

# **Основы информационной безопасности**

**Индивидуальный проект. Этап № 5. Использование Burp Suite**

Нзита Диатезилуа Катенди

# Содержание

Постановка задачи	4
Теоретические сведения	5
Выполнение лабораторной работы	8
Выводы	13
Список литературы	14

# Список иллюстраций

1	Установка ПО . . . . .	8
2	Создание проекта . . . . .	9
3	Установка параметров . . . . .	9
4	Включение Burp Proху . . . . .	10
5	Настройка HTTP Proху браузера . . . . .	10
6	Установка флага allow_hijacking_localhost . . . . .	10
7	Перехват запроса на вход на сайт . . . . .	11
8	Запрос на аутентификацию . . . . .	11
9	Функция повторения запроса . . . . .	11
10	Изучение ответа на запрос с функцией повторения запроса . . . . .	12

## Постановка задачи

Целью данной работы является использование Burp Suite для перехвата, изменения и изучения HTTP запросов и ответов.

# Теоретические сведения

Damn Vulnerable Web Application (DVWA) — это чрезвычайно уязвимое веб-приложение PHP/MySQL[~@dvwa]. Его основная цель — помочь специалистам по безопасности проверить свои навыки и инструменты в юридической среде, помочь веб-разработчикам лучше понять процессы безопасности веб-приложений, а также помочь студентам и преподавателям узнать о безопасности веб-приложений в контролируемой среде класса.

Цель DVWA — устранить некоторые наиболее распространенные веб-уязвимости различного уровня сложности с помощью простого и понятного интерфейса. В этом программном обеспечении имеются документированные и недокументированные уязвимости.

DVWA имеет три уровня безопасности, они меняют уровень безопасности каждого веб-приложения в DVWA:

- Невозможно. Этот уровень должен быть защищен от всех уязвимостей. Он используется для сравнения уязвимого исходного кода с безопасным исходным кодом.
- Высокий. Это расширение среднего уровня сложности с сочетанием более сложных или альтернативных плохих методов в попытке защитить код. Уязвимости не допускают такого широкого использования, как на других уровнях.
- Средний. Этот уровень безопасности в первую очередь предназначен для того, чтобы дать пользователю пример плохих методов обеспечения безопас-

ности, когда разработчик пытался обеспечить безопасность приложения, но потерпел неудачу.

- Низкий – этот уровень безопасности полностью уязвим и не имеет защиты. Он призван стать примером уязвимых веб-приложений, примером плохих методов программирования и служить платформой для изучения основных методов эксплуатации.

Burp Suite — интегрированная платформа для тестирования безопасности веб-приложений в ручном и автоматическом режимах[~@bs].

Пакет состоит из набора утилит, включая инструменты для сбора и анализа информации, моделирования различных типов атак, перехвата запросов и ответов от сервера и т.д.

- Target – создает карту сайта с подробной информацией о тестируемом приложении. Показывает, какие цели тестируются, и позволяет управлять процессом обнаружения уязвимостей.
- Прокси – находится между браузером пользователя и тестируемым веб-приложением. Он перехватывает все сообщения, передаваемые по протоколу HTTP(S).
- Spider – автоматически собирает данные о функциях и компонентах веб-приложения.
- Clickbandit – имитирует кликджекинг-атаки, при которых поверх страницы приложения загружается невидимая страница, подготовленная злоумышленниками.

— DOM Invader — проверяет веб-приложение на уязвимость к межсайтовому скриптингу на основе DOM (на основе объектной модели документа), внедряя на страницу вредоносный код.

- Сканер (в профессиональной и корпоративной редакциях) — автоматически сканирует веб-приложения на наличие уязвимостей. Он также существует в бесплатной версии, но там представлено лишь описание возможностей.

Intruder — осуществляет автоматические атаки различных типов, от перебора открытых веб-каталогов до SQL-инъекций.

— Повторитель — утилита для ручного манипулирования и перевыпуска отдельных HTTP-запросов и анализа ответов приложений. Запрос в Повторитель можно отправить из любой другой утилиты Burp Suite.

- Секвенсор — анализирует качество случайности в выборке элементов данных. Его можно использовать для тестирования токенов сеанса приложения или других важных элементов данных, которые, как ожидается, будут непредсказуемыми, таких как токены защиты от CSRF, токены сброса пароля и т. д.

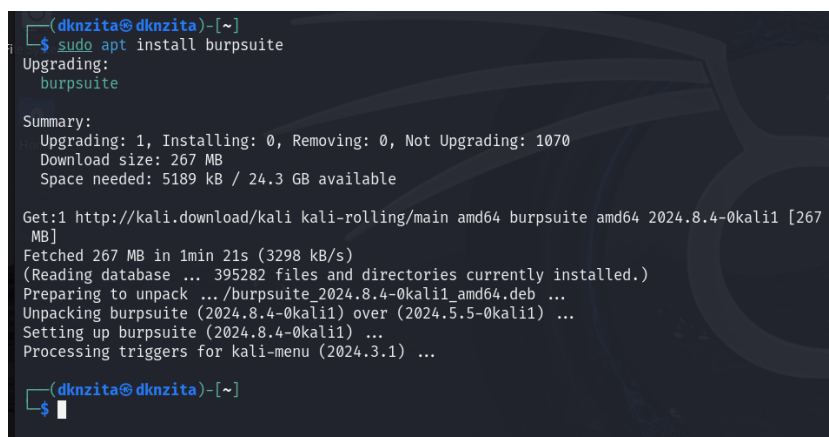
Декодер — преобразует закодированные данные в исходный формат или необработанные данные в различные хешированные и закодированные форматы. Вы можете распознавать различные форматы кодирования с помощью эвристики.

Компаратор — обеспечивает функцию визуального сравнения различий в данных.

# Выполнение лабораторной работы

## Intercept HTTP traffic with Burp Proxy

Установим Burp Suit с официального сайта(рис. @fig:001)

A terminal window showing the installation of Burp Suite. The user runs the command 'sudo apt install burpsuite'. The terminal output shows the package being upgraded, a summary of the installation (1 package to be installed, 267 MB download size, 5189 kB space needed), and the progress of downloading and unpacking the package. The installation is successful, and the terminal returns to the prompt.

```
(dknzita@dknzita)-[~]  
$ sudo apt install burpsuite  
Upgrading:  
  burpsuite  
  
Summary:  
  Upgrading: 1, Installing: 0, Removing: 0, Not Upgrading: 1070  
  Download size: 267 MB  
  Space needed: 5189 kB / 24.3 GB available  
  
Get:1 http://kali.download/kali kali-rolling/main amd64 burpsuite amd64 2024.8.4-0kali1 [267 MB]  
Fetched 267 MB in 1min 21s (3298 kB/s)  
(Reading database ... 395282 files and directories currently installed.)  
Preparing to unpack .../burpsuite_2024.8.4-0kali1_amd64.deb ...  
Unpacking burpsuite (2024.8.4-0kali1) over (2024.5.5-0kali1) ...  
Setting up burpsuite (2024.8.4-0kali1) ...  
Processing triggers for kali-menu (2024.3.1) ...  
  
(dknzita@dknzita)-[~]  
$
```

Рис. 1: Установка ПО

Откроем приложение и создадим временный проект с параметрами по умолчанию(рис. @fig:002).



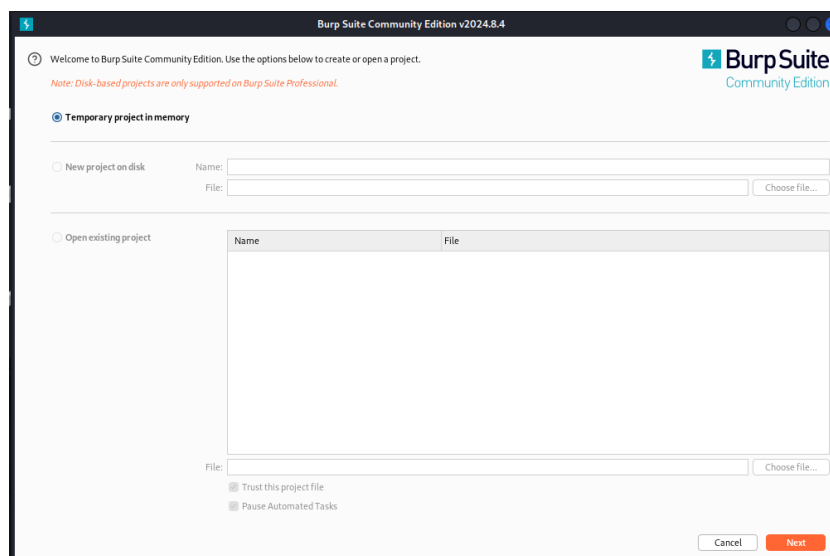


Рис. 2: Создание проекта

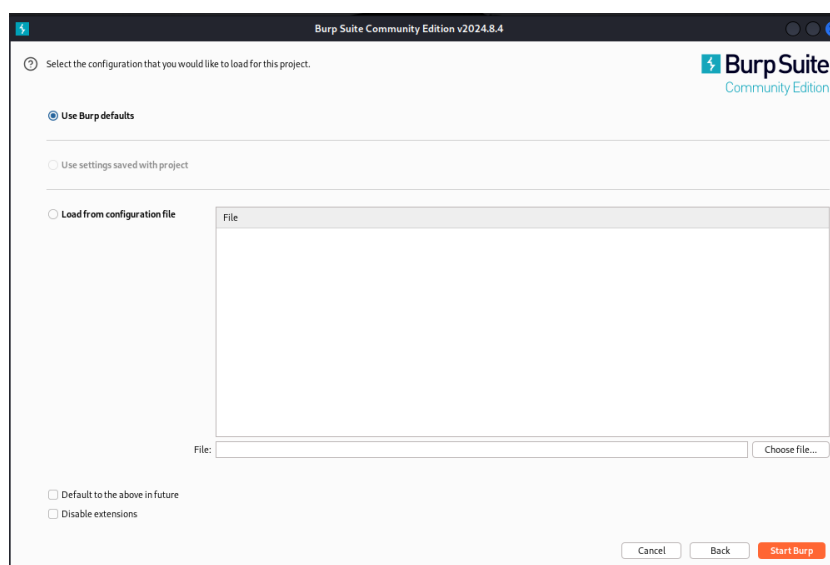


Рис. 3: Установка параметров

Теперь попробуем перехватить http запрос с помощью Burp Проxy. Включим перехват, а в браузере включим прокси и укажем для него адрес локального хоста, а также установим параметр, разрешающий перехват запросов локального хоста(рис. @fig:004 - @fig:006).

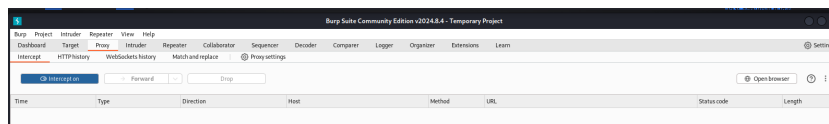


Рис. 4: Включение Burp Proxy

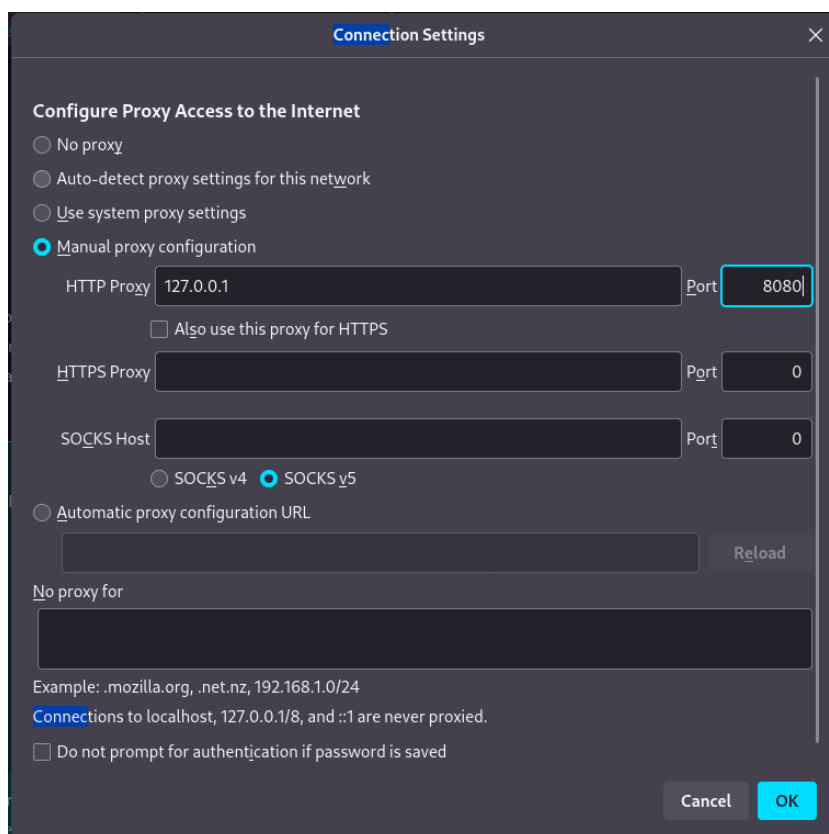


Рис. 5: Настройка HTTP Proxy браузера

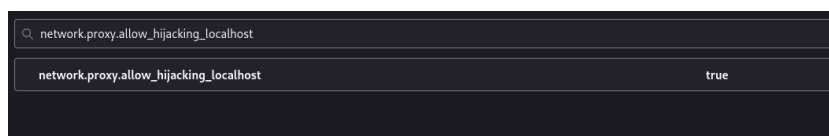


Рис. 6: Установка флага allow\_hijacking\_localhost

Можем увидеть первый перехваченный запрос: вход на сайт DVWA. Указаны адрес локального хоста, версия браузера, ОС устройства и другая информация(рис.

@fig:007):

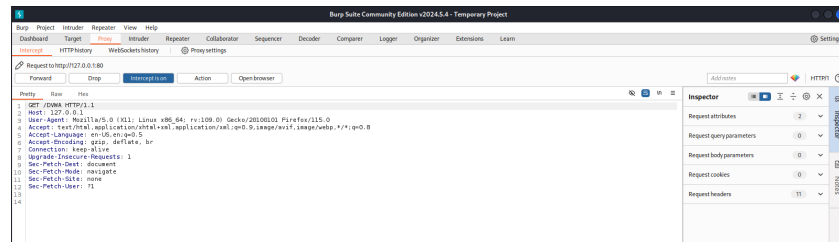


Рис. 7: Перехват запроса на вход на сайт

Рассмотрим перехват запроса аутентификации(рис. @fig:008):

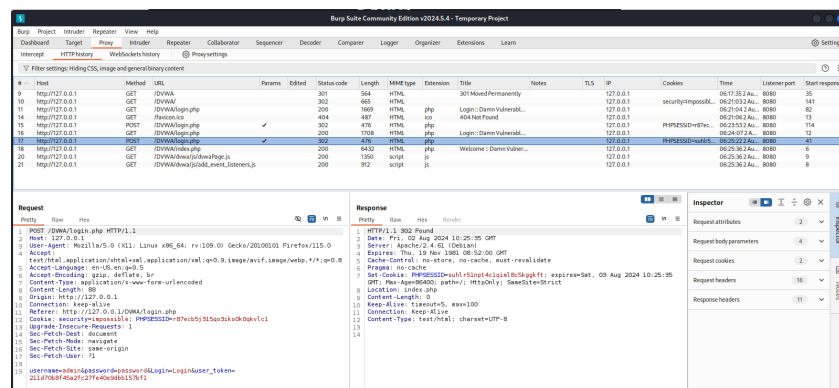


Рис. 8: Запрос на аутентификацию

Здесь дополнительно указываются куки запроса, а также выдается сам запрос с указанием введенного имени пользователя и пароля.

Кроме того уже совершенный запрос можно отпправить на повтор для того чтобы изучить ответы(рис. @fig:009):

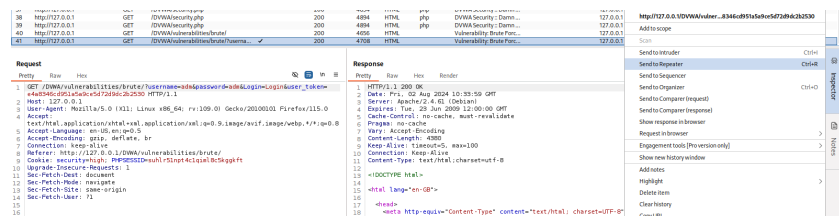


Рис. 9: Функция повторения запроса

В запросах можно изменять вводимую информацию и сравнивать ответы(рис. @fig:010):

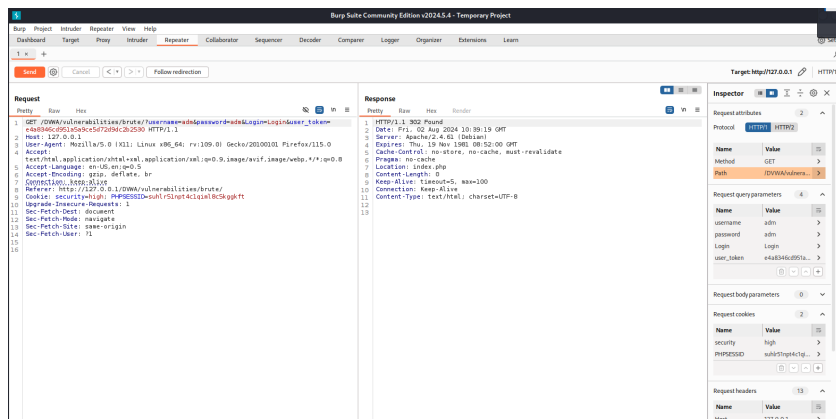


Рис. 10: Изучение ответа на запрос с функцией повторения запроса

## **Выводы**

В результате выполнения работы научились на практике использовать ПО Burp Suit для перехвата, изменения и изучения HTTP запросов и ответов.

## Список литературы

.....