

Индивидуальный проект - этап 5

Использование BurpSuite

Овезов Мерген

Содержание

1 Цель работы

Освоить работу с инструментом Burp Suite для перехвата, анализа и модификации HTTP-запросов/ответов веб-приложения DVWA, а также для поиска уязвимостей.

2 Выполнение работы

Запустить Burp Suite. Настроить прокси в Burp Suite на `127.0.0.1:8080`. В браузере включить ручной прокси `127.0.0.1:8080` и установить сертификат Burp в «Доверенные центры сертификации». Включить перехват. В браузере открыть DVWA, выполнить вход и зафиксировать перехваченные запросы. Вкладка “HTTP history” — выбрать запрос. Отправить запрос в Repeater.

Time	Type	Direction	Method	URL	Status code	Length
20:23:0...	WS	→ To server		https://kws2.web.telegram.org/apiws		153
20:23:0...	WS	← To client		https://kws2.web.telegram.org/apiws		393
20:23:0...	HT...	→ Request	GET	https://kws2.web.telegram.org/apiws		
20:23:0...	HT...	→ Request	GET	https://kws2.web.telegram.org/apiws		
20:23:3...	HT...	→ Request	GET	https://kws2.web.telegram.org/apiws		
20:23:4...	HT...	→ Request	GET	https://kws2.web.telegram.org/apiws		
20:23:5...	HT...	→ Request	GET	https://kws2.web.telegram.org/apiws		
20:24:4...	HT...	→ Request	GET	https://kws2.web.telegram.org/apiws		
20:33:2...	HT...	→ Request	POST	http://o.pki.goog/we2		

Figure 1: окно Intercept с перехваченным запросом

# ^	Host	Method	URL	Params	Edited	Status
1	http://detectportal.firefox.c...	GET	/success.txt?ipv6	✓		200
2	http://detectportal.firefox.c...	GET	/success.txt?ipv4	✓		200
3	https://push.services.mozill...	GET	/			101
7	https://kws2.web.telegram....	GET	/apiws			101
8	https://kws2.web.telegram....	GET	/apiws			101
9	https://kws2-1.web.telegra...	GET	/apiws			101
10	https://kws4-1.web.telegra...	GET	/apiws			101
11	https://kws2.web.telegram....	GET	/apiws			
12	https://kws2.web.telegram....	GET	/apiws			
13	https://kws2.web.telegram....	GET	/apiws			
14	https://kws2.web.telegram....	GET	/apiws			
15	https://kws2.web.telegram....	GET	/apiws			

Figure 2: вкладка HTTP history с выбранным запросом

The screenshot shows the Burp Suite Repeater tab with a target URL of `https://kws2.web.telegram.org`. The selected request is a GET request to `/apiws` with various headers. The Inspector panel on the right shows the selected text `kws2.web.telegram.org` and the decoded from field also containing `kws2.web.telegram.org`. The Request attributes section shows 2 attributes, Request query parameters shows 0, Request body parameters shows 0, Request cookies shows 0, and Request headers shows 16.

Request

```

1 GET /apiws HTTP/1.1
2 Host: kws2.web.telegram.org
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept: */*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Sec-WebSocket-Version: 13
8 Origin: https://web.telegram.org
9 Sec-WebSocket-Protocol: binary
10 Sec-WebSocket-Key: SjQdaeAf4vui80LWjjsvuw==
11 Connection: keep-alive, Upgrade
12 Sec-Fetch-Dest: empty
13 Sec-Fetch-Mode: websocket
14 Sec-Fetch-Site: same-site
15 Pragma: no-cache
16 Cache-Control: no-cache
17 Upgrade: websocket

```

Inspector

Selection: 21 (0x15)

Selected text: `kws2.web.telegram.org`

Decoded from: `kws2.web.telegram.org`

Request attributes: 2

Request query parameters: 0

Request body parameters: 0

Request cookies: 0

Request headers: 16

Figure 3: вкладка Repeater с изменённым запросом и ответом сервера

3 Вывод

Burp Suite позволяет детально анализировать взаимодействие клиента и сервера. - Инструмент полезен для поиска уязвимостей, тестирования валидации данных и проведения атак типа SQL Injection, XSS и др. - Для защиты необходимо фильтровать и проверять все входящие данные на стороне сервера.