

Основы информационной безопасности

Индивидуальный проект. Этап № 5. Использование Burp Suite

-Сунгурова М.

Российский университет дружбы народов, Москва, Россия

Информация

- Сунгурова Мариян Мухсиновна
- НКНбд-01-21
- Российский университет дружбы народов

Постановка задачи

Целью данной работы является использование Burp Suite для перехвата, изменения и изучения HTTP запросов и ответов.

Теоретические сведения

Damn Vulnerable Web Application (DVWA) – это веб-приложение PHP/MySQL, которое чертовски уязвимо. 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.

Цель DVWA – отработать некоторые из наиболее распространенных веб-уязвимостей различного уровня сложности с помощью простого и понятного интерфейса. В этом программном обеспечении существуют как задокументированные, так и недокументированные уязвимости.

DVWA имеет три уровня безопасности, они меняют уровень безопасности каждого веб приложения в DVWA:

- Невозможный — этот уровень должен быть безопасным от всех уязвимостей. Он используется для сравнения уязвимого исходного кода с безопасным исходным кодом.
- Высокий — это расширение среднего уровня сложности, со смесью более сложных или альтернативных плохих практик в попытке обезопасить код. Уязвимости не позволяют такой простор эксплуатации как на других уровнях.
- Средний — этот уровень безопасности предназначен главным образом для того, чтобы дать пользователю пример плохих практик безопасности, где разработчик попытался сделать приложение безопасным, но потерпел неудачу.
- Низкий — этот уровень безопасности совершенно уязвим и совсем не имеет защиты. Его предназначение быть примером среди уязвимых веб приложений, примером плохих практик программирования и служить платформой обучения базовым техникам эксплуатации

Пакет состоит из набора утилит, среди которых есть инструменты для сбора и анализа информации, моделирования разных типов атак, перехвата запросов и ответов сервера и так далее.

- Target – создает карту сайта с подробной информацией о тестируемом приложении. Показывает, какие цели находятся в процессе тестирования, и позволяет управлять процессом обнаружения уязвимостей.
- Proxy – находится между браузером пользователя и тестируемым веб-приложением. Перехватывает все сообщения, передаваемые по протоколу HTTP(S).
- Spider – автоматически собирает данные о функциях и компонентах веб-приложения.
- Clickbandit – моделирует кликджекинг-атаки (clickjacking attacks), при которых поверх страницы приложения загружается невидимая страница, подготовленная злоумышленниками.
- DOM Invader – проверяет веб-приложение на уязвимость DOM-based межсайтовому скриптинг (основанному на объектной модели документа) внедрению вредоносного

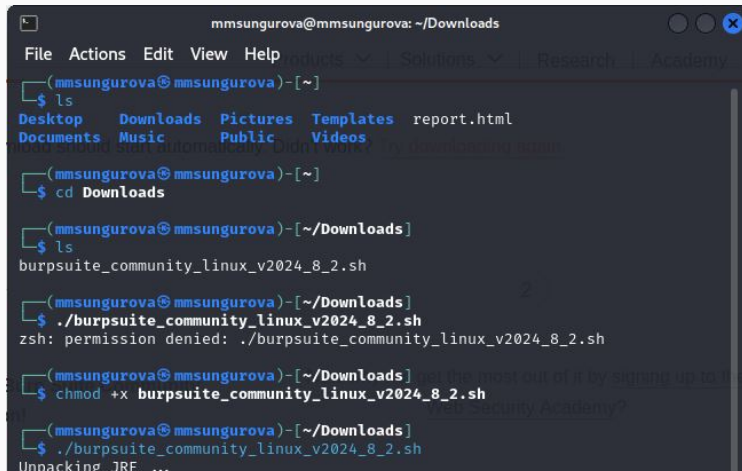
- Scanner (в профессиональной и корпоративной редакциях) — автоматически сканирует уязвимости в веб-приложениях. Также существует в бесплатной версии, но, предоставляет только описание возможностей. Intruder – проводит автоматические атаки различного типа, от перебора открытых веб-директорий до внедрения SQL-кода.
- Repeater – утилита для ручного манипулирования и повторной выдачи отдельных HTTP-запросов и анализа ответов приложения. Отправить запрос в Repeater можно из любой другой утилиты Burp Suite.
- Sequencer – анализирует качество случайности в выборке элементов данных. Можно использовать для тестирования сеансовых маркеров приложения или других важных элементов данных, которые должны быть непредсказуемыми, например маркеров анти-CSRF, маркеров сброса пароля и так далее. Decoder— преобразовывает закодированные данные в исходную форму или необработанные в различные закодированные и хешированные формы. Способен распознавать несколько форматов кодирования, используя эвристические методы. Comparer – предоставляет функцию

Выполнение лабораторной работы

Выполнение лабораторной работы

Intercept HTTP traffic with Burp Proxy

Установим Burp Suit с официального сайта(рис. (fig:001?))



```
mmsungurova@mmsungurova: ~/Downloads
File Actions Edit View Help
(mmsungurova@mmsungurova)-[~]
$ ls
Desktop Downloads Pictures Templates report.html
Documents Music Public Videos

(mmsungurova@mmsungurova)-[~]
$ cd Downloads

(mmsungurova@mmsungurova)-[~/Downloads]
$ ls
burpsuite_community_linux_v2024_8_2.sh

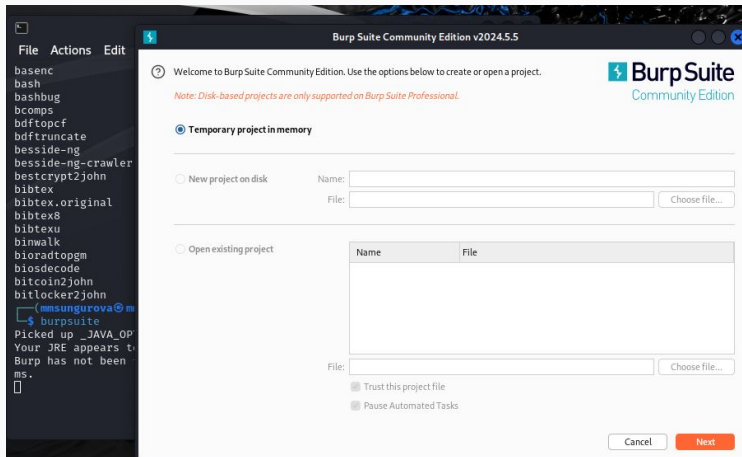
(mmsungurova@mmsungurova)-[~/Downloads]
$ ./burpsuite_community_linux_v2024_8_2.sh
zsh: permission denied: ./burpsuite_community_linux_v2024_8_2.sh

(mmsungurova@mmsungurova)-[~/Downloads]
$ chmod +x burpsuite_community_linux_v2024_8_2.sh

(mmsungurova@mmsungurova)-[~/Downloads]
$ ./burpsuite_community_linux_v2024_8_2.sh
Unpacking JRE ...
```

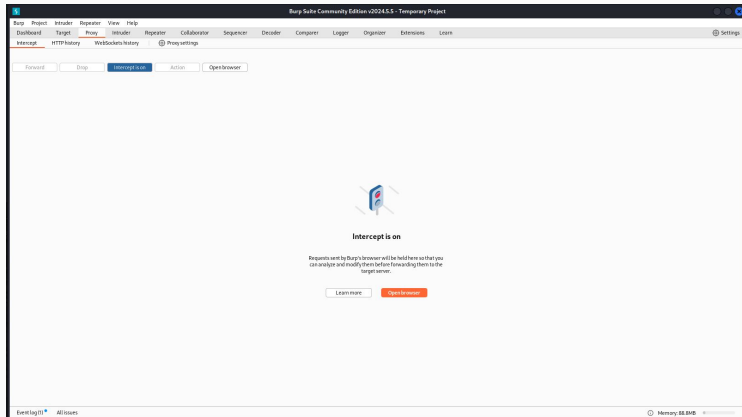
Выполнение лабораторной работы

Откроем приложение и создадим временный проект с параметрами по умолчанию (рис. (fig:002?)).



Выполнение лабораторной работы

Теперь попробуем перехватить http запрос с помощью Vurp Proху. Включим перехват, а в браузере включим прокси и укажем для него адрес локального хоста, а также установим параметр, разрешающий перехват запросов локального хоста(рис. (fig:004?)).



Выполнение лабораторной работы

Можем увидеть первый перехваченный запрос: вход на сайт DVWA. Указаны адрес локального хоста, версия браузера, ОС устройства и другая информация(рис. (fig:007?)):

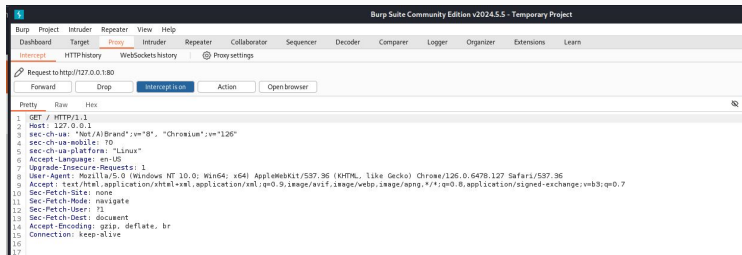


Рис. 4: Перехват запроса на вход на сайт

В запросах можно изменять вводимую информацию и сравнивать ответы(рис. (fig:010?)):

The screenshot displays the 'Network' tab of a web browser's developer tools. It shows a list of seven requests. The first two are POST requests to a Google URL. The next three are GET requests to a local host. The last two are GET requests to a local host, with the second one highlighted. Below the list, the 'Request' details for the highlighted request are shown in 'Pretty' format.

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP	Cookies
1	https://sb-ssl.google.com	POST	/safebrowsing/clientreport/download...		✓	400	765	JSON				✓	142.250.150.91	
2	https://sb-ssl.google.com	POST	/safebrowsing/clientreport/download...		✓							✓	142.250.150.91	
3	http://127.0.0.1	GET	/										127.0.0.1	
4	http://127.0.0.1	GET	/DVWA										127.0.0.1	
5	http://127.0.0.1	GET	/DVWA										127.0.0.1	
6	http://127.0.0.1	GET	/DVWA/login.php					HTML	php				127.0.0.1	
7	http://127.0.0.1	GET	/DVWA/login.php					HTML	php				127.0.0.1	

Request

Pretty Raw Hex

```
1 GET /DVWA/login.php HTTP/1.1
2 Host: 127.0.0.1
3 sec-ch-ua: "Not(A)Brand";v="8", "Chromium";v="126"
4 sec-ch-ua-mobile: ?0
5 sec-ch-ua-platform: "Linux"
6 Accept-Language: en-US
7 Upgrade-Insecure-Requests: 1
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.6478.127 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
10 Sec-Fetch-Site: none
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-User: ?1
13 Sec-Fetch-Dest: document
14 Accept-Encoding: gzip, deflate, br
15 Connection: keep-alive
16
17
```

Рис. 5: Изучение ответа на запрос с функцией повторения запроса

Выводы

В результате выполнения работы научились на практике использовать ПО Burp Suit для перехвата, изменения и изучения HTTP запросов и ответов.