ta sẽ mở rộng kiến thức này trong Chương 10.

Để thực hiện thêm, hãy thử thực hiện cùng thao tác quét này bằng Wfuzz.

5

THIẾT LẬP MỤC TIÊU API

Trong chương này, bạn sẽ xây dựng phòng thí nghiệm mục tiêu API của riêng mình để tấn công trong các chương tiếp theo. Bằng cách nhắm mục tiêu vào một hệ thống mà bạn kiểm soát, bạn sẽ có thể thực hành các kỹ thuật của mình một cách an toàn và xem tác động của chúng từ cả góc độ tấn công và phòng thủ. Bạn cũng sẽ có thể phạm sai lầm và thử nghiệm các khai thác mà bạn có thể chưa cảm thấy thoải mái khi sử dụng trong các cam kết thực tế.

Bạn sẽ nhắm mục tiêu các máy này trong suốt các phần phòng thí nghiệm trong tài liệu này. cuốn sách để tìm hiểu cách thức hoạt động của các công cụ, khám phá các điểm yếu của API, tìm hiểu cách làm mờ đầu vào và khai thác tất cả các phát hiện của bạn. Phòng thí nghiệm sẽ có những lỗ hổng vượt xa những gì được đề cập trong cuốn sách này, vì vậy tôi khuyến khích bạn tìm kiếm chúng và phát triển các kỹ năng mới thông qua thử nghiệm.

Chương này hướng dẫn bạn thiết lập các điều kiện tiên quyết trong máy chủ Linux, cài đặt Docker, tải xuống và khởi chạy ba hệ thống dễ bị tấn công sẽ được sử dụng làm mục tiêu của chúng tôi và tìm tài nguyên bổ sung cho các mục tiêu hack API.

LƯU Ý ng thí nghiệm này chứa các hệ thống dễ bị tổn thương có chủ ý. Những thứ này có thể thu hút những kẻ tấn công và gây ra những rủi ro mới cho mạng gia đình hoặc cơ quan của bạn. Không kết nối các máy này với phần còn lại của mạng của bạn; đảm bảo phòng thí nghiệm hack được cách ly và bảo vệ. Nói chung, hãy lưu ý nơi bạn lưu trữ một mạng lưới các máy dễ bị tấn công.

Tạo máy chủ Linux

Bạn sẽ cần một hệ thống máy chủ để có thể chạy ba ứng dụng dễ bị tấn công. Để đơn giản, tôi khuyên bạn nên giữ các ứng dụng dễ bị tấn công trên các hệ thống máy chủ khác nhau. Khi chúng được lưu trữ cùng nhau, bạn có thể gặp xung đột về tài nguyên mà ứng dụng sử dụng và một cuộc tấn công vào một ứng dụng web dễ bị tấn công có thể ảnh hưởng đến các ứng dụng khác. Sẽ dễ dàng hơn nếu có thể có từng ứng dụng dễ bị tấn công trên hệ thống máy chủ của chính nó.

Tôi khuyên bạn nên sử dụng hình ảnh Ubuntu gần đây được lưu trữ trên siêu visor (chẳng hạn như VMware, Hyper-V hoặc VirtualBox) hoặc trên đám mây (chẳng hạn như AWS, Azure hoặc Google Cloud). Những kiến thức cơ bản về thiết lập hệ thống máy chủ và kết nối chúng với nhau nằm ngoài phạm vi của cuốn sách này và được đề cập rộng rãi ở những nơi khác. Bạn có thể tìm thấy nhiều hướng dẫn miễn phí tuyệt vời để thiết lập những điều cơ bản của phòng thí nghiệm hack tại nhà hoặc đám mây. Dưới đây là một số tôi khuyên bạn nên:

Cybrary, “Hướng dẫn: Thiết lập Phòng thí nghiệm Pentesting ảo tại nhà,” https://www.cybrary.it/blog/0p3n/tutorial-for-setting-up-a-virtual-penetration-kiểm tra-phòng thí nghiệm-tại-nhà-của-bạn

Bảo mật thông tin Black Hills, “Webcast: Cách xây dựng phòng thí nghiệm tại nhà,” https://www.blackhillsinfosec.com/webcast-how-to-build-a-home-lab

Null Byte, “Cách tạo phòng thí nghiệm hack ảo,” https://null-byte.wonderhowto.com/how-to/hack-like-pro-create-virtual-hacking-lab-0157333

Các bài báo về hack, “Thiết lập phòng thí nghiệm Pentest ứng dụng web trên AWS,” https://www.hackingarticles.in/web-application-pentest-lab-setup-on-aws

Sử dụng các hướng dẫn này để thiết lập máy Ubuntu của bạn.

Cài đặt Docker và Docker Compose

Khi bạn đã định cấu hình hệ điều hành máy chủ của mình, bạn có thể sử dụng Docker để lưu trữ các ứng dụng dễ bị tấn công ở dạng vùng chứa. Docker và Docker Compose sẽ giúp việc tải xuống các ứng dụng dễ bị tổn thương và khởi chạy chúng trong vòng vài phút trở nên vô cùng dễ dàng.

Làm theo hướng dẫn chính thức tại https://docs.docker.com/engine/install/Ubuntu để cài đặt Docker trên máy chủ Linux của bạn. Bạn sẽ biết rằng Docker Engine đã được cài đặt đúng cách khi bạn có thể chạy hình ảnh hello-world:

```
$ sudo docker chạy hello-world
```

Nếu bạn có thể chạy bộ chứa hello-world, bạn đã thiết lập Docker thành công. Chúc mừng! Nếu không, bạn có thể khắc phục sự cố bằng hướng dẫn chính thức của Docker.

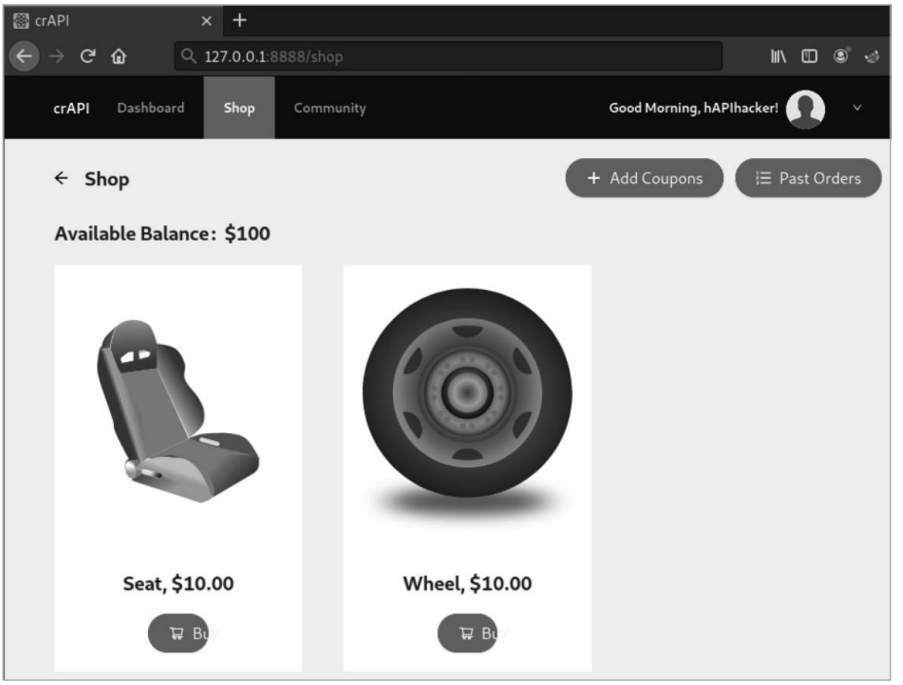
Docker Compose là một công cụ cho phép bạn chạy nhiều vùng chứa từ một tệp YAML. Tùy thuộc vào thiết lập phòng thí nghiệm hack của bạn, Docker Compose có thể cho phép bạn khởi chạy các hệ thống dễ bị tấn công bằng lệnh đơn giản docker-compose up. Bạn có thể tìm thấy tài liệu chính thức để cài đặt Docker Compose tại https://docs.docker.com/compose/install.

Cài đặt ứng dụng dễ bị tấn công

Tôi đã chọn những ứng dụng dễ bị tổn thương này để chạy trong phòng thí nghiệm: OWASP crAPI, OWASP Juice Shop, OWASP DevSlop's Pixi và Dam Vulnerable GraphQL. Các ứng dụng này sẽ giúp bạn phát triển các kỹ năng hack API thiết yếu như khám phá API, fuzzing, định cấu hình tham số, kiểm tra xác thực, khám phá 10 lỗ hổng Bảo mật API hàng đầu của OWASP và tấn công các lỗ hổng đã phát hiện. Phần này mô tả cách thiết lập các ứng dụng này.

API hoàn toàn lố bịch (crAPI)

API hoàn toàn lố bịch, được hiển thị trong Hình 5-1, là API dễ bị tổn thương do Dự án bảo mật API OWASP phát triển và phát hành. Như đã lưu ý trong lời cảm ơn của cuốn sách này, dự án này do Inon Shkedy, Erez Yalon và Paolo Silva phụ trách. API dễ bị tổn thương crAPI được thiết kế để chứng minh các lỗ hổng API nghiêm trọng nhất. Chúng tôi sẽ tập trung vào hack crAPI trong hầu hết các phòng thí nghiệm của chúng tôi.

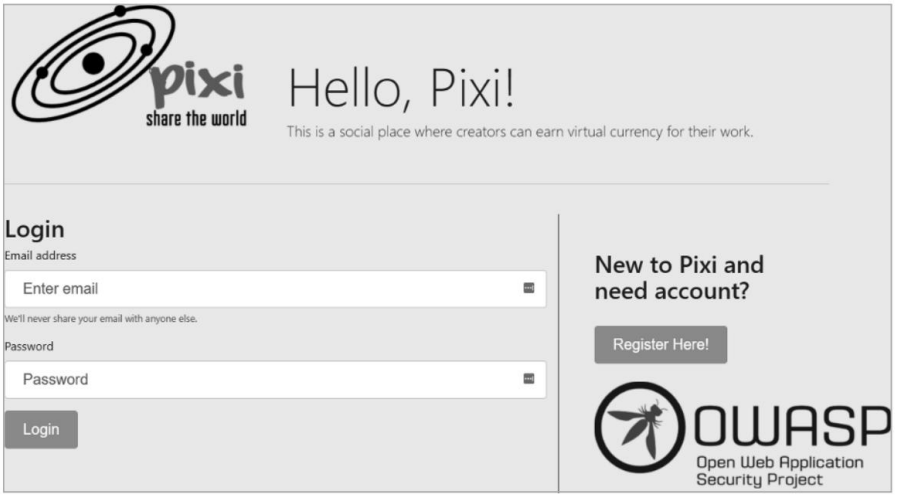


Hình 5-1: Cửa hàng crAPI

Ứng dụng crAPI chứa ứng dụng web hiện đại, API và máy chủ email Mail Hog. Trong ứng dụng này, bạn có thể mua các bộ phận của xe, sử dụng tính năng trò chuyện cộng đồng và liên kết xe để tìm các cửa hàng sửa chữa tại địa phương.
Ứng dụng crAPI được xây dựng với các triển khai thực tế của 10 lỗ hổng bảo mật API hàng đầu của OWASP. Bạn sẽ học được khá nhiều từ cái này.

Pixi của OWASP DevSlop

Pixi là một ứng dụng web ngăn xếp MongoDB, Express.js, Angular, Node (MEAN) được thiết kế với các API có lỗ hổng cố ý (xem Hình 5-2).
Nó được tạo ra tại OWASP DevSlop, một dự án vườn ươm OWASP làm sáng tỏ những sai lầm liên quan đến DevOps, bởi Nicole Becher, Nancy Gariché, Mordecai Kraushar và Tanya Janca.



Hình 5-2: Trang đích Pixi

Bạn có thể coi ứng dụng Pixi như một nền tảng truyền thông xã hội với hệ thống thanh toán ảo. Là một kẻ tấn công, bạn sẽ thấy thông tin người dùng, chức năng quản trị và hệ thống thanh toán của Pixi đặc biệt thú vị.

Một tính năng tuyệt vời khác của Pixi là rất dễ khởi động và chạy ninh. Chạy các lệnh sau từ thiết bị đầu cuối Ubuntu:

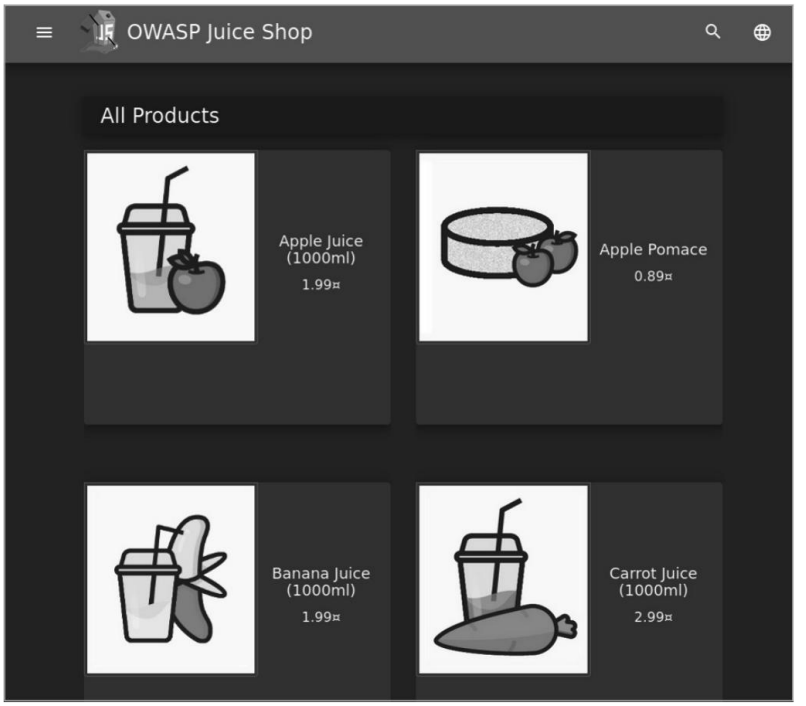
```
$ git bản sao https://github.com/DevSlop/Pixi.git
$ cd Pixi
$ sudo docker-compose up
```

Sau đó sử dụng trình duyệt và truy cập http://localhost:8000 để xem trang đích. Nếu Docker và Docker Compose đã được thiết lập, như đã mô tả trước đây trong chương này, thì việc khởi chạy Pixi thực sự sẽ dễ dàng như vậy.

Cửa hàng nước trái cây OWASP

Cửa hàng nước trái cây OWASP, được hiển thị trong Hình 5-3, là một dự án hàng đầu của OWASP do Björn Kimminich tạo ra. Nó được thiết kế để bao gồm các lỗ hổng từ cả hai

Top 10 của OWASP và Top 10 của Bảo mật API OWASP. Một tính năng tuyệt vời có trong Juice Shop là nó theo dõi tiến trình hack của bạn và bao gồm một bảng điểm ẩn. Juice Shop được xây dựng bằng Node.js, Express và Angular. Nó là một ứng dụng JavaScript được cung cấp bởi API REST.



Hình 5-3: Cửa hàng nước trái cây OWASP

Trong số tất cả các ứng dụng chúng tôi sẽ cài đặt, Juice Shop hiện là ứng dụng được hỗ trợ nhiều nhất, với hơn 70 người đóng góp. Để tải xuống và khởi chạy Juice Shop, hãy chạy các lệnh sau:

```
$ docker pull bkimminich/cửa hàng nước trái cây
$ docker run --rm -p 80:3000 bkimminich/juice-shop
```

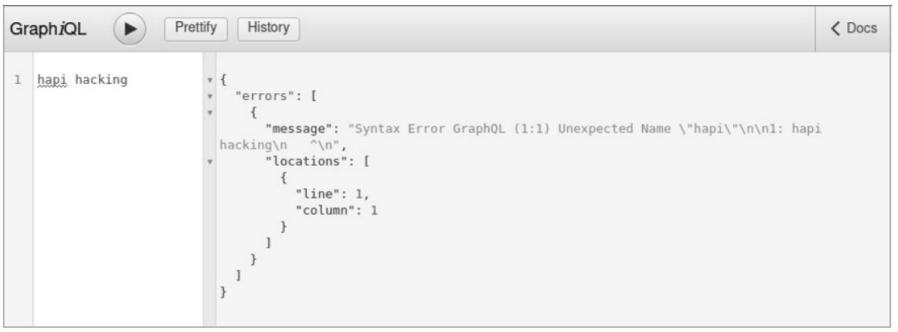
Cửa hàng nước trái cây và Ứng dụng GraphQL dễ bị tổn thương chết tiệt (DVGA) đều chạy trên cổng 3000 theo mặc định. Để tránh xung đột, đối số -p 80:3000 trong lệnh docker-run sẽ đặt Juice Shop chạy qua cổng 80 thay thế.

Để truy cập Juice Shop, duyệt đến http://localhost. (Trên macOS và Windows, duyệt đến http://192.168.99.100 nếu bạn đang sử dụng Docker Machine thay vì cài đặt Docker gốc.)

Ứng dụng GraphQL dễ bị tổn thương chết tiệt

DVGA là một ứng dụng GraphQL có lỗ hổng cố ý được phát triển bởi Dolev Farhi và Connor McKinnon. Tôi đưa DVGA vào phòng thí nghiệm này vì GraphQL ngày càng phổ biến và được các tổ chức như Facebook, Netflix, AWS và IBM áp dụng. Ngoài ra, bạn có thể ngạc nhiên

theo tần suất môi trường phát triển tích hợp GraphQL (IDE) được hiển thị cho tất cả mọi người sử dụng. GraphiQL là một trong những IDE GraphQL phổ biến hơn mà bạn sẽ bắt gặp. Hiểu cách tận dụng GraphiQL IDE sẽ chuẩn bị cho bạn tương tác với các API GraphQL khác có hoặc không có giao diện người dùng thân thiện (xem Hình 5-4).



Hình 5-4: Trang web GraphiQL IDE được lưu trữ trên cổng 5000

Để tải xuống và khởi chạy DVGA, hãy chạy các lệnh sau từ thiết bị đầu cuối máy chủ Ubuntu của bạn:

```
$ sudo docker pull dolevf/dvga
$ sudo docker run -t -p 5000:5000 -e WEB_HOST=0.0.0.0 dolevf/dvga
```

Để truy cập nó, hãy sử dụng trình duyệt và truy cập http://localhost:5000.

Thêm các ứng dụng dễ bị tổn thương khác

Nếu muốn thử thách thêm, bạn có thể thêm các máy khác vào phòng thí nghiệm hack API của mình. GitHub là một nguồn tuyệt vời chứa các API dễ bị tấn công có chủ ý để củng cố phòng thí nghiệm của bạn. Bảng 5-1 liệt kê thêm một số hệ thống có API dễ bị tấn công mà bạn có thể dễ dàng sao chép từ GitHub.

Bảng 5-1: Các hệ thống bổ sung có API dễ bị tổn thương

Tên	người đóng góp	URL GitHub
VAmPI	Erev0s	https://github.com/erev0s/VAmPI
DVWS-nút	Snoopysecurity	https://github.com/snoopysecurity/dvws-node
chết tiệt dễ bị tổn thương dịch vụ vi mô	ne0z	https://github.com/ne0z/ Dịch vụ vi mô dễ bị tổn thương chết tiệt
Nút-API-dê	Layro01	https://github.com/layro01/node-api-goat
Dễ bị tổn thương API đồ thị	Aidan Noll	https://github.com/CarveSystems/vulnerable-graphql-api
Generic-University	InsiderPhD	https://github.com/InsiderPhD/Generic-University
vulnapi	tkisason	https://github.com/tkisason/vulnapi

Hack API trên TryHackMe và HackTheBox

TryHackMe (https://tryhackme.com) và HackTheBox (https://www.hackthebox
.com) là các nền tảng web cho phép bạn hack các máy dễ bị tấn công, tham gia các cuộc thi cướp cờ (CTF), giải quyết các thử thách hack và leo lên bảng xếp hạng hack. TryHackMe có một số nội dung miễn phí và nhiều nội dung khác với phí đăng ký hàng tháng. Bạn có thể triển khai các máy hack dựng sẵn của nó trên trình duyệt web và tấn công chúng. Nó bao gồm một số cỗ máy tuyệt vời có API dễ bị tấn công:

•Hiệu sách (miễn phí)

•Carpe Diem 1 (miễn phí)

•ZTH: Lỗ hổng web tối nghĩa (trả phí)

•ZTH: Web2 (trả phí)

•GraphQL (trả phí)

Các máy TryHackMe dễ bị tấn công này bao gồm nhiều cách tiếp cận cơ bản để hack API REST, API GraphQL và các cơ chế xác thực API phổ biến. Nếu bạn chưa quen với việc hack, TryHackMe đã giúp việc triển khai một cỗ máy tấn công trở nên đơn giản bằng cách nhấp vào Bắt đầu hộp tấn công. Trong vòng vài phút, bạn sẽ có một cỗ máy tấn công dựa trên trình duyệt với nhiều công cụ mà chúng ta sẽ sử dụng xuyên suốt cuốn sách này.

HackTheBox (HTB) cũng có nội dung miễn phí và mô hình đăng ký nhưng giả sử bạn đã có kỹ năng hack cơ bản. Ví dụ: HTB hiện không cung cấp cho người dùng các phiên bản máy tấn công, vì vậy HTB yêu cầu bạn phải chuẩn bị sẵn máy tấn công của riêng mình. Để hoàn toàn sử dụng HTB, bạn cần có khả năng chấp nhận thử thách và hack quy trình mã lời mời của nó để giành được quyền tham gia.

Sự khác biệt chính giữa bậc miễn phí HTB và bậc trả phí của nó là quyền truy cập vào các máy dễ bị tấn công. Với quyền truy cập miễn phí, bạn sẽ có quyền truy cập vào 20 máy dễ bị tổn thương gần đây nhất, có thể bao gồm hệ thống liên quan đến API. Tuy nhiên, nếu bạn muốn truy cập vào thư viện các máy dễ bị tổn thương có lỗ hổng API của HTB, bạn sẽ phải trả phí để trở thành thành viên VIP cho phép bạn truy cập vào các máy đã ngừng hoạt động của HTB.

Tất cả các máy đã ngừng hoạt động được liệt kê trong Bảng 5-2 đều bao gồm các khía cạnh của hack API.

Bảng 5-2: Các máy đã ngừng hoạt động với các thành phần tấn công API

thủ công	Người phát thơ	máy đập 2
JSON	Nút	Giúp đỡ
người chơi hai	Lu-ca	Chơi với vớ bẩn

HTB cung cấp một trong những cách tốt nhất để cải thiện kỹ năng hack của bạn và mở rộng trải nghiệm phòng thí nghiệm hack của bạn ngoài tường lửa của riêng bạn. Ngoài các máy HTB, các thử thách như Fuzzy có thể giúp bạn cải thiện các kỹ năng hack API quan trọng.

Các nền tảng web như TryHackMe và HackTheBox là những phần bổ sung tuyệt vời cho phòng thí nghiệm hack của bạn và sẽ giúp nâng cao khả năng hack API của bạn.
Khi bạn không tham gia hack trong thế giới thực, bạn nên giữ kỹ năng của mình sắc bén với các cuộc thi CTF như thế này.

Bản tóm tắt

Trong chương này, tôi đã hướng dẫn bạn cách thiết lập bộ ứng dụng dễ bị tổn thương của riêng bạn mà bạn có thể lưu trữ trong phòng thí nghiệm tại nhà. Khi bạn học các kỹ năng mới, các ứng dụng trong phòng thí nghiệm này sẽ đóng vai trò là nơi để thực hành tìm và khai thác các lỗ hổng API. Với những ứng dụng dễ bị tổn thương này đang chạy trong phòng thí nghiệm tại nhà của bạn, bạn sẽ có thể làm theo các công cụ và kỹ thuật được sử dụng trong các chương và bài tập trong phòng thí nghiệm sau đây. Tôi khuyến khích bạn vượt xa các đề xuất của tôi và tự mình tìm hiểu những điều mới bằng cách mở rộng hoặc khám phá ngoài phòng thí nghiệm hack API này.

Lab #2: Tìm các API dễ bị tổn thương của bạn

Hãy để ngón tay của bạn trên bàn phím. Trong phòng thí nghiệm này, chúng ta sẽ sử dụng một số công cụ cơ bản của Kali để khám phá và tương tác với các API dễ bị tấn công mà bạn vừa thiết lập. Chúng tôi sẽ tìm kiếm ứng dụng phòng thí nghiệm Juice Shop trên mạng cục bộ của mình bằng cách sử dụng Netdetect, Nmap, Nikto và Burp Suite.

LƯU Ý ng thí nghiệm này giả định rằng bạn đã lưu trữ các ứng dụng dễ bị tổn thương trên mạng cục bộ của mình hoặc trên một trình ảo hóa. Nếu bạn đã thiết lập phòng thí nghiệm này trên đám mây, bạn sẽ không cần khám phá địa chỉ IP của hệ thống máy chủ vì bạn sẽ có thông tin đó.

Trước khi khởi động phòng thí nghiệm của bạn, tôi khuyên bạn nên tìm hiểu xem có thể tìm thấy những thiết bị nào trên mạng của bạn. Sử dụng Netdetect trước khi khởi động phòng thí nghiệm dễ bị tấn công và sau khi bạn khởi động phòng thí nghiệm:

```
$ Sudo netDiscover
Đang quét: 172.16.129.0/16 | Chế độ xem màn hình: Máy chủ duy nhất

 13 gói ARP Req/Rep đã chụp, từ 4 máy chủ. Tổng kích thước: 780

                Tại địa chỉ MAC         Đếm         Len MAC Nhà cung cấp / Tên máy chủ

 192.168.195.2 00:50:56:f0:23:20              6       360 VMware, Inc.
 192.168.195.130 00:0c:29:74:7c:5d            4       240 VMware, Inc.
 192.168.195.132 00:0c:29:85:40:c0            2       120 VMware, Inc.
 192.168.195.254 00:50 : 56 :ed:c0:7c         1        60 VMware, Inc.
```

Bạn sẽ thấy một địa chỉ IP mới xuất hiện trên mạng. Khi bạn đã phát hiện ra IP phòng thí nghiệm dễ bị tấn công, bạn có thể sử dụng CTRL-C để dừng Netdetect.

Bây giờ bạn đã có địa chỉ IP của máy chủ dễ bị tấn công, hãy tìm hiểu những gì các dịch vụ và cổng đang được sử dụng trên thiết bị ảo đó bằng lệnh Nmap đơn giản:

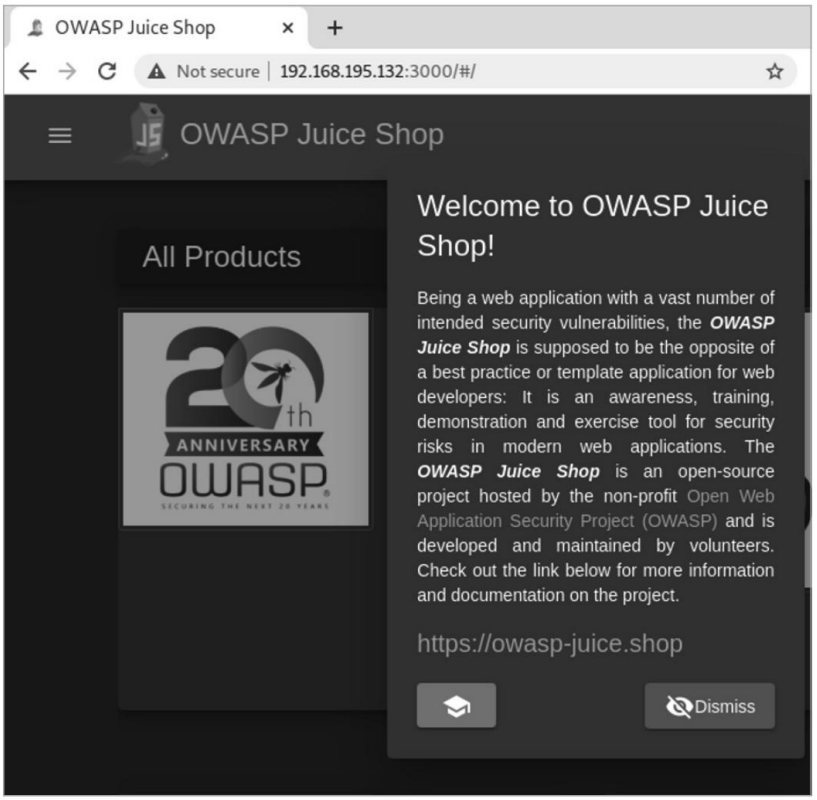
```
$ bản đồ 192.168.195.132
Báo cáo quét Nmap cho 192.168.195.132
Máy chủ đang hoạt động (độ trễ 0,00046 giây).
Không hiển thị: 999 cổng đã đóng
DỊCH VỤ NHÀ NƯỚC CẢNG
3000/tcp mở ppp

Nmap đã hoàn thành: 1 địa chỉ IP (1 máy chủ lưu trữ) được quét trong 0,14 giây
```

Chúng tôi có thể thấy rằng địa chỉ IP được nhắm mục tiêu chỉ có cổng 3000 mở (phù hợp với những gì chúng tôi mong đợi dựa trên thiết lập ban đầu của chúng tôi về Juice Shop). Để tìm hiểu thêm thông tin về mục tiêu, chúng ta có thể thêm các cờ -sC và -sV vào quá trình quét của mình để chạy các tập lệnh Nmap mặc định và để thực hiện liệt kê dịch vụ:

```
$ nmap -sC -sV 192.168.195.132
Báo cáo quét Nmap cho 192.168.195.132
Máy chủ đang hoạt động (độ trễ 0,00047 giây).
Không hiển thị: 999 cổng đã đóng
HẢI CẢNG PHIÊN BẢN DỊCH VỤ NHÀ NƯỚC
3000/tcp mở ppp?
| chuỗi dấu vân tay:
| DNSStatusRequestTCP, DNSVersionBindReqTCP, Trợ giúp, NCP, RPCCheck, RTSPRequest:
HTTP/1.1 400 Yêu cầu không hợp lệ
Kết nối: đóng
| | | Nhận yêu cầu:
HTTP/1.1 200 OK
--snip--
Bản quyền (c) Bjoern Kimminich.
SPDX-License-Identifier: MIT
<!doctypehtml>
<html lang="vi">
<đầu>
<meta charset="utf-8">
<title>Cửa hàng nước trái cây OWASP </title>
```

Bằng cách chạy lệnh này, chúng tôi biết rằng HTTP đang chạy trên cổng 3000. Chúng tôi đã tìm thấy một ứng dụng web có tiêu đề “Cửa hàng nước trái cây OWASP”. Bây giờ chúng ta có thể sử dụng trình duyệt web để truy cập Juice Shop bằng cách điều hướng đến URL (xem Hình 5-5). Trong trường hợp của tôi, URL là http://192.168.195.132:3000.



Hình 5-5: Cửa hàng nước trái cây OWASP

Tại thời điểm này, bạn có thể khám phá ứng dụng web bằng trình duyệt web của mình, xem các tính năng khác nhau của ứng dụng và tìm các loại nước trái cây hảo hạng của Cửa hàng nước trái cây. Nói chung, hãy nhấp vào mọi thứ và chú ý đến các URL mà những lần nhấp này tạo ra để biết các dấu hiệu của API đang hoạt động. Bước đầu tiên điển hình sau khi khám phá ứng dụng web là kiểm tra lỗ hổng bảo mật. Sử dụng lệnh Nikto sau để quét ứng dụng web trong phòng thí nghiệm của bạn:

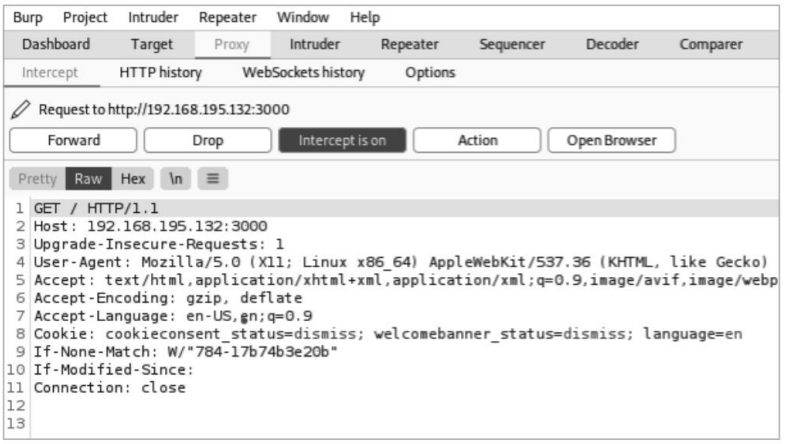
```
$ nikto -h http://192.168.195.132:3000

+ IP mục tiêu:          192.168.195.132
+ Tên máy chủ mục tiêu: 192.168.195.132
+ Cảng mục tiêu:        3000

+ Máy chủ: Không lấy được banner
+ Đã lấy tiêu đề access-control-allow-origin: *
+ Tiêu đề X-XSS-Protection không được xác định. Tiêu đề này có thể gợi ý cho tác nhân người dùng để bảo vệ chống lại một số dạng XSS
+ Tìm thấy tiêu đề 'feature-policy' không phổ biến, với nội dung: thanh toán 'self'
+ Không tìm thấy Thư mục CGI nào (sử dụng '-C all' để buộc kiểm tra tất cả các thư mục có thể có)
+ Mục '/ftp/' trong robots.txt trả về mã HTTP chuyển hướng hoặc không bị cấm (200)
+ "robots.txt" chứa 1 mục cần xem thủ công.
```

Nikto làm nổi bật một số thông tin thú vị, chẳng hạn như tệp robots.txt và một mục nhập hợp lệ cho FTP. Tuy nhiên, không có gì ở đây tiết lộ rằng một API đang hoạt động.

Vì chúng tôi biết rằng API hoạt động ngoài GUI, nên bắt đầu nắm bắt lưu lượng truy cập web bằng cách ủy quyền lưu lượng truy cập của chúng tôi thông qua Burp Suite. Đảm bảo đặt FoxyProxy thành mục Burp Suite của bạn và xác nhận rằng Burp Suite đã bật tùy chọn Chặn (xem Hình 5-6). Tiếp theo, làm mới trang web Juice Shop.



Hình 5-6: Yêu cầu HTTP của Juice Shop bị chặn

Khi bạn đã chặn một yêu cầu với Burp Suite, bạn sẽ thấy một cái gì đó tương tự như trong Hình 5-6. Tuy nhiên, vẫn không có API!
Tiếp theo, từ từ nhấp vào Chuyển tiếp để gửi hết yêu cầu được tạo tự động này đến yêu cầu khác đến ứng dụng web và để ý xem GUI của trình duyệt web được xây dựng từ từ như thế nào.

Sau khi bắt đầu chuyển tiếp yêu cầu, bạn sẽ thấy các điểm cuối API chỉ báo sau:

```
NHẬN /rest/admin/application-configuration
NHẬN /api/Challenges/?name=Score%20Board
NHẬN /api/Số lượng/
```

Đẹp! Phòng thí nghiệm ngắn này đã chứng minh cách bạn có thể tìm kiếm một lỗ hổng máy trong môi trường mạng cục bộ của bạn. Chúng tôi đã thực hiện một số cách sử dụng cơ bản các công cụ mà chúng tôi đã thiết lập trong Chương 4 để giúp chúng tôi tìm thấy một trong những ứng dụng dễ bị tổn thương và nắm bắt một số yêu cầu API có vẻ thú vị được gửi ngoài những gì chúng tôi thường thấy trong GUI của trình duyệt web.