

# Cybersecurity

Brute Force Online Lab



# Brute Force Online Materials

- Materials needed
  - Kali Linux Virtual Machine
- Software Tool used
  - OWASP ZAP
    - Tool pre-installed on Kali Linux
  - DVWA
    - Installed on the Kali Machine



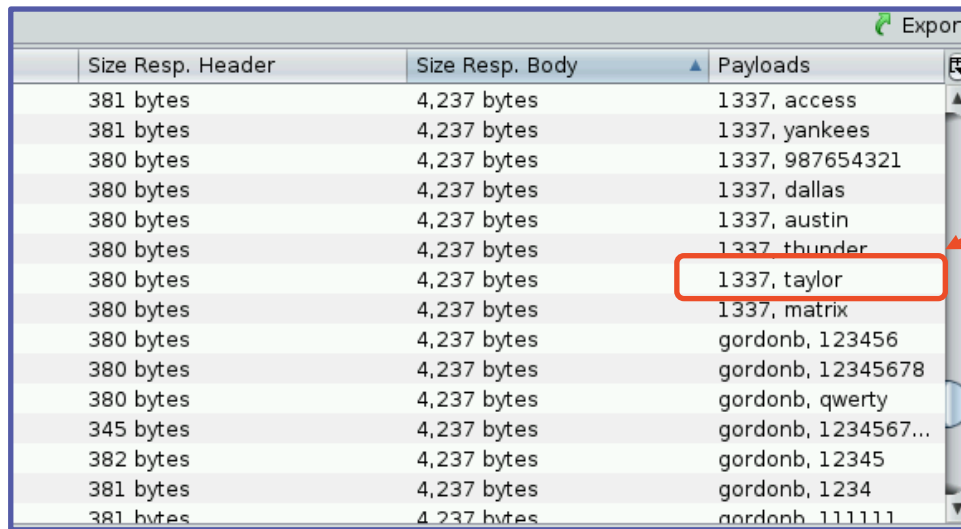
# Objectives Covered

- Security+ Objectives (SY0-701)
  - Objective 2.4 - Given a scenario, analyze indicators of malicious activity.
    - Physical attacks
      - Brute force



# What is a Brute Force Attack?

- A brute force attack is a form of password attack where the attack attempts to guess a password by trying many passwords in the attempt to guess the correct password



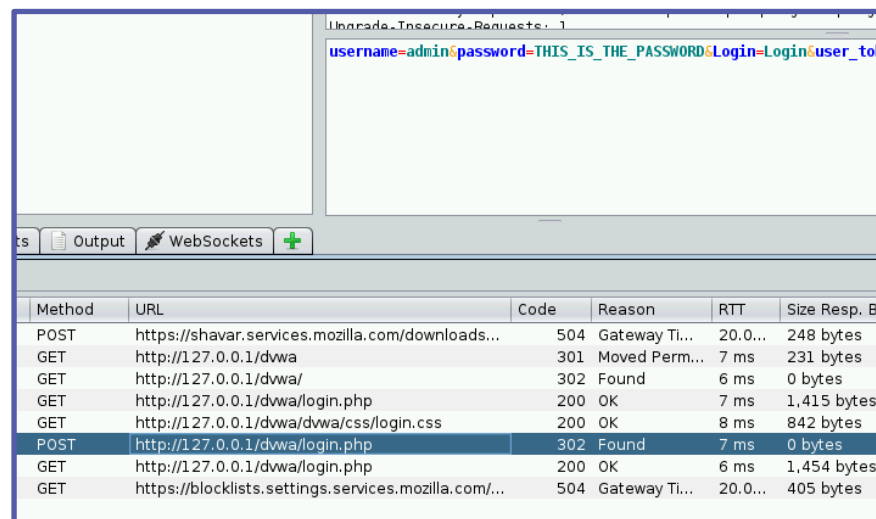
Size Resp. Header	Size Resp. Body	Payloads
381 bytes	4,237 bytes	1337, access
381 bytes	4,237 bytes	1337, yankees
380 bytes	4,237 bytes	1337, 987654321
380 bytes	4,237 bytes	1337, dallas
380 bytes	4,237 bytes	1337, austin
380 bytes	4,237 bytes	1337, thunder
380 bytes	4,237 bytes	1337, taylor
380 bytes	4,237 bytes	1337, matrix
380 bytes	4,237 bytes	gordonb, 123456
380 bytes	4,237 bytes	gordonb, 12345678
380 bytes	4,237 bytes	gordonb, qwerty
345 bytes	4,237 bytes	gordonb, 1234567...
382 bytes	4,237 bytes	gordonb, 12345
381 bytes	4,237 bytes	gordonb, 1234
381 bytes	4,237 bytes	gordonb, 111111

Notice all the usernames/passwords being used in hopes of finding the right password for the system

**Please Note: The attack in this lab uses a dictionary attack to help perform the brute force attack**

# Brute Force Online Lab Overview

1. Set up Environment
2. Download Password List
3. Start DVWA Servers
4. Open OWASP ZAP
5. Launch the Web Browser
6. Enter False Credentials
7. Get the GET Request
8. Brute Force the Password
9. Log into DVWA



```
Unbrute-Insecure_Requests: 1  
username=admin&password=THIS_IS_THE_PASSWORD&Login=Login&user_tok
```

Method	URL	Code	Reason	RTT	Size Resp. B
POST	https://shavar.services.mozilla.com/downloads...	504	Gateway Ti...	20.0...	248 bytes
GET	http://127.0.0.1/dwa	301	Moved Perm...	7 ms	231 bytes
GET	http://127.0.0.1/dwa/	302	Found	6 ms	0 bytes
GET	http://127.0.0.1/dwa/login.php	200	OK	7 ms	1,415 bytes
GET	http://127.0.0.1/dwa/dwa/css/login.css	200	OK	8 ms	842 bytes
POST	http://127.0.0.1/dwa/login.php	302	Found	7 ms	0 bytes
GET	http://127.0.0.1/dwa/login.php	200	OK	6 ms	1,454 bytes
GET	https://blocklists.settings.services.mozilla.com/...	504	Gateway Ti...	20.0...	405 bytes

# Set up Environment

- Log into your range
- Open the Kali Linux Environment
  - You should be on your Kali Linux Desktop



# Start DVWA Servers

- Start up the web servers (on the Kali machine)
  - If you used the DVWA Setup Lab, use the following command to start XAMPP (then start/restart all the servers under the Manage Servers tab):

```
sudo /opt/lampp/xampp start
```



# Open OWASP ZAP

- Start the OWASP ZAP application

`owasp-zap`

```
(kali@10.15.22.173)-[~]  
$ owasp-zap  
Found Java version 17.0.8  
Available memory: 1947 MB  
Using JVM args: -Xmx486m
```

Select the top option  
and the hit start

OWASP ZAP

**Do you want to persist the ZAP Session?**

Yes, I want to persist this session with name based on the current timestamp

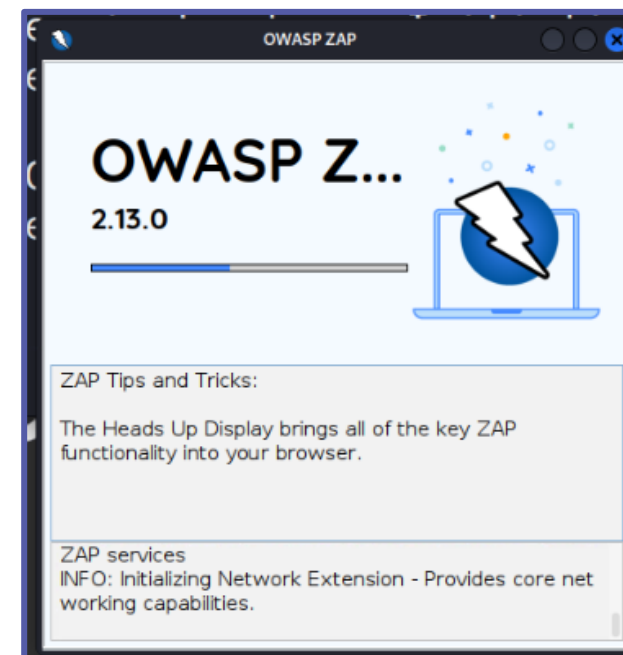
Yes, I want to persist this session but I want to specify the name and location

No, I do not want to persist this session at this moment in time

Remember my choice and do not ask me again.

You can always change your decision via the Options / Database screen

Help Start



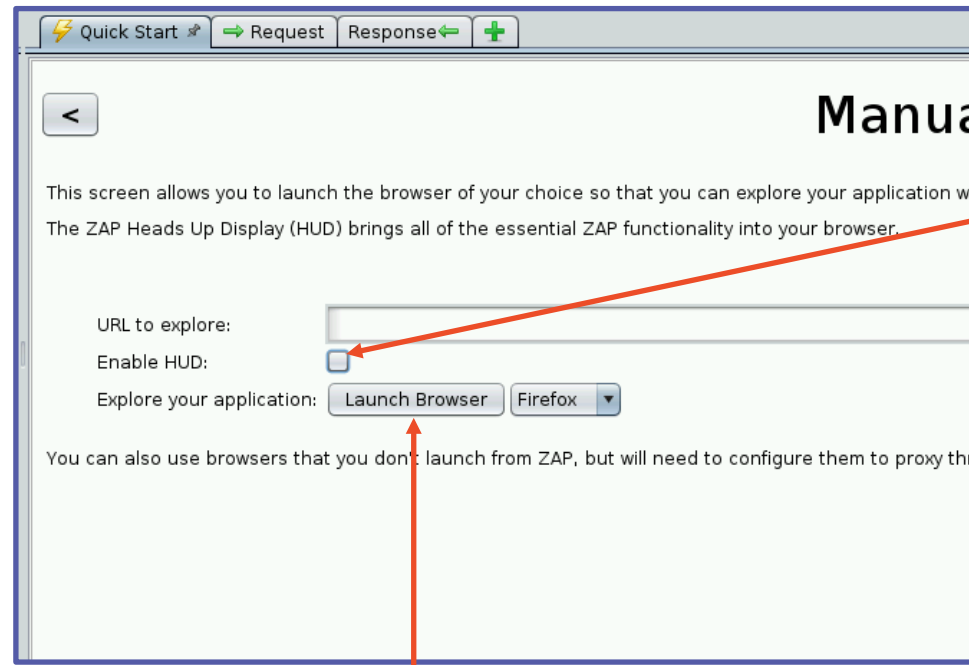


# Launch the Web Browser

- Click on Manual Explore
- Unclick the Enable HUD option
- Click on Launch Browser



1. Select "Manual Explore"




2. Unselect the "Enable HUD:" option

3. Then, select "Launch Browser"



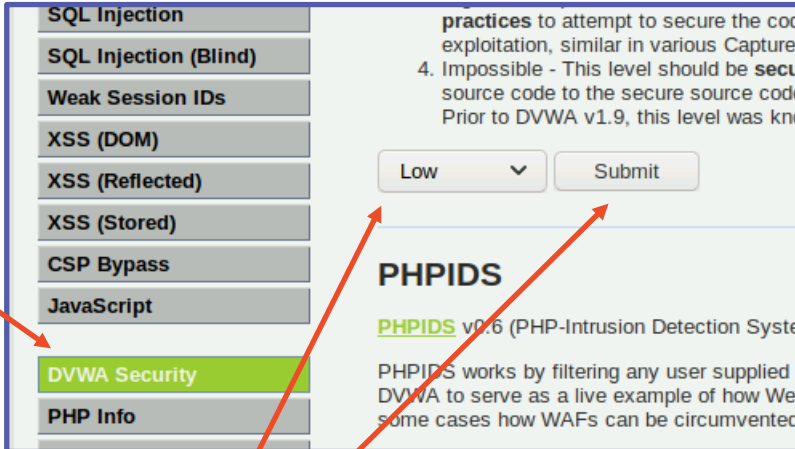
# Log Into DVWA

- Access DVWA
  - Go to the URL **127.0.0.1/dvwa**
- Enter the following credentials
  - Username: **admin**
  - Password: **password**
- Click on **DVWA Security**
- Set the Security to **LOW**
  - Then click on **Submit**



The image shows the DVWA login page. At the top is the DVWA logo. Below it are two input fields: 'Username' with the text 'admin' and 'Password' with masked characters. A 'Login' button is positioned below the password field.

Click on DVWA Security



The image shows the DVWA Security settings page. On the left is a sidebar menu with items like 'SQL Injection', 'XSS (DOM)', and 'DVWA Security' (which is highlighted in green). The main content area shows a dropdown menu set to 'Low' and a 'Submit' button. Below this is the 'PHPIDS' section with descriptive text.

Set to Low and Click Submit

# Enter False Credentials

- Click on **Brute Force** option
- For the Username, enter **THISISTHEUSERNAME**
- For the Password, enter **THISISTHEPASSWORD**
- Click the **Login** button

Select Brute Force

Home  
Instructions  
Setup / Reset DB  
**Brute Force**  
Command Injection  
CSRF  
File Inclusion  
File Upload  
Insecure CAPTCHA

### Vulnerability: Brute Force

#### Login

Username:  
THISISTHEUSERNAME

Password:  
.....

Login

Enter wrong  
credentials

Then, click  
Login



# View the GET Request

- Navigate back to OWASP-ZAP application
- Click on History
- Double click on the last GET under the Method column

Verify the username and password that was input are shown

History

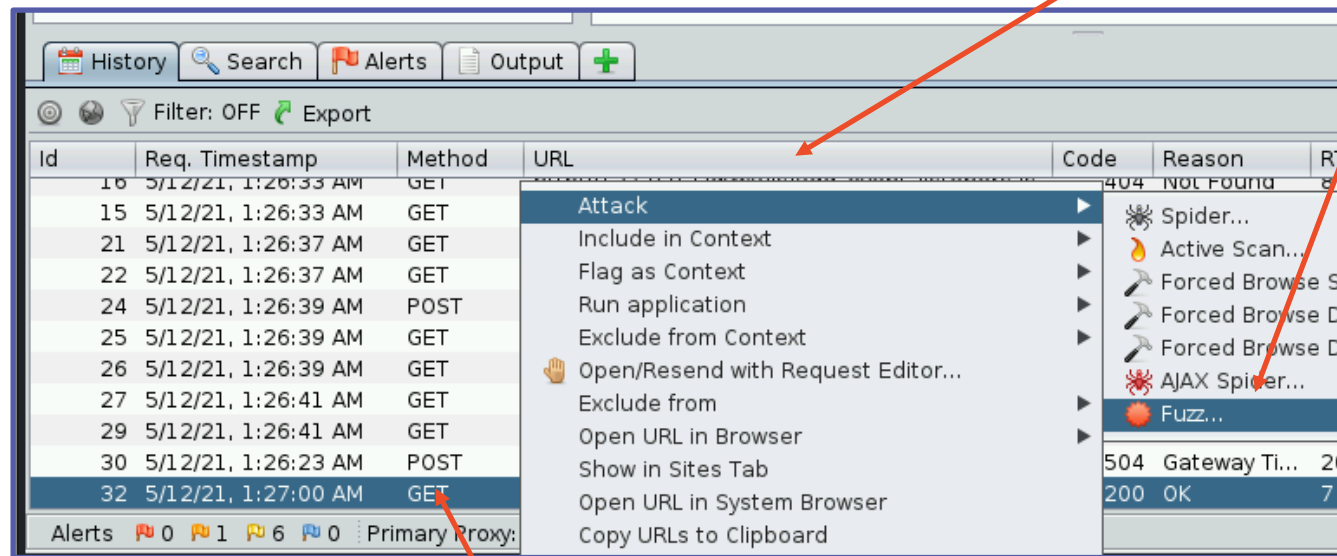
The screenshot shows the OWASP ZAP History tab. The left sidebar shows 'Contexts' and 'Sites'. The main area displays a list of requests with columns: Req. Timestamp, Method, URL, Code, Reason, RTT, Size Resp. Body, Highest Alert, Note, and Tags. The last request is highlighted in blue.

Req. Timestamp	Method	URL	Code	Reason	RTT	Size Resp. Body	Highest Alert	Note	Tags
16 5/12/21, 1:26:33 AM	GET	http://127.0.0.1/dwaw/js/add_event_listeners.js	404	Not Found	8 ms	1,174 bytes	Low		MailTo, Comment
15 5/12/21, 1:26:33 AM	GET	http://127.0.0.1/dwaw/dwaw/js/dwawPage.js	200	OK	2 ms	1,030 bytes	Low		Comment
21 5/12/21, 1:26:37 AM	GET	http://127.0.0.1/dwaw/security.php	200	OK	4 ms	5,283 bytes	Medium		Form, Hidden, Scr...
22 5/12/21, 1:26:37 AM	GET	http://127.0.0.1/dwaw/js/add_event_listeners.js	404	Not Found	3 ms	1,180 bytes	Low		MailTo, Comment
24 5/12/21, 1:26:39 AM	POST	http://127.0.0.1/dwaw/security.php	302	Found	4 ms	0 bytes	Low		SetCookie
25 5/12/21, 1:26:39 AM	GET	http://127.0.0.1/dwaw/security.php	200	OK	6 ms	5,352 bytes	Medium		Form, Hidden, Scr...
26 5/12/21, 1:26:39 AM	GET	http://127.0.0.1/dwaw/js/add_event_listeners.js	404	Not Found	3 ms	1,180 bytes	Low		MailTo, Comment
27 5/12/21, 1:26:41 AM	GET	http://127.0.0.1/dwaw/vulnerabilities/brute/	200	OK	4 ms	4,185 bytes	Medium		Form, Password, ...
29 5/12/21, 1:26:41 AM	GET	http://127.0.0.1/dwaw/dwaw/js/add_event_list...	200	OK	2 ms	593 bytes	Low		
30 5/12/21, 1:26:23 AM	POST	https://shavar.services.mozilla.com/download...	504	Gateway Ti...	20....	248 bytes			
32 5/12/21, 1:27:00 AM	GET	http://127.0.0.1/dwaw/vulnerabilities/brute/?u...	200	OK	7 ms	4,237 bytes	Medium		Form, Password, ...

Double click the last GET under the Method column

# Brute Force the Password

- When you see the username and password that you entered, attempt to brute force the password
- Right click on the GET packet
- Select “Attack” and then “Fuzz...”

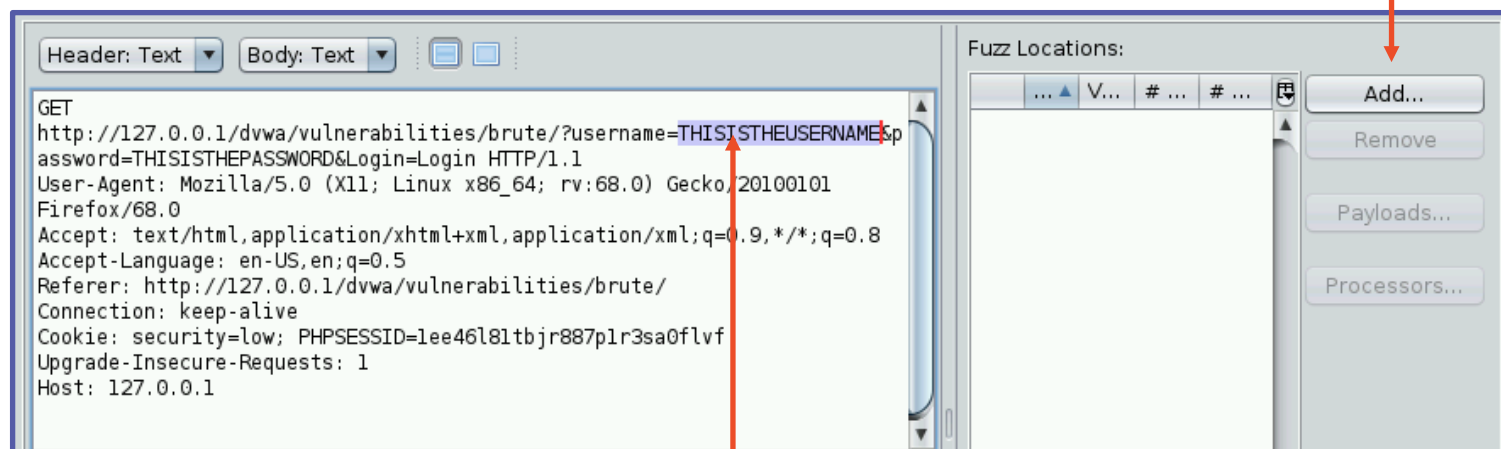


Click on Attack and then Fuzz...

Right-Click on the GET packet

# Brute Force the User/Password

- Highlight all of the Username entered
- Then click on “Add...”



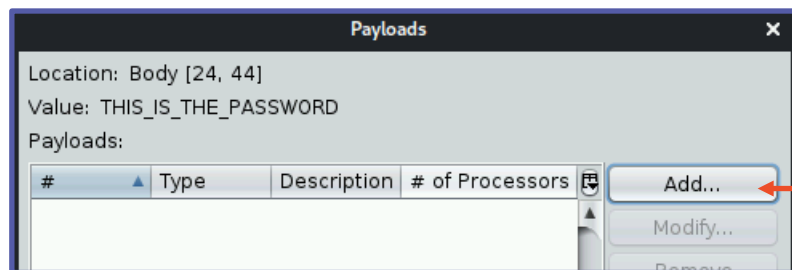
The click on Add

Highlight the entire password

Here, “THISISTHEUSERNAME” is highlighted

# Brute Force the User/Password

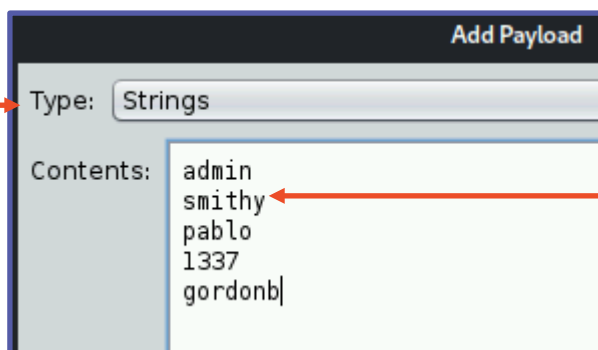
- Click on “Add...” in the Payloads



Click on Add

- With the Type: set as Strings, enter the following 5 usernames
  - **admin, smithy, pablo, 1337, and gordonb**

Make sure Type is set as Strings

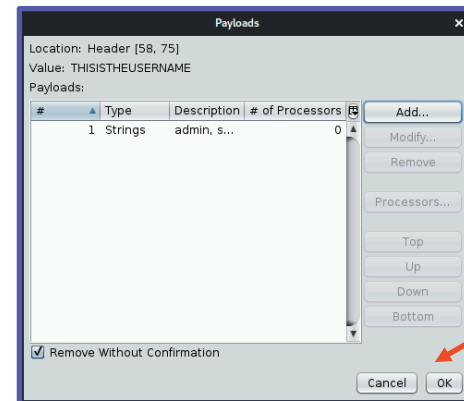


Enter the 5 usernames

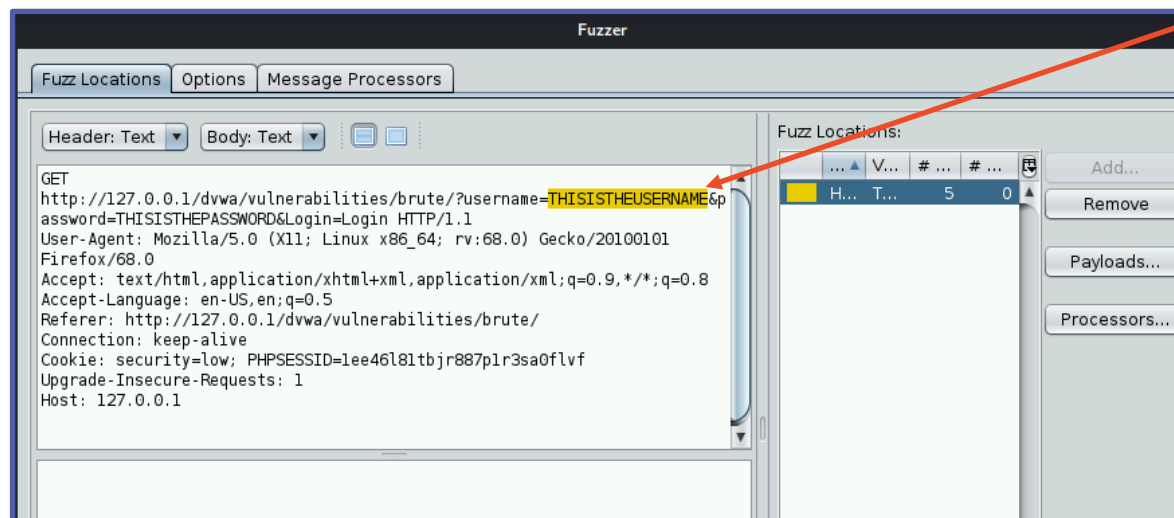
Please Note: These Usernames can be found using the SQL Injection Lab

# Brute Force the User/Password

- Then select **Add**
- Then select **Ok**
  - This will bring you back to the Fuzzer screen



Select Ok

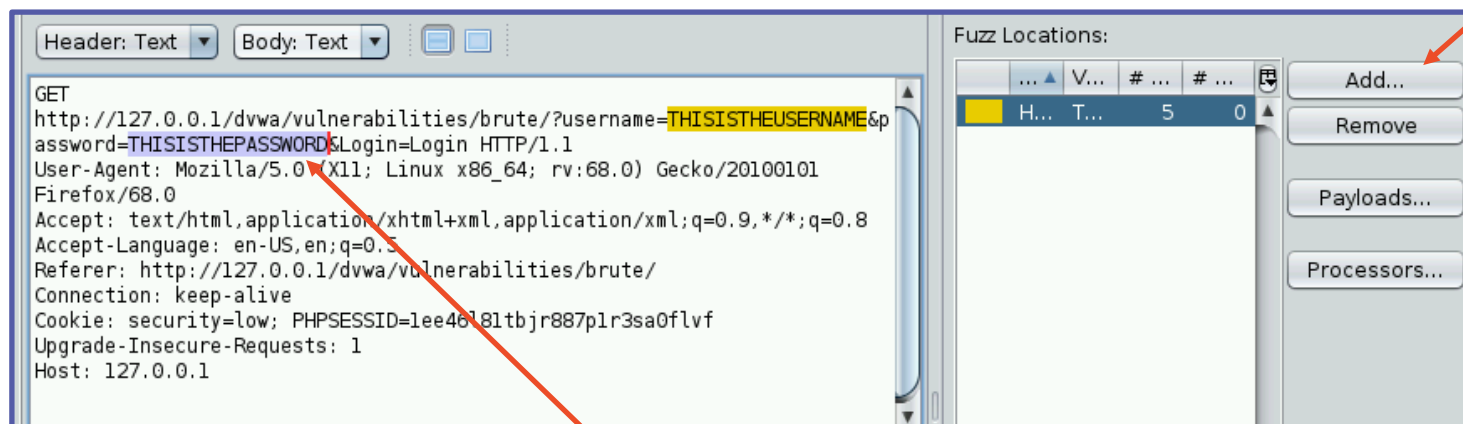


Verify that the Username has been highlighted a color



# Brute Force the User/Password

- Highlight all of the Password entered
- Then click on “Add...”



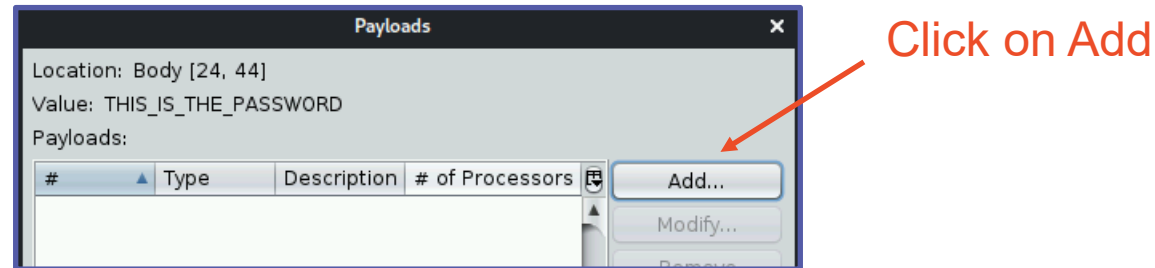
The click on Add

Highlight the entire password

Here, “THISISTHEPASSWORD” is highlighted

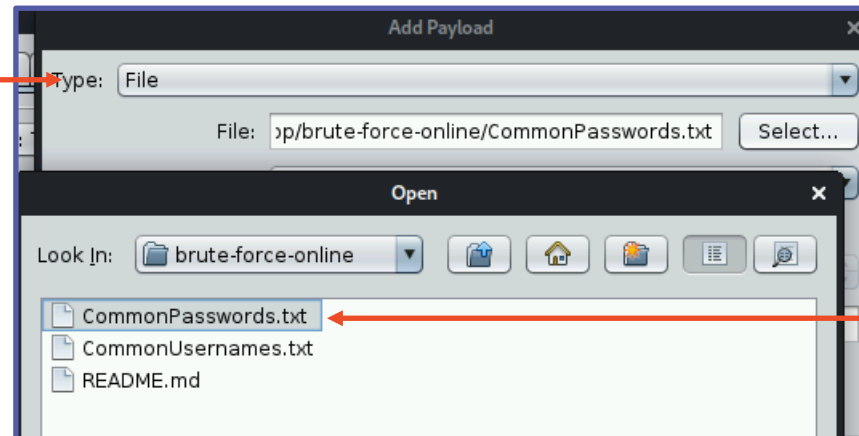
# Brute Force the Password

- Click on “Add...” in the Payloads



- In the Drop-down menu next to “Type:” select “File”
- Search for the file in the brute-force-online folder in the CourseFiles/Cybersecurity folder named CommonPasswords.txt, select it and then click Open

Select the  
“File” for  
Type

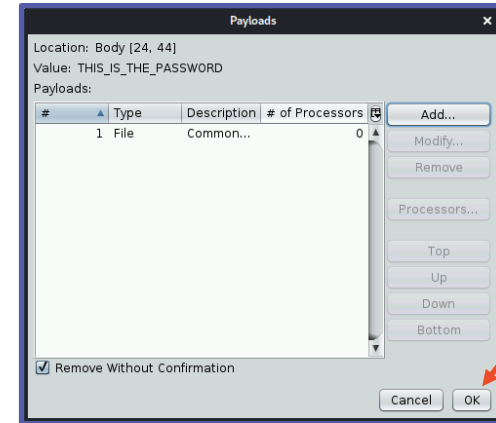


Select the  
CommonPasswords.txt  
file



# Brute Force the Password

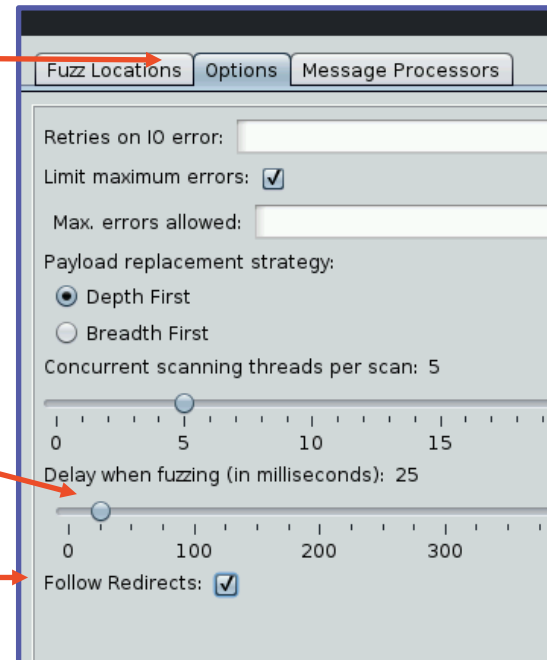
- Click on “Add” and then click “Ok” in the Payloads menu
- In the Fuzzer, select the Options Tab
- Set the delay when fuzzing to 25ms
- Select the “Follow Redirects” option



Click Options tab

Set the delay to 25ms

Select the Follow Redirects option



# Brute Force the Password

- Click on the Start Fuzzer
- This starts the Brute Force attempt
- Organize the data by **Size Resp. Body**
- Look for the largest responses

Size Resp. Body	Payloads
4,237 bytes	gordonb, dallas
4,237 bytes	gordonb, austin
4,237 bytes	gordonb, thunder
4,237 bytes	gordonb, taylor
4,237 bytes	gordonb, matrix
4,280 bytes	admin, password
4,280 bytes	pablo, letmein
4,282 bytes	smithy, password
4,284 bytes	gordonb, abc123

Largest Responses

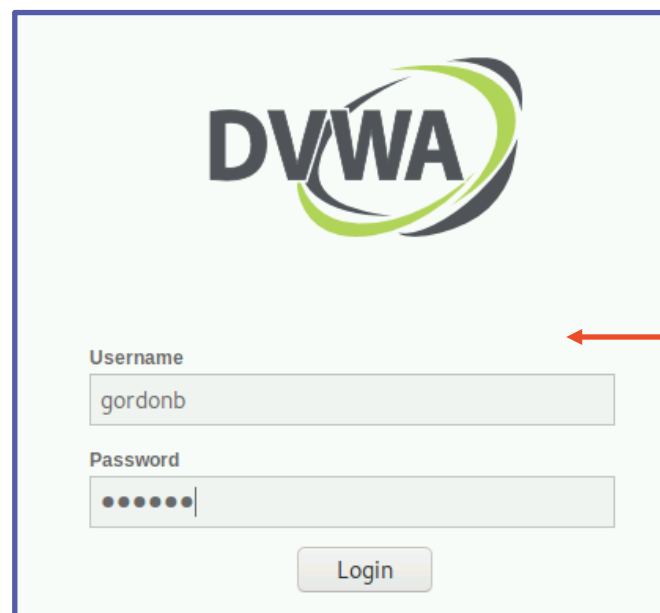
Cracked  
Username/Passwords

The 4 passwords with the largest Size Resp. Body should be the username/password combinations

Which username did not correctly crack a password?

# Log into DVWA

- Go back to the browser
- Log out of DVWA
- Attempt to login as another user
  - Use the captured/cracked credentials



The screenshot shows the DVWA login interface. At the top center is the DVWA logo, which consists of the letters 'DVWA' in a bold, dark font, with a green and grey swoosh graphic to its right. Below the logo are two input fields: 'Username' and 'Password'. The 'Username' field contains the text 'gordonb'. The 'Password' field contains six dots, indicating a masked password. Below these fields is a 'Login' button.

Log back in using  
another account's  
credentials

# How to Defend Against a Brute Force Attack

- Strong Passwords
  - Why is a longer password stronger? (D0e5 w31rd sp3LLing M4tt3r?)
  - Why were some passwords solved before others?
- Increasingly longer delay between failed attempts
  - Slow down the attacker. (10s, 15s, 30s, 45s, 1minute between attempts.)



# How to Defend Against a Brute Force Attack

- Lockout after \_\_\_ failed attempts
  - Account will eventually lock. User will need contact support to regain access.
- Two-Factor Authentication
  - Why would these help secure your password?
- What are some other ways of defending against a brute force attack?

