

## Задание 1.

В зависимости от того, где, когда, какими силами во время разработки/жизни ресурса проводится исследование, соответственно заданным условиям и будет предоставляться отчёт об уязвимости ресурса. За основу, в том числе в методичке, рассмотрена форма исследования на базе Microsoft's SDL (Security Development Lifecycle). В данном примере подобного рода отчёт не будет иметь смысла, т.к. ресурс уже в стадии эксплуатации. Поэтому можно будет воспользоваться любым более-менее адекватным шаблоном для составления подобного рода отчёта.

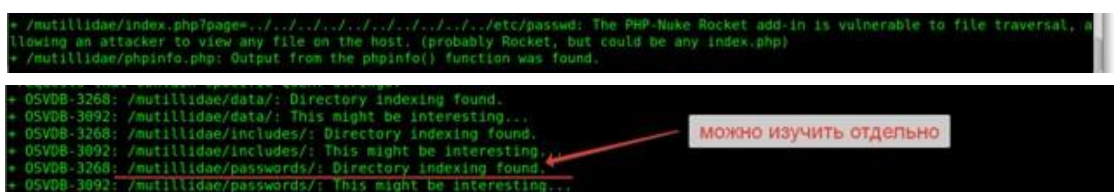
Рассматриваемый ресурс:

<http://192.168.56.11/mutillidae>

Используемое ПО:

Nikto (Open Source ( PL) веб-сервер сканер).

Детали обнаружения и воспроизведения:



```
+ /mutillidae/index.php?page=../../../../../../../../etc/passwd: The PHP-Nuke Rocket add-in is vulnerable to file traversal, allowing an attacker to view any file on the host, (probably Rocket, but could be any index.php)
+ /mutillidae/phpinfo.php: Output from the phpinfo() function was found.

+ OSVDB-3268: /mutillidae/data/: Directory indexing found.
+ OSVDB-3892: /mutillidae/data/: This might be interesting...
+ OSVDB-3268: /mutillidae/includes/: Directory indexing found.
+ OSVDB-3892: /mutillidae/includes/: This might be interesting...
+ OSVDB-3268: /mutillidae/passwords/: Directory indexing found.
+ OSVDB-3892: /mutillidae/passwords/: This might be interesting...
```

можно изучить отдельно

1. Path Traversal - уязвимость, позволяющая получить доступ к файлам и директориям сервера за пределами корневой директории сайта. Так же может быть использована с URLкодированием для обхода безопасности. На данном примере был получен доступ к файлу "/etc/passwd"- своего рода "открывашка" для данного рода уязвимостей.
2. Файл "phpinfo.php" - вся поднаготная сервера, настройки, конфигурации, а если при включённых allow\_url\_fopen/allow\_url\_include - прямое приглашение к LFI RFI уязвимостям.
3. Индексация директорий, в которых возможно хранится информация о паролях на данном ресурсе.

Выводы и рекомендации по устранению уязвимостей:

1. Запрет возможностей уязвимостей Path Traversal, зависит от типа, возможностей сервера, чаще всего достаточно обновления версии сервера для устранения подобной уязвимости.
2. Скрытие, переименование, закрытие доступа к данному файлу. Как минимум это усложнит работу злоумышленнику и даст возможное время для работы BlueTeam.
3. Скрытие, закрытие прямого доступа к файлам, относящимся к чувствительной информации ресурса, хэширование данных. Аналогично, полностью почти невозможно закрыть доступ, но увеличить время для реагирования - предоставит).

## Задание 2

Ресурс:

DVWA brute force

<http://192.168.56.103/>

Найденная уязвимость:

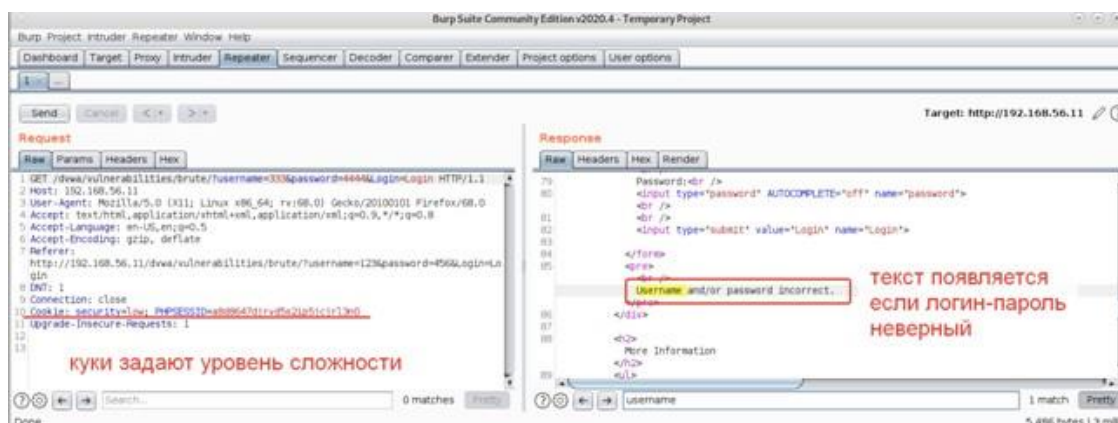
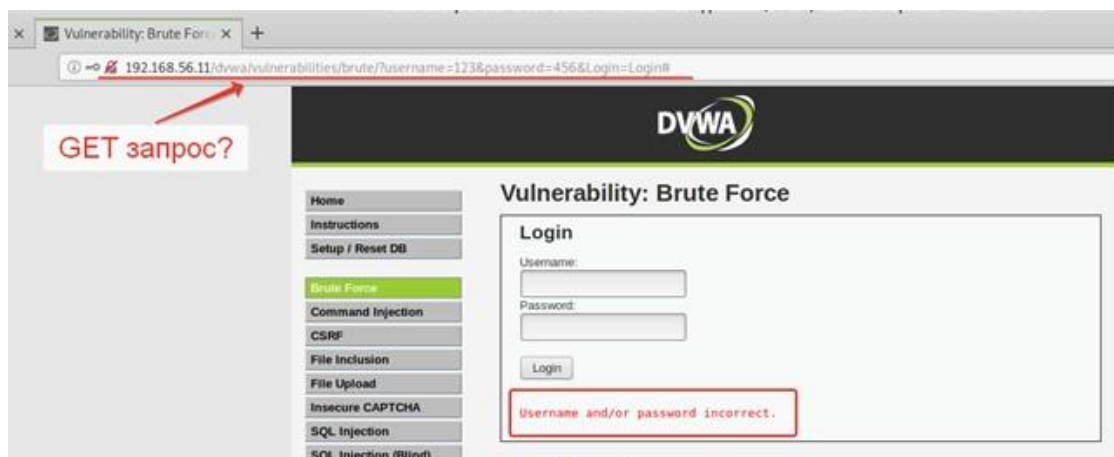
<http://192.168.56.103/mutillidae/index.php?page=login.php>

Детали обнаружения и воспроизведения уязвимости:

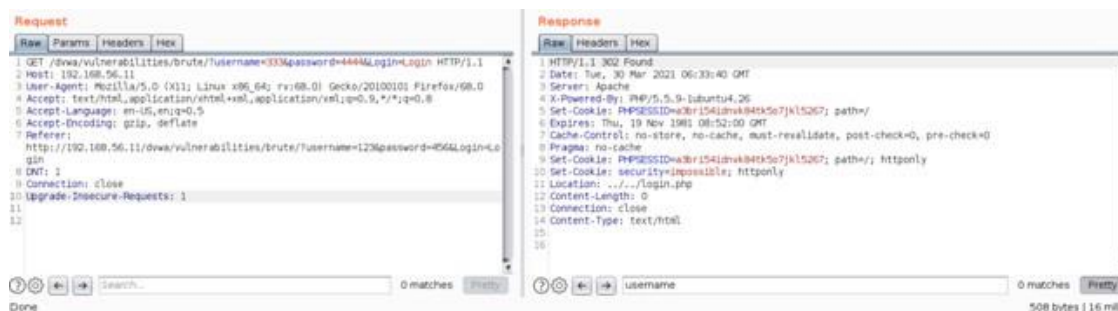
Уязвимость обнаружена при вводе пары логин/пароль.

Ответ в зависимости от корректности ввода пары.

Для передачи данных используется GET-запрос без шифрования передаваемых данных.

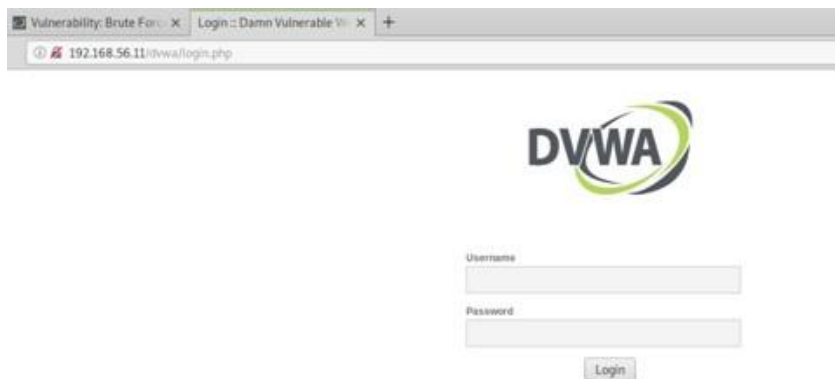


Изучая логику работы ресурса на этапе ввода пары через BurpSuite можно заметить определённое использование cookie.



При удалении cookie и отправке нового запроса - ресурс обрабатывает редиректом (код - 302), назначают новые cookie для пользователя. Перенаправление на новую страницу аутентификации.

Далее при повторном вводе пары логин/пароль возвращаюсь в Бург. Можно увидеть, что значение сессионной cookie поменялось. Без данного заголовка выполнение запроса не проходит напрямую.



Далее выполнялись атаки с помощью утилиты rotator

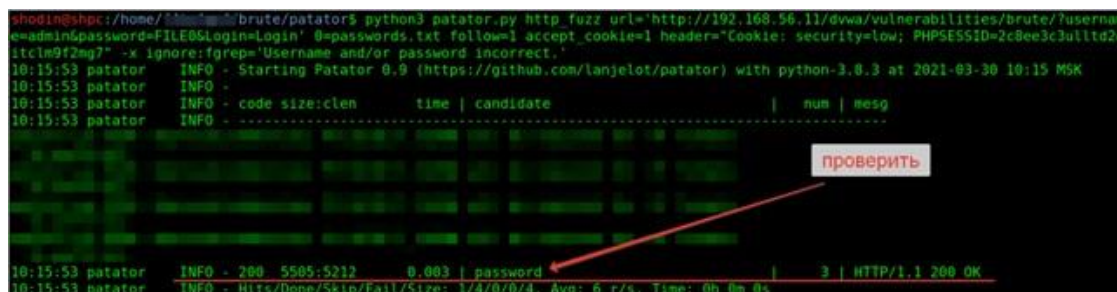
Исходные данные:

- Это GET запрос.
- Целевые параметры – username и password.
- Без кук запрос не работает.

Запрос:

```
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.'
```

Результат:



Выводы и рекомендации по устранению:

Уязвимость позволяет выполнить подбор логина/пароля для любой учётной записи. В итоге, получим доступ к конфиденциальной информации. Не требует дополнительных уязвимостей для эксплуатации.

Рекомендации по устранению:

- Использовать шифрование при передаче логина/пароля на сервер.
- Удалить различие ответа сервера при неудачной аутентификации между неверный «логин и пароль» и неверный «пароль».
- Установить ограничение на кол-во попыток в кол-ве 5 штук,
- Добавить двухфакторную аутентификацию.

Используемое программное обеспечение:

- Firefox web browser
- Burp Suite
- Сканер patat