

Отчёт по индивидуальному проекту этап №5

Дисциплина: Основы информационной безопасности

Паращенко Антонина Дмитриевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Вывод	19
	Список литературы	20

Список иллюстраций

2.1	Запуск Apache2	6
2.2	Запуск Burp Suite	7
2.3	Intercept is on	7
2.4	Настройки сервера в браузере	8
2.5	network_allow_hijacking_localhost	8
2.6	Захваченный запрос	9
2.7	Страница авторизации	9
2.8	Target	9
2.9	случайный логин и пароль	10
2.10	Intruder	10
2.11	Cluster bomb	11
2.12	Логин	12
2.13	Пароль	13
2.14	Результаты атаки	14
2.15	location: login.php	14
2.16	admin password	15
2.17	Send to Repeater	16
2.18	location: index.php	16
2.19	Follow redirection	17
2.20	Render	18

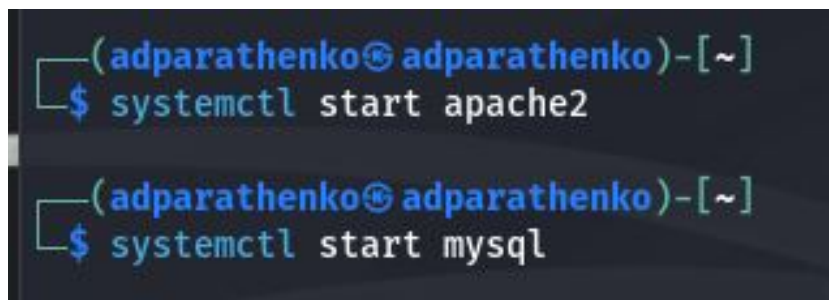
Список таблиц

1 Цель работы

Научиться использовать на практике Burp Suite.

2 Выполнение лабораторной работы

- 1) Запускаем локальный сервер, на котором откроем веб-приложение DVWA для тестирования инструмента Burp Suite. (рис. 2.1)



```
(adparathenko@adparathenko)~  
$ systemctl start apache2  
  
(adparathenko@adparathenko)~  
$ systemctl start mysql
```

Рис. 2.1: Запуск Apache2

- 2) Запускаем Burp Suite. (рис. 2.2)

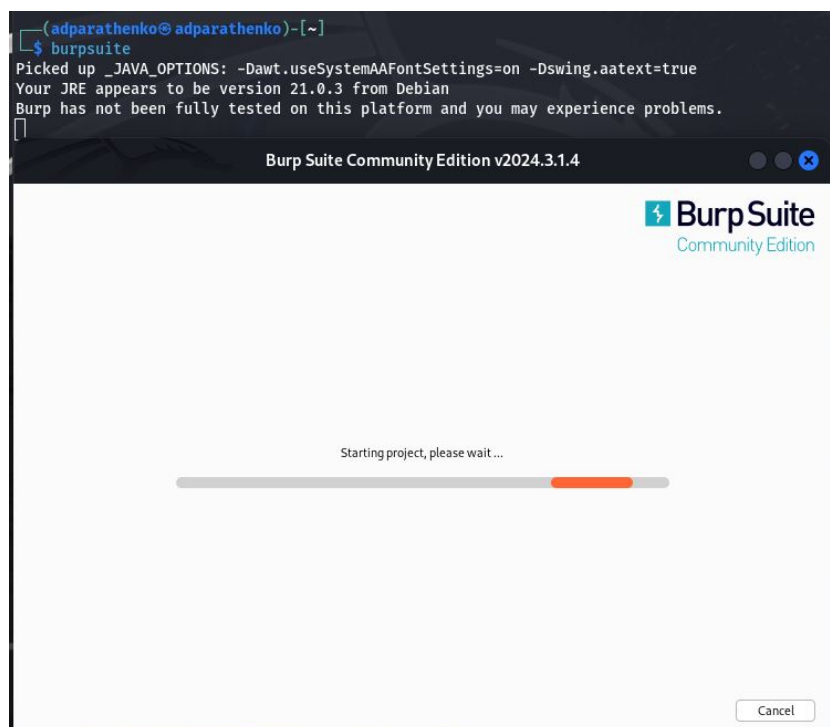


Рис. 2.2: Запуск Burp Suite

3) Изменяем настройки в Проху на *Intercept is on*. рис. 2.3)

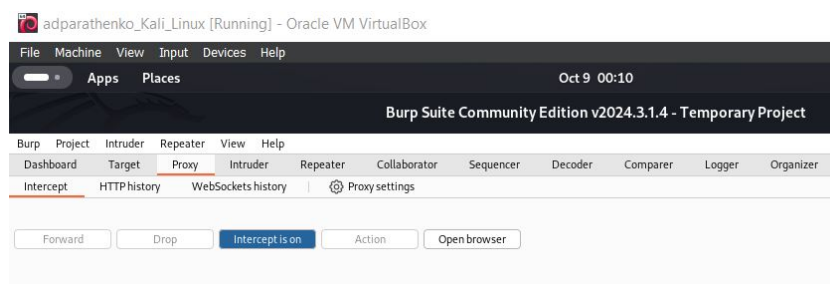


Рис. 2.3: Intercept is on

4) Изменяем настройки сервера в браузере для работы с Проху и захватом данных с помощью Burp Suite. (рис. 2.4)

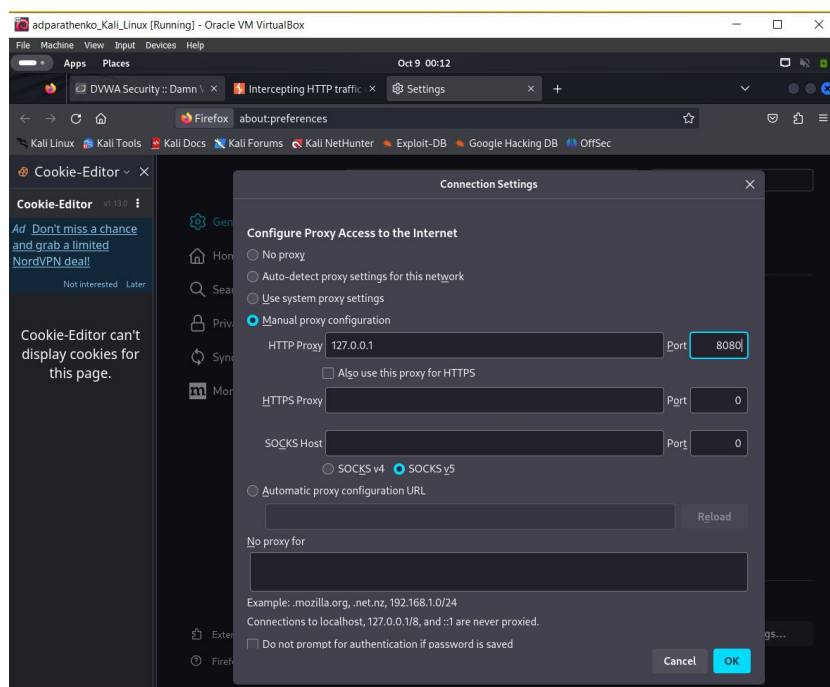


Рис. 2.4: Настройки сервера в браузере

5) Устанавливаем параметр `network_allow_hijacking_localhost` на `true`. (рис. 2.5)

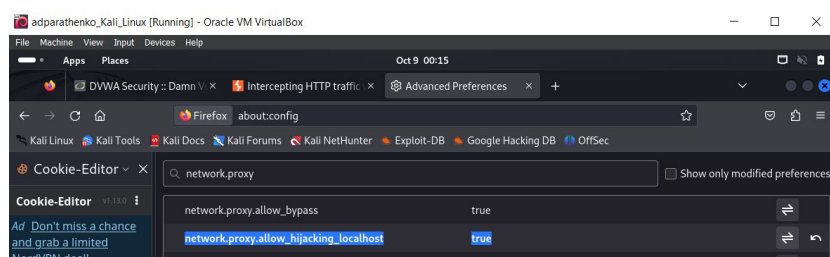


Рис. 2.5: `network_allow_hijacking_localhost`

6) В браузере заходим на DVWA и во вкладке Proxy появляется захваченный запрос. нажимаем Forward, чтобы загрузить страницу. (рис. 2.6)

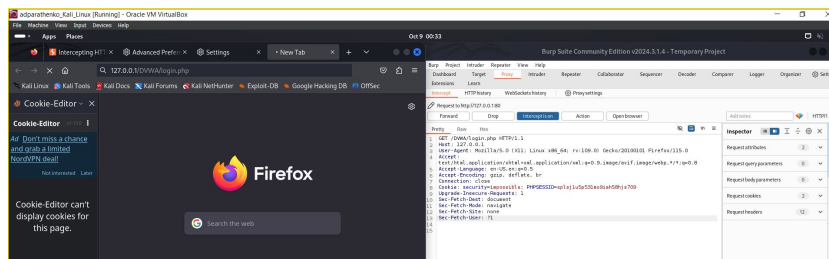


Рис. 2.6: Захваченный запрос

7) Загрузилась страница авторизации. (рис. 2.7)

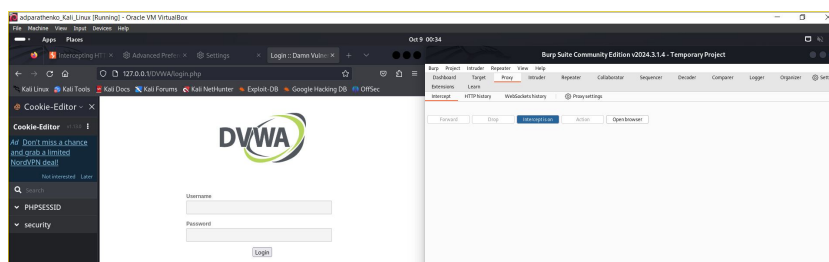


Рис. 2.7: Страница авторизации

8) Историю запросов можно посмотреть во вкладке Target. (рис. 2.8)

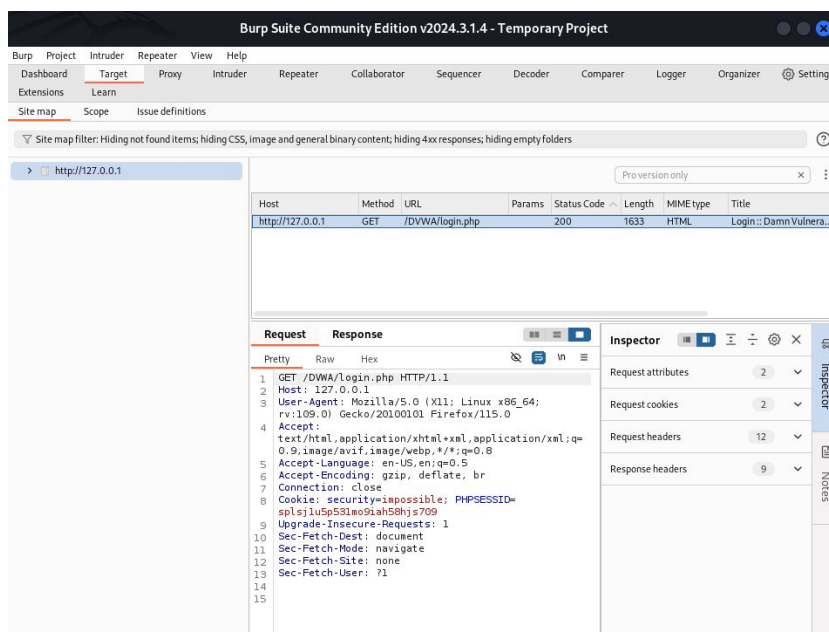


Рис. 2.8: Target

- 9) Вводя случайный логин и пароль, в запросе мы увидим введенные данные.
(рис. 2.9)

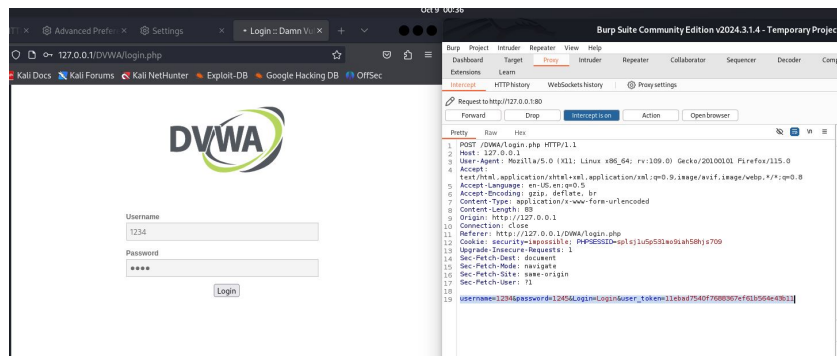


Рис. 2.9: случайный логин и пароль

- 10) Находим этот запрос во вкладке Target и, нажимая на правую кнопку мыши, нажимаем на Send to Intruder. Попадая во вкладку мы видим вид атаки.
(рис. 2.10)

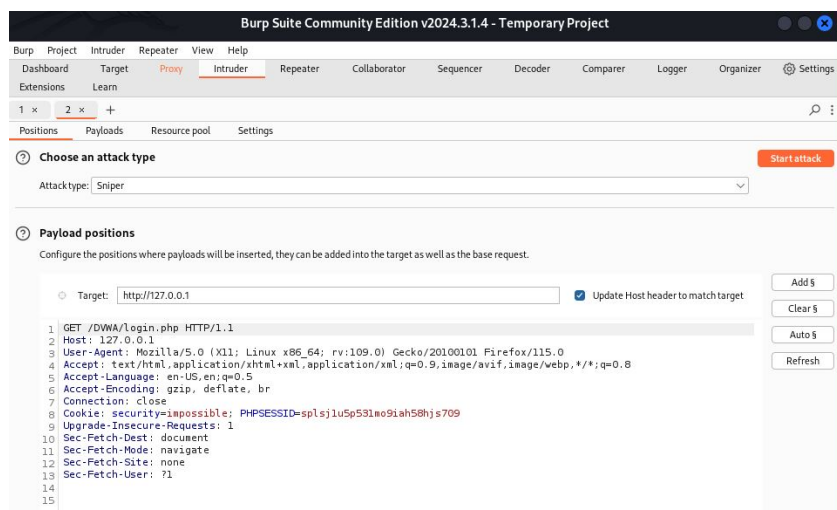


Рис. 2.10: Intruder

- 11) Меняем вид атаки на Cluster bomb и выделяем специальными знаками данные ввода, которые хотим подбирать, в нашем случае, это логин и пароль.
(рис. 2.11)

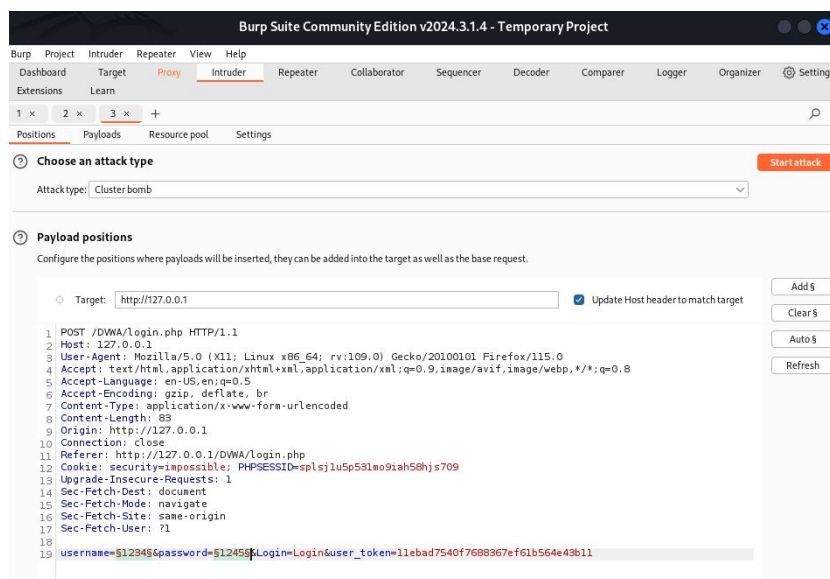


Рис. 2.11: Cluster bomb

- 12) Далее добавляем 2 списка параметров для подбора логина и пароля. (рис. 2.12) -(рис. 2.13)

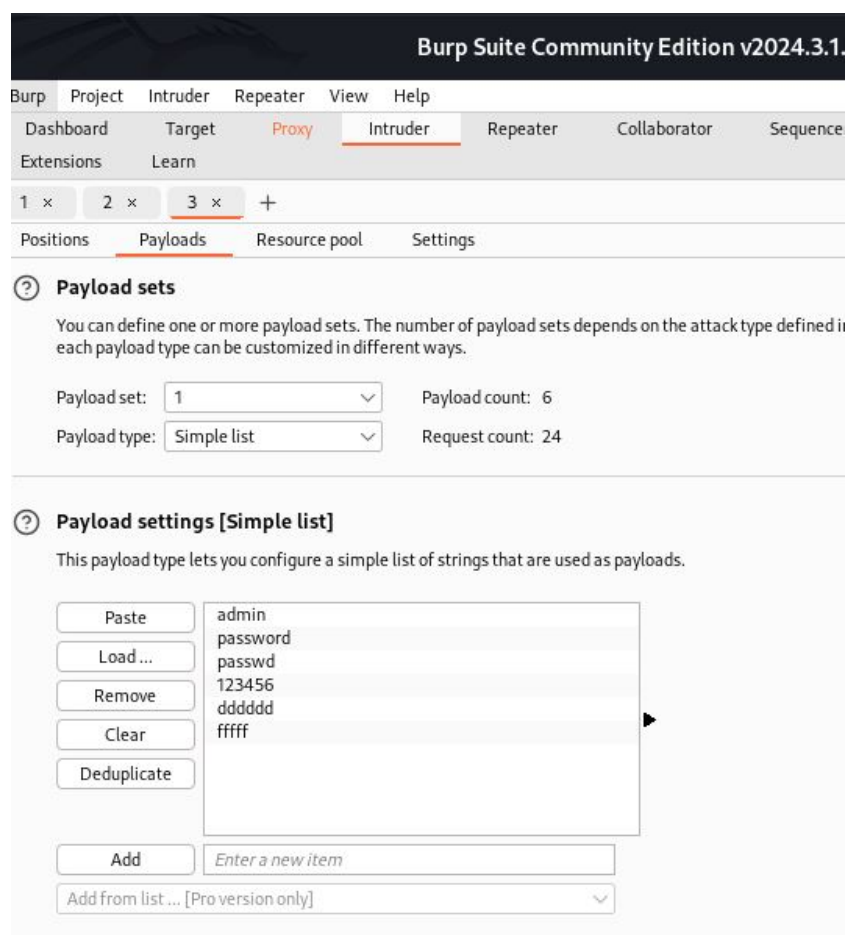


Рис. 2.12: Логин

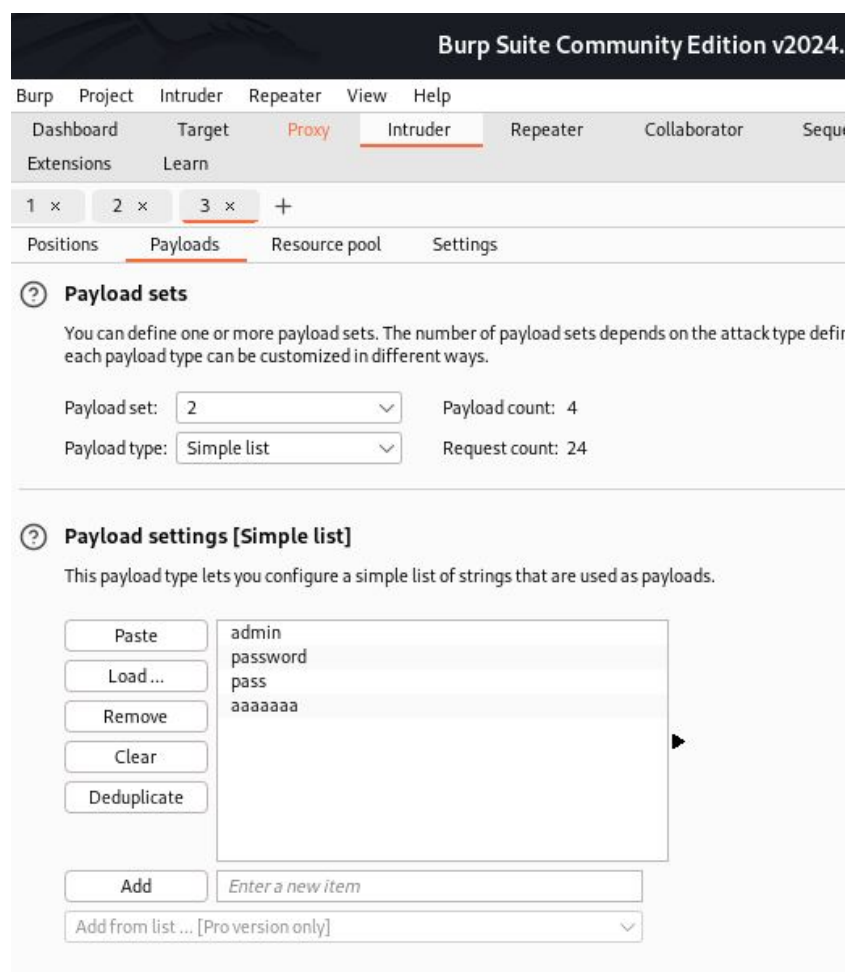


Рис. 2.13: Пароль

13) Запускаем атаку и получаем результаты перебора. (рис. 2.14)

2. Intruder attack of http://127.0.0.1							
Attack Save							
2. Intruder attack of http://127.0.0.1							
Results Positions Payloads Resource pool Settings							
Intruder attack results filter: Showing all items							
Request	Payload 1	Payload 2	Status code	Response ...	Error	Timeout	Length
0			302	15			476
1	admin	admin	302	28			475
2	password	admin	302	29			476
3	passwd	admin	302	46			475
4	123456	admin	302	46			476
5	dddddd	admin	302	56			475
6	fffff	admin	302	185			476
7	admin	password	302	23			476
8	password	password	302	11			475
9	passwd	password	302	26			476
10	123456	password	302	18			475
11	dddddd	password	302	26			475
12	fffff	password	302	27			476
13	admin	pass	302	23			476
14	password	pass	302	19			475
15	passwd	pass	302	17			475
16	123456	pass	302	10			475
17	dddddd	pass	302	19			476
18	fffff	pass	302	12			475
19	admin	aaaaaaa	302	28			476
20	password	aaaaaaa	302	25			476
21	passwd	aaaaaaa	302	25			476
22	123456	aaaaaaa	302	24			476
23	dddddd	aaaaaaa	302	31			476
24	fffff	aaaaaaa	302	19			476

Рис. 2.14: Результаты атаки

14) У всех вариантов перебора, кроме одного, *location: login.php*. (рис. 2.15)

2. Intruder attack of http://127.0.0.1							
Attack Save							
2. Intruder attack of http://127.0.0.1							
Results Positions Payloads Resource pool Settings							
Intruder attack results filter: Showing all items							
Request	Payload 1	Payload 2	Status code	Response ...	Error	Timeout	Length
0			302	15			476
1	admin	admin	302	28			475
2	password	admin	302	29			476
3	passwd	admin	302	46			475
4	123456	admin	302	46			476
5	dddddd	admin	302	56			475
6	fffff	admin	302	185			476
7	admin	password	302	23			476
8	password	password	302	11			475
9	passwd	password	302	26			476
10	123456	password	302	18			475
11	dddddd	password	302	26			475
12	fffff	password	302	27			476
13	admin	pass	302	23			476
14	password	pass	302	19			475
15	passwd	pass	302	17			475
16	123456	pass	302	10			475
17	dddddd	pass	302	19			476
18	fffff	pass	302	12			475
19	admin	aaaaaaa	302	28			476
20	password	aaaaaaa	302	25			476
21	passwd	aaaaaaa	302	25			476
22	123456	aaaaaaa	302	24			476
23	dddddd	aaaaaaa	302	31			476
24	fffff	aaaaaaa	302	19			476

Request	Response
F	Raw Hex Render
1	HTTP/1.1 302 Found
2	Date: Tue, 08 Oct 2024 21:46:35 GMT
3	Server: Apache/2.4.59 (Debian)
4	Expires: Thu, 19 Nov 1981 08:52:00 GMT
5	Cache-Control: no-store, no-cache, must-revalidate
6	Pragma: no-cache
7	Set-Cookie: PHPSESSID=krchjm2sresktqcafmsdvq8bj4; expires=Wed, 09 Oct 2024 21:46:35 GMT; Max-Age=86400; path=/
8	Location: login.php
9	Content-Length: 0
10	Keep-Alive: timeout=5, max=100
11	Connection: Keep-Alive
12	Content-Type: text/html; charset=UTF-8
13	
14	

Рис. 2.15: location: login.php

15) А у пары **admin password** результат *location: index.php*. Это показывает нам,

что это верная пара логин-пароль. (рис. 2.16)

2. Intruder attack of http://127.0.0.1

AttackSave

2. Intruder attack of http://127.0.0.1

Attack

ResultsPositionsPayloadsResource poolSettings

Intruder attack results filter: Showing all items

Request	Payload1	Payload2	Status code	Response ...	Error	Timeout	Length	Con
0			302	15			476	
1	admin	admin	302	28			475	
2	password	admin	302	29			476	
3	passwd	admin	302	46			475	
4	123456	admin	302	46			476	
5	ddddd	admin	302	56			475	
6	fffff	admin	302	185			476	
7	admin	password	302	23			476	
8	password	password	302	11			475	

RequestResponse

PrettyRawHexRender

```
1 HTTP/1.1 302 Found
2 Date: Tue, 08 Oct 2024 21:46:38 GMT
3 Server: Apache/2.4.59 (Debian)
4 Expires: Thu, 19 Nov 1981 08:52:00 GMT
5 Cache-Control: no-store, no-cache, must-revalidate
6 Pragma: no-cache
7 Set-Cookie: PHPSESSID=lb236j3gdmejhnb98ns6cr742q; expires=Wed, 09 Oct 2024 21:46:38 GMT; Max-Age=86400; path=/;
8 SameSite=Strict
9 Location: index.php
10 Content-Length: 0
11 Keep-Alive: timeout=5, max=100
12 Connection: Keep-Alive
13 Content-Type: text/html; charset=UTF-8
14
```

Рис. 2.16: admin password

16) Чтобы ещё раз проверить результат мы отправляем эту пару на повторную проверку *Send to Repeater*. (рис. 2.17)

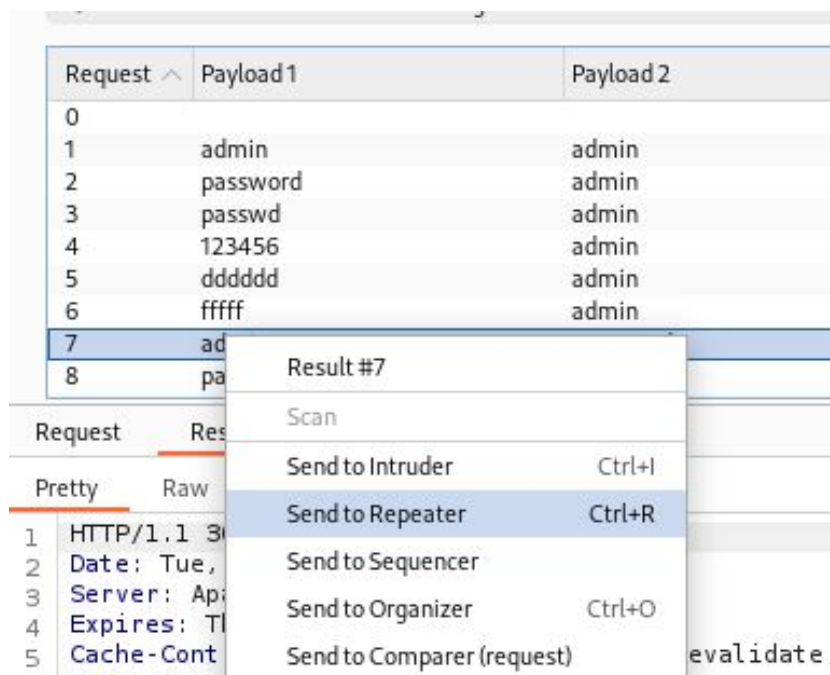


Рис. 2.17: Send to Repeater

17) Получаем тот же результат *location: index.php*. (рис. 2.18)

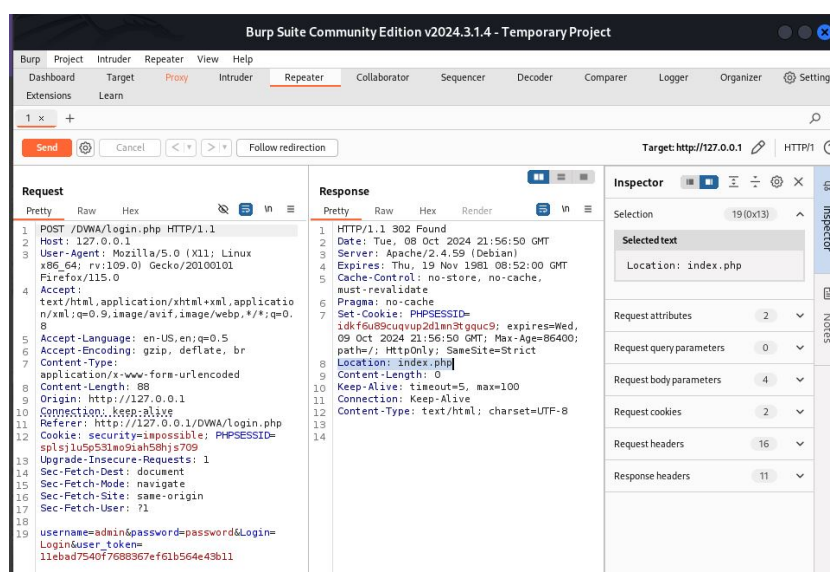


Рис. 2.18: location: index.php

18) Нажимаем на *Follow redirection* и получаем некомпилитированный html код в окне Response. (рис. 2.19)

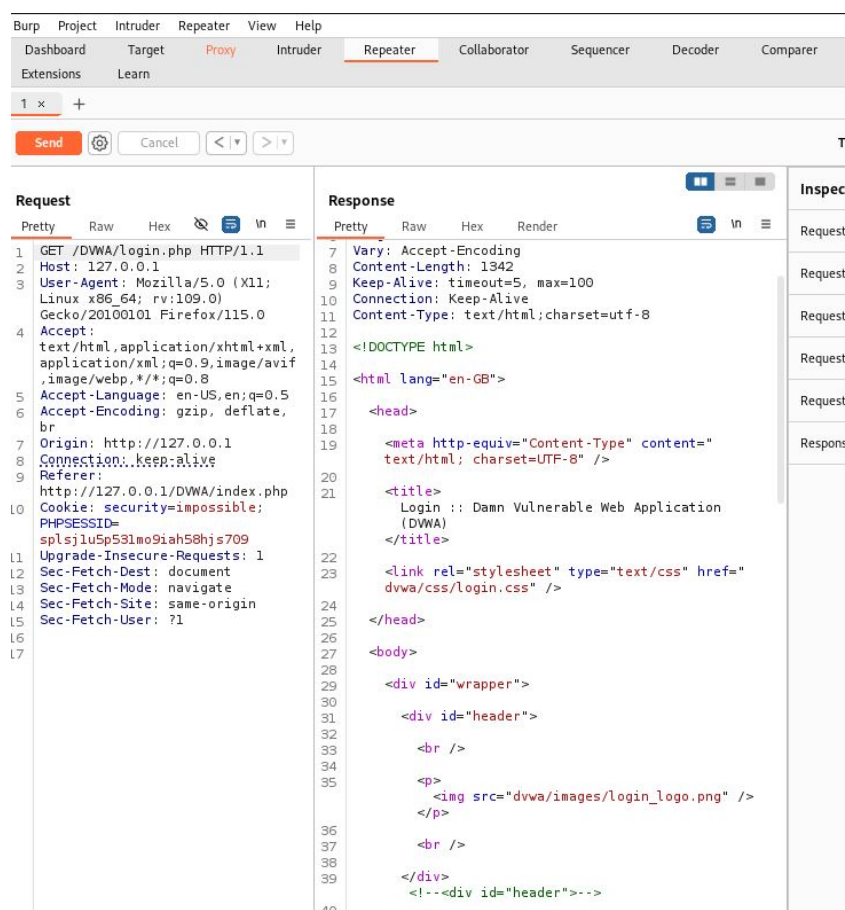


Рис. 2.19: Follow redirection

19) В подокне Render получаем вид страницы в браузере. (рис. 2.20)

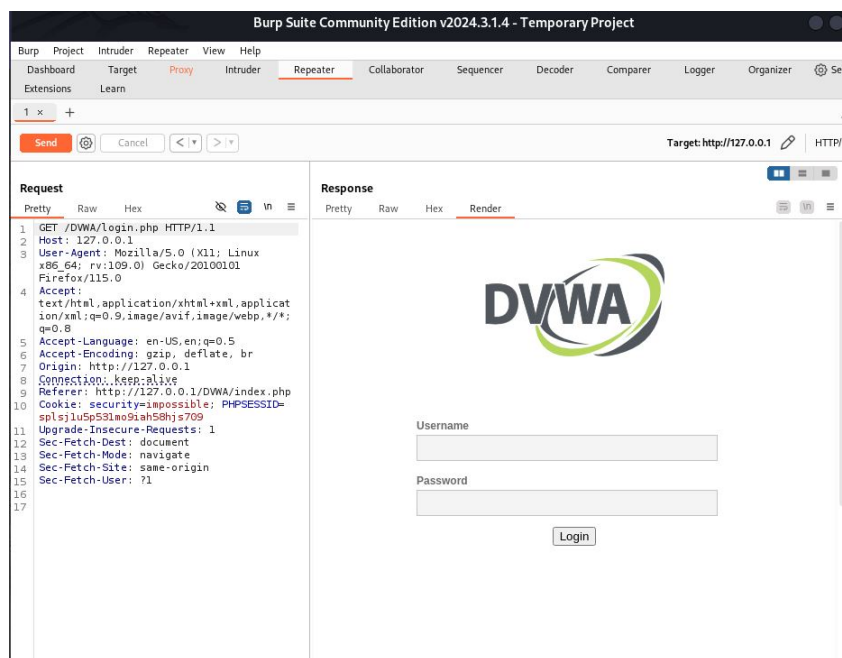


Рис. 2.20: Render

3 Вывод

В результате выполнения работы мы научились использовать инструмент Burp Suite и перебором возможных пар подобрали пару логин-пароль для входа на сайт.

Список литературы

- 1) Парасрам, Ш. Kali Linux: Тестирование на проникновение и безопасность : Для профессионалов. Kali Linux / Ш. Парасрам, А. Замм, Т. Хериянто, и др. – Санкт-Петербург : Питер, 2022. – 448 сс.