Основы информационной безопасности

Индивидуальный проект. Этап №3. Использование Hydra

Нзита Диатезилуа Катенди

Содержание

# Постановка задачи

Целью данной работы являестся использование Hydra для подбора пароля.

# Теоретические сведения

Damn Vulnerable Web Application (DVWA) – это веб-приложение PHP/MySQL, которое чертовски уязвимо[~@dvwa]. Его основная цель – помочь специалистам по безопасности проверить свои навыки и инструменты в правовой среде, помочь веб-разработчикам лучше понять процессы обеспечения безопасности веб-приложений, а также помочь студентам и преподавателям узнать о безопасности веб-приложений в контролируемом классе. обстановка помещения.

DVWA (Damn Vulnerable Web Application) — это приложение, предназначенное для практики работы с распространёнными веб-уязвимостями, предлагая пользователю интуитивно понятный интерфейс. Оно содержит как документированные, так и недокументированные уязвимости, позволяя исследовать безопасность веб-приложений на разных уровнях сложности.

Некоторые из уязвимостей, представленных в DVWA, включают:

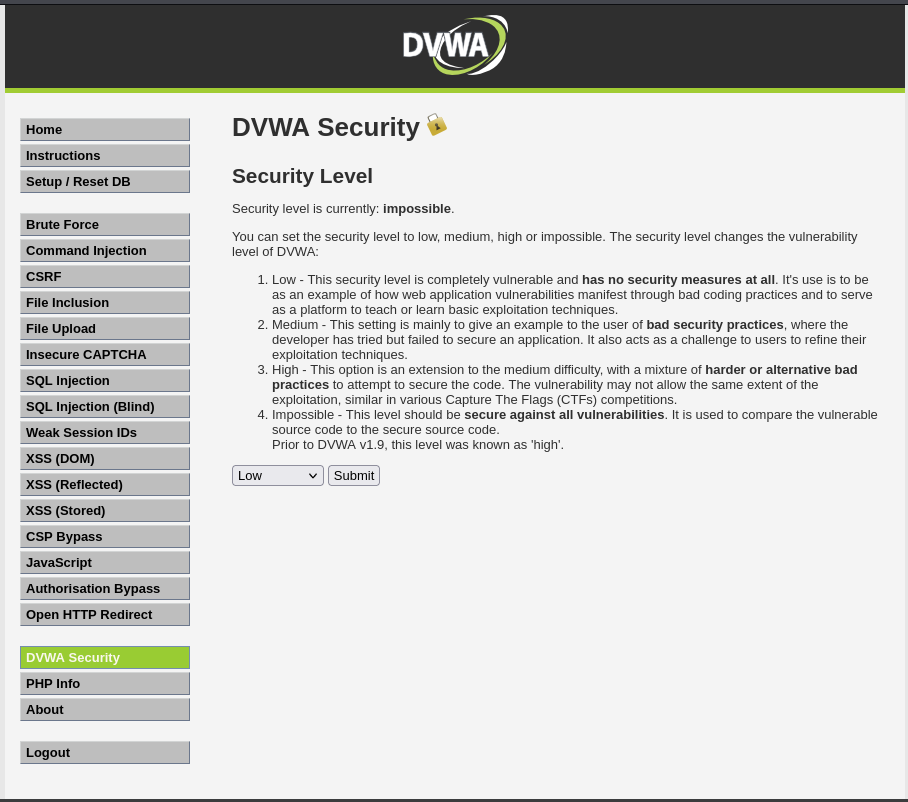
* **Брутфорс**: Атака на формы входа, используемая для тестирования инструментов, позволяющих подбирать пароли, и демонстрации уязвимости слабых паролей.
* **Выполнение команд**: Позволяет злоумышленнику исполнять команды на уровне операционной системы.
* **Межсайтовая подделка запроса (CSRF)**: Позволяет злоумышленнику изменять пароль администратора.
* **Внедрение файлов**: Злоумышленник может подключать удалённые или локальные файлы к веб-приложению.
* **SQL-внедрение**: Позволяет вставлять SQL-код в запросы через поля ввода, включая слепое и основанное на ошибках внедрение.
* **Небезопасная выгрузка файлов**: Позволяет загружать вредоносные файлы на сервер.
* **Межсайтовый скриптинг (XSS)**: Злоумышленник может внедрять свои скрипты в веб-приложение или базу данных, включая отражённые и сохранённые XSS.
* **Пасхальные яйца**: Раскрытие путей к файлам, обход аутентификации и другие уязвимости.

DVWA предлагает четыре уровня безопасности, которые меняют уязвимость веб-приложений:

* **Невозможный**: Уровень безопасности, при котором приложение защищено от всех уязвимостей. Используется для сравнения уязвимого кода с безопасным.
* **Высокий**: Уровень сложности с элементами более сложных и альтернативных плохих практик, который снижает возможности эксплуатации.
* **Средний**: Уровень, показывающий примеры плохих практик безопасности, где разработчик пытался обеспечить безопасность, но не смог.
* **Низкий**: Полностью уязвимый уровень, предназначенный для демонстрации плохих практик программирования и обучения базовым методам эксплуатации.

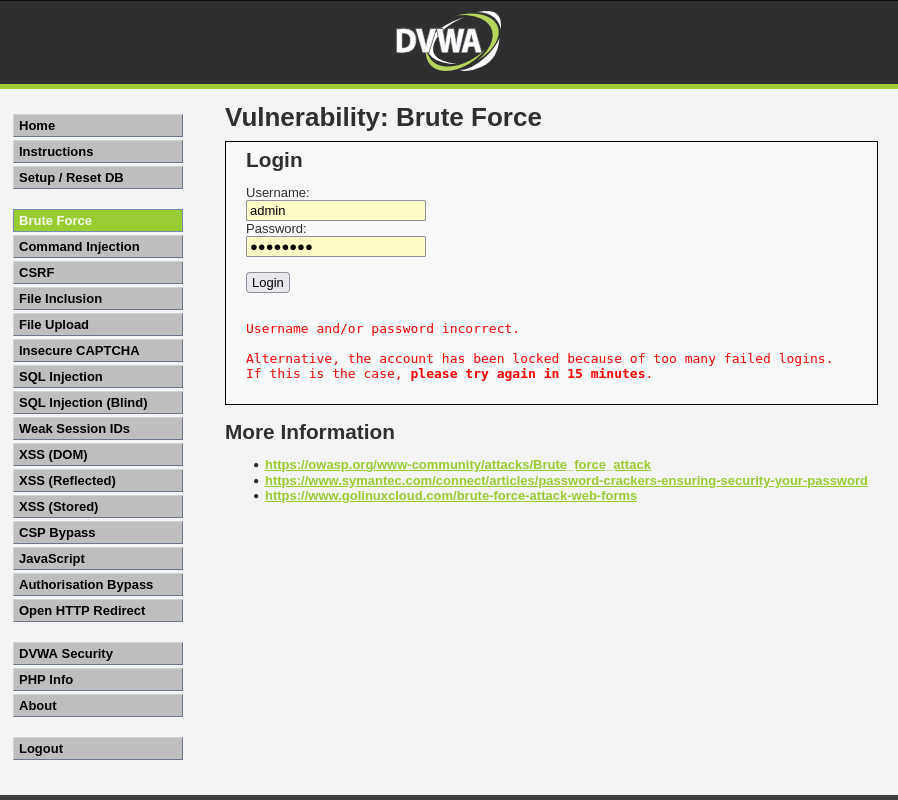
# Выполнение лабораторной работы

Установим самый низкий уровень защиты DVWA (рис. @fig:001)



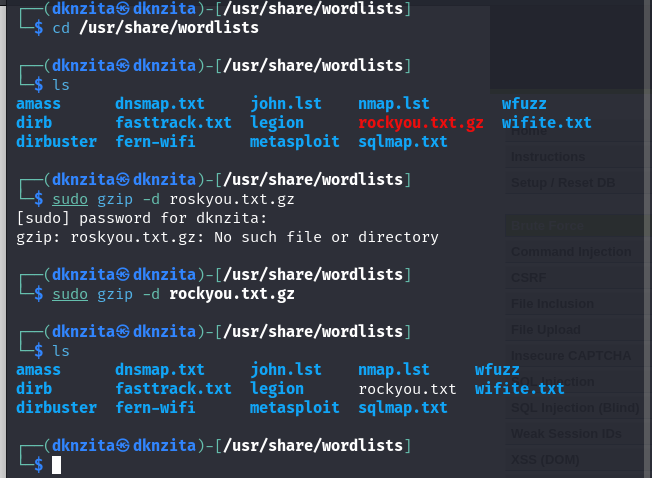
Уровень защиты DVWA

Откроем страницу для проведения атаки brute force, которая представляет собой простейшую уязвимую форму с паролем (рис. @fig:002).



Уязвимая форма для ввода пароля

В Kali лежит файл с найболее популярными паролями, который мы распакуем(рис. @fig:003).



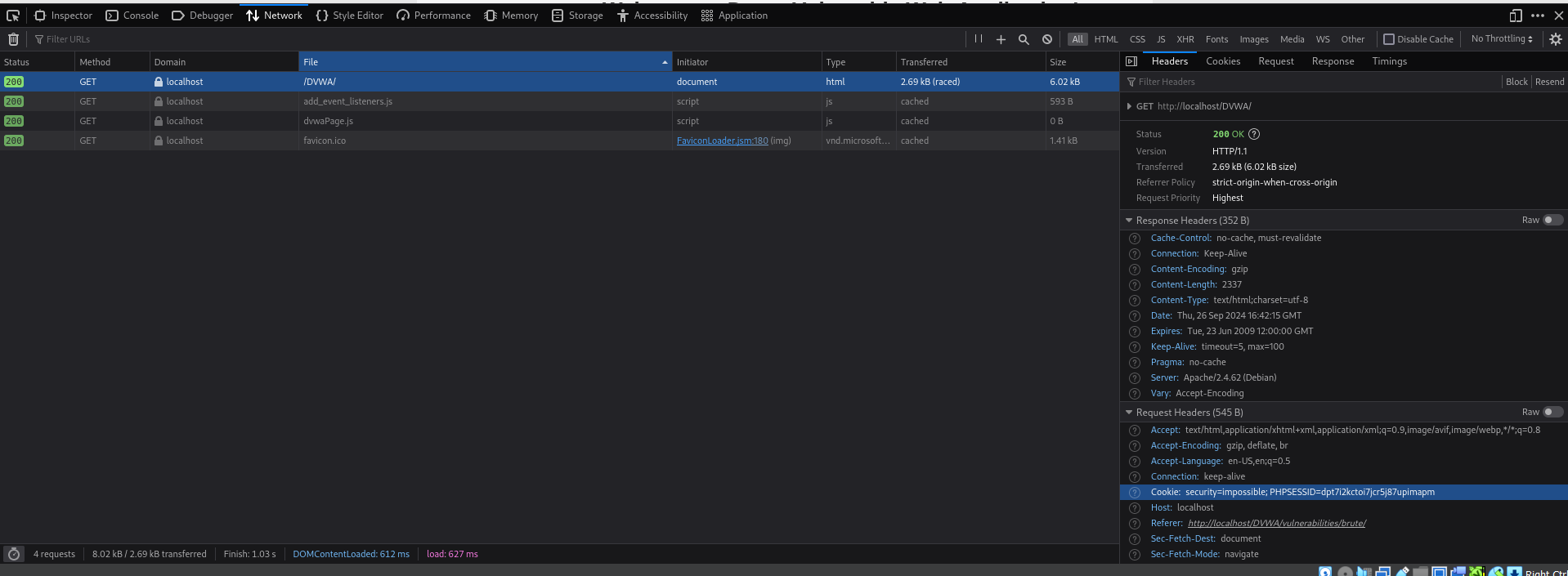
Распаковка rockyou.txt.gz

Можно увидеть, что уже в начале есть пароль, который установлен по умолчанию для нашего аккаунта(рис. @fig:004, @fig:005).



файл rockyou.txt. с наиболее популярными паролями

Рассмотрим данные о запросе на вход(рис. @fig:005).

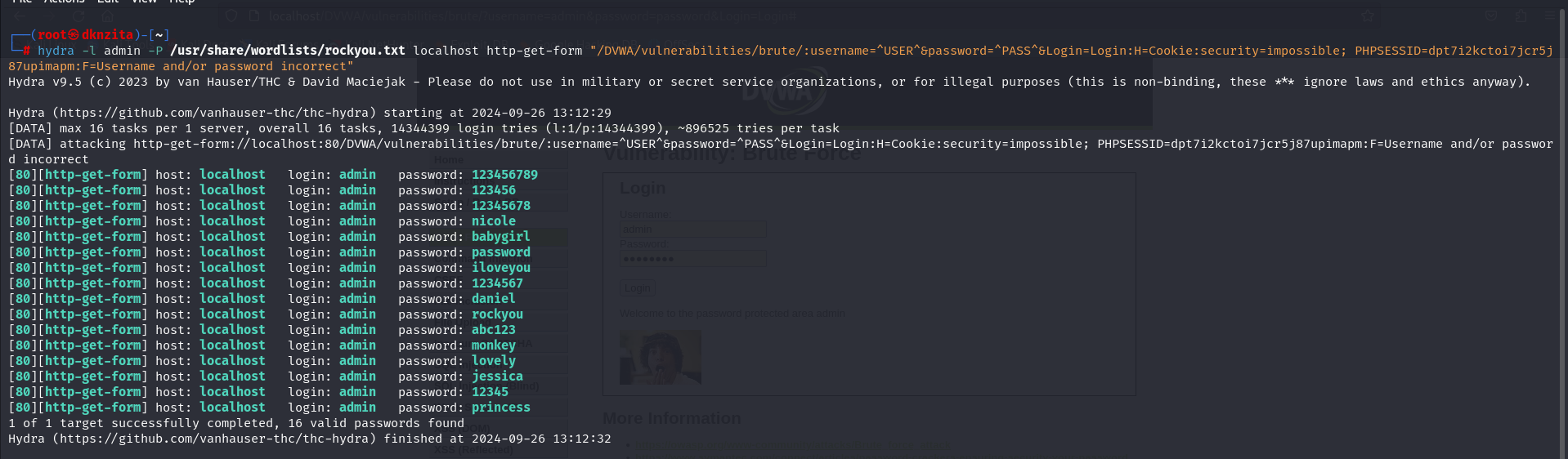


Данные о запросе на вход

Исходные данные:

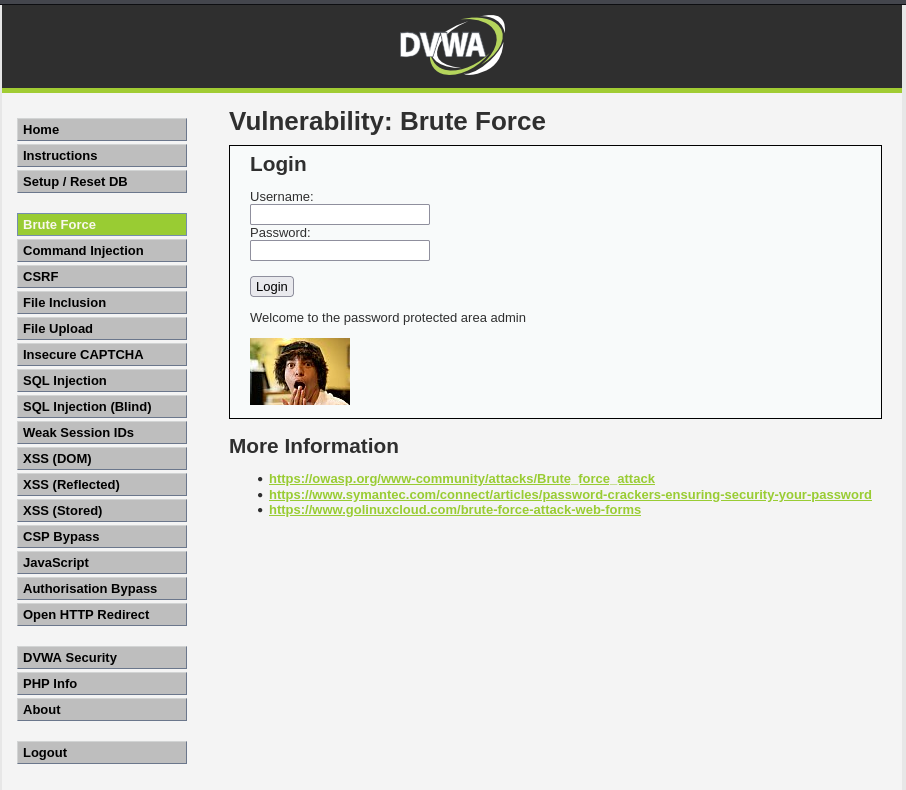
* IP сервера 127.0.0.1(localhost);
* Сервис http на стандартном 80 порту;
* Для авторизации используется html форма, которая отправляет по адресу http://localhost/DVWA/vulnerabilities/brute методом GET запрос вида username=admin&password=test\_password;
* В случае неудачной аутентификации пользователь наблюдает сообщение Username and/or password incorrect.

Запрос к Hydra будет выглядеть так(рис. @fig:006):



Запрос к Hydra

В результате получим нужный пароль(рис. @fig:007):



Проверка полученного пароля

# Выводы

В результаты выпольнения работы была использована Hydra, для атаки типа brute force.

# Список литературы

:::::: :::