

# **Отчет по индивидуальной проекте 5**

**Основы информационной безопасности**

Нджову Нелиа

# Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	7
4	Выводы	20
	Список литературы	21

## Список иллюстраций

3.1	Запуск локального сервера . . . . .	7
3.2	Сетевые настройки браузера . . . . .	7
3.3	Настройки сервера . . . . .	8
3.4	Настройки Burp Suite . . . . .	8
3.5	Настройки Proxy . . . . .	9
3.6	Настройки параметров . . . . .	9
3.7	Получаемые запросы сервера . . . . .	9
3.8	Получаемые запросы сервера . . . . .	10
3.9	Страница авторизации . . . . .	10
3.10	Страница авторизации . . . . .	11
3.11	История запросов . . . . .	11
3.12	Ввод случайных данных . . . . .	12
3.13	Ввод случайных данных . . . . .	12
3.14	POST-запрос с вводом пароля и логина . . . . .	13
3.15	Вкладка Intruder . . . . .	13
3.16	Изменение типа атаки . . . . .	14
3.17	Первый Simple list . . . . .	15
3.18	Запуск атаки . . . . .	15
3.19	Результат запроса . . . . .	16
3.20	Результат запроса . . . . .	16
3.21	Дополнительная проверка результата . . . . .	17
3.22	Вкладка Repeater . . . . .	17
3.23	Окно Response . . . . .	18
3.24	Изменение в окне Response . . . . .	18
3.25	Полученная страница . . . . .	19

## Список таблиц

# 1 Цель работы

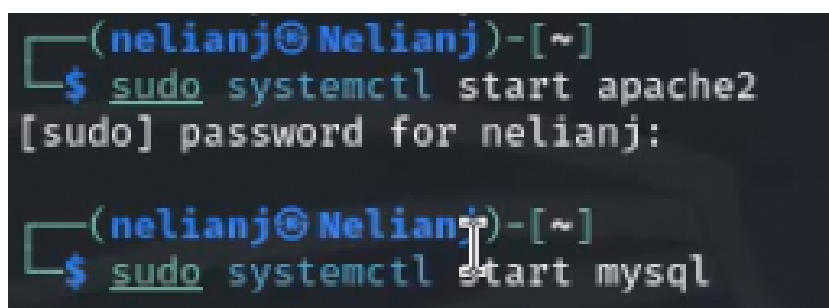
Научиться использовать Burp Suite.

## 2 Теоретическое введение

Burp Suite представляет собой набор мощных инструментов безопасности веб-приложений, которые демонстрируют реальные возможности злоумышленника, проникающего в веб-приложения. Эти инструменты позволяют сканировать, анализировать и использовать веб-приложения с помощью ручных и автоматических методов. Интеграция интерфейсов этих инструментов обеспечивает полную платформу атаки для обмена информацией между одним или несколькими инструментами, что делает Burp Suite очень эффективной и простой в использовании платформой для атаки веб-приложений.

### 3 Выполнение лабораторной работы

Я запускаю локальный сервер, на котором открою веб-приложение DVWA для тестирования инструмента Burp Suite(рис.1).



```
(nelianj@Nelianj)-[~]  
$ sudo systemctl start apache2  
[sudo] password for nelianj:  
  
(nelianj@Nelianj)-[~]  
$ sudo systemctl start mysql
```

Рис. 3.1: Запуск локального сервера

Я запускаю инструмент Burp Suite. После этого я открываю сетевые настройки браузера, для подготовке к работе(рис.2).

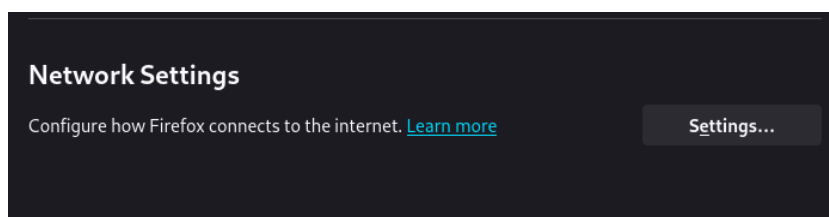


Рис. 3.2: Сетевые настройки браузера

Я изменяю настройки сервера для работы с проху и захватом данных с помощью Burp Suite(рис.3)

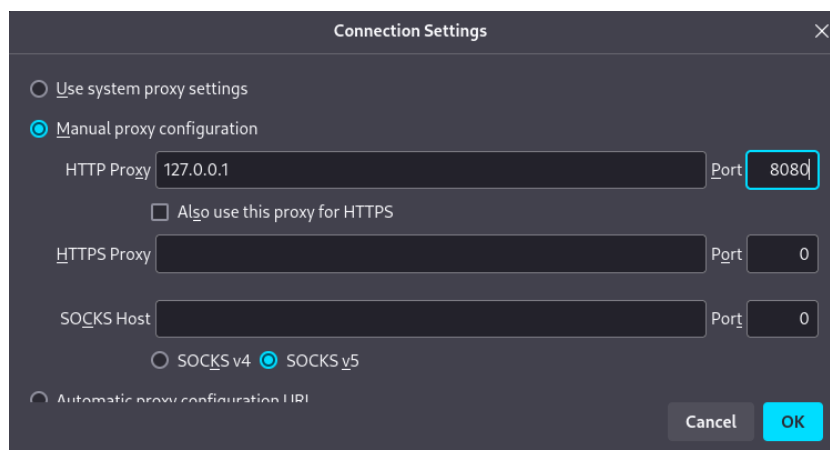


Рис. 3.3: Настройки сервера

Я изменяю настройки Proxy инструмента Burp Suite для дальнейшей работы(рис.4).

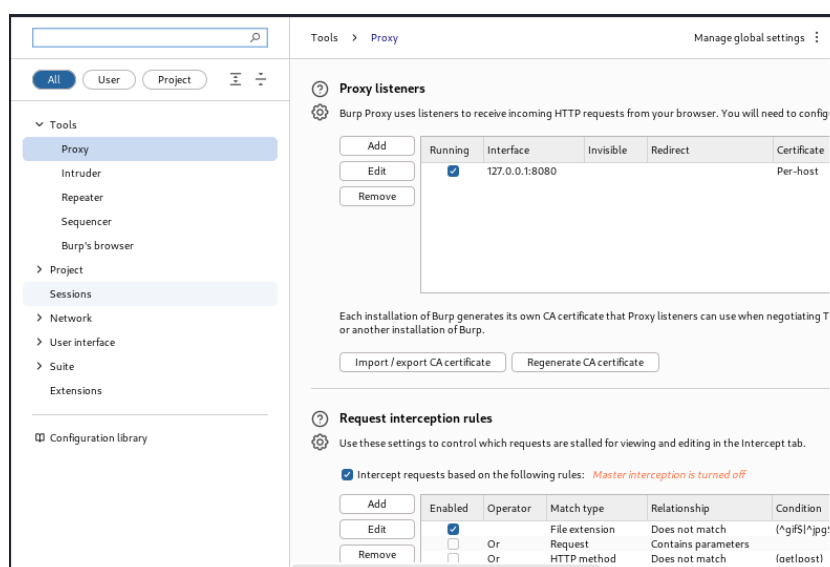


Рис. 3.4: Настройки Burp Suite

Во вкладке Proxy устанавливаю “Intercept is on”(рис.5).



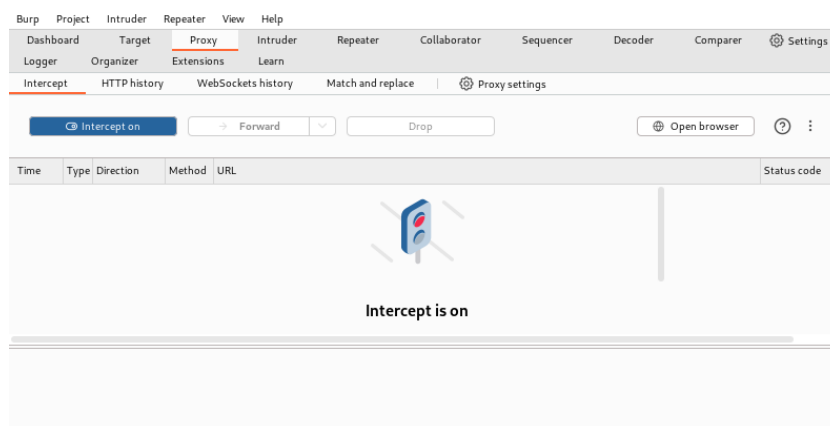


Рис. 3.5: Настройки Proxy

Чтобы Burp Suite исправно работал с локальным сервером, необходимо установить параметр `network.allow_hijacking_localhost` на `true` (рис.6).

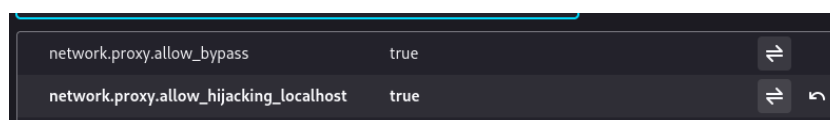


Рис. 3.6: Настройки параметров

Я пытаюсь зайти в браузере на DVWA, тут же во вкладки Proxy появляется захваченный запрос. Нажимаем “Forward”, чтобы загрузить страницу (рис.7 и рис.8).

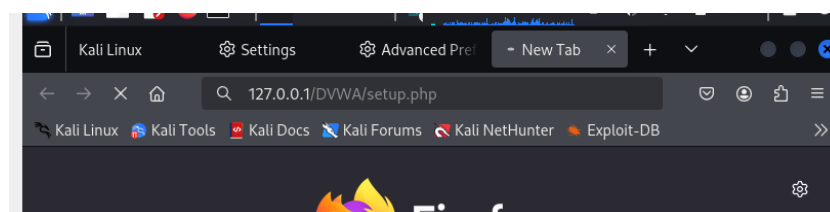


Рис. 3.7: Получаемые запросы сервера

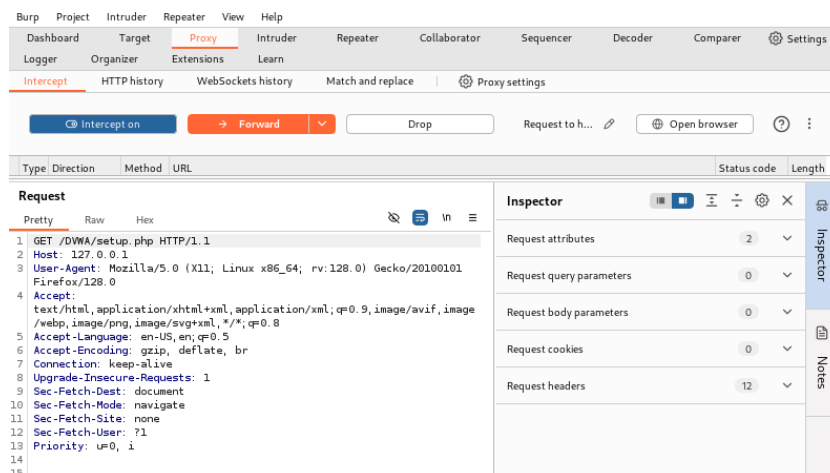


Рис. 3.8: Получаемые запросы сервера

Загрузилась страница авторизации, текст запроса поменялся(рис.9 и рис.10).

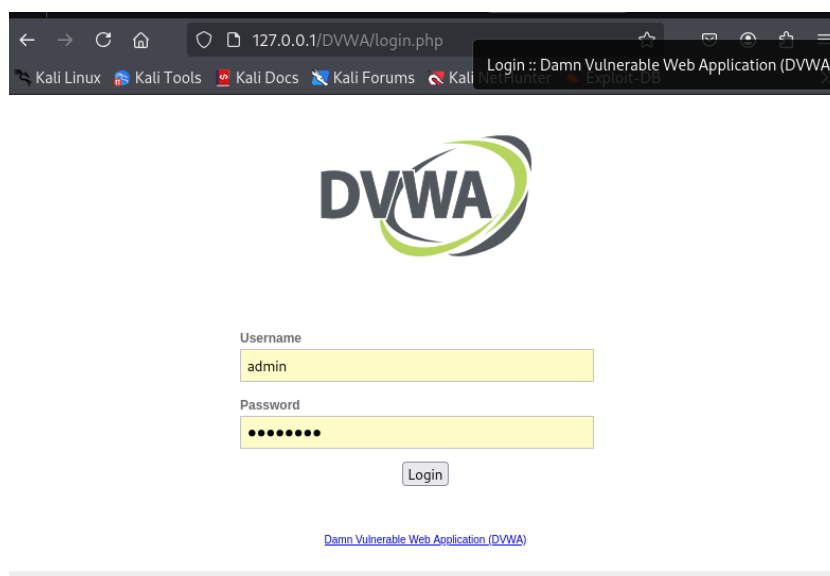


Рис. 3.9: Страница авторизации

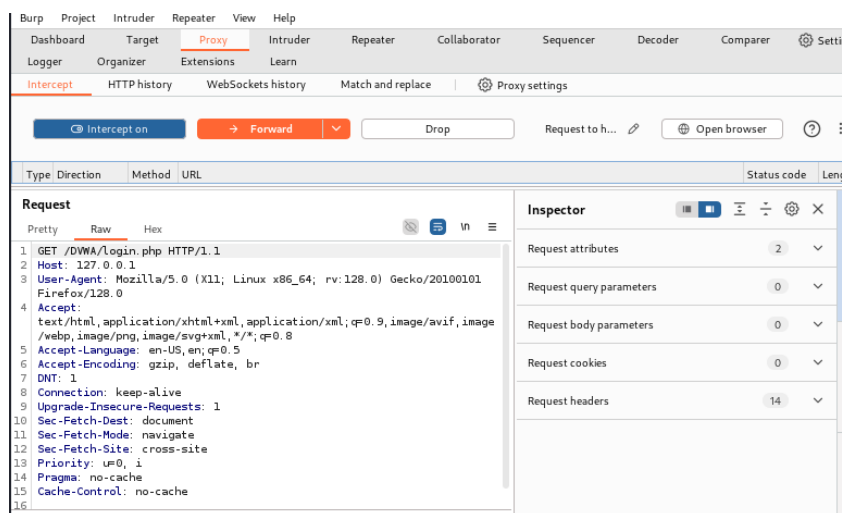


Рис. 3.10: Страница авторизации

История запросов хранится во вкладке Target (рис.11).

Host	Method	URL	Params	Status code
http://127.0.0.1	GET	/DWWA/dvwa/js/add_e...		200
http://127.0.0.1	GET	/DWWA/dvwa/js/dvwa...		200
http://127.0.0.1	GET	/DWWA/login.php		200
http://127.0.0.1	GET	/DWWA/setup.php		200
http://127.0.0.1	POST	/DWWA/setup.php	✓	302
http://127.0.0.1	GET	/DWWA/dvwa/images/l...		304
http://127.0.0.1	GET	/DWWA/dvwa/images/l...		304

Рис. 3.11: История запросов

Я попробую ввести неправильные, случайные данные в веб-приложении и нажмем Login. В запросе увидим строку, в которой отображаются введенные нами данные, то есть поле для ввода(рис.12 и рис.13).



Username  
aaaaa

Password  
.....

Login

Рис. 3.12: Ввод случайных данных

```
Request
Pretty Raw Hex
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 83
9 Origin: http://127.0.0.1
10 Connection: keep-alive
11 Referer: http://127.0.0.1/DVWA/login.php
12 Cookie: security=impossible; PHPSESSID=2g1g7427mLm9ngnm0co2f12ph
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18 Priority: u=0, i
19
20 username=aaa&password=aaaaa&Login=Login&user_token=33adb71dfc58d5a21b5c2ebdf1023337
```

Рис. 3.13: Ввод случайных данных

Этот запрос так же можно найти во вкладке Target, там же жмем правой кнопкой мыши на хост нужного запроса, и далее нажимаем “Send to Intruder”(рис.14).

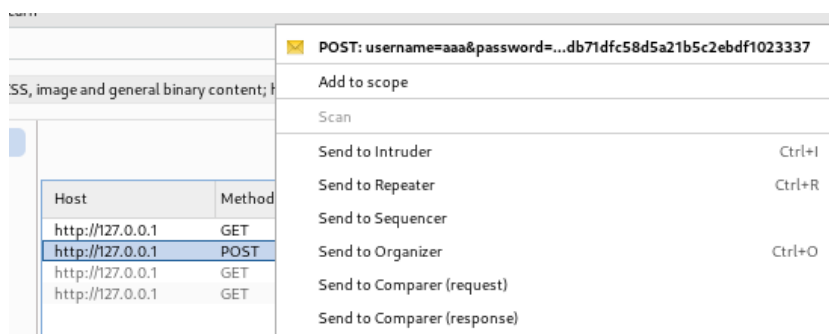


Рис. 3.14: POST-запрос с вводом пароля и логина

Я попадаю на вкладку Intruder, вижу значения по умолчанию у типа атаки и наш запрос(рис.15).

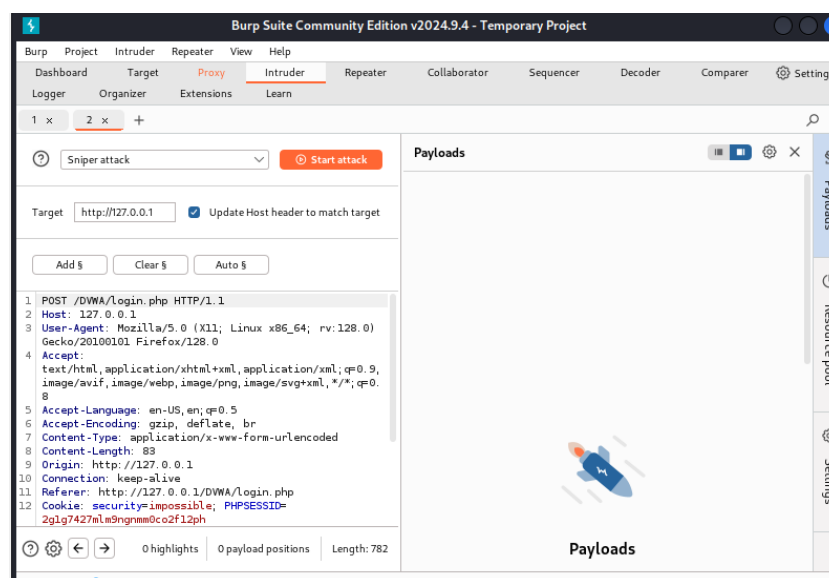


Рис. 3.15: Вкладка Intruder

Я изменяю значение типа атаки на Cluster bomb и проставляю специальные символы у тех данных в форме для ввода, которые буду пробивать(рис.16).

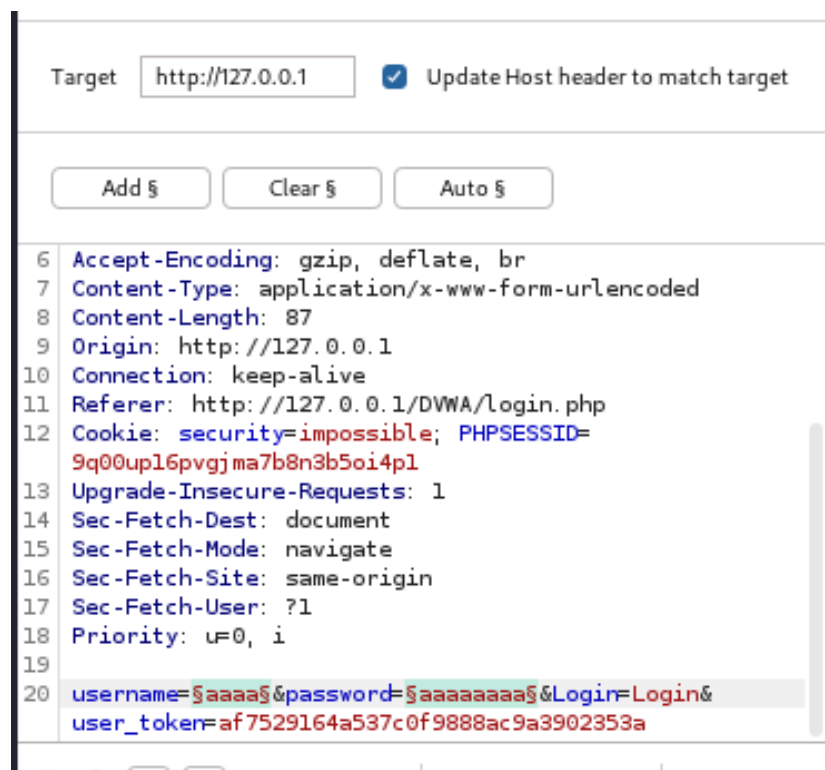


Рис. 3.16: Изменение типа атаки

Так как мне нужно параметра для подбора, то нужно списка со значениями для подбора(рис.17).

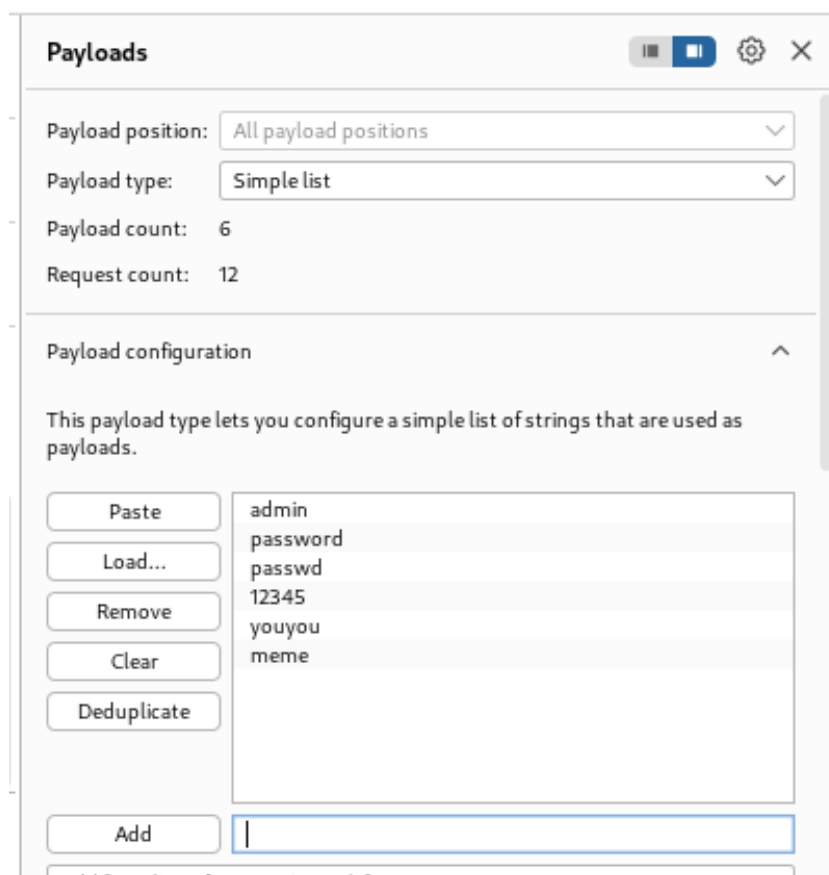


Рис. 3.17: Первый Simple list

Я запускаю атаку и начинаю подбор(рис.18).

2. Intruder attack of http://127.0.0.1

Results Positions

Intruder attack results filter: Showing all items

Request	Position	Payload	Status code	Response received	Error
3	1	passwd	302	2	
4	1	12345	302	4	
5	1	youyou	302	5	
6	1	meme	302	3	
7	2	admin	302	4	
8	2	password	302	0	
9	2	passwd	302	2	
10	2	12345	302	1	
11	2	youyou	302	3	
12	2	meme	302	1	

Рис. 3.18: Запуск атаки

При открытии результата каждого post-запроса можно увидеть полученный get-запрос, в нем видно, куда нас перенаправило после выполнения

ввода пары пользователь-пароль. В представленном случае с подбором пары passwd-password нас перенаправило на login.php, это значит, что пара не подходит(рис.19).

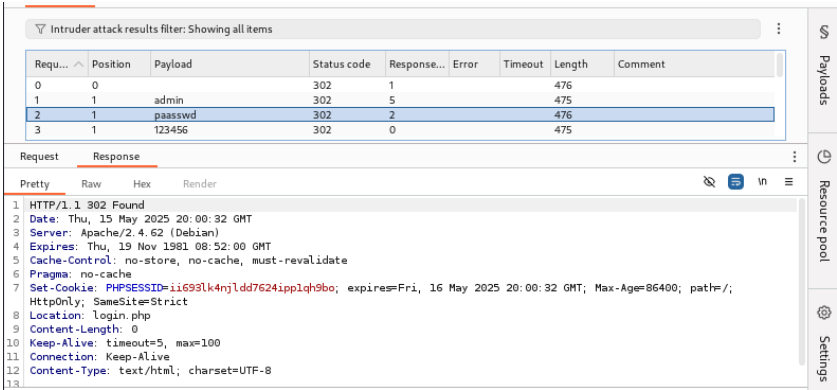


Рис. 3.19: Результат запроса

Я проверяю результат пары admin-password во вкладке Response, теперь нас перенаправляет на страницу index.php, значит пара должна быть верной(рис.20).

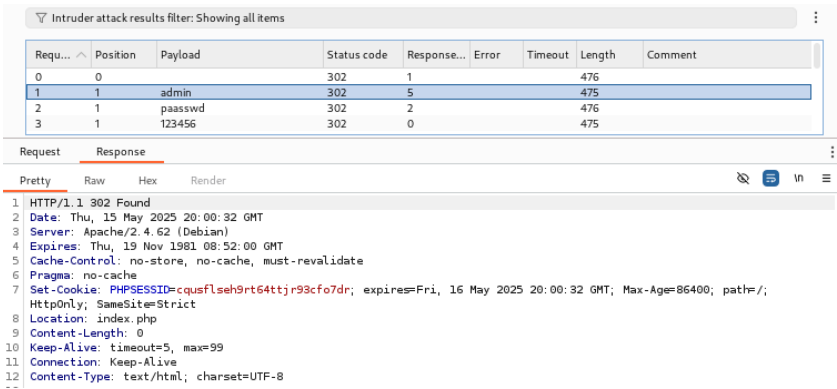


Рис. 3.20: Результат запроса

Дополнительная проверка с использованием Repeater, нажимаю на нужный мне запрос правой кнопкой мыши и жмем “Send to Repeater” (рис.21).



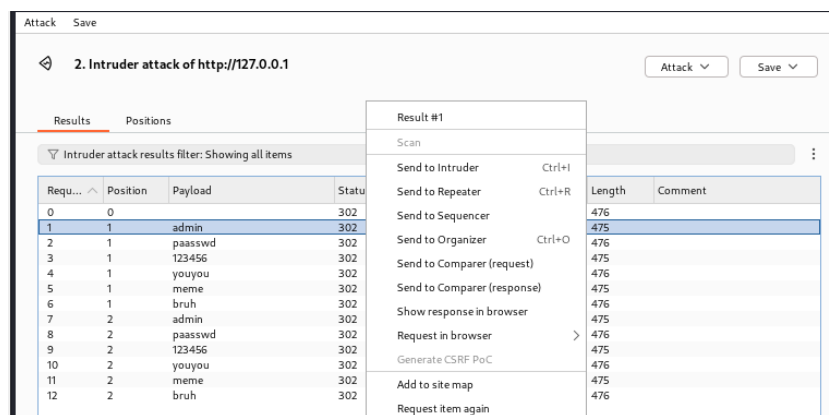


Рис. 3.21: Дополнительная проверка результата

Я перехожу во вкладку “Repeater”(рис.22).

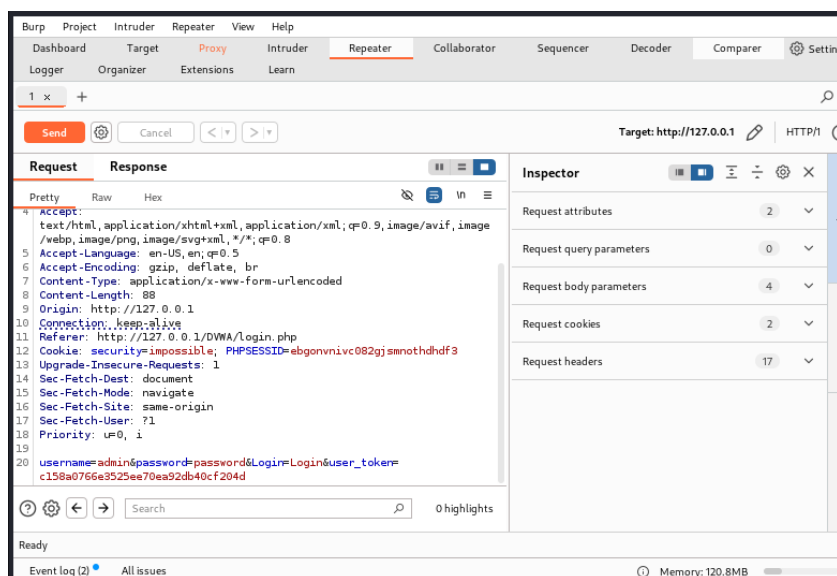


Рис. 3.22: Вкладка Repeater

Нажимаю”send”, получаю в Response в результат перенаправление на index.php(рис.23).

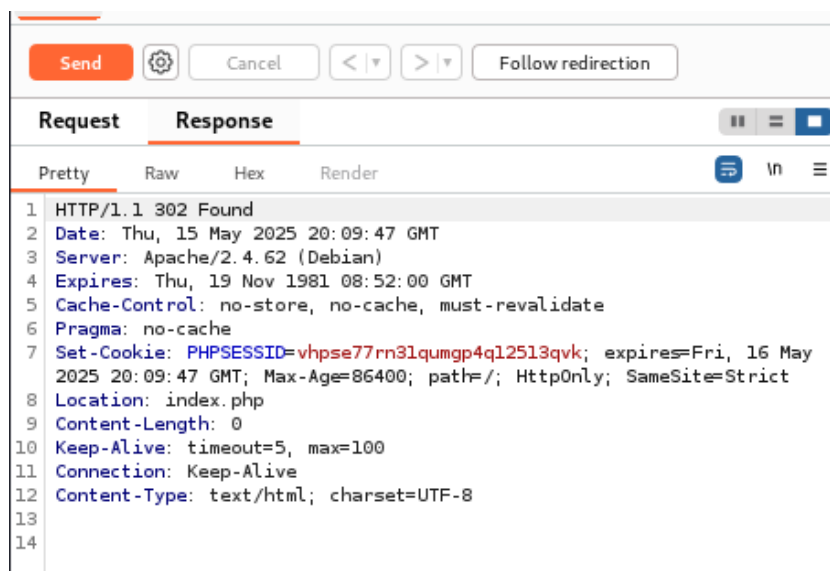


Рис. 3.23: Окно Response

После нажатия на Follow redirection, получим неcompiled html код в окне Response(рис.24).

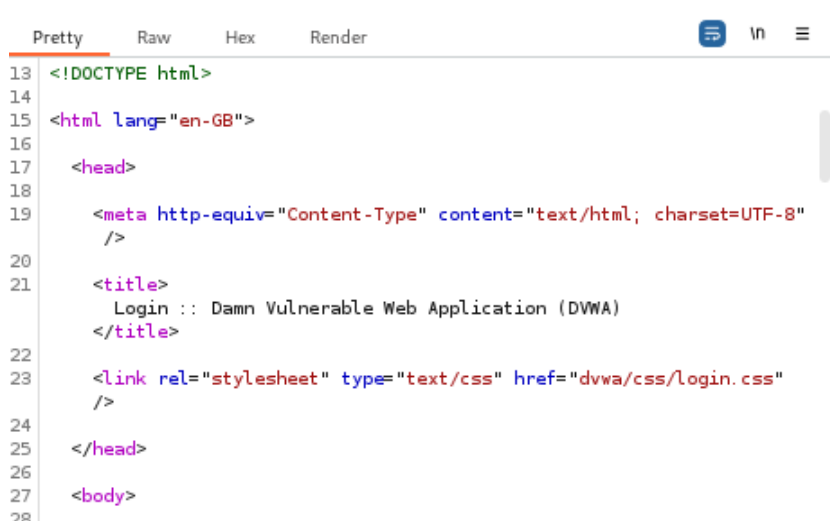


Рис. 3.24: Изменение в окне Response

Далее в подокне Render получаю то, как выглядит полученная страница (рис.25).

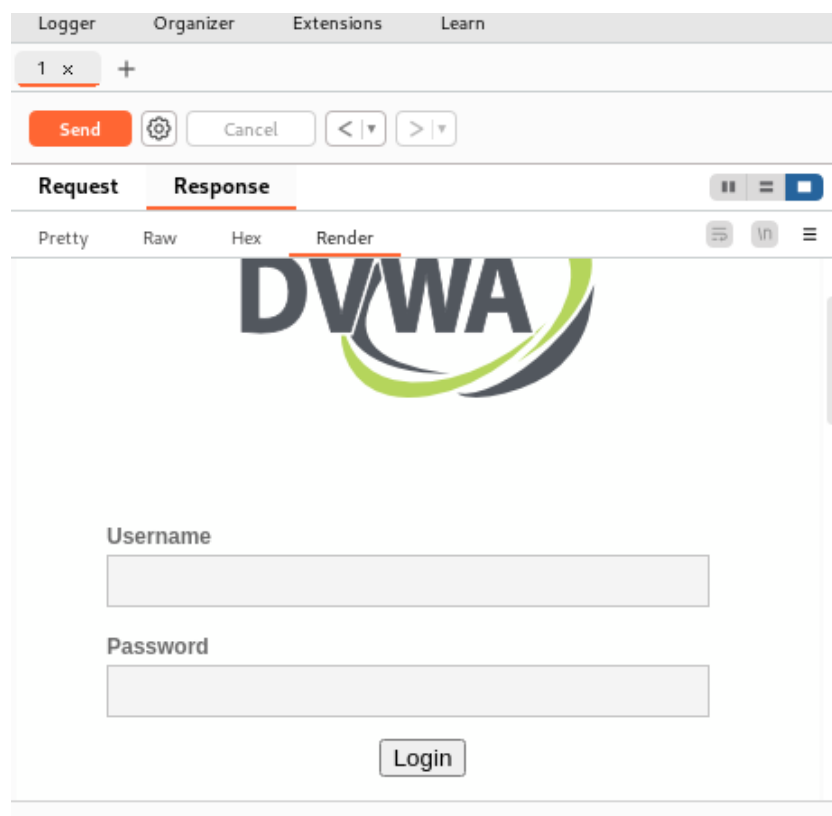


Рис. 3.25: Полученная страница

## 4 Выводы

При выполнении лабораторной работы научилась использовать инструмент Burp Suite.

## Список литературы

Ш. Парасрам Т.Х.и.др. А. Замм. Kali Linux: Тестирование на проникновение и безопасность: для профессионалов. Питер, 2022. 448 с.