Индивидуальный проект - Этап 5

Основы информационной безопасности

Чистов Даниил Максимович

Содержание

Список иллюстраций

1 Цель работы

Получение навыков пользование Burp Suite.

1.1 Введение

Burp Suite - инструмент для тестирования безопасности веб-приложений, позволяющий множеством функций перехватывать, анализировать, модифицировать разные HTTP-запросы между клиентом и сервером.

2 Выполнение лабораторной работы

Запускаю Burp Suite, прохожу через пару диалоговых окон, где спрашивают, как будет устроен проект, над которым мы будем работать (рис. 2.1).

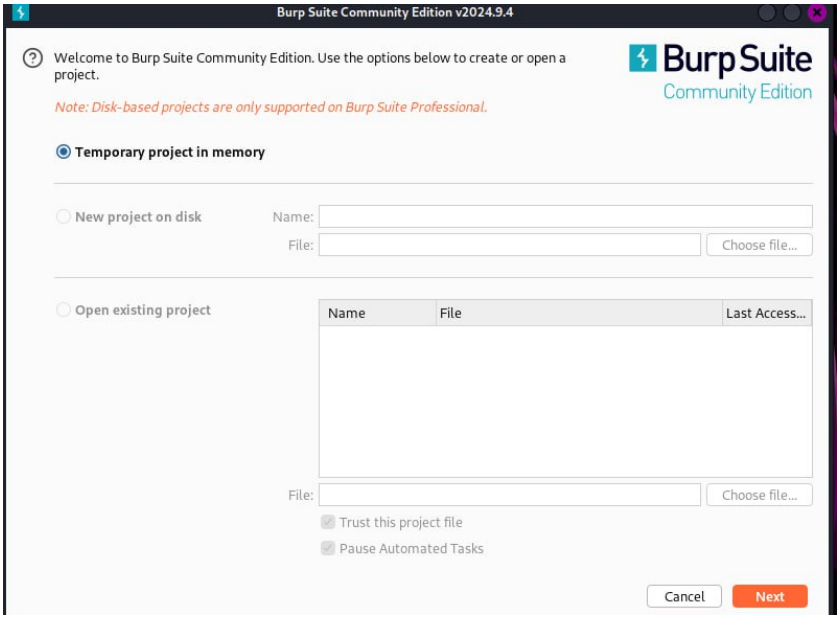

Рис. 2.1: Инициализация Burp Suite

Открываю встроенный в Burp Suite браузер и открываю в нём DVWA - всё как обычно (рис. 2.2).

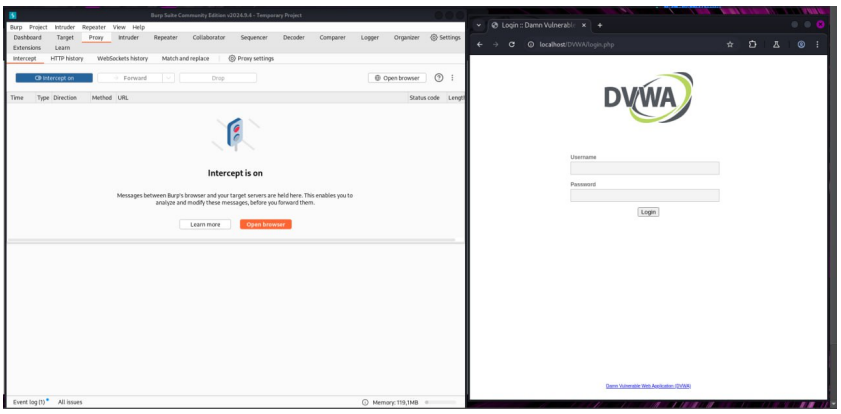

Рис. 2.2: DVWA через встроенный браузер

Перед работой надо запустить apache2 и mysql, буду тестировать Burp Suite на dvwa - брут форс пароля, как в этапе про Hydra, только в этот раз у DVWA будет уровень защиты “Высокий” (рис. 2.3).

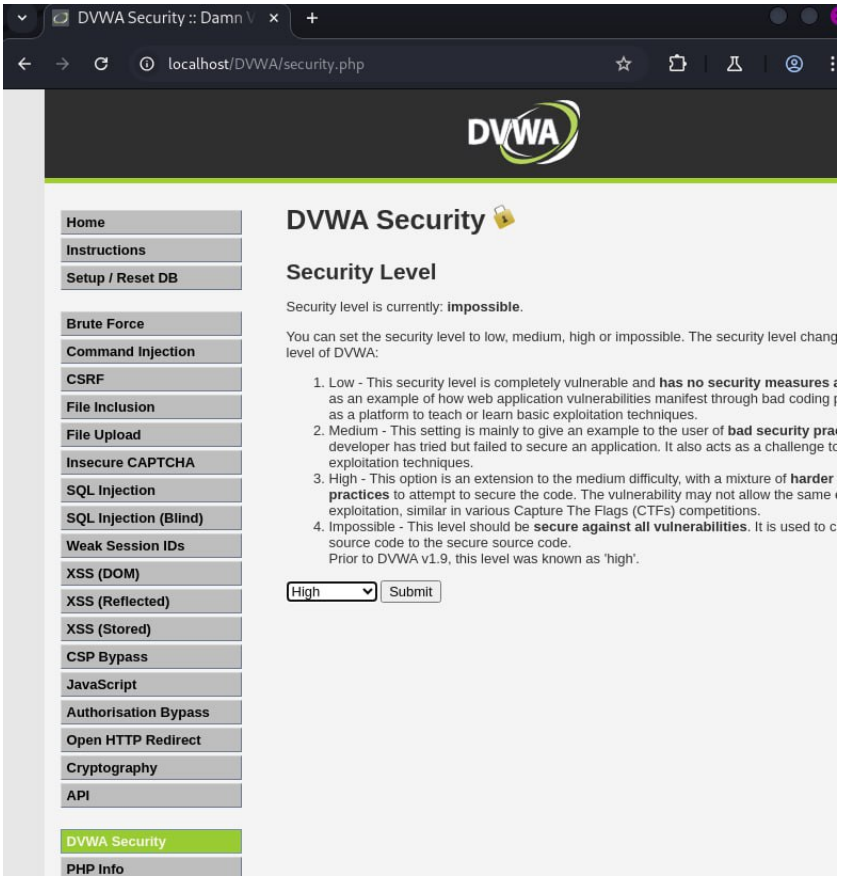

Рис. 2.3: Высокий уровень защиты DVWA

Перехожу на страничку Brute Force DVWA, там есть кнопку view source, которая позволяет посмотреть код данной странички. Такая страничка различается на разных уровнях сложности - на уровне сложности High появляется user_token, который совсем чуток усложняет брут форс (рис. 2.4).

vulnerabilities/brute/source/high.php

```
<?php

if( isset( $_GET[ 'Login' ] ) ) {
    // Check Anti-CSRF token
    checkToken( $_REQUEST[ 'user_token' ], $_SESSION[ 'session_token' ], 'index.php' );

    // Sanitise username input
    $user = $_GET[ 'username' ];
    $user = stripslashes( $user );
    $user = ((isset($GLOBALS["___mysqli_ston"]) && is_object($GLOBALS["___mysqli_ston"])) ? mysqli_real_escape_string($GLOBALS["_
[MySQLConverterToo] Fix the mysql_escape_string() call! This code does not work.", E_USER_ERROR)) ? "" : "));

    // Sanitise password input
    $pass = $_GET[ 'password' ];
    $pass = stripslashes( $pass );
    $pass = ((isset($GLOBALS["___mysqli_ston"]) && is_object($GLOBALS["___mysqli_ston"])) ? mysqli_real_escape_string($GLOBALS["_
[MySQLConverterToo] Fix the mysql_escape_string() call! This code does not work.", E_USER_ERROR)) ? "" : "));
    $pass = md5( $pass );

    // Check database
    $query  = "SELECT * FROM `users` WHERE user = '$user' AND password = '$pass';";
    $result = mysqli_query($GLOBALS["___mysqli_ston"],  $query ) or die( '<pre>' . ((is_object($GLOBALS["___mysqli_ston"])) ? mys

    if( $result && mysqli_num_rows( $result ) == 1 ) {
        // Get users details
        $row    = mysqli_fetch_assoc( $result );
        $avatar = $row["avatar"];

        // Login successful
        echo "<p>Welcome to the password protected area {$user}</p>";
        echo "<img src=\"{$avatar}\" />";
    }
    else {
        // Login failed
        sleep( rand( 0, 3 ) );
        echo "<pre><br />Username and/or password incorrect.</pre>";
    }

    ((is_null($___mysqli_res = mysqli_close($GLOBALS["___mysqli_ston"]))) ? false : $___mysqli_res);
}

// Generate Anti-CSRF token
generateSessionToken();

?>
```

Рис. 2.4: Код странички входа на сложности High

2.0.1 О user_token

При каждом обновлении страницы меняется и user_token (а страница будет много обновляться при множестве неудачных попыток брут форса), сервер в свою очередь не пропускает реквесты, у которых уже устарел user_token, т.е. взломщику нужно придумать способ, как этот user_token получать автоматически при каждой попытке брут форса.

Идём далее, с помощью Burp Suite мы можем автоматизировать процесс нахождения user_token (он вшит в страничку). Захожу в настройки и во вкладке Sessions создаю новое правило (рис. 2.5).

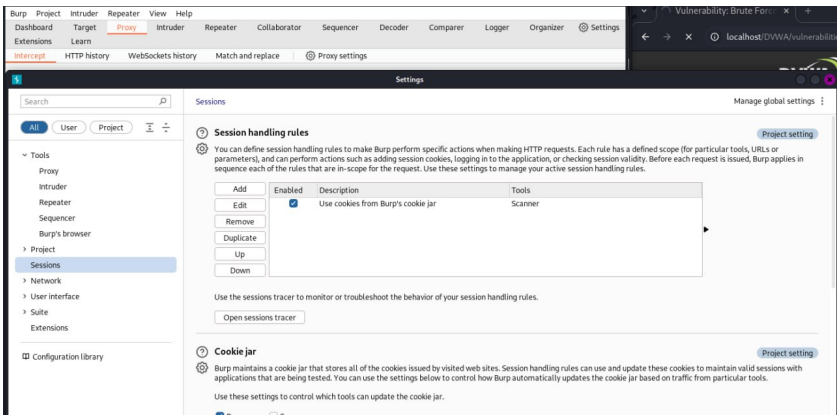

Рис. 2.5: Новое правило в Burp Suite

В новом правиле мы добавляем новое “макро действие”, и затем настраиваем его - открывается Macro Recorder, где мы выбираем наш последний реквест - попытку входа в DVWA, оттуда мы можем посмотреть на наш реквест в виде кода и найти строчки с user_token (рис. 2.6).

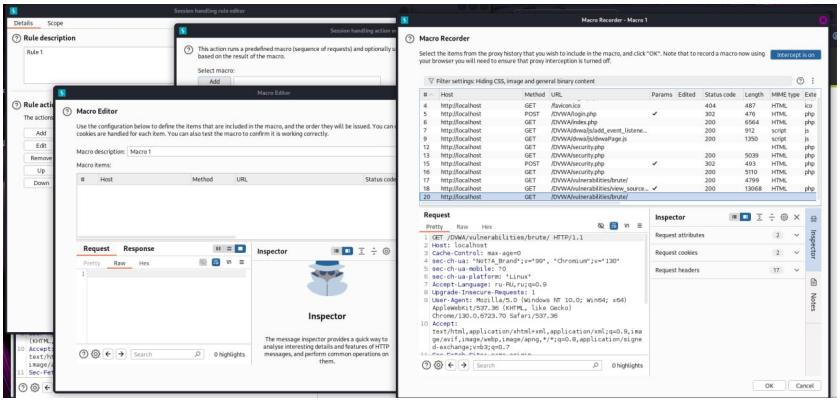

Рис. 2.6: Новое макро действие в Burp Suite

В открытом коде реквеста находим нужный параметр, за которым мы будем следить и запоминать - user_token (рис. 2.7).

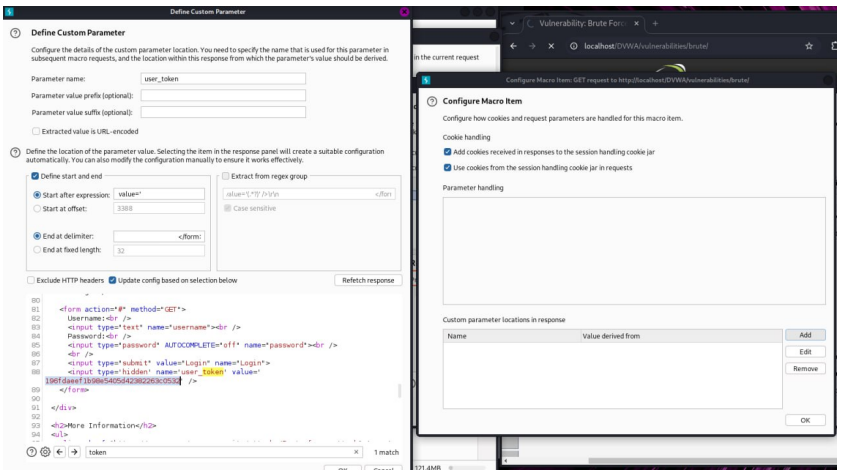

Рис. 2.7: Отслеживание параметра user_token в каждом реквесте

Сохраняем наше макро действие, ставим галочку “Tolerate URL mismatch when matching parameters (Use for URL-agnostic CRSF tokens)” - тут написано ставить, если мы имеем дело с юзер токенами (рис. 2.8).

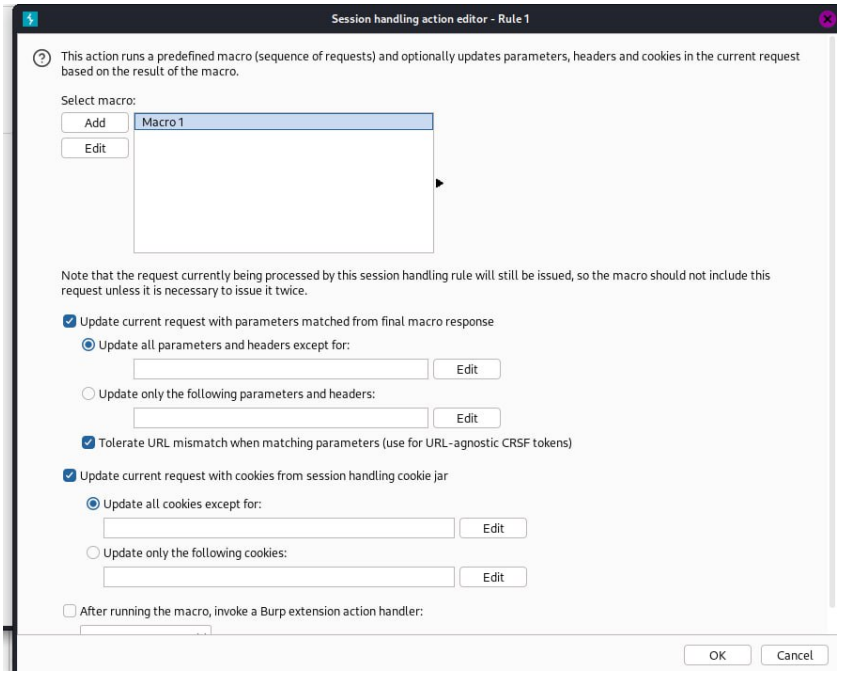

Рис. 2.8: Сохранение макро действия

Возвращаемся в настройку правила, выставляем галочки так, чтобы это правило применялось исключительно к инструменту Intruder - им мы будем пользо-

ваться для брут форса приложения (рис. 2.9).

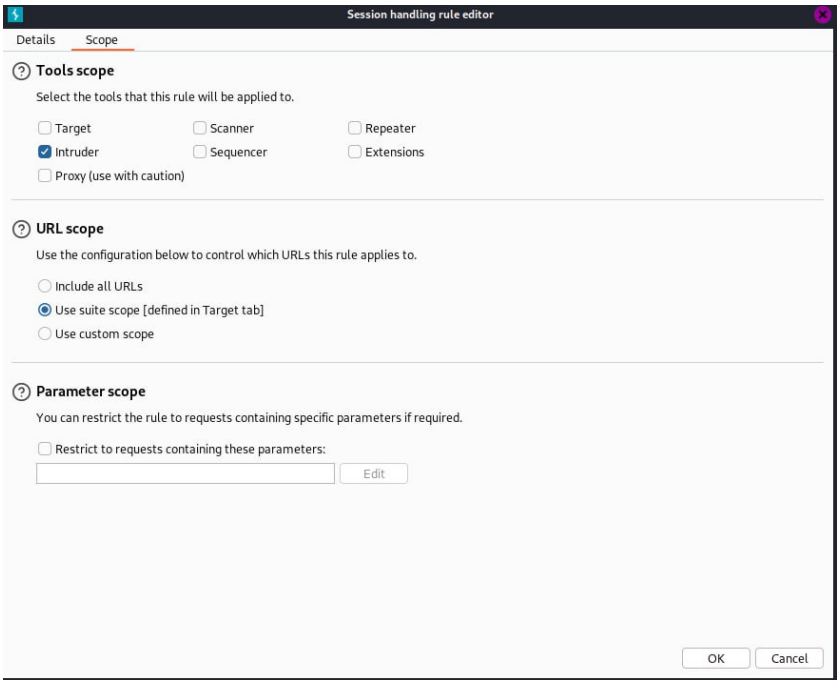

Рис. 2.9: Сохранение нового правила

Начнём. Включаем Intercepter - перехватываем реквест с попыткой входа (рис. 2.10).

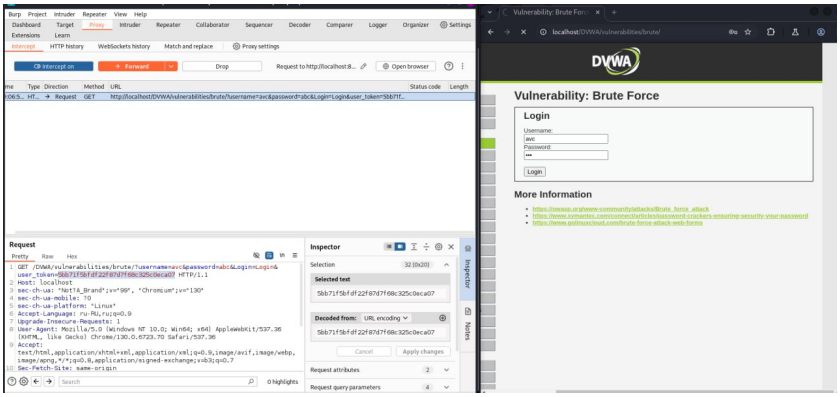

Рис. 2.10: Перехват реквеста с попыткой входа

Открываем вкладку HTTP-history и находим перехваченный реквест, нажимаем на него правой кнопкой и “Send to Intruder” (отправляем в инструмент взломащика), а затем “Add to scope” (рис. 2.11).

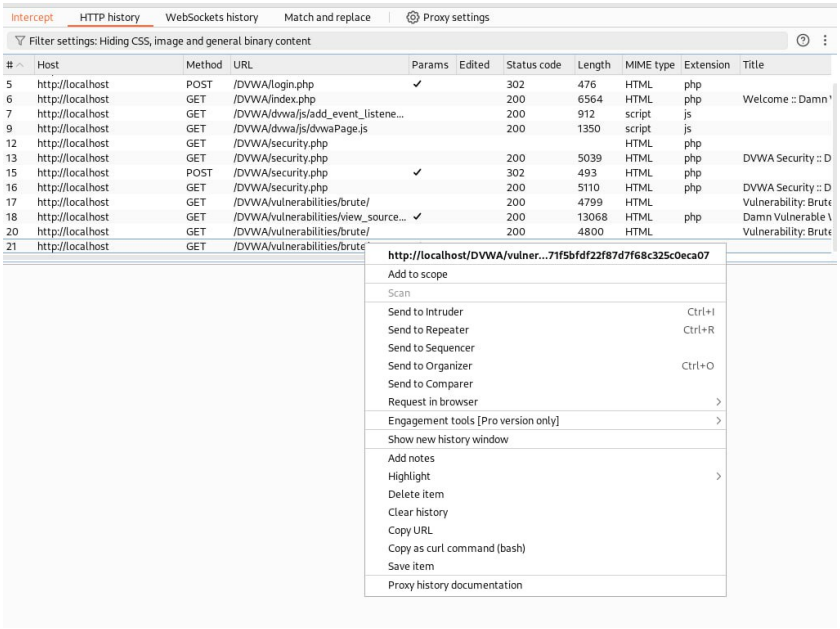

Рис. 2.11: Отправляем реквест взломщику

Теперь открываем вкладку “Intruder” - находим посланный нами реквест, выбираем тип атаки “Cluster Bomb” - стандартный брут форс - постоянный перебор и отправка реквестов, также выделяем значения параметра username и нажимаем “Add $”, так мы выделили первый параметр, который мы будем перебирать и посылать каждый реквест - аналогично делаем и со значением переменной password (рис. 2.12).

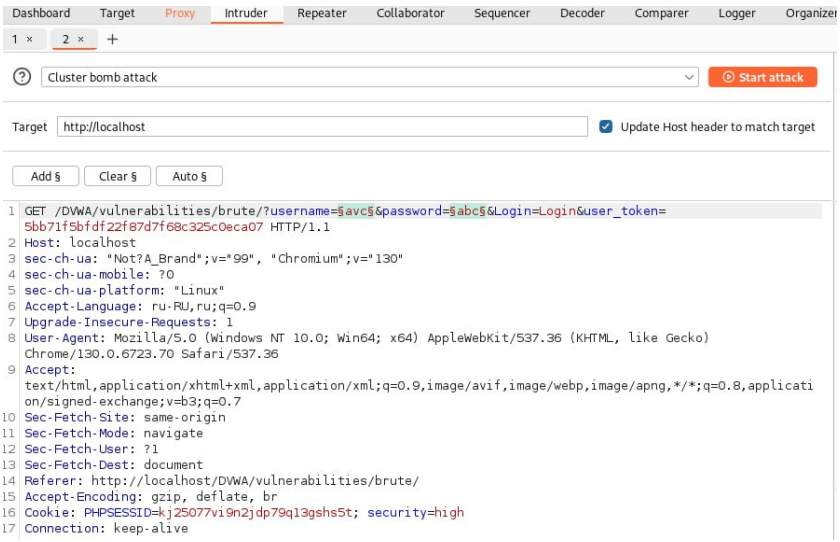

Рис. 2.12: Выделяем переменные для подбора пароля и логина

Открываем Payloads (тут мы настраиваем переменные, которые будем перебирать, т.к. мы перебираем логин и пароль, у нас их 2). Первый пейлоуд - выбираем, что будем перебирать: значения из файла, выбираем файл - в Kali есть стандартный список дефолтных логинов и паролей - они лежат в /usr/share/wordlists/metasploit. Для списка логинов выбираем http_default_users.txt (рис. 2.13).

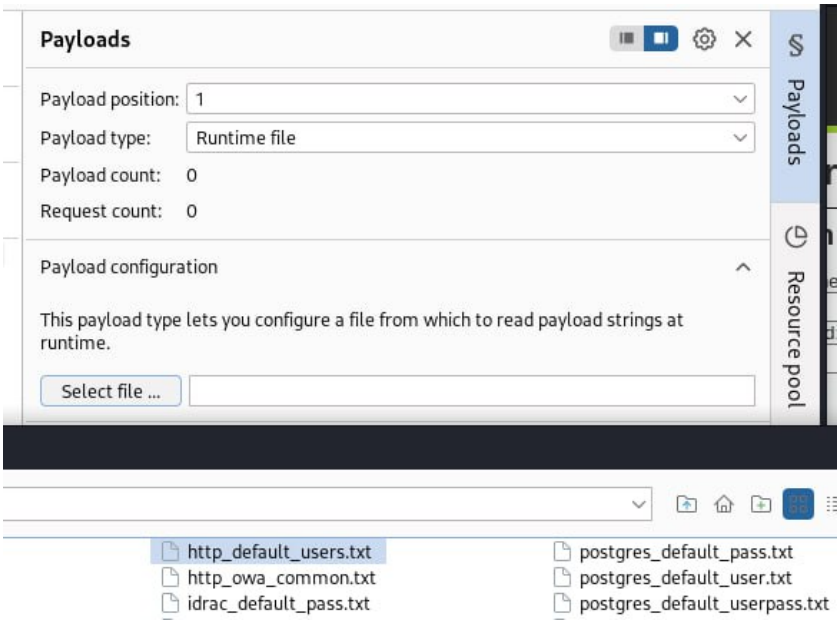

Рис. 2.13: Выбираем файл для перебора логинов

Аналогично делаем и для второго пейлоуда - перебор паролей - http_default_pass.txt (рис. 2.14).

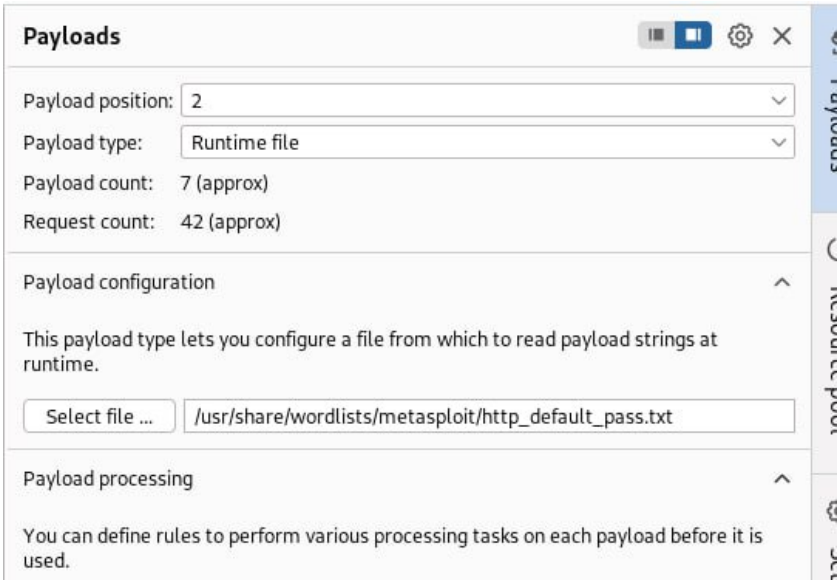

Рис. 2.14: Выбираем файл для перебора паролей - http_default_pass.txt

Открываем настройки Intruder, для наглядности добавим слово, за которым мы будем следить, и если оно появляется в коде странички - то мы ставим нашему реквесту флажок. Выбираем слово “incorrect”, тогда мы обратим внимание, что при правильном наборе логина и пароля флажка не будет (рис. 2.15).

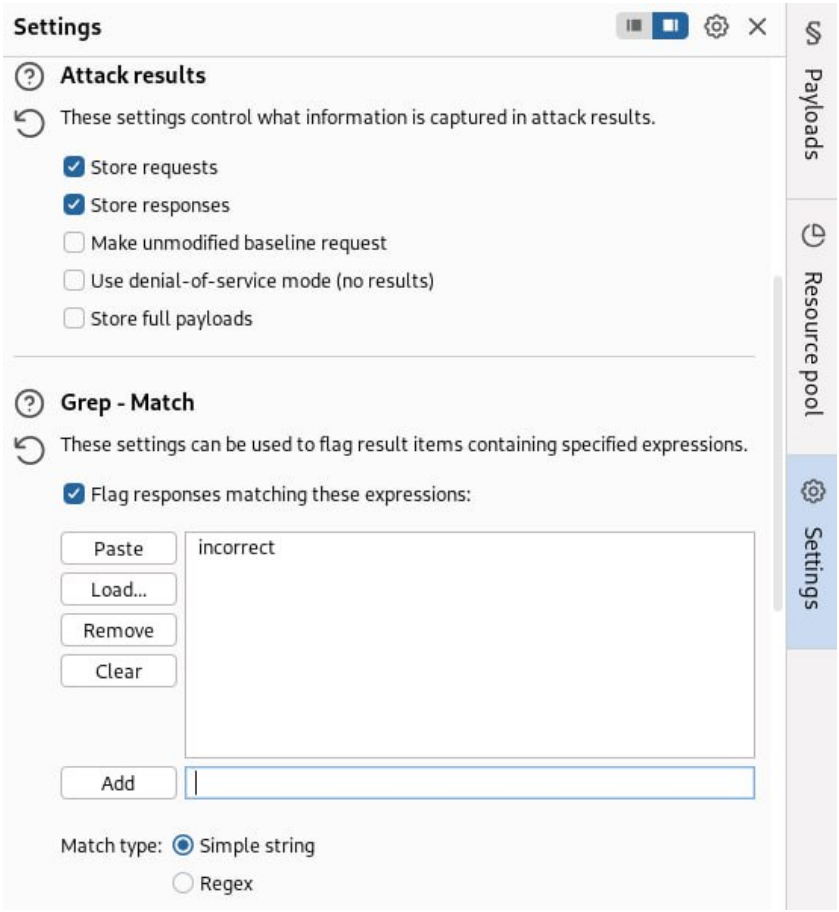

Рис. 2.15: Ставим маркер на слово incorrect

Запускаем нашего атакующего - начинаем брут форс. Наглядно видно, как посылается много реквестов. На фото я также их отсортировал по длине кода в страчничке. Обратим внимание, что тут в первой строке при логине admin и пароле password мало того, нету флажка Incorrect, так ещё и длина кода страничке значительно отличается от всех остальных - явно что-то особенное случилось при таком наборе логина и пароля. Обычно, взломщик в таком случае сам попробует такой набор логина и пароля (рис. 2.16).

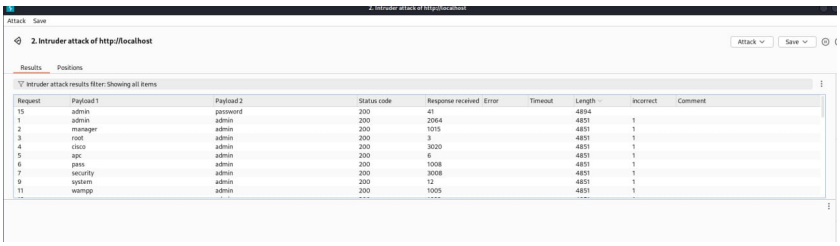

Рис. 2.16: Задокументированная брут форс атака

Вставляем такую комбинацию логина и пароля в страчнику входа и видим, что мы успешно прорвались в чужок аккаунт (рис. 2.17).

Рис. 2.17: Атака прошла успешно

3 Выводы

При выполнении данной работы я успешно получил навыки работы с Burp Suite.

4 Список литературы

Индивидуальный проект

Brute Force DVWA разной сложности с использованием Burp Suite (На английском)