Vulnerabilities Findings Report for cms.bjitacademy.com

Prepared By

Mir Mohaiminul Islam

Id: 00-30111

Trainee SQA Engineer

BJIT Academy

Date of Submission: 24 November, 2023

Index

SL No.	Vulnerabilities Findings	Page Number
01	User get access of the all SEO pages content and adding course by changing the authorization token of super Admin with normal user's authorization token (Vertical Privilege Escalation).	03-13
02	User (Admin, SEO Manager, Trainer, Content Manager) gains unauthorized permission to delete a subscriber by replacing the super admin's authorization token with the user's authorization token (Vertical Privilege Escalation).	13-30
03	Normal User (Trainer, Content Manager, SEO Manager) has unauthorized access of adding information by replacing the user_id of Super Admin (Access control violation).	30-53
04	Normal User's (Admin, Trainer, SEO Manager, Content Manager) user_id can be manipulated to make a new user.	53-65
05	Unauthorized role assignment (Horizontal Privilege Escalation) to the new users (Add User) and the existing users (Edit User).	65-73
06	Non-Admin users (Content Manager, SEO Manager, Trainer) can update some unauthorized information such as client's information.	74-83

1. **Title:** User get access of the all SEO pages content and adding course by changing the authorization token of super Admin with normal user's authorization token (Vertical Privilege Escalation).

   Target: cms.bjitacademy.com

   Affected URL/API:

   POST /academysite/api/public/api/v1/pages/update-page/ (SEO Page Content edit)

   POST /academysite/api/public/api/v1/course/create-popular-course/ (Create a new course)

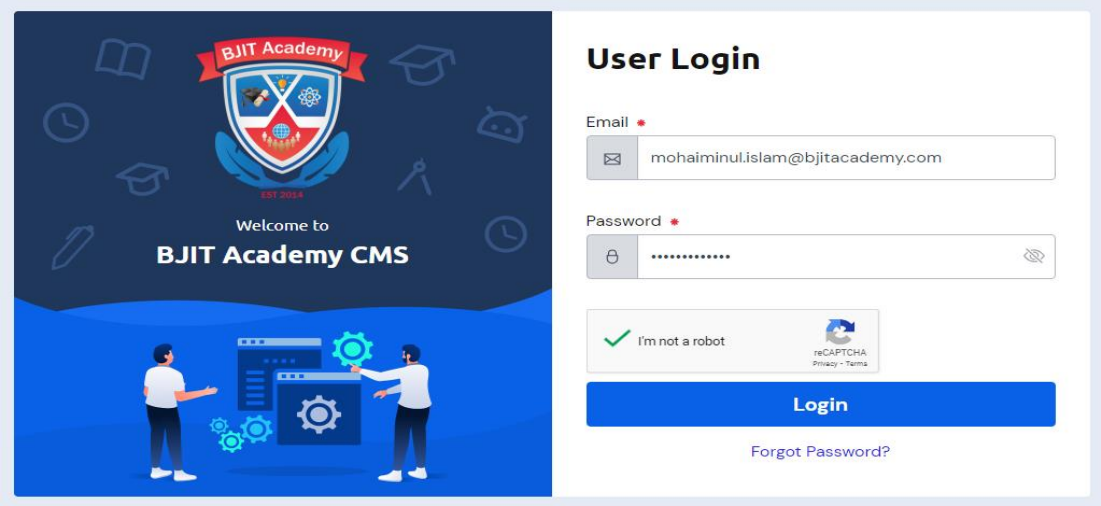
   Summary: Super Admin, Content Manager, SEO Manager & Admin they have access to the SEO pages module but trainers don't have access. By replacing the Super Admin's Authorization token a Trainer level user can get vertical access to see all the SEO pages information which is a critical Vulnerability.

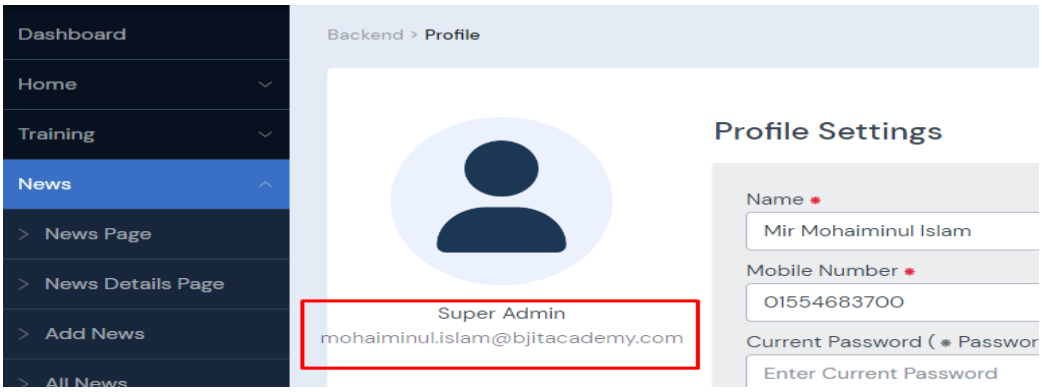
   Proof of Concept:

   i) POST /academysite/api/public/api/v1/pages/update-page/

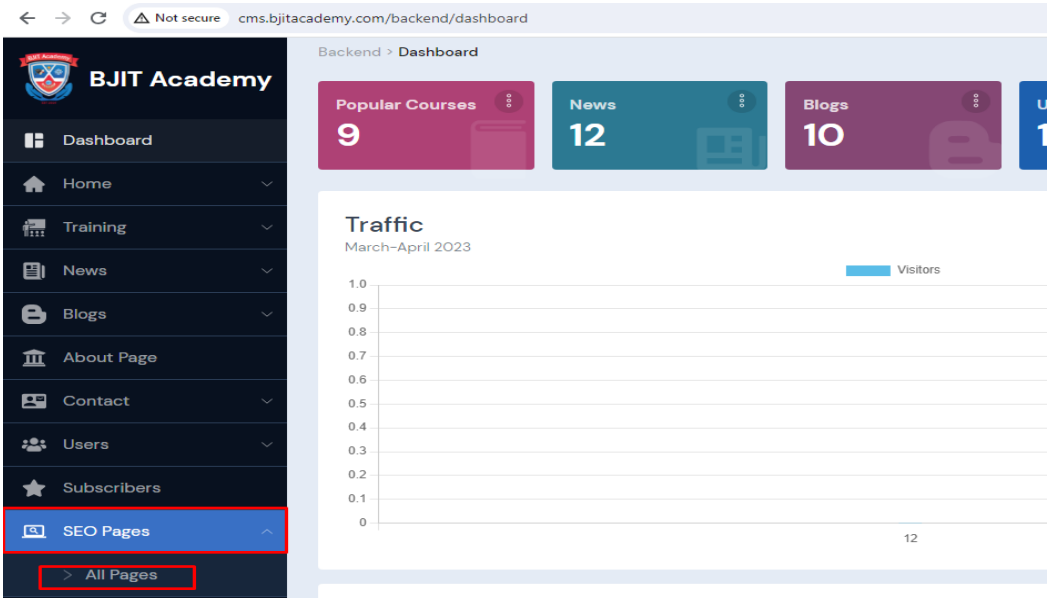
   At first go to http://cms.bjitacademy.com/login and login with super admin credential with built in chromium browser in the burpsuite.

   Here the Super-Admin User: Mir Mohaiminul Islam (mohaiminul.islam@bjitacademy.com)



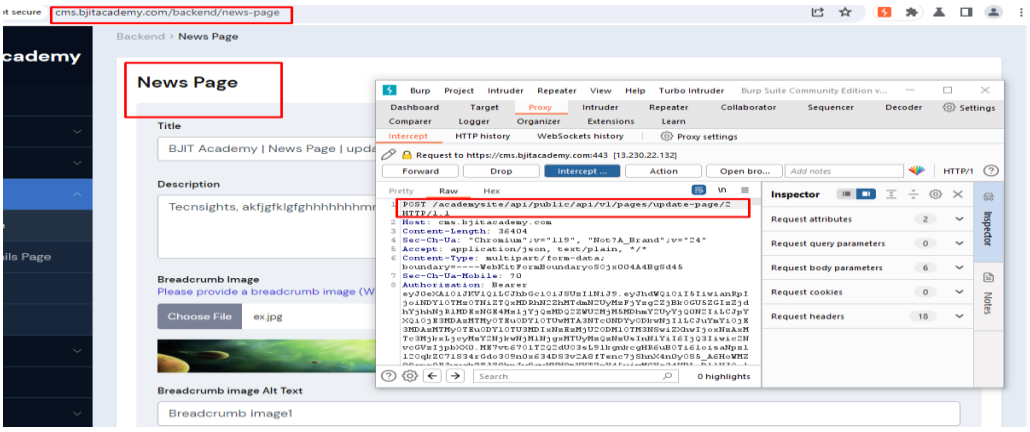


After login go to the Seo page → All pages.

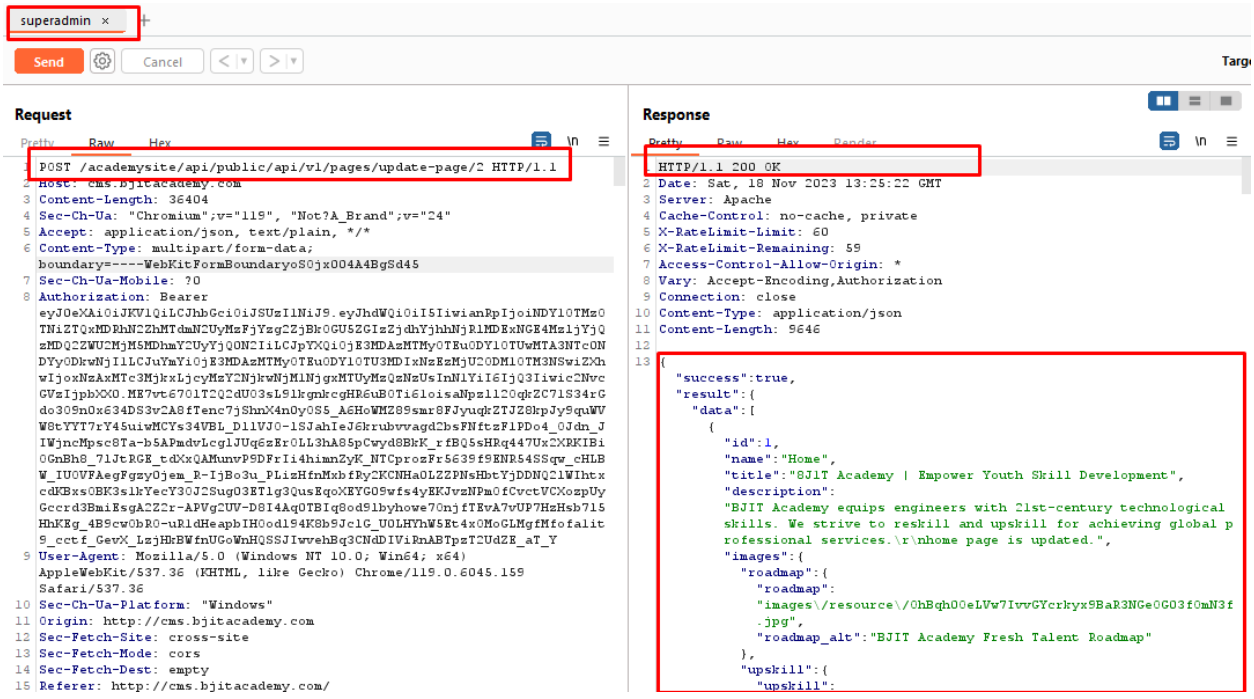


After that all pages are loaded

Let's jump into the **News** page and try to update it then capture request in the burp Repeater.



Here I try to update the existing news page's content. Then I click the save button and capture the request and send **POST /academysite/api/public/ api/v1/pages/update-page/2** to the burp Repeater.

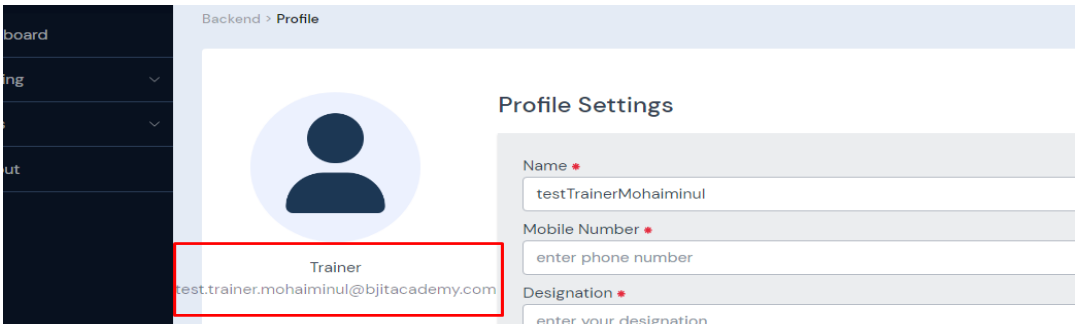


Now observe the super admin's authorization token.

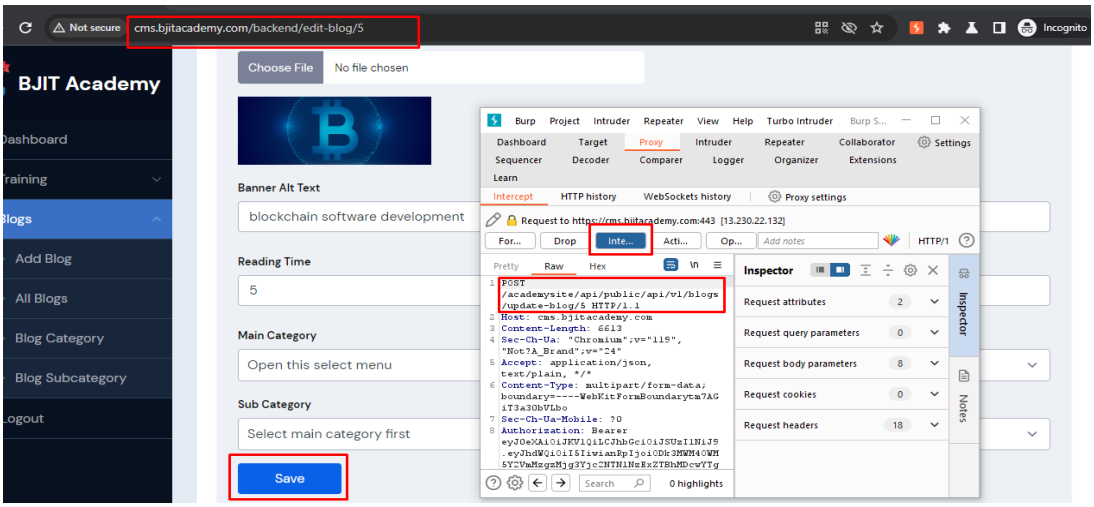
```
eyJ0eXAi0iJKV1QiLCJhbGci0iJSUzI1NiJ9.eyJhdWQi0iI5IiwianRpIjoiND
Yl0TMz0TNiZTQxMDRhN2ZhMTdmN2UyMzFjYzg2ZjBk0GU5ZGIzZjdhYjhhNjRlM
DExNGE4MzljYjQzMDQ2ZWU2MjM5MDhmY2UyYjQ0N2IiLCJpYXQi0jE3MDAzMTMy
0TEu0DYl0TUwMTA3NTc0NDYy0DkwNjI1LCJuYmYi0jE3MDAzMTMy0TEu0DYl0TU
3MDIxNzEzMjU20DM10TM3NSwiZXhwIjoxNzAxMTc3MjkxLjcyMzY2NjkwNjM1Nj
gxMTUyMzQzNzUsInN1YiI6IjQ3Iiwic2NvcGVzIjpbXX0.ME7vt6701T2Q2dU03
sL9lkgnkcgHR6uB0Ti6loisaNpzll20qkZC7lS34rGdo309n0x634DS3v2A8fTe
nc7jShnX4n0y0S5_A6HoWMZ89smr8FJyuqkZTJZ8kpJy9quWVW8tYYT7rY45uiw
MCYs34VBL_DllVJ0-lSJahIeJ6krubvvagd2bsFNftzFlPDo4_0Jdn_JIWjncMp
sc8Ta-b5APmdvLcglJUq6zEr0LL3hA85pCwyd8BkK_rfBQ5sHRq447Ux2XRKIBi
0GnBh8_7lJtRGE_tdXxQAMunvP9DFrIi4himnZyK_NTCprozFr5639f9ENR54SS
qw_cHLBW_IU0VFAegFgzy0jem_R-IjBo3u_PLizHfnMxbfRy2KCNHa0LZZPNsHb
tYjDDNQ2lWIhtxcdKBxs0BK3slkYecY30J2Sug03ETlg3QusEqoXEYG09wfs4yE
KJvzNPm0fCvctVCXozpUyGccrd3BmiEsgA2Z2r-APVg2UV-D8I4Aq0TBIq8od9l
byhowe70njfTEvA7vUP7HzHsb7l5HhKEg_4B9cw0bR0-uRldHeapbIH0odl94K8
b9JclG_U0LHYhW5Et4x0MoGLMgfMfofalit9_cctf_GevX_LzjHrBWfnUGoWnHQ
SSJIwvehBq3CNdDIViRnABTpzT2UdZE_aT_Y
```


Again login with another user whose role is Trainer with the build in chromium's incognito browser.

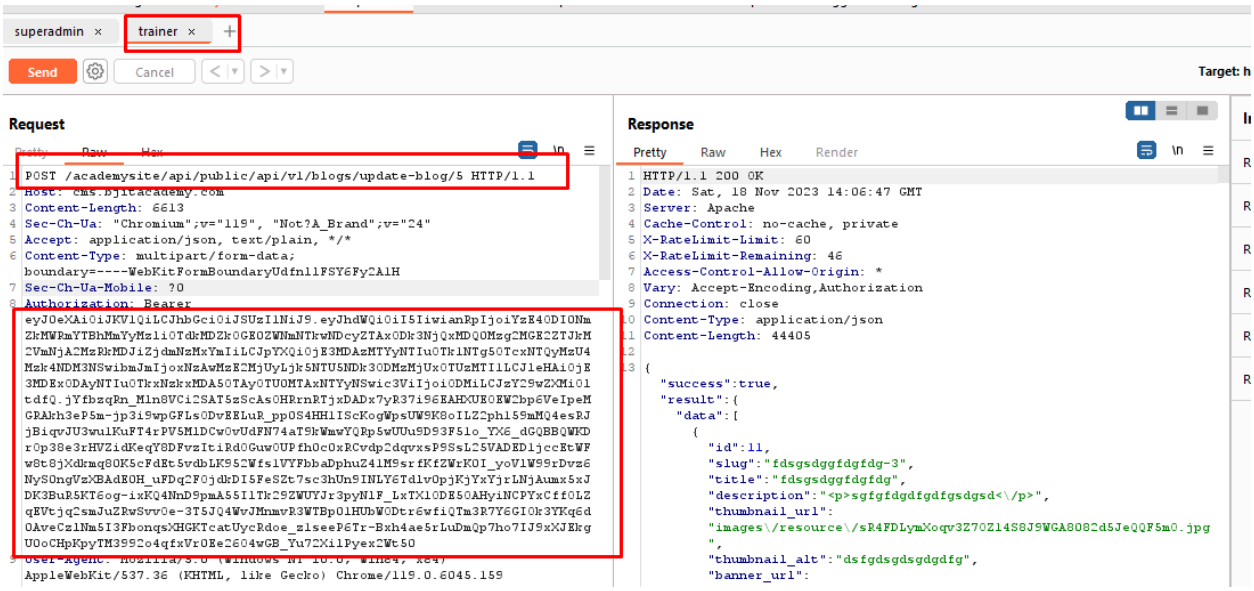
Here the Trainer Account: test.trainer.mohaiminul@bjitacademy.com



Now I will to capture the authorization token of that Trainer. If I capture a request from the trainer account then I will get the authorization token. So I have to go to **backend/edit-blog/5** to capture the token.



Now I will take this request to the Repeater

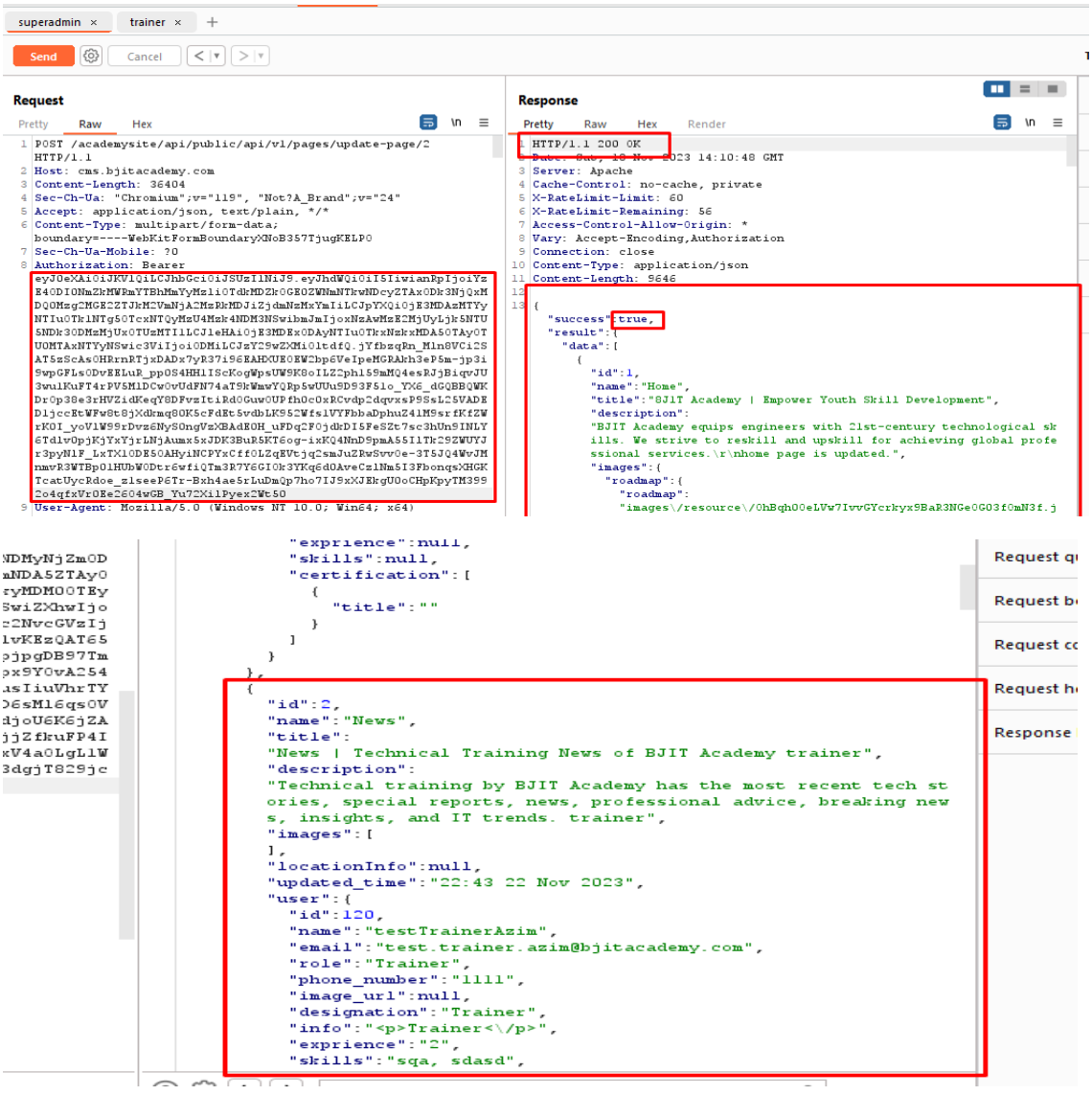


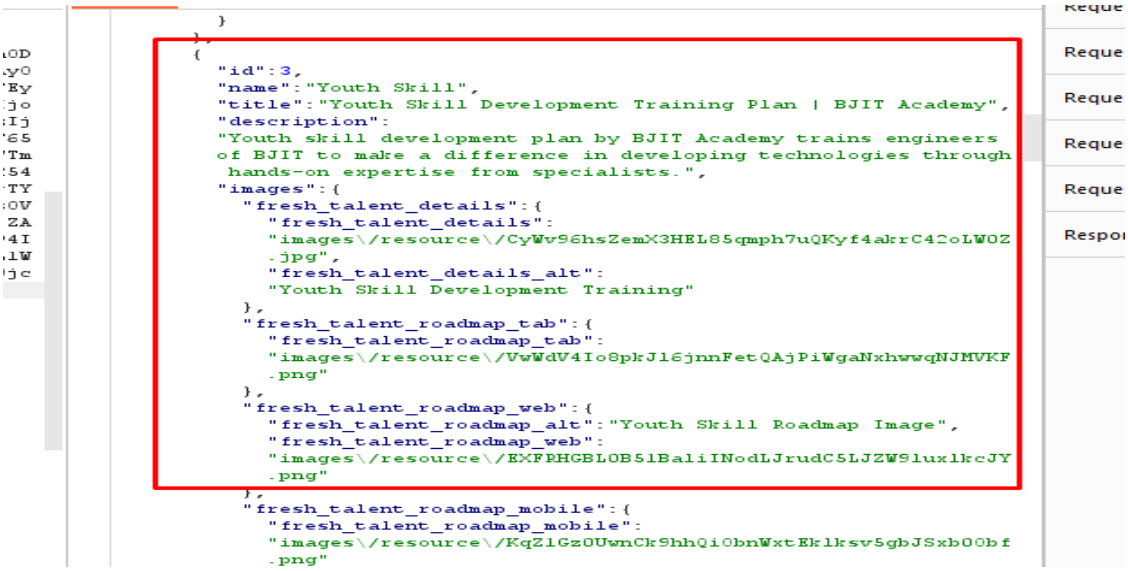
In the above picture this is trainer authorizationtoken which is

```
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJhdWQiOiI5IiwianRpIjoiYzE4ODI0Nm
ZkMWRmYTBhMmYyMzli0TdkMDZk0GE0ZWNmNTkwNDcyZTAx0Dk3NjQxMDQ0Mzg2MGE2ZTJkM
2VmNjA2MzRkMDJiZjdmNzMxYmIiLCJpYXQiOjE3MDAzMTYyNTIu0Tk1NTg50TcxNTQyMzU4
Mzk4NDM3NSwibmJmIjoxNzAwMzE2MjUyLjk5NTU5NDk30DMzMjUx0TUzMTIlLCJleHAiOjE
3MDEx0DAyNTIu0TkxNzkxMDA50TAy0TU0MTAxNTYyNSwic3ViIjoi0DMiLCJzY29wZXMi0l
tdfQ.jYfbzqRn_Mln8VCi2SAT5zScAs0HRrnRTjxDADx7yR37i96EAHXUE0EW2bp6VeIpeM
GRAkh3eP5m-jp3i9wpGFLs0DvEELuR_pp0S4HHlIScKogWpsUW9K8oILZ2phl59mMQ4esRJ
jBiqvJU3wulKuFT4rPV5MlDCw0vUdFN74aT9kWmwYQRp5wUUu9D93F5lo_YX6_dGQBBQWKD
r0p38e3rHVZidKeqY8DFvzItiRd0Guw0UPfh0c0xRCvdp2dqvxsP9SsL25VADED1jccEtWF
w8t8jXdkmq8OK5cFdEt5vdbLK952Wfs1VYFbbaDphuZ41M9srfKfZWrKOI_yoVlW99rDvz6
NyS0ngVzXBAdEOH_uFDq2F0jdkDI5FeSZt7sc3hUn9INLY6TdlvOpjKjYxYjrLNjAumx5xJ
DK3BuR5KT6og-ixKQ4NnD9pmA55IlTk29ZWUYJr3pyNlF_LxTXl0DE50AHyiNCPYxCff0LZ
qEVtjq2smJuZRwSvv0e-3T5JQ4WvJMnmvR3WTBp0lHUbW0Dtr6wfiQTm3R7Y6GI0k3YKq6d
0AveCzlNm5I3FbonqsXHGKTcatUycRdoe_zlseeP6Tr-Bxh4ae5rLuDmQp7ho7IJ9xXJEkg
UOoCHpKpyTM3992o4qfxVr0Ee2604wGB_Yu72XilPyex2Wt50
```

Now Replace the super admin token with the trainer token in that API in the repeater:

POST /academysite/api/public/api/v1/pages/update-page/2 HTTP/1.1





After replacing the super admin token with the trainer token the response shows 200 OK and also shows all pages information. Trainer has no access of the SEO pages but changing the authorization token gives the trainer that kind of access and disclose all the page information to the trainer level users which is a critical vulnerability.

Expected Result: Trainer can't access the SEO page's content.

Actual Result: Trainer can access the SEO page with replacing the super admin's authorization token with trainer token.

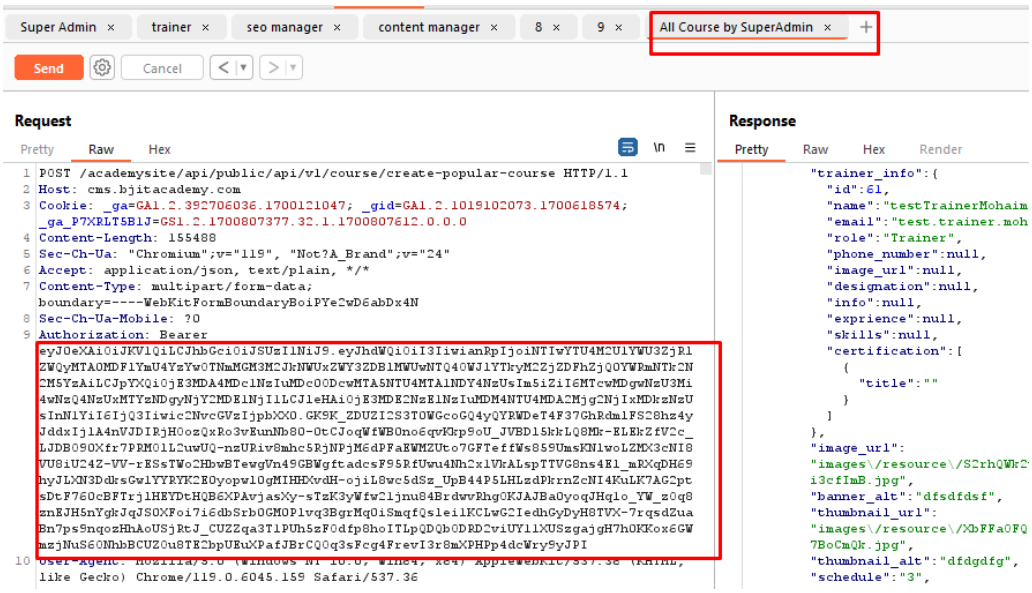
Similarly For creating course:

The API for creating course:

ii) **POST /academysite/api/public/api/v1/course/create-popular-course/**

First login with the super admin credential → go to home→ add course → capture the course request by intercept on→ take the request to Repeater

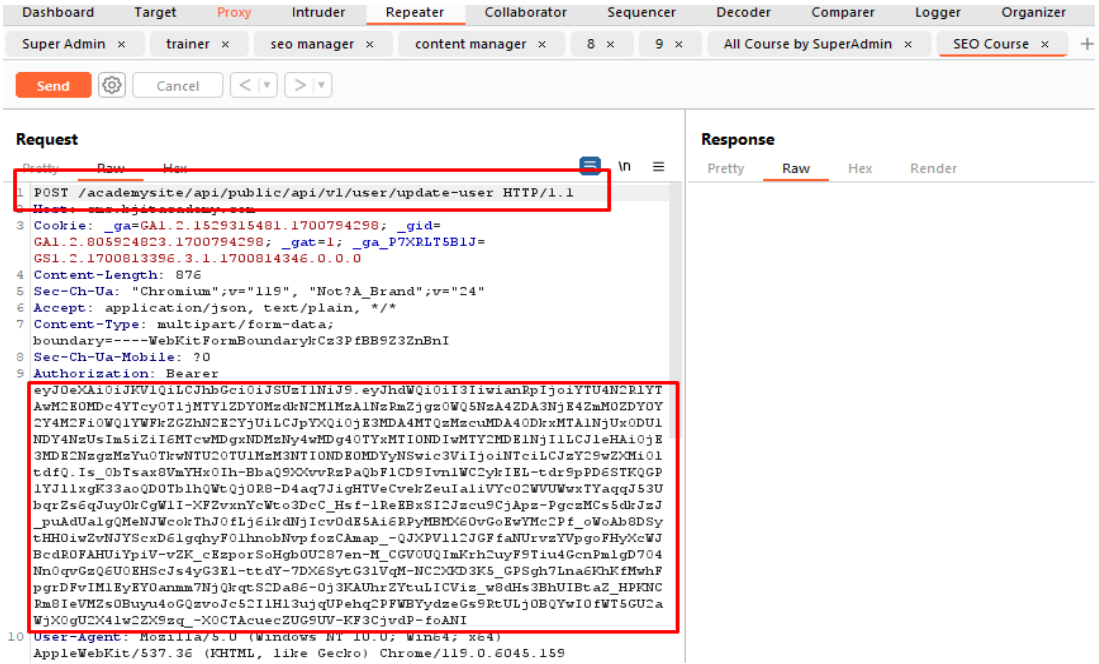
The api of add course is: **POST /academysite/api/public/api/v1/course/create-popular-course**



Now SEO Manager and Content Manager don't have access of adding course. So now I will replace the Super Admin Authorization token with the seo and content manager.

First SEO Manager:

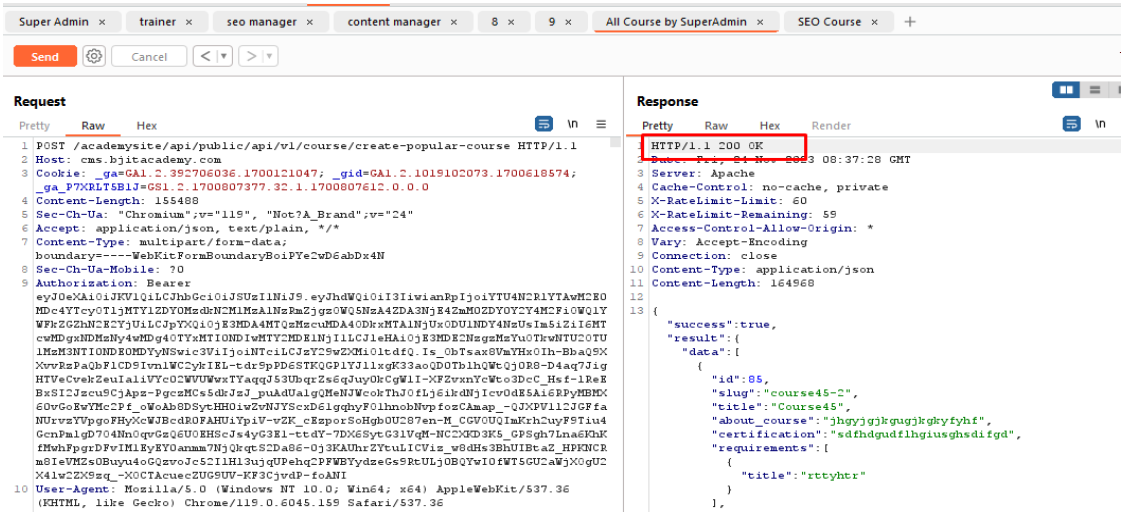
For SEO manager Authorization token, login with SEO Manager → go to profile → update profile and capture the request in to burp suite → take it to repeater.



Here the API for taking the Authorization token of a SEO manager is : **POST /academysite/api/public/api/v1/user/update-user**

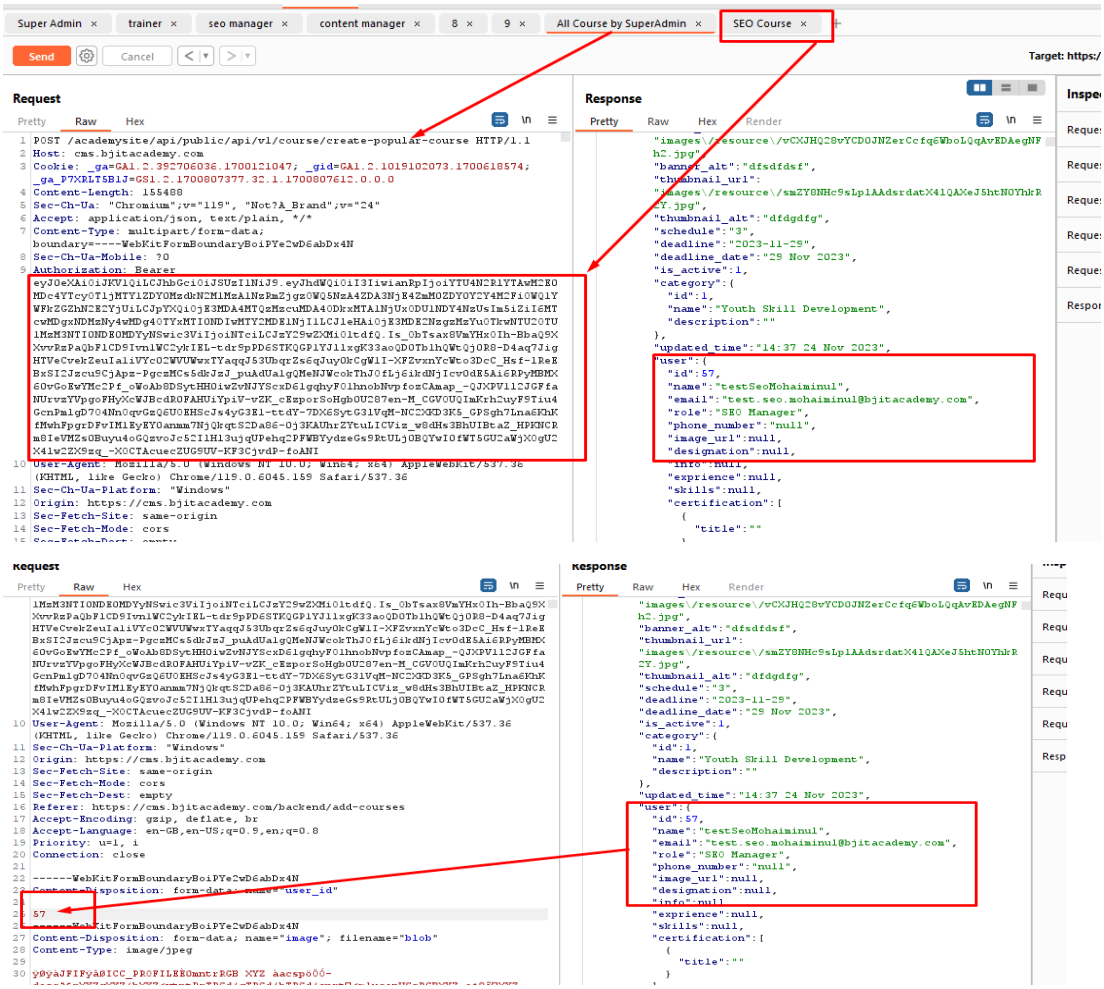
Now Replace the Authorization Token of super admin with seo manager Authorization token. Here I also need to change the **user_id** of the super admin and replace it with seo manager user_id.

```
Content-Disposition: form-data; name="email"

test.seo.mohaiminul@bjitacademy.com
------WebKitFormBoundaryrCz3PfBB9Z3ZnBnI
Content-Disposition: form-data; name="image"


------WebKitFormBoundaryrCz3PfBB9Z3ZnBnI
Content-Disposition: form-data; name="phone_number"

null
------WebKitFormBoundaryrCz3PfBB9Z3ZnBnI
Content-Disposition: form-data; name="password"


------WebKitFormBoundaryrCz3PfBB9Z3ZnBnI
Content-Disposition: form-data; name="new_password"


------WebKitFormBoundaryrCz3PfBB9Z3ZnBnI
Content-Disposition: form-data; name="new_password_confirmation"


------WebKitFormBoundaryrCz3PfBB9Z3ZnBnI
Content-Disposition: form-data; name="user_id"

57
------WebKitFormBoundaryrCz3PfBB9Z3ZnBnI--
```



After Replacing the the authentication token and user_id I got 200 OK.

And also I found that the course is created by replacing authorization token and user_id.



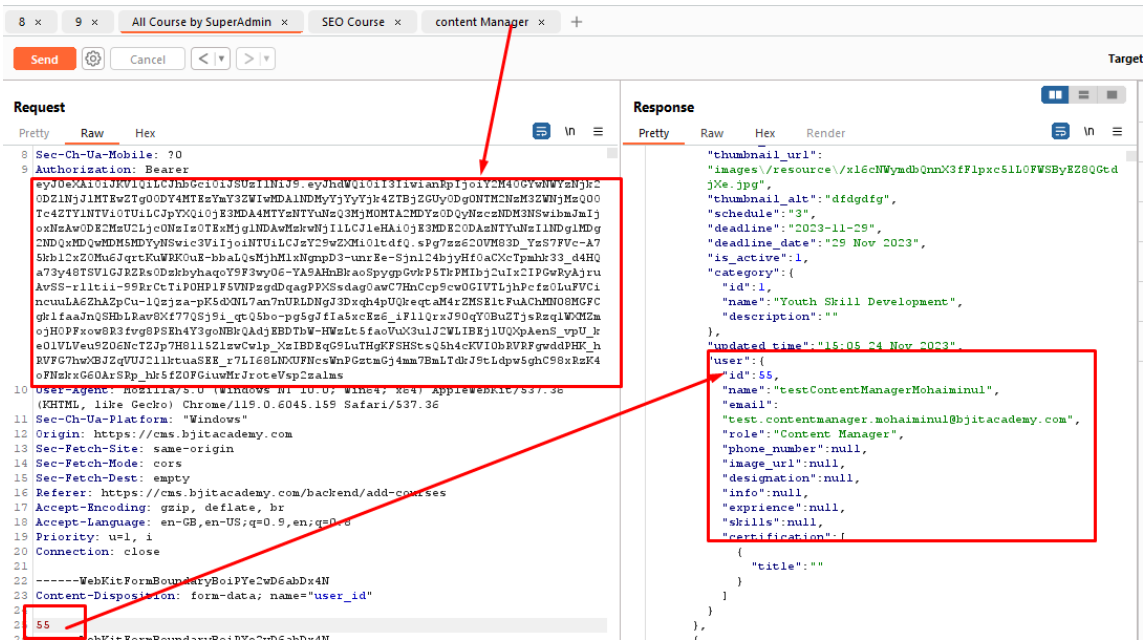
From Content Manager:

For Content manager Authorization token, login with Content Manager → go to profile → update profile and capture the request in to burp suite → take **POST /academysite/api/public/api/v1/user/update-user** to repeater.

Here the content Manager account:
test.contentmanager.mohaiminul@bjitacademy.com

User_id: 55

Now replace the super admin user_id and authorization token with the content manager user_id and authorization token.



Expected Result: Super Admin, Admin, Trainer can create a new course

Actual result: SEO Manager and Content Manager also has unauthorized access to create a new course.

So Here for the first API trainer don't have access to see all content of SEO page

But from the backend by using API trainer can access the seo page by manipulating the authorization token.

For second API SEO Manager and Content Manager don't have access to create course from frontend but from backend API they can create course by manipulating the authorization token and user_id. So it is a vertical privilege escalation type vulnerability.

2. **Title:** User (Admin, SEO Manager, Trainer, Content Manager) gains unauthorized permission to delete a subscriber by replacing the super admin's authorization token with the user's authorization token (Vertical Privilege Escalation).

   Target: cms.bjitacademy.com

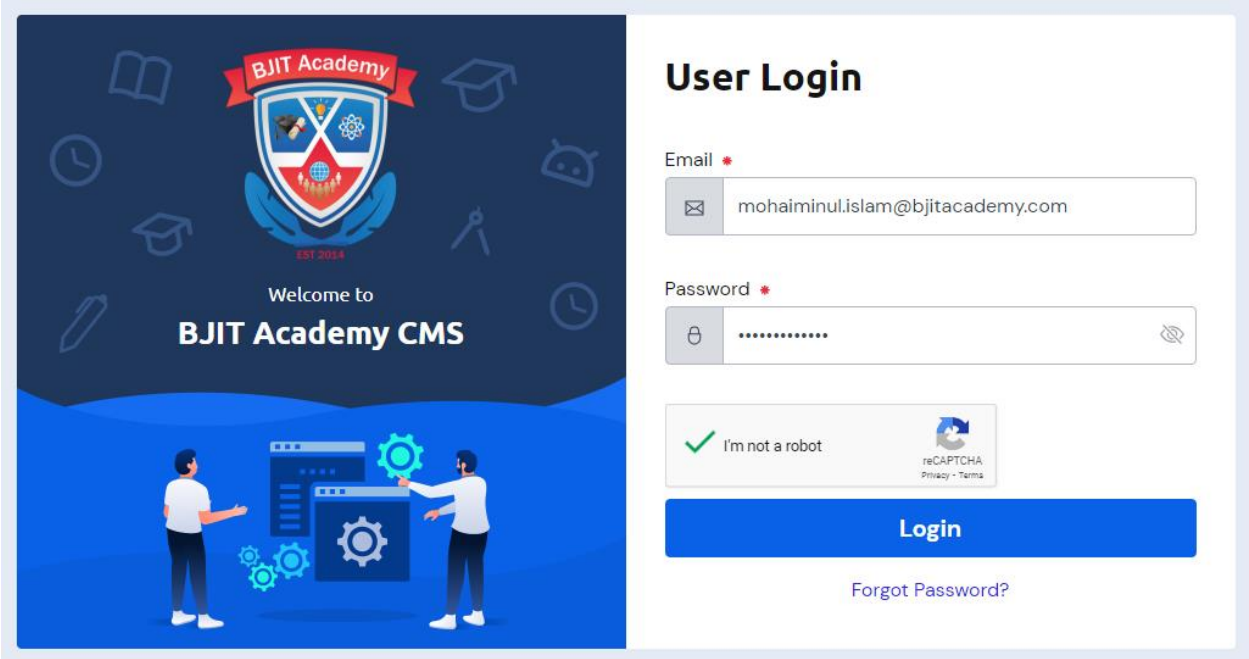
   URL/API: DELETE /academysite/api/public/api/v1/contact/delete-email/ (from backend)

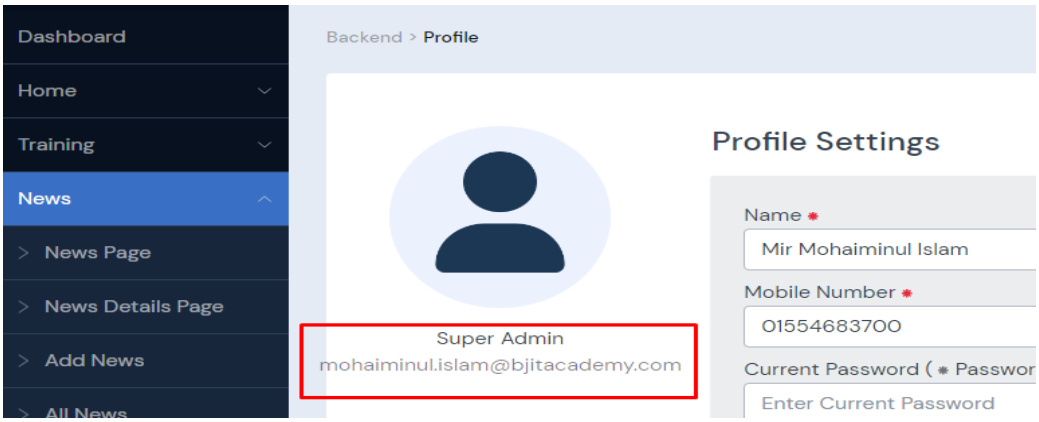
Summary: Generally a normal user can use their email id to subscribe in the BJIT academy's website. Only Super Admin have the access to remove a user email. But here by manipulating the super admin's authorization token user can delete a subscriber in this website.

POC:

At first go to http://cms.bjitacademy.com/login and login with super admin credential with built in chromium browser in the burpsuite.

Here the Super-Admin User: Mir Mohaiminul Islam (mohaiminul.islam@bjitacademy.com)

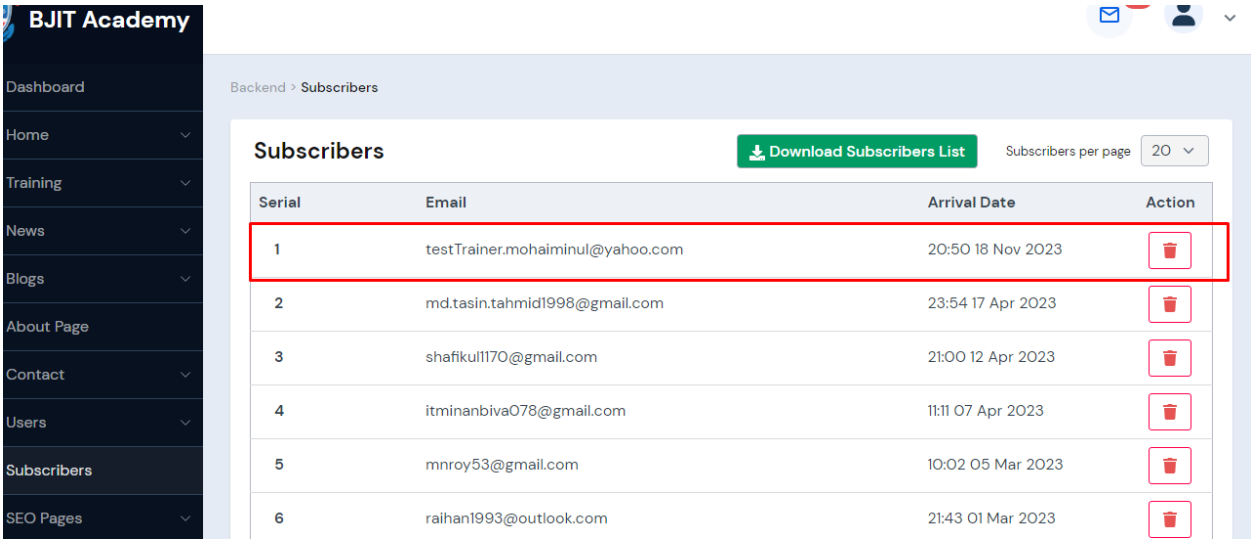




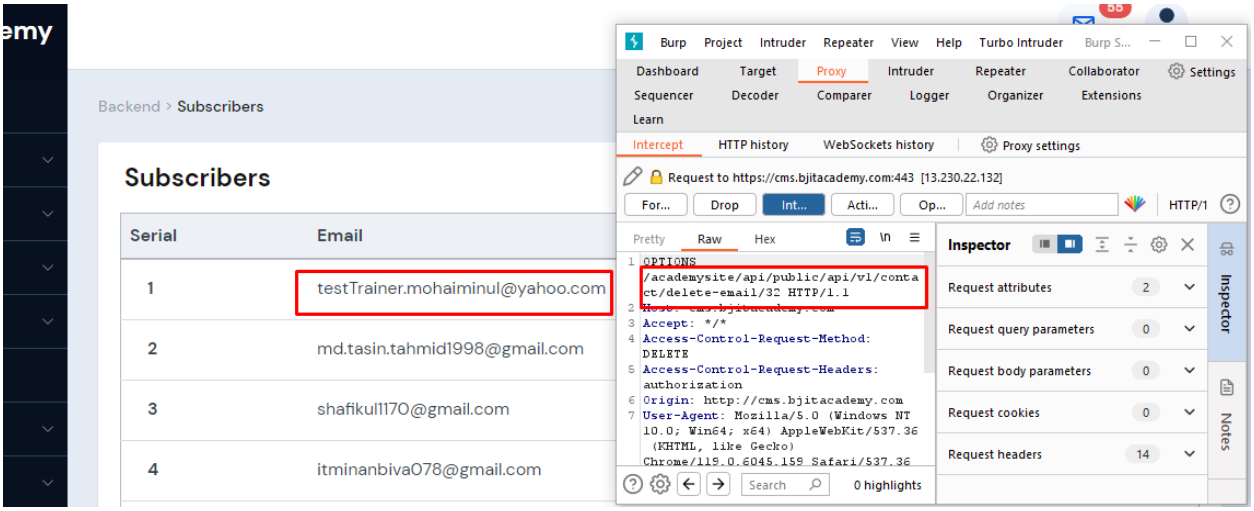
Now click on the subscriber tab.

Super Admin has the access to remove any subscriber but trainers don't have this access.

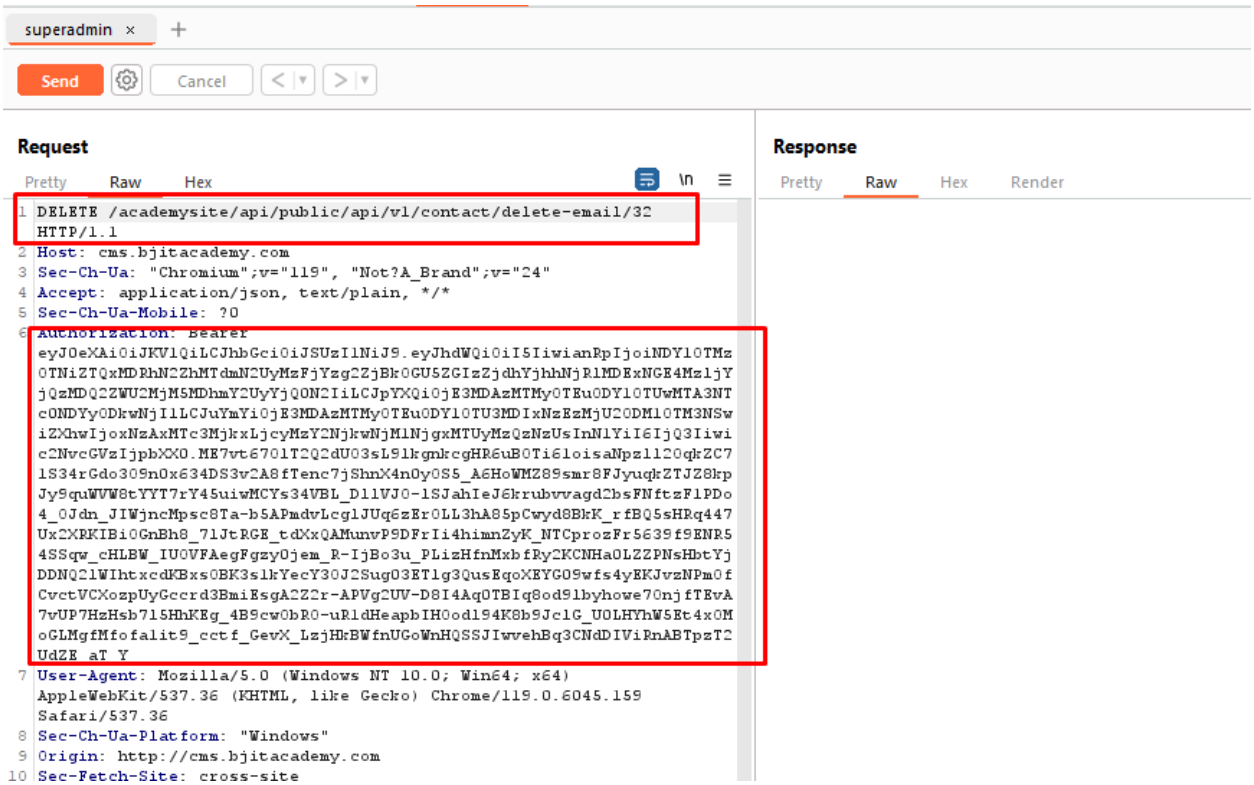
Let's try to delete a subscriber as Super Admin when burp intercept is on & capture the request and send it to the burp Repeater.



Delete this subscriber and capturing the request.



Now take this response in the Repeater and take a closer look to the Authorization Token.



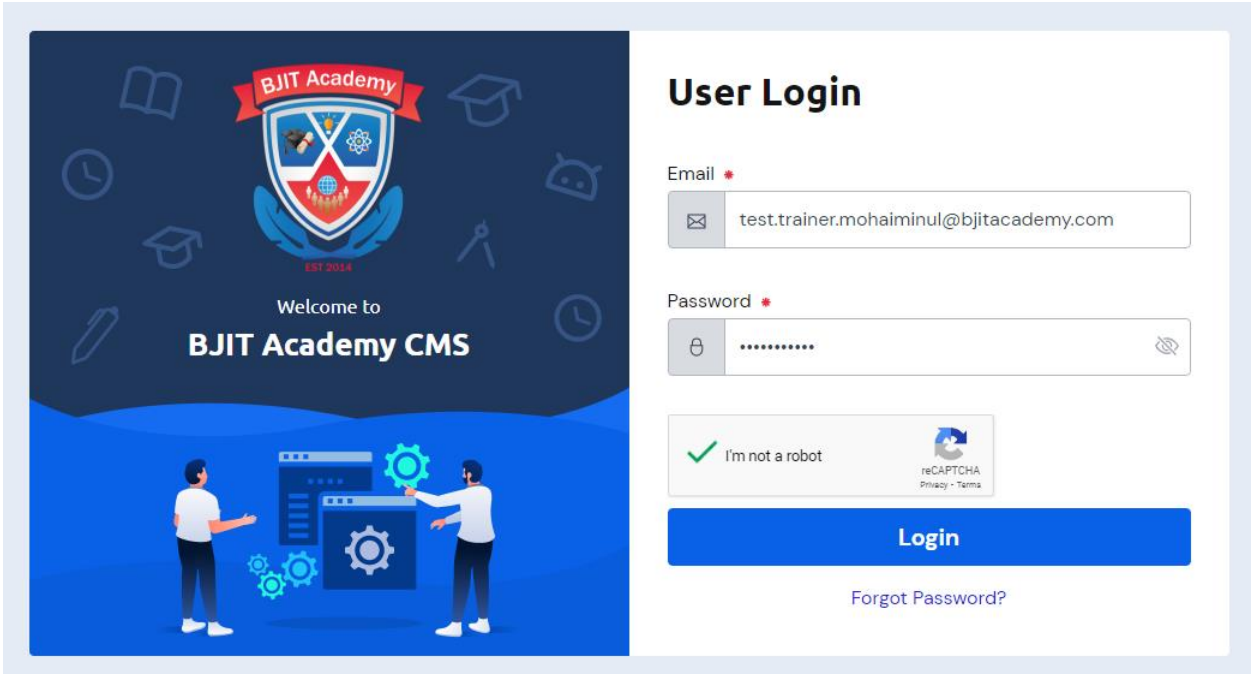
The token:

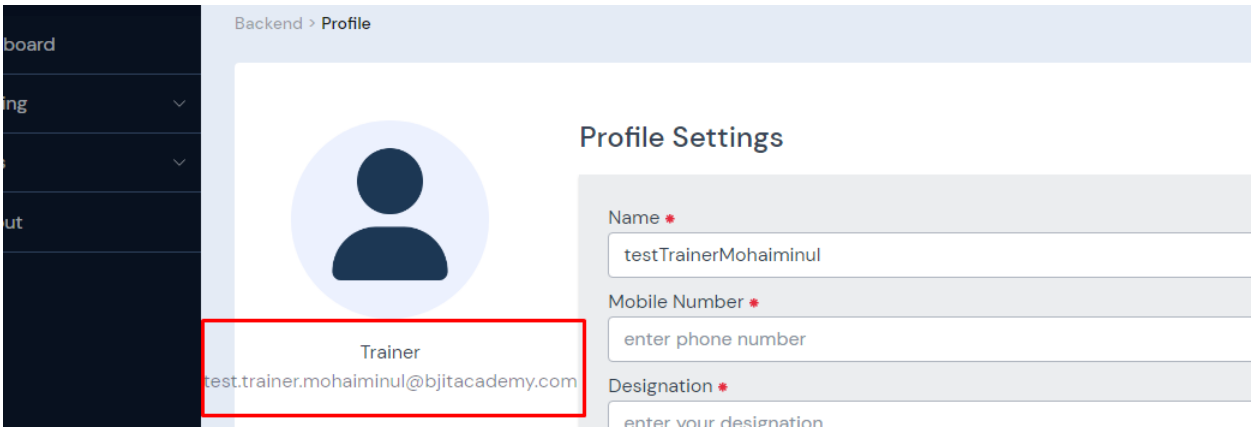
```
Authorization: Bearer
eyJ0eXAi0iJKV1QiLCJhbGci0iJSUzI1NiJ9.eyJhdWQi0iI5IiwianRpIjoiNDY10TMz
0TNiZTQxMDRhN2ZhMTdmN2UyMzFjYzg2ZjBk0GU5ZGIzZjdhYjhhNjR1MDExNGE4MzljY
jQzMDQ2ZWU2MjM5MDhmY2UyYjQ0N2IiLCJpYXQi0jE3MDAzMTMy0TEu0DY10TUwMTA3NT
c0NDYy0DkwNjI1LCJuYmYi0jE3MDAzMTMy0TEu0DY10TU3MDIxNzEzMjU20DM10TM3NSw
iZXhwIjoxNzAxMTc3MjkxLjcyMzY2NjkwNjM1NjgxMTUyMzQzNzUsInN1YiI6IjQ3Iiwi
c2NvcGVzIjpbXX0.ME7vt6701T2Q2dU03sL9lkgnkcgHR6uB0Ti6loisaNpzll20qkZC7
lS34rGdo309n0x634DS3v2A8fTenc7jShnX4n0y0S5_A6HoWMZ89smr8FJyuqkZTJZ8kp
Jy9quWVW8tYYT7rY45uiwMCYs34VBL_DllVJ0-lSJahIeJ6krubvvagd2bsFNftzF1PDo
4_0Jdn_JIWjncMpsc8Ta-b5APmdvLcglJUq6zEr0LL3hA85pCwyd8BkK_rfBQ5sHRq447
Ux2XRKIBi0GnBh8_7lJtRGE_tdXxQAMunvP9DFrIi4himnZyK_NTCprozFr5639f9ENR5
4SSqw_cHLBW_IU0VFAegFgzy0jem_R-IjBo3u_PLizHfnMxbfRy2KCNHa0LZZPNsHbtYj
DDNQ2lWIhtxcdKBxs0BK3slkYecY30J2Sug03ETlg3QusEqoXEYGO9wfs4yEKJvzNPm0f
CvctVCXozpUyGccrd3BmiEsgA2Z2r-APVg2UV-D8I4Aq0TBIq8od9lbyhowe70njfTEvA
7vUP7HzHsb7l5HhKEg_4B9cw0bR0-uRldHeapbIH0odl94K8b9JclG_U0LHYhW5Et4x0M
oGLMgfMfofalit9_cctf_GevX_LzjHrBWfnUGoWnHQSSJIwvehBq3CNdDIViRnABTpzT2
UdZE_aT_Y
```

Now I need a trainer's authorization token to check if he has any unauthorized access to delete the subscriber.

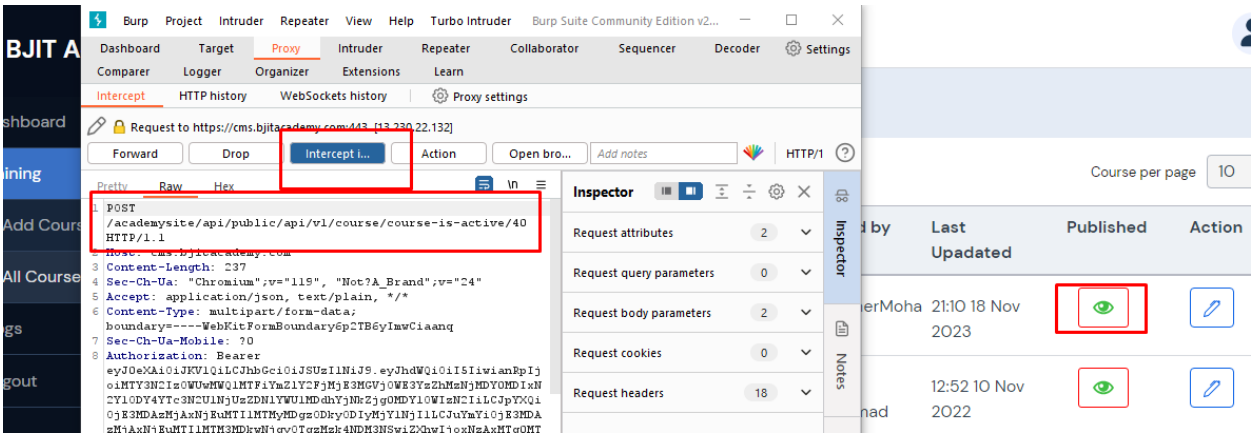
Now login with another user whose role is Trainer with the chromium browser's incognito tab.

Trainer Account Name: test.trainer.mohaiminul@bjitacademy.com

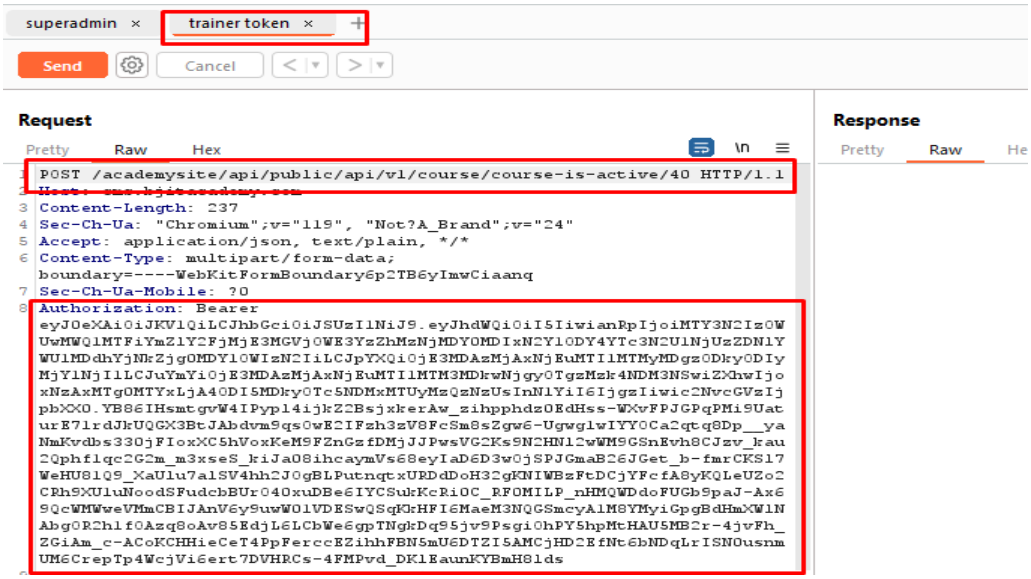




Capturing request for token:



Send it to the Repeater

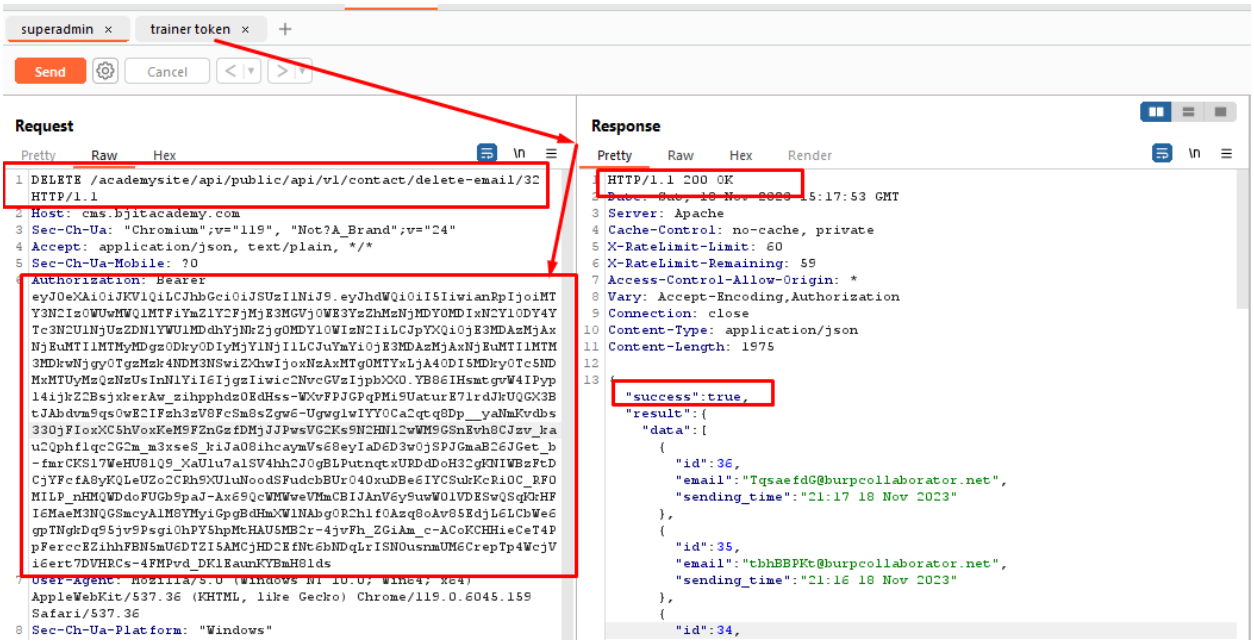


The trainer token:

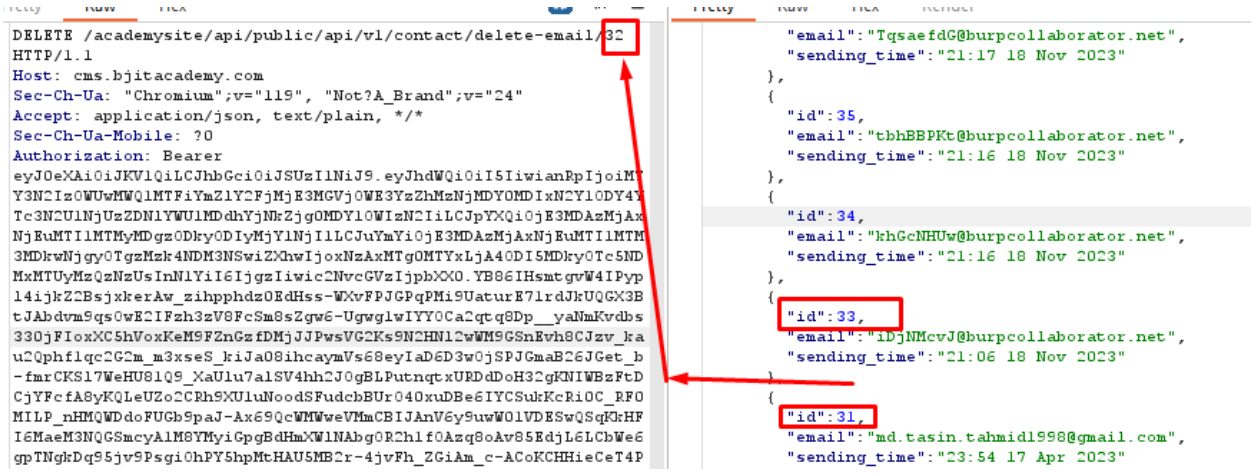
Authorization: Bearer
eyJ0eXAi0iJKV1QiLCJhbGci0iJSUzI1NiJ9.eyJhdWQi0iI5IiwianRpIjoiMTY3N2Iz0W
UwMWQ1MTFiYmZlY2FjMjE3MGVj0WE3YzZhMzNjMDY0MDIxN2Y10DY4YTc3N2U1NjUzZDN1Y
WU1MDdhYjNkZjg0MDY10WIzN2IiLCJpYXQi0jE3MDAzMjAxNjEuMTI1MTMyMDgz0Dky0DIy
MjY1NjI1LCJuYmYi0jE3MDAzMjAxNjEuMTI1MTM3MDkwNjgy0TgzMzk4NDM3NSwiZXhwIjo
xNzAxMTg0MTYxLjA40DI5MDky0Tc5NDMxMTUyMzQzNzUsInN1YiI6IjgzIiwic2NvcGVzIj
pbXX0.YB86IHsmtgvW4IPypl4ijkZ2BsjxkerAw_zihpphdz0EdHss-WXvFPJGPqPMi9Uat
urE7lrdJkUQGX3BtJAbdvm9qs0wE2IFzh3zV8FcSm8sZgw6-UgwglwIYY0Ca2qtq8Dp__ya
NmKvdbs330jFIoxXC5hVoxKeM9FZnGzfDMjJJPwsVG2Ks9N2HNl2wWM9GSnEvh8CJzv_kau
2Qphflqc2G2m_m3xseS_kiJa08ihcaymVs68eyIaD6D3w0jSPJGmaB26JGet_b-fmrCKSl7
WeHU81Q9_XaUlu7alSV4hh2J0gBLPutnqtxURDdDoH32gKNIWBzFtDCjYFcfA8yKQLeUZo2
CRh9XUluNoodSFudcbBUr040xuDBe6IYCSukKcRi0C_RF0MILP_nHMQWDdoFUGb9paJ-Ax6
9QcWMWweVMmCBIJAnV6y9uwW0lVDESwQSqKrHFI6MaeM3NQGSmcyAlM8YMyiGpgBdHmXWlN
Abg0R2hlf0Azq8oAv85EdjL6LCbWe6gpTNgkDq95jv9Psgi0hPY5hpMtHAU5MB2r-4jvFh_
ZGiAm_c-ACoKCHHieCeT4PpFerccEZihhFBN5mU6DTZI5AMCjHD2EfNt6bNDqLrISN0usnm
UM6CrepTp4WcjVi6ert7DVHRCs-4FMPvd_DKlEaunKYBmH8lds

Now I will copy the trainer token and replace the super admin token with it. Let's delete the subscriber with the modified request

Delete operation is performed successfully with trainer authentication token



Here 32 number subscribers are the victims of that operation which is shown in the below picture.



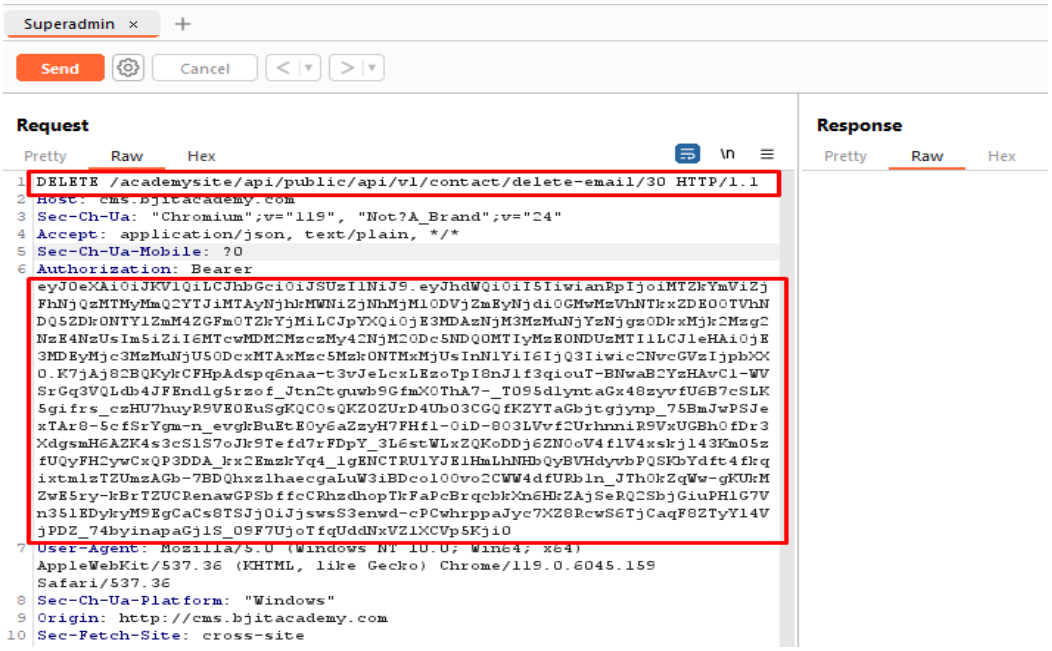
So, by changing the authorization token trainer level user get unauthorized access.

Similarly SEO Manager, Admin and Content Manager's authorization token can be used to perform this task.

From content Manager:

Here delete request is taken to the burp repeater with the similar way of the previous task.

The below page has the super admin's authorization token

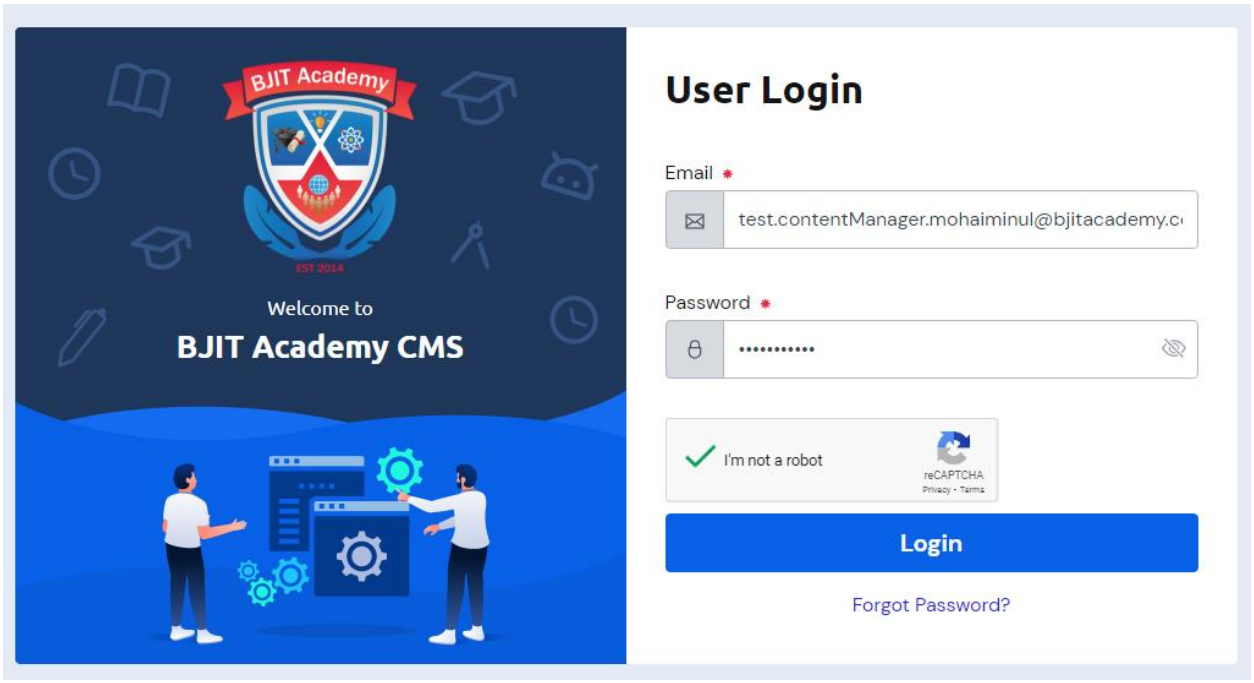


This is the Super Admin Authorization Token

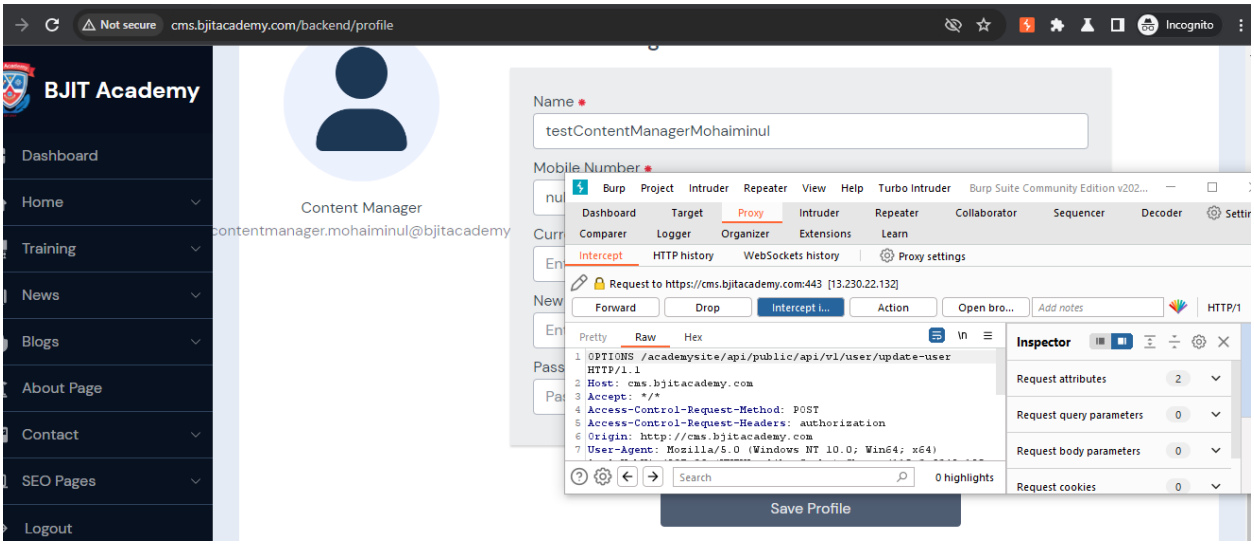
Now Collect Content Manager Authorization Token.

From Content Manager,

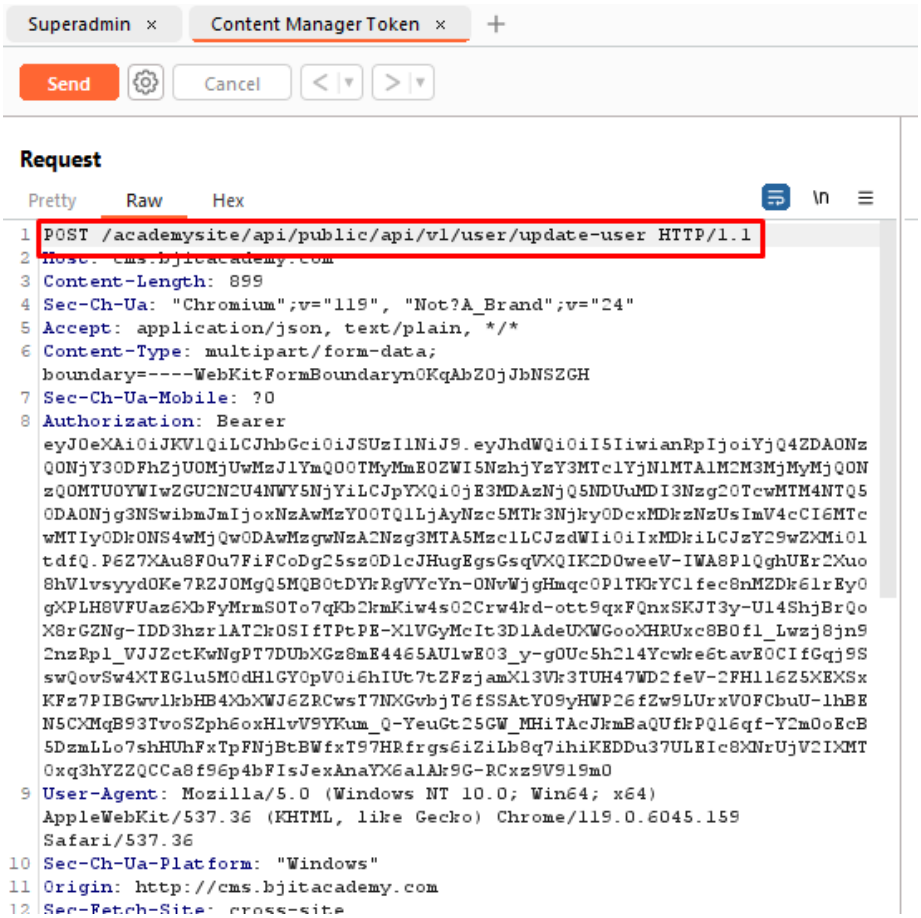
Account of Content Manager:
test.contentmanager.mohaiminul@bjitacademy.com



Go to the profile page then capture the request by intercept on.



Now take this request to the repeater.

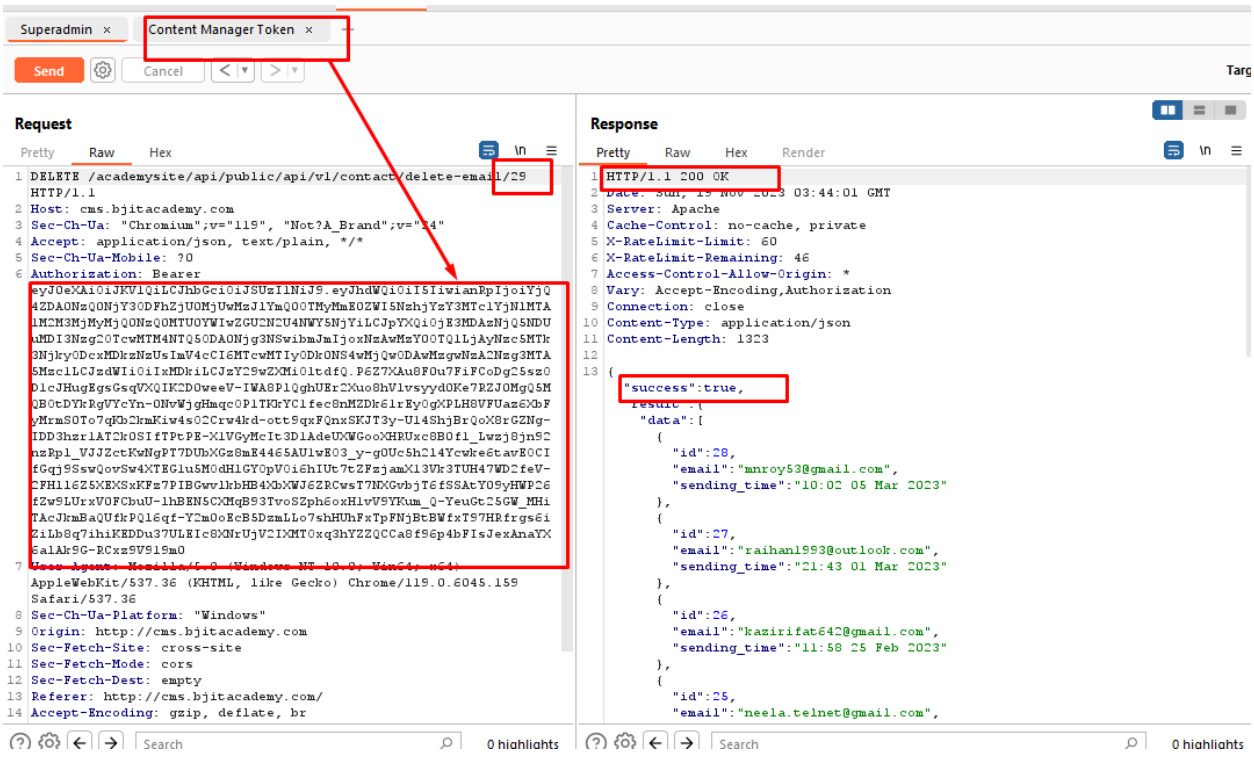


The Token of Content Manager:


```
eyJ0eXAi0iJKV1QiLCJhbGci0iJSUzI1NiJ9.eyJhdWQi0iI5IiwianRpIjoiYjQ4ZDA0Nz
Q0NjY30DFhZjU0MjUwMzJlYmQ00TMyMmE0ZWI5NzhjYzY3MTclYjN1MTA1M2M3MjMyMjQ0N
zQ0MTU0YWIwZGU2N2U4NWY5NjYiLCJpYXQi0jE3MDAzNjQ5NDUuMDI3Nzg20TcwMTM4NTQ5
0DA0Njg3NSwibmJmIjoxNzAwMzY00TQ1LjAyNzc5MTk3Njky0DcxMDkzNzUsImV4cCI6MTc
wMTIy0Dk0NS4wMjQw0DAwMzgwNzA2Nzg3MTA5Mzc1LCJzdWIi0iIxMDkiLCJzY29wZXMi0l
tdfQ.P6Z7XAu8F0u7FiFCoDg25sz0DlcJHugEgsGsqVXQIK2D0weeV-IWA8PlQghUEr2Xuo
8hVlvsyyd0Ke7RZJ0MgQ5MQB0tDYkRgVYcYn-0NvWjgHmqc0PlTKkYClfec8nMZDk6lrEy0
gXPLH8VFUaz6XbFyMrmS0To7qKb2kmKiw4s02Crw4kd-ott9qxFQnxSKJT3y-Ul4ShjBrQo
X8rGZNg-IDD3hzrlAT2k0SIfTPtPE-XlVGyMcIt3DlAdeUXWGooXHRUxc8B0fl_Lwzj8jn9
2nzRpl_VJJZctKwNgPT7DUbXGz8mE4465AUlwE03_y-gOUc5h2l4Ycwke6tavE0CIfGqj9S
swQovSw4XTEGlu5M0dHlGY0pV0i6hIUt7tZFzjamXl3Vk3TUH47WD2feV-2FHll6Z5XEXSx
KFz7PIBGwvlkbHB4XbXWJ6ZRCwsT7NXGvbjT6fSSAtY09yHWP26fZw9LUrxV0FCbuU-lhBE
N5CXMqB93TvoSZph6oxHlvV9YKum_Q-YeuGt25GW_MHiTAcJkmBaQUfkPQl6qf-Y2m0oEcB
5DzmLLo7shHUhFxTpFNjBtBWfxT97HRfrgs6iZiLb8q7ihiKEDDu37ULEIc8XNrUjV2IXMT
0xq3hYZZQCCa8f96p4bFIsJexAnaYX6alAk9G-RCxz9V919m0
```


Now Replace the super admin's Authorization token with the Content Manager's Authorization token in that API in the repeater:

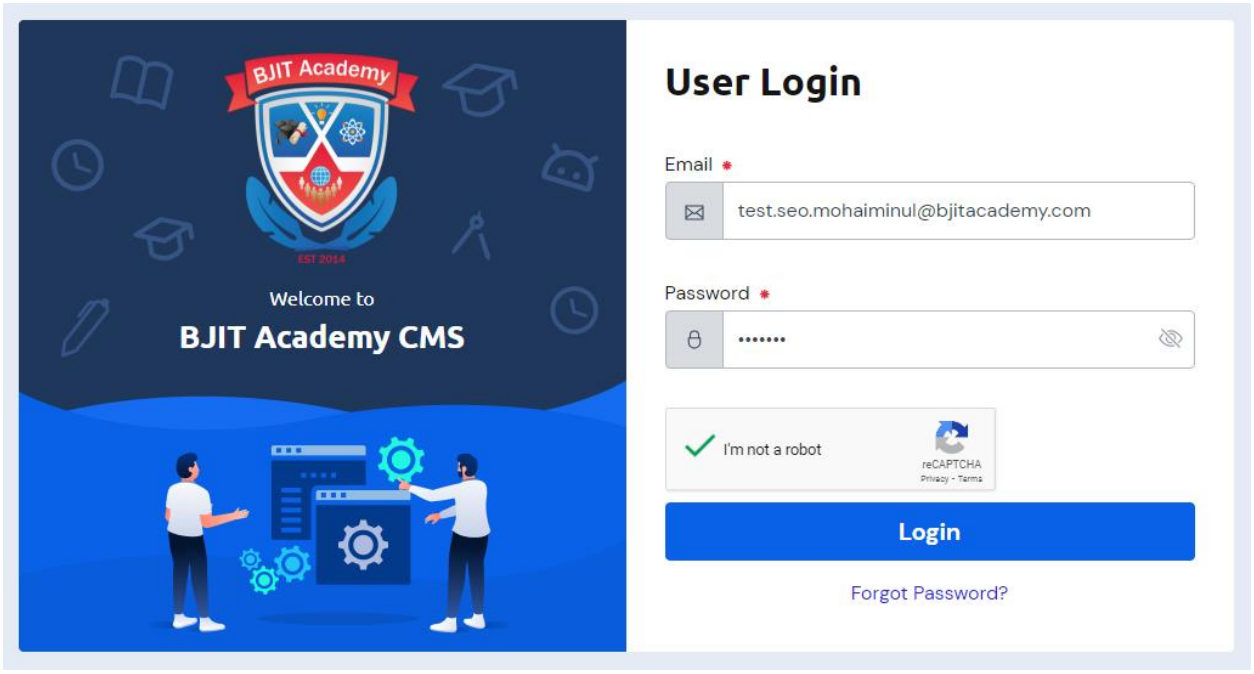
DELETE /academysite/api/public/api/v1/contact/delete-email/29



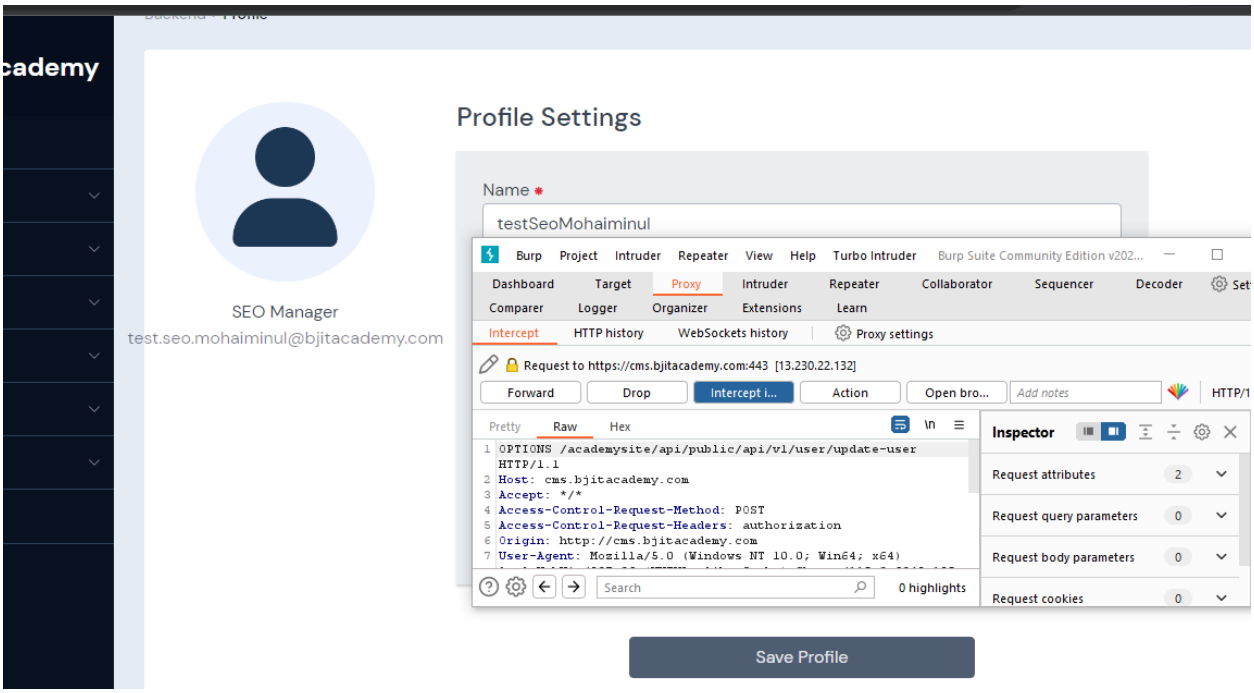
So the Content Manager has unauthorized access here.

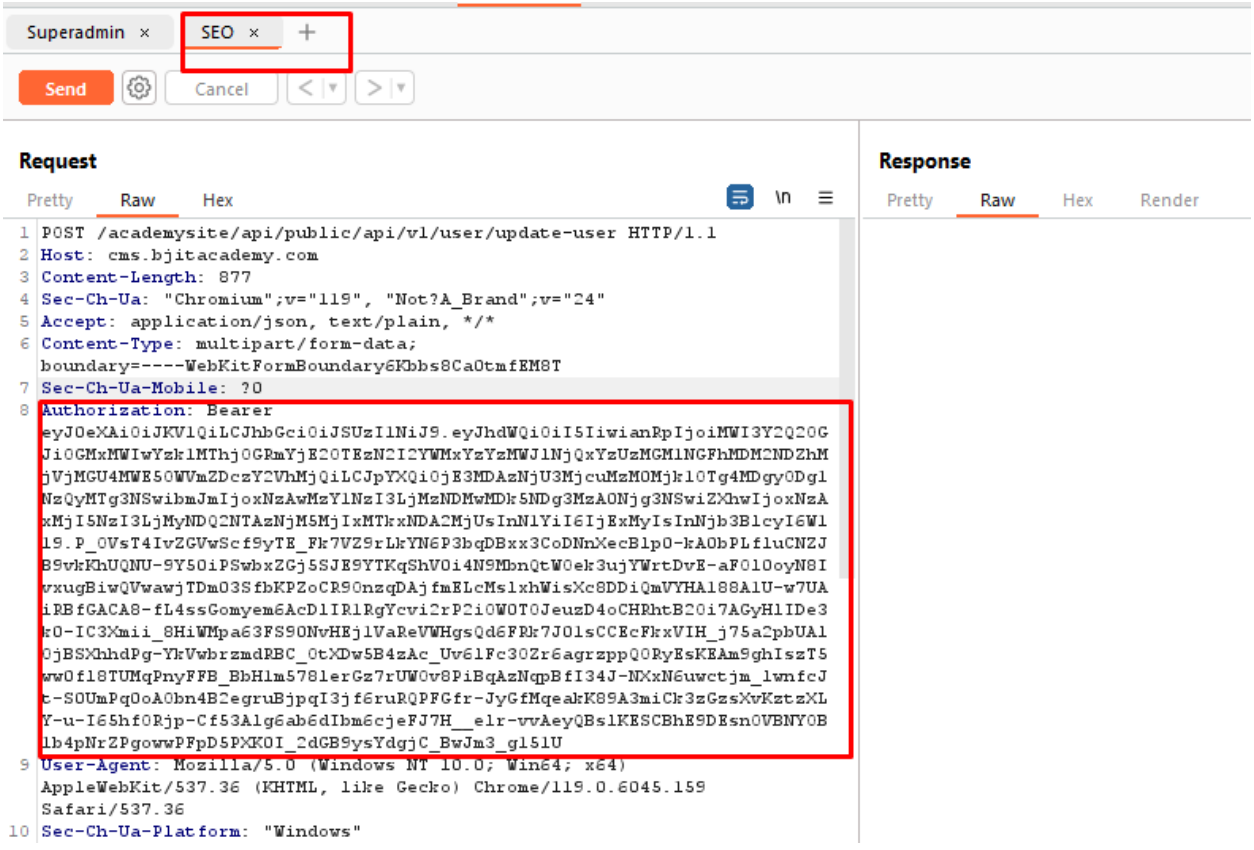
Now From SEO Manager credential:

Account of SEO Manager: test.seo.mohaiminul@bjitacademy.com



Go to profile page ,capture the request and take the request in the burp repeater like previous way,



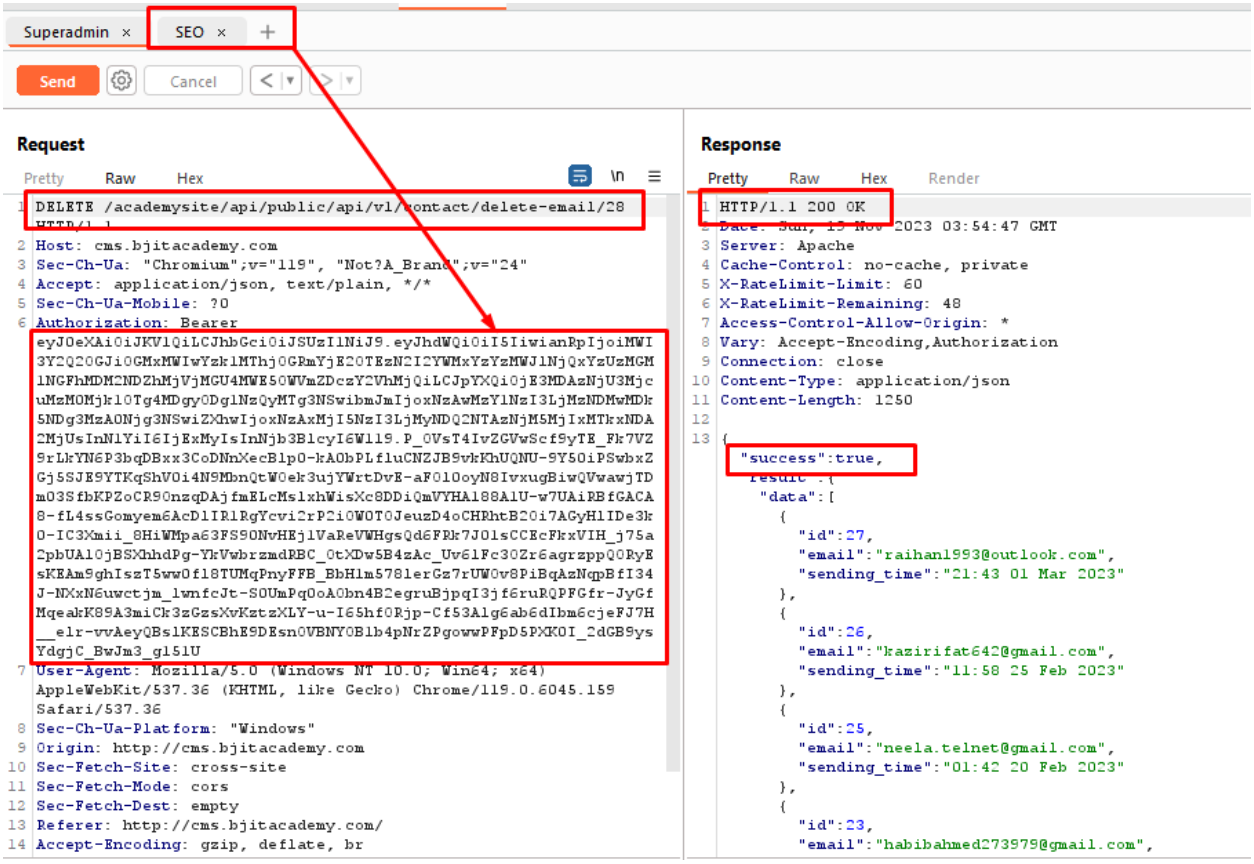


Here SEO Manager Authorization Token is :

```
eyJ0eXAi0iJKV1QiLCJhbGci0iJSUzI1NiJ9.eyJhdWQi0iI5IiwianRpIjoiMWI3Y2Q20G
Ji0GMxMWIwYzk1MThj0GRmYjE20TEzN2I2YWMxYzYzMWJlNjQxYzUzMGM1NGFhMDM2NDZhM
jVjMGU4MWE50WVmZDczY2VhMjQiLCJpYXQi0jE3MDAzNjU3MjcuMzM0Mjk10Tg4MDgy0Dgl
NzQyMTg3NSwibmJmIjoxNzAwMzY1NzI3LjMzNDMwMDk5NDg3MzA0Njg3NSwiZXhwIjoxNzA
xMjI5NzI3LjMyNDQ2NTAzNjM5MjIxMTkxNDA2MjUsInN1YiI6IjExMyIsInNjb3BlcyI6Wl
19.P_0VsT4IvZGVwScf9yTE_Fk7VZ9rLkYN6P3bqDBxx3CoDNnXecBlp0-kA0bPLfluCNZJ
B9vkKhUQNU-9Y50iPSwbxZGj5SJE9YTKqShV0i4N9MbnQtW0ek3ujYWrtDvE-aF0l0oyN8I
vxugBiwQVwawjTDm03SfbKPZoCR90nzqDAjfmELcMslxhWisXc8DDiQmVYHA188AlU-w7UA
iRBfGACA8-fL4ssGomyem6AcDlIRlRgYcvi2rP2i0W0T0JeuzD4oCHRhtB20i7AGyHlIDe3
k0-IC3Xmii_8HiWMpa63FS90NvHEjlVaReVWHgsQd6FRk7J0lsCCEcFkxVIH_j75a2pbUAl
0jBSXhhdPg-YkVwbrzmdRBC_0tXDw5B4zAc_Uv6lFc30Zr6agrzppQ0RyEsKEAm9ghIszT5
ww0fl8TUMqPnyFFB_BbHlm578lerGz7rUW0v8PiBqAzNqpBfI34J-NXxN6uwctjm_lwnfcJ
t-S0UmPq0oA0bn4B2egruBjpqI3jf6ruRQPFGfr-JyGfMqeakK89A3miCk3zGzsXvKztzXL
Y-u-I65hf0Rjp-Cf53Alg6ab6dIbm6cjeFJ7H__elr-vvAeyQBslKESCBhE9DEsn0VBNY0B
lb4pNrZPgowwPFpD5PXK0I_2dGB9ysYdgjC_BwJm3_gl51U
```


Now Replace the super admin's Authorization token with the SEO Manager's Authorization token in that API in the repeater:

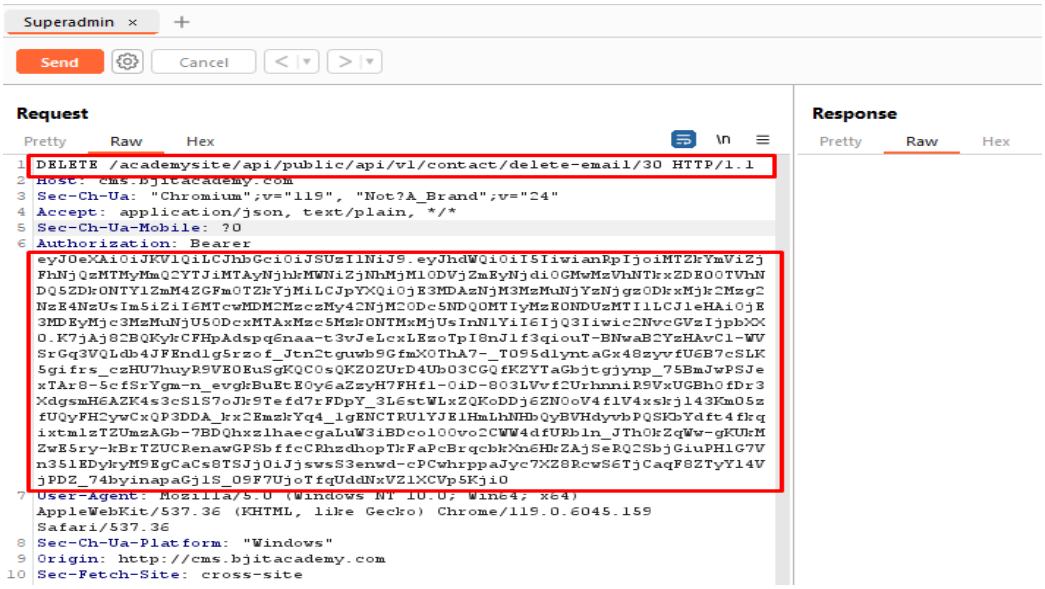
DELETE /academysite/api/public/api/v1/contact/delete-email/28



So SEO Manager also has unauthorized access to delete a subscriber.

Now from admin:

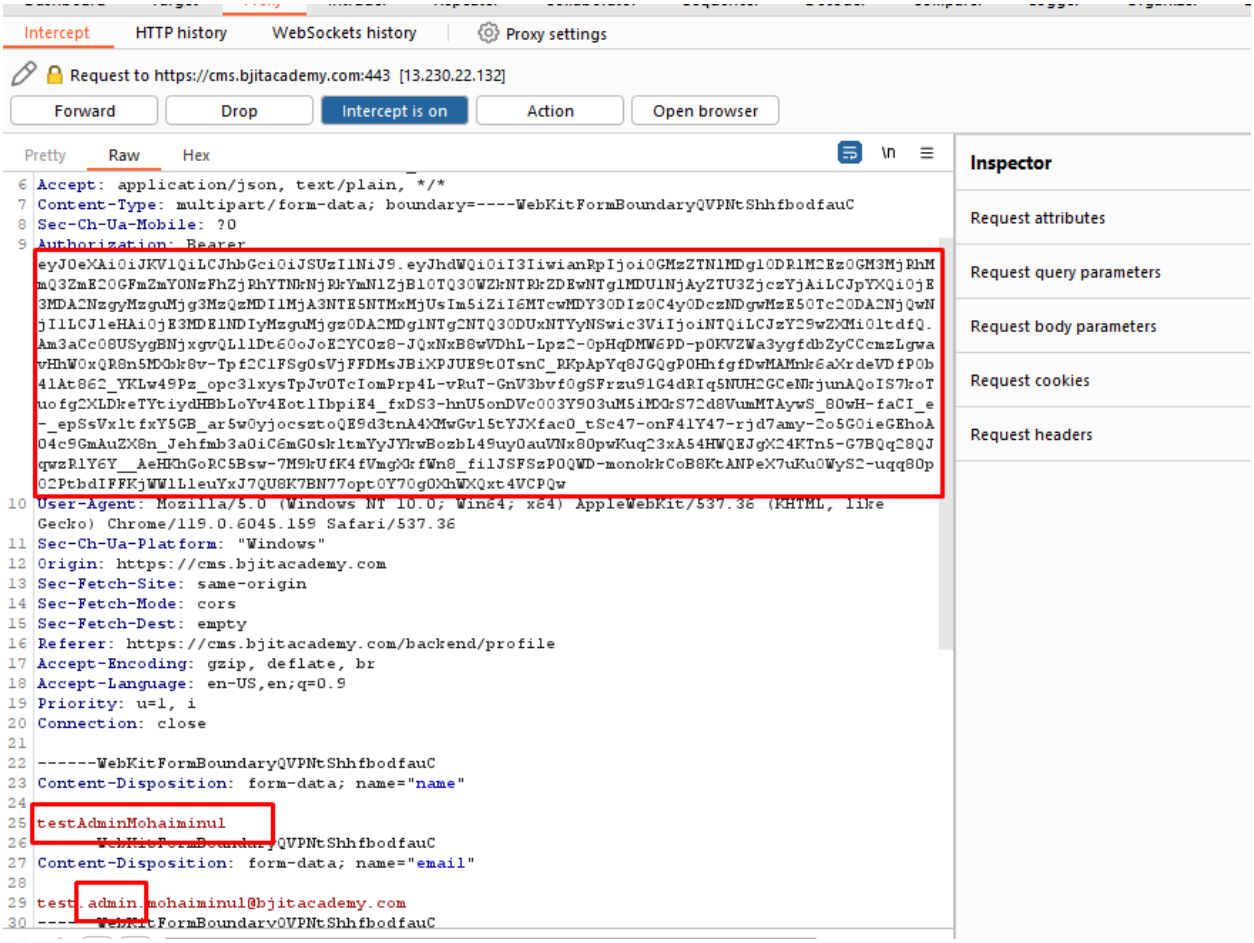
The below page has the super admin's authorization token



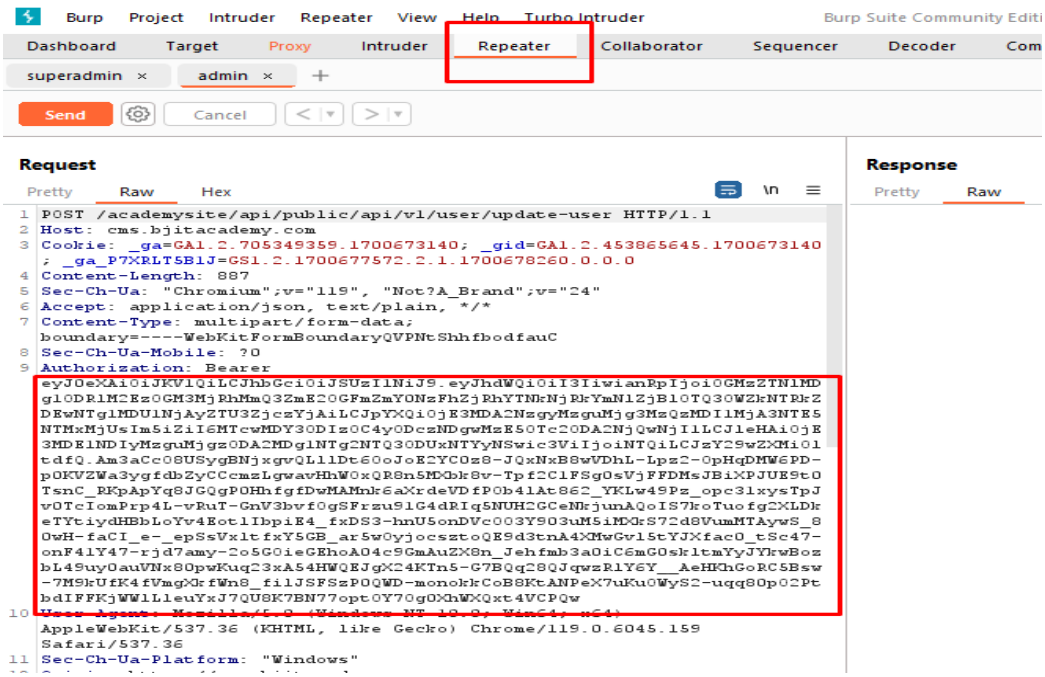
This is the Super Admin Authorization Token

Now I have to collect the admin token. For that I logged in as admin and then went to profile and captured the profile update request. My main purpose is to collect the admin authentication token.

Account of Content Manager: test.admin.mohaiminul@bjitacademy.com



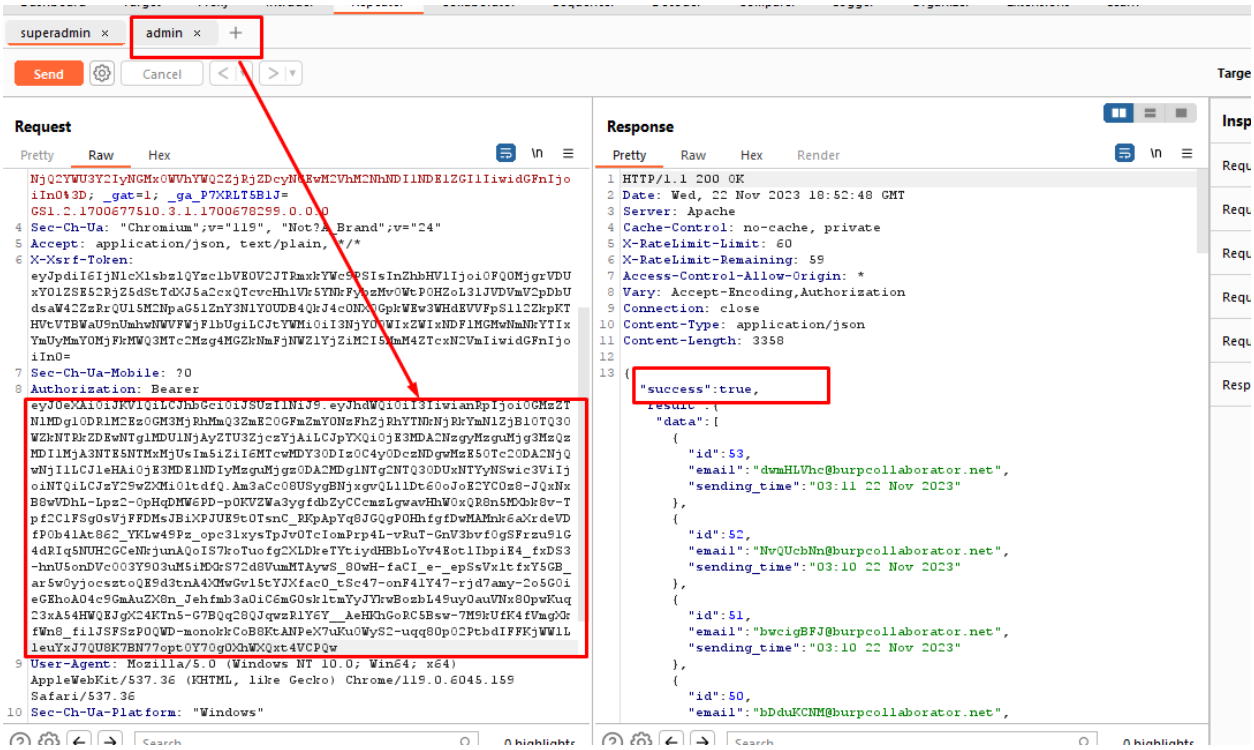
Now take that request to the burp repeater.

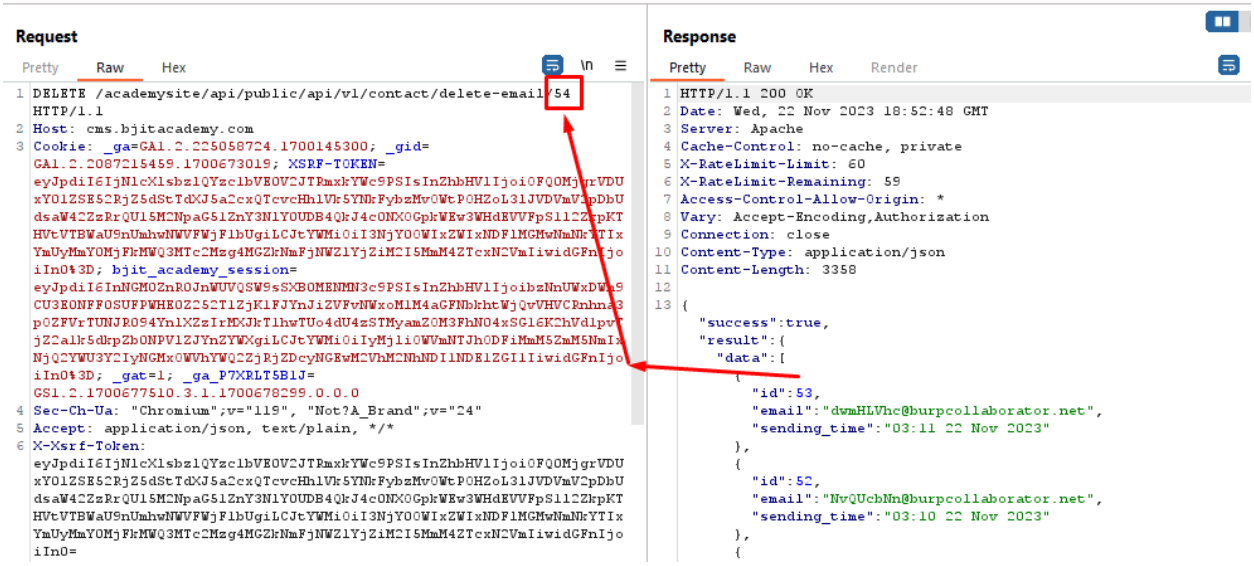


Here the admin token:

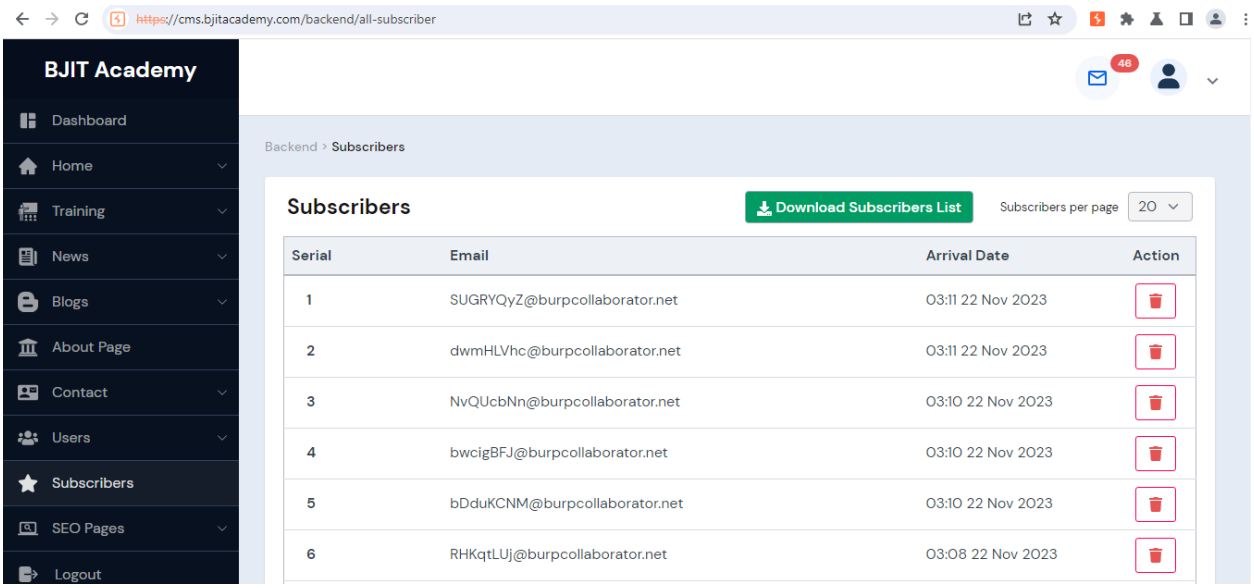

```
eyJ0eXAi0iJKV1QiLCJhbGci0iJSUzI1NiJ9.eyJhdWQi0iI3IiwianRpIjoi0GMzZTNlMD
gl0DRlM2Ez0GM3MjRhMmQ3ZmE20GFmZmY0NzFhZjRhYTNkNjRkYmNlZjB10TQ30WZkNTRkZ
DEwNTglMDUlNjAyZTU3ZjczYjAiLCJpYXQi0jE3MDA2NzgyMzguMjg3MzQzMDIlMjA3NTE5
NTMxMjUsIm5iZiI6MTcwMDY30DIz0C4y0DczNDgwMzE50Tc20DA2NjQwNjIlLCJleHAi0jE
3MDElNDIyMzguMjgz0DA2MDglNTg2NTQ30DUxNTYyNSwic3ViIjoiNTQiLCJzY29wZXMi0l
tdfQ.Am3aCc08USygBNjxgvQLllDt60oJoE2YCOz8-JQxNxB8wVDhL-Lpz2-0pHqDMW6PD-
p0KVZWa3ygfdbZyCCcmzLgwavHhW0xQR8n5MXbk8v-Tpf2ClFSg0sVjFFDMsJBiXPJUE9t0
TsnC_RKpApYq8JGQgP0HhfgfDwMAMnk6aXrdeVDfP0b4lAt862_YKLw49Pz_opc3lxysTpJ
v0TcIomPrp4L-vRuT-GnV3bvf0gSFrzu91G4dRIq5NUH2GCeNrjunAQoIS7koTuofg2XLDk
eTYtiydHBbLoYv4EotlIbpiE4_fxDS3-hnU5onDVc003Y903uM5iMXkrS72d8VumMTAywS_8
0wH-faCI_e-_epSsVxltfxY5GB_ar5w0yjocsztoQE9d3tnA4XMwGvl5tYJXfac0_tSc47-
onF41Y47-rjd7amy-2o5G0ieGEhoA04c9GmAuZX8n_Jehfmb3aOiC6mG0skltmYyJYkrwBoz
bL49uy0auVNx80pwKuq23xA54HWQEJgX24KTn5-G7BQq28QJqwzRlY6Y__AeHKhGoRC5Bsw
-7M9kUfK4fVmgXkfWn8_filJSFSzP0QWD-monokkCoB8KtANPeX7uKu0WyS2-uqq80p02Pt
bdIFFKjWWlLleuYxJ7QU8K7BN77opt0Y70g0XhWXQxt4VCPQw
```


Now replace the super admin's authorization token with admin's authorization token.

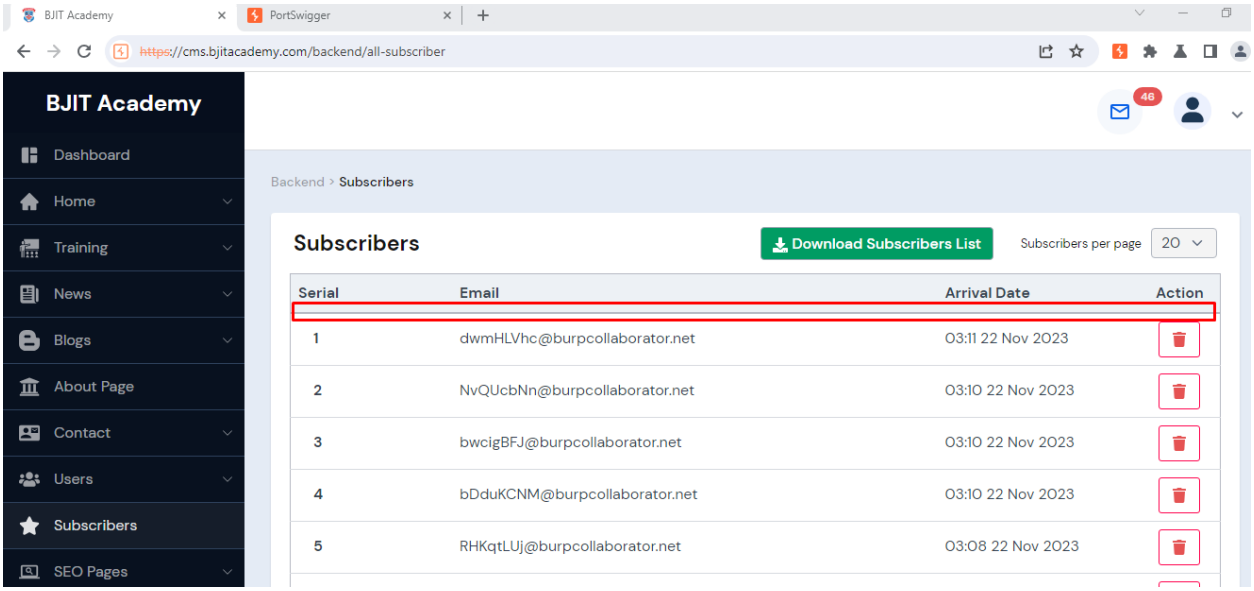




Before delete



After delete



The three subscriber deleted by SEO and Content Manager, Admin and also deleted from the frontend.

3. **Title:** Normal User (Trainer, Content Manager, SEO Manager) has unauthorized access of adding information by replacing the user_id of Super Admin (Access control violation).

   Target: cms.bjitacademy.com

   Affected URL/API:

   POST /academysite/api/public/api/v1/client/store-client(Add Client)

   POST /academysite/api/public/api/v1/post/create-slider-post (Add Banner)

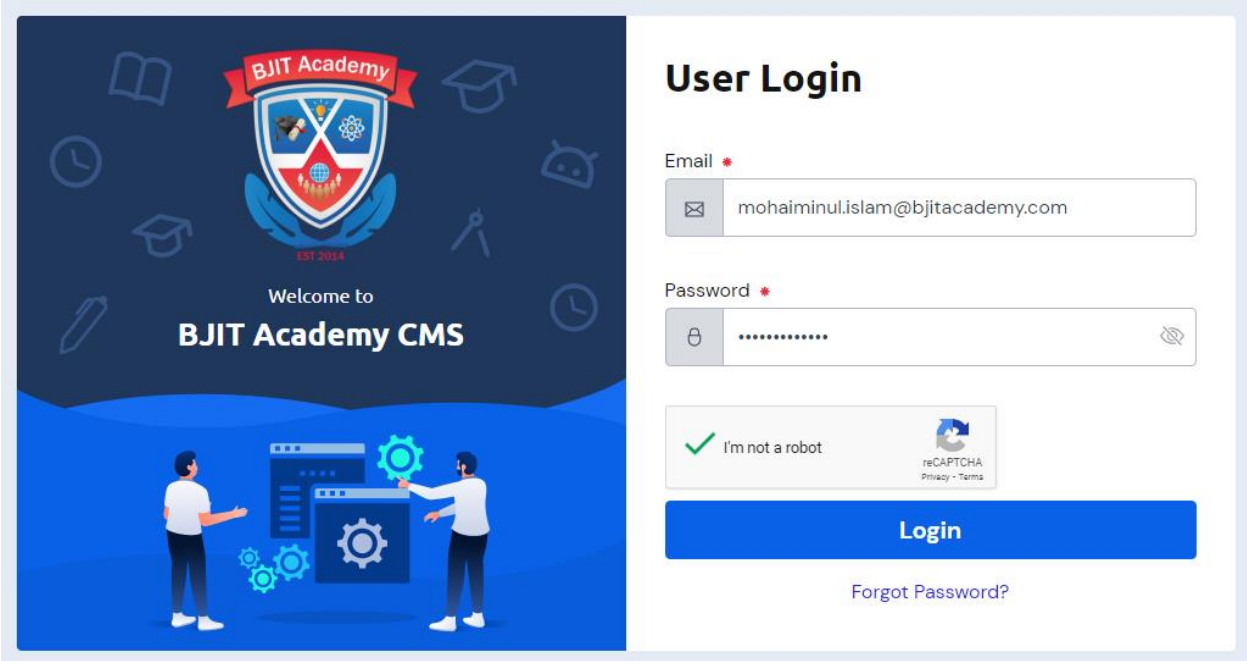
   POST /academysite/api/public/api/v1/post/create-fresh-talent-scope (youth skill page)

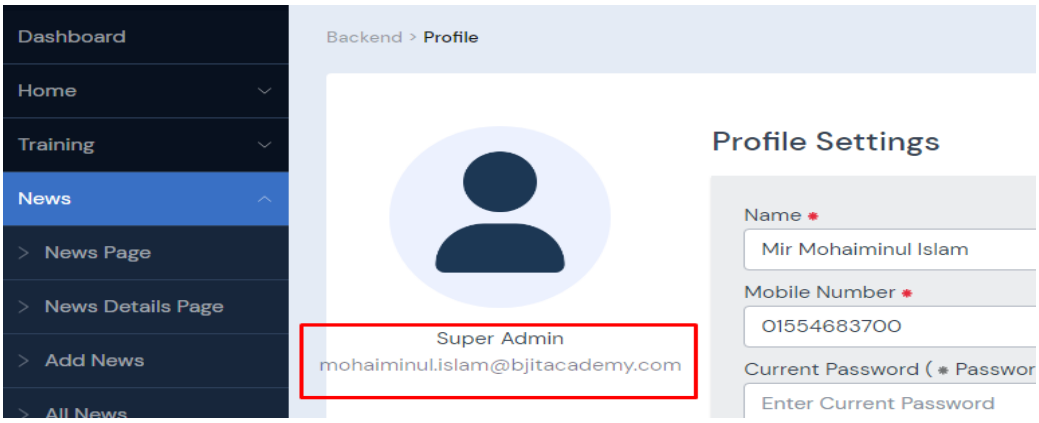
   POST /academysite/api/public/api/v1/testimonial/create-testimonial (For testimonial page)

   Summary: In the website normal user can add information of client adding, add banner, content of youth skill page and testimonial page content from the backend API which is restricted from the frontend of the website.

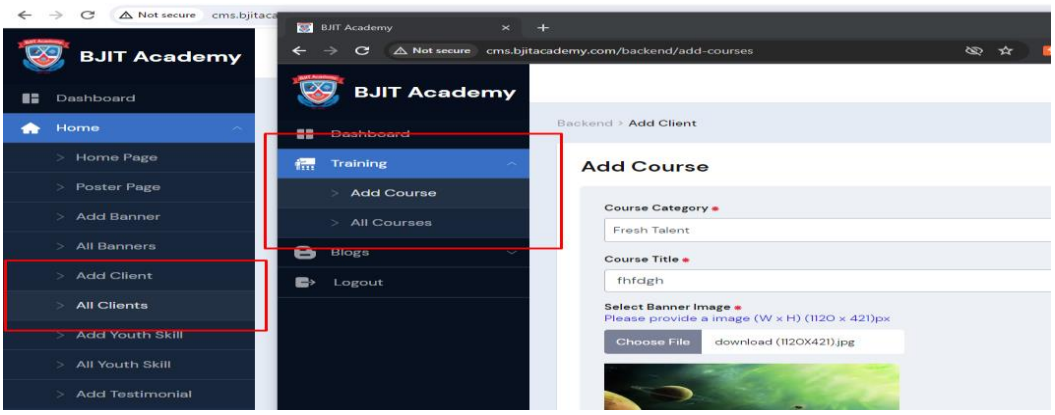
   Proof of Concept:

   At first go to http://cms.bjitacademy.com/login and login with super admin credential

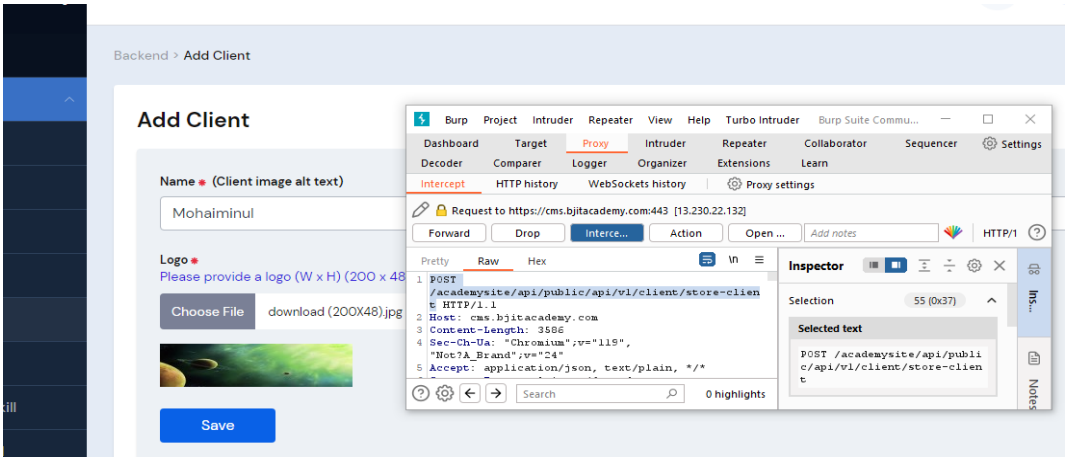




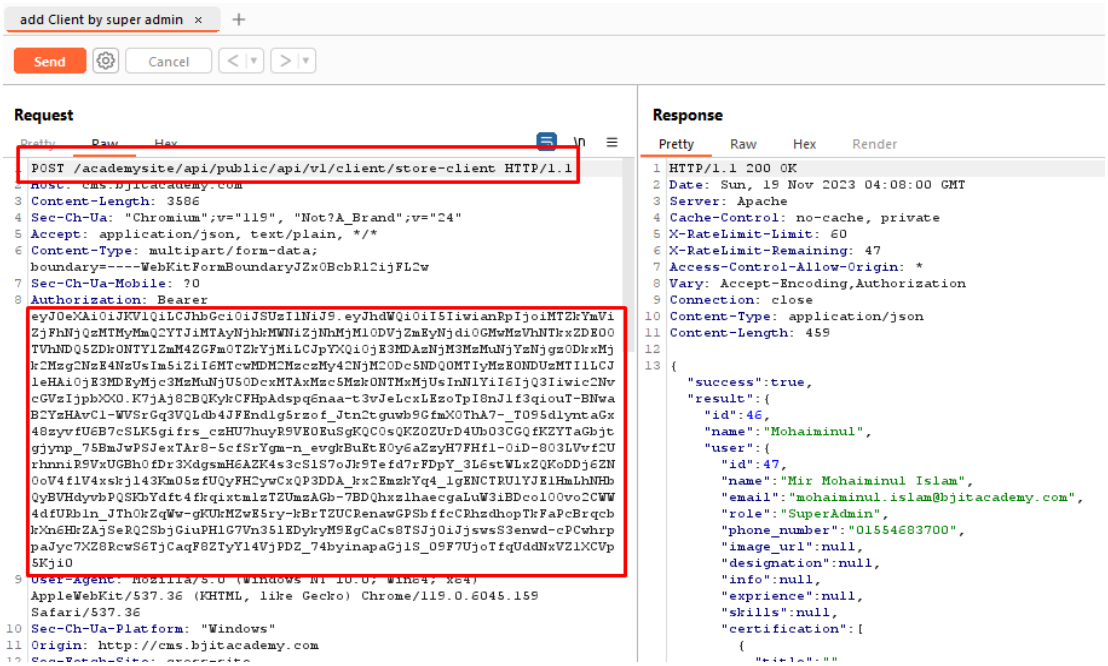
Here only super admin has the access of adding client information



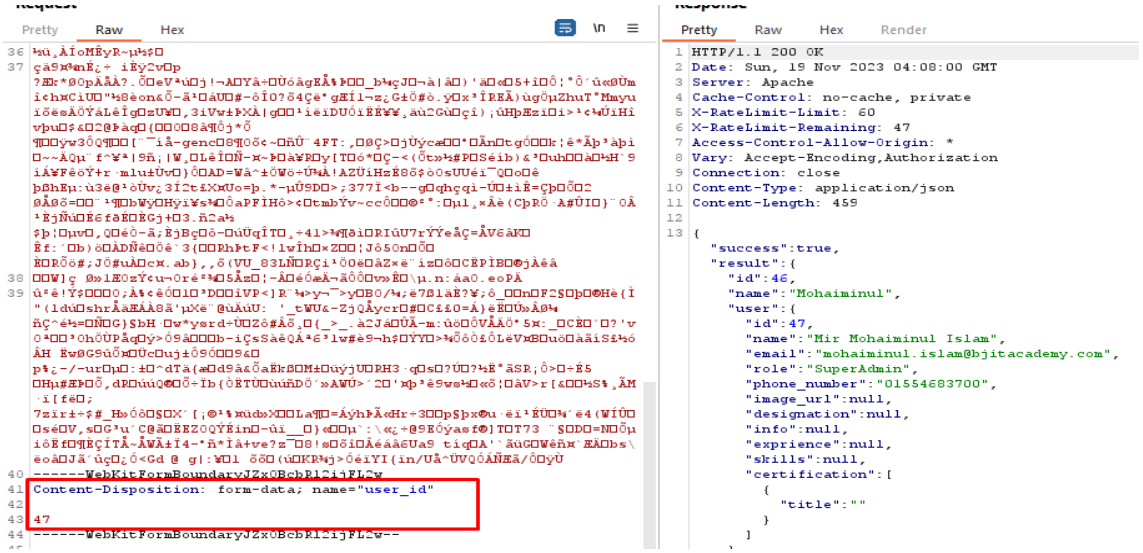
After login go to the Home page → Add Client & add a client information and capture the request.



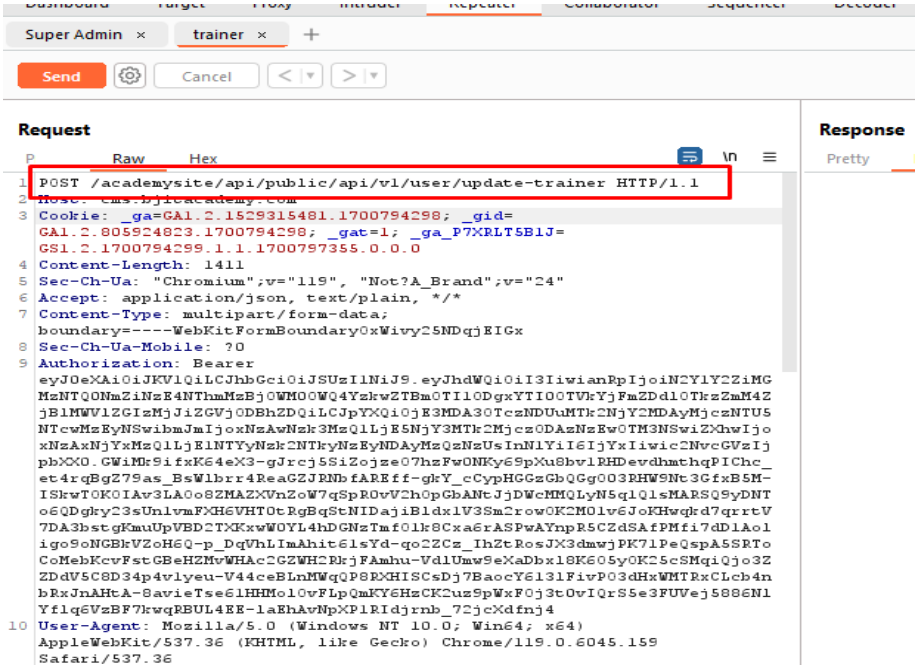
Now take this request in the burp repeater

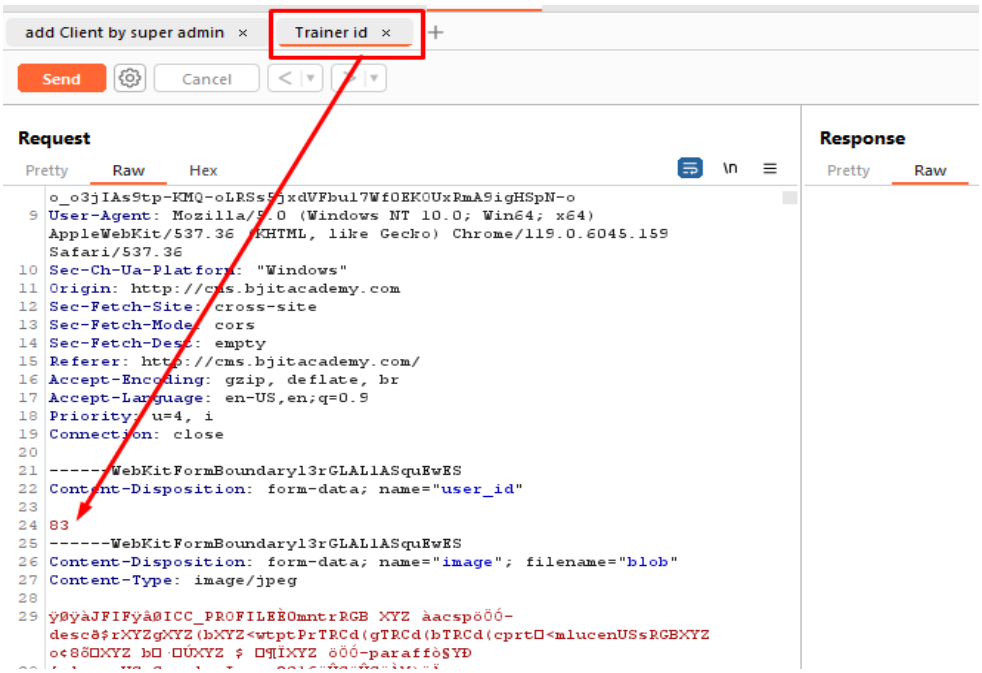


Now get the user_id of the super admin which is 47

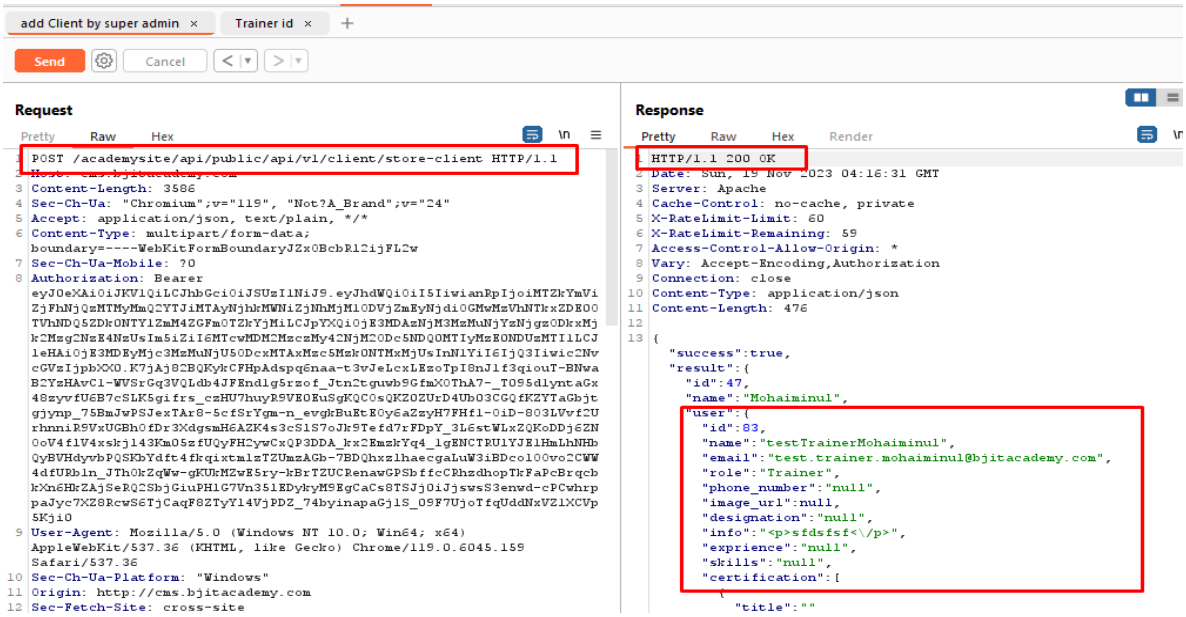


Now login with another user(Trainer) and capture its user_id.

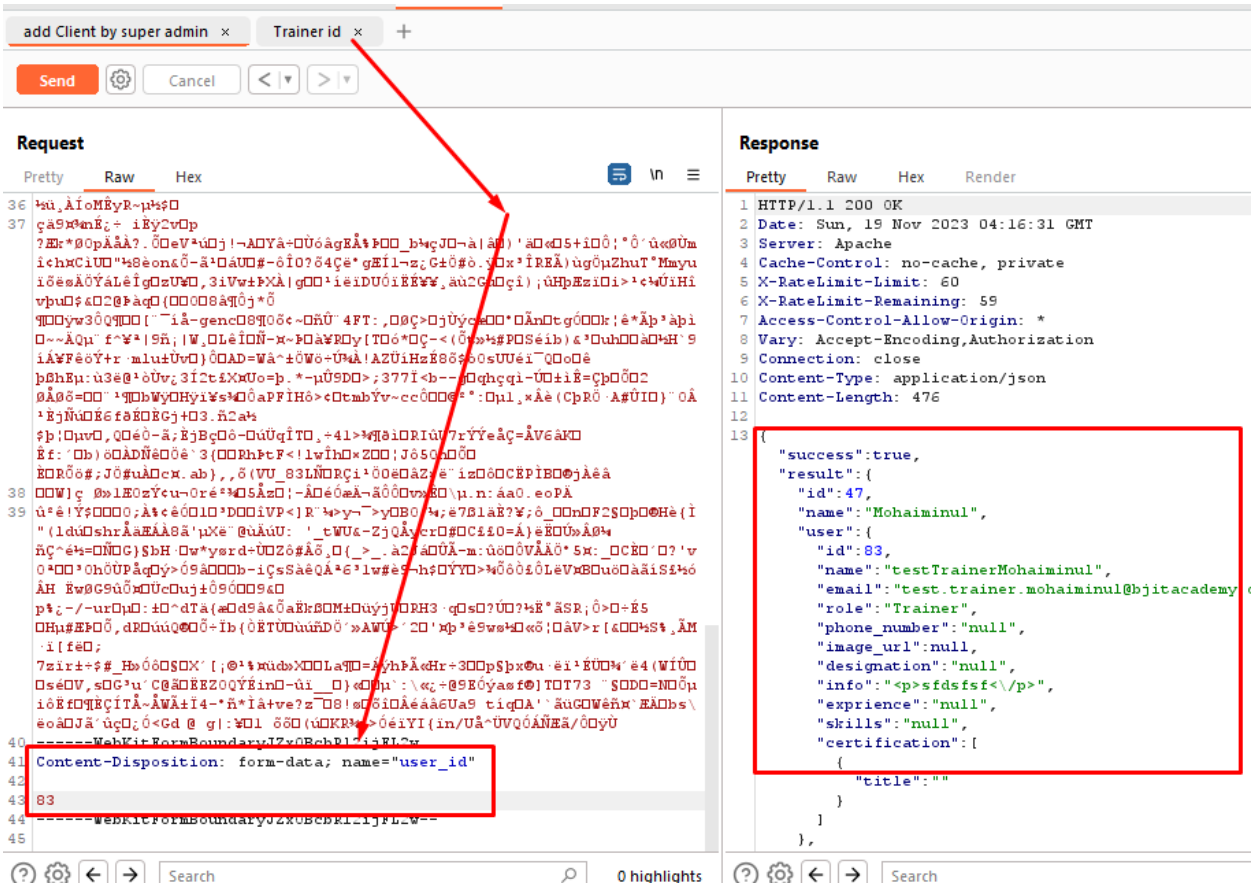




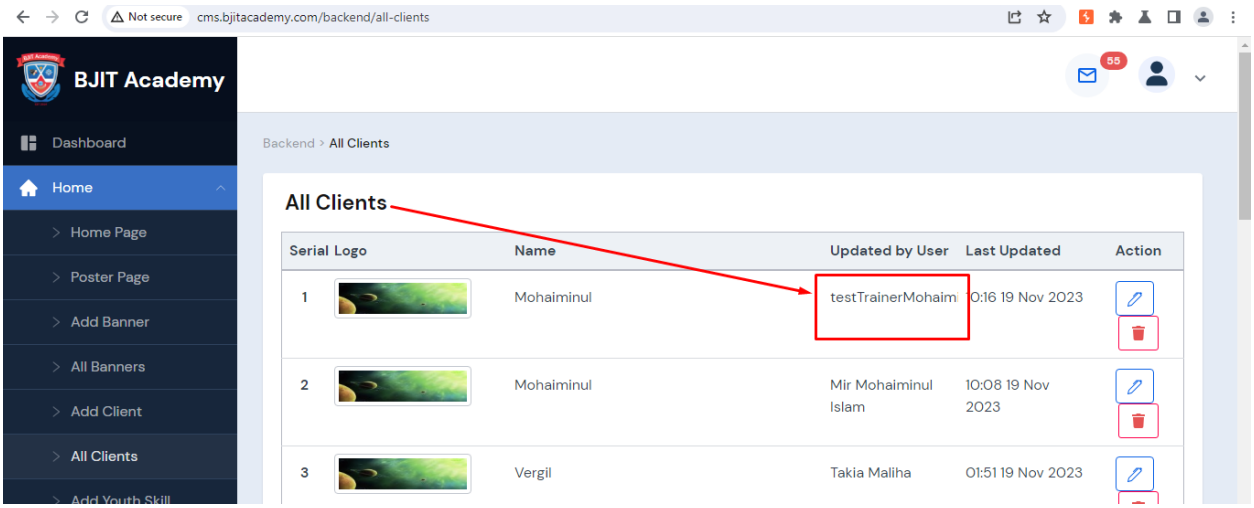
Here trainer id is 83. And the API where I get that trainer id: **POST /academysite/api/public/api/v1/user/update-trainer** Let's replace the super admin user id with that user_id in that API **POST /academysite/api/public/api/v1/client/store-client**.



Here I use super admin Authorization token and the trainer's user_id(83).



So user get unauthorized access in the adding client information by manipulating user_id.

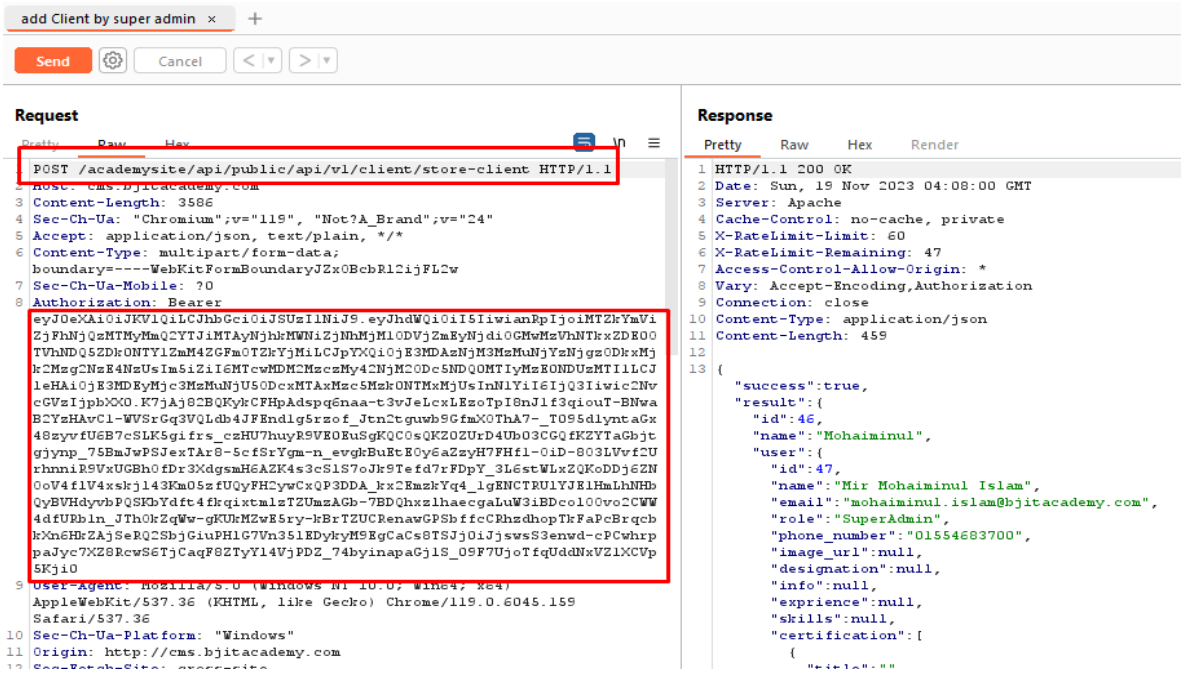


For SEO Manager:

SEO account: test.seo.mohaiminul@bjitacademy.com

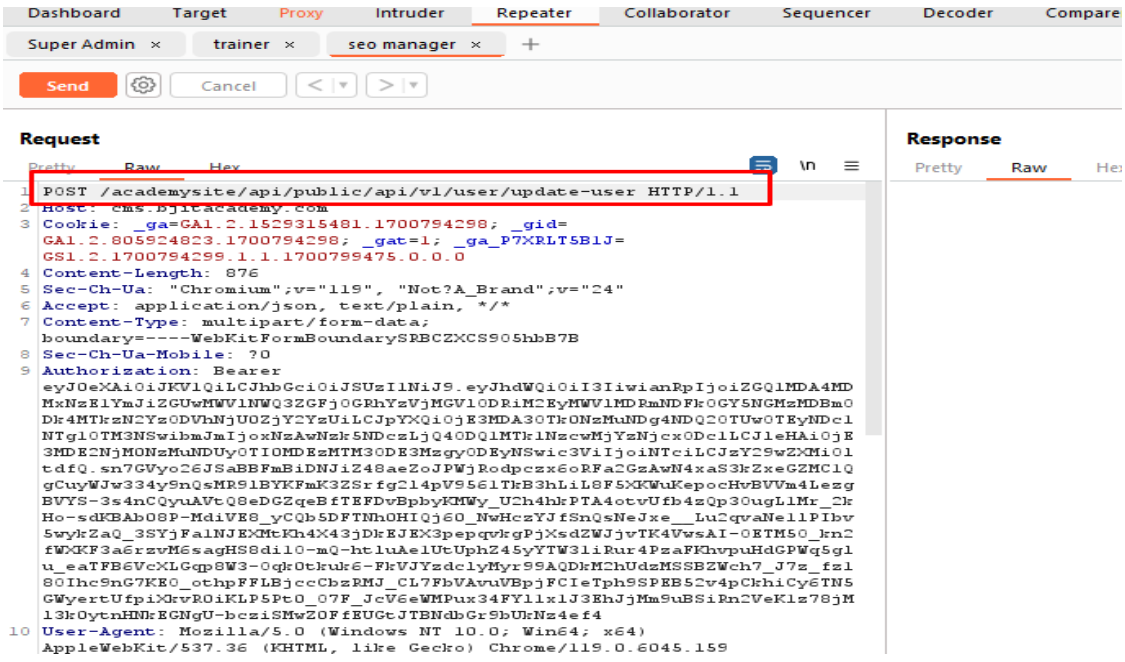
User_id: 57

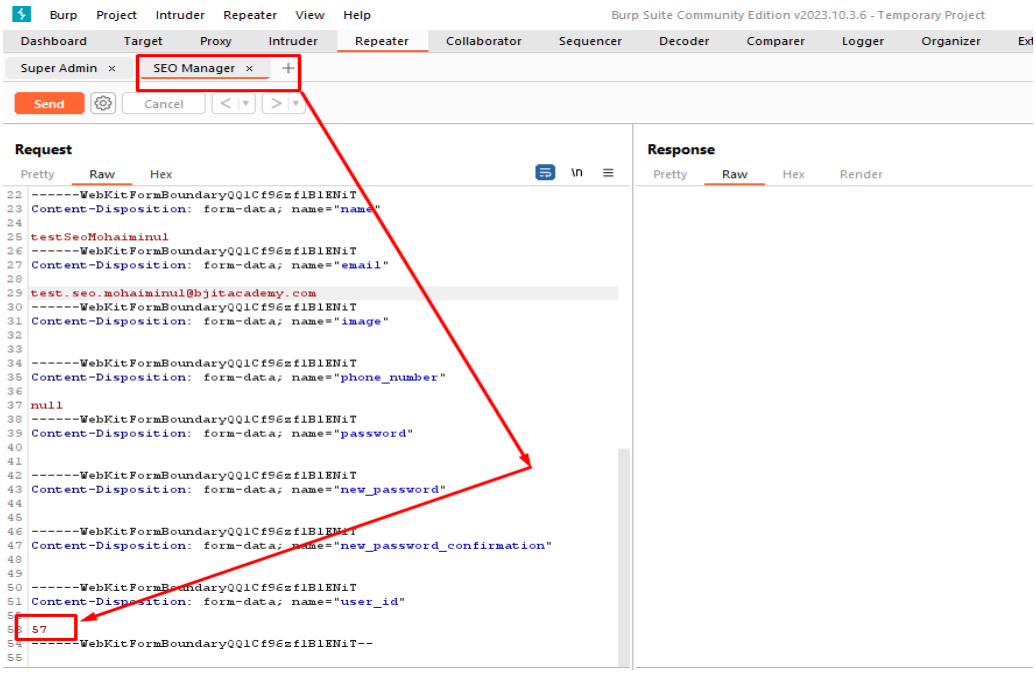
Login→ dashboard → home→ add client→ capture request to the repeater



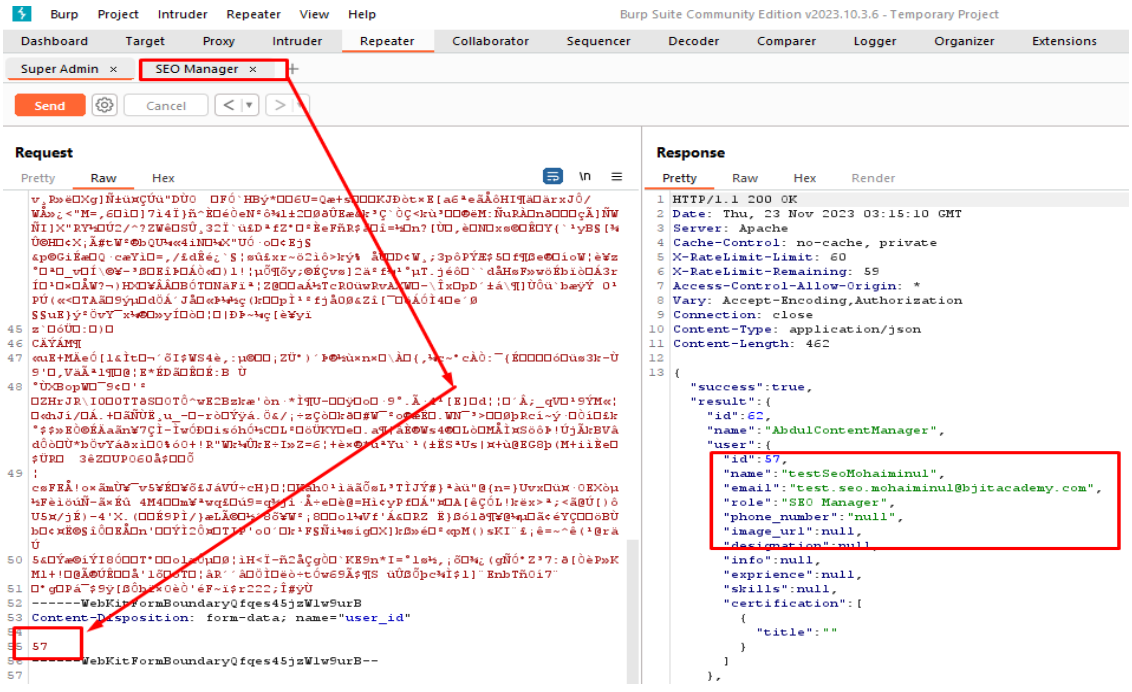
The below screen shot shows the API from where the SEO Manager id is found

POST /academysite/api/public/api/v1/user/update-user

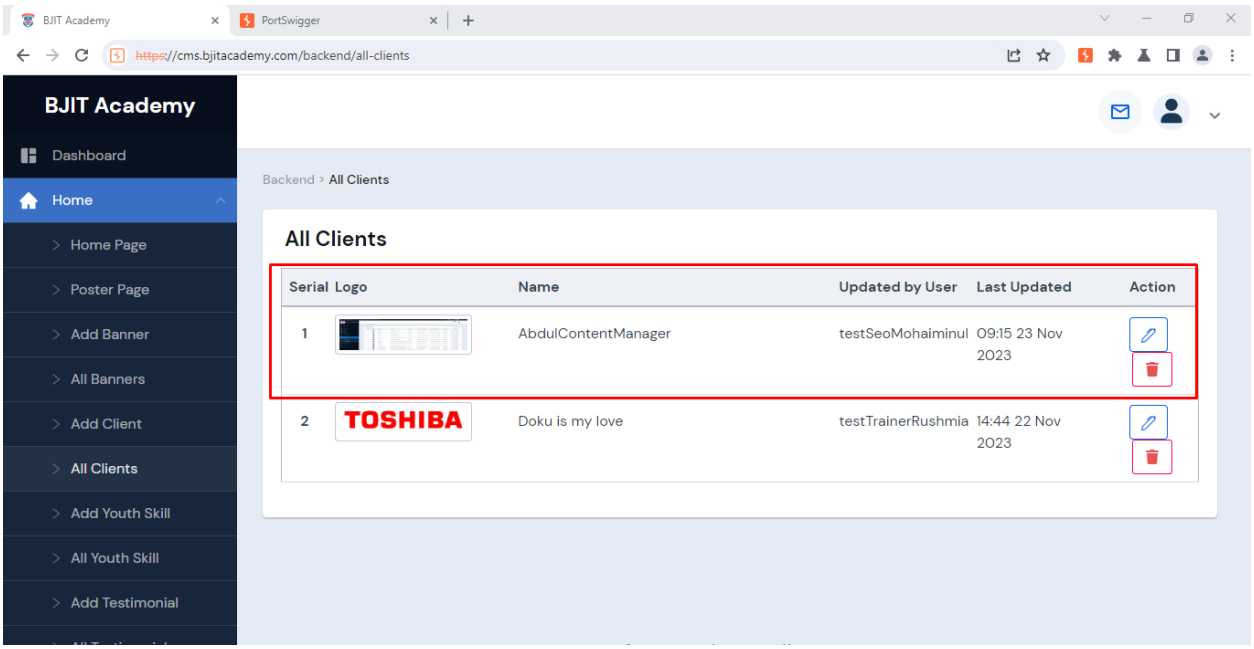




Replacing the user_id of super admin with the SEO Manager user_id.



From the frontend,

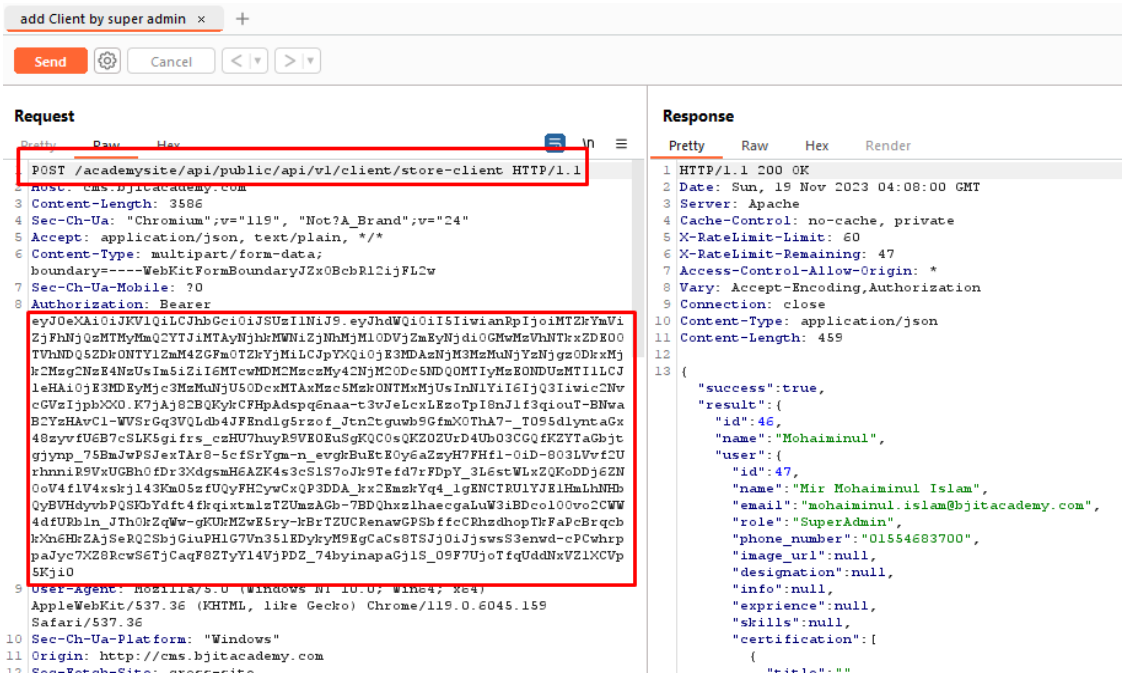


From Content Manager:

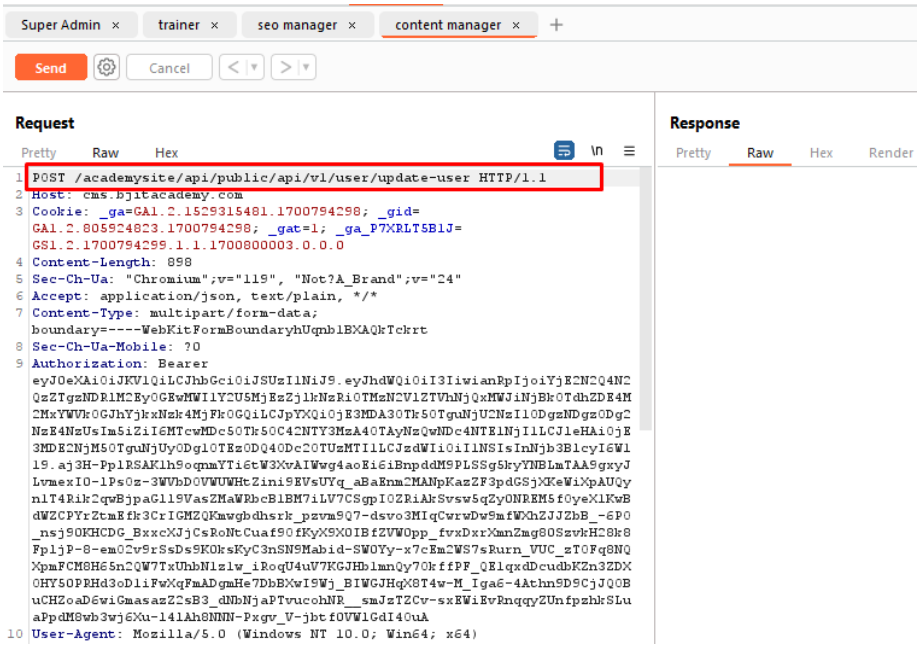
Content Manager account: test.contentmanager.mohaiminul@bjitacademy.com

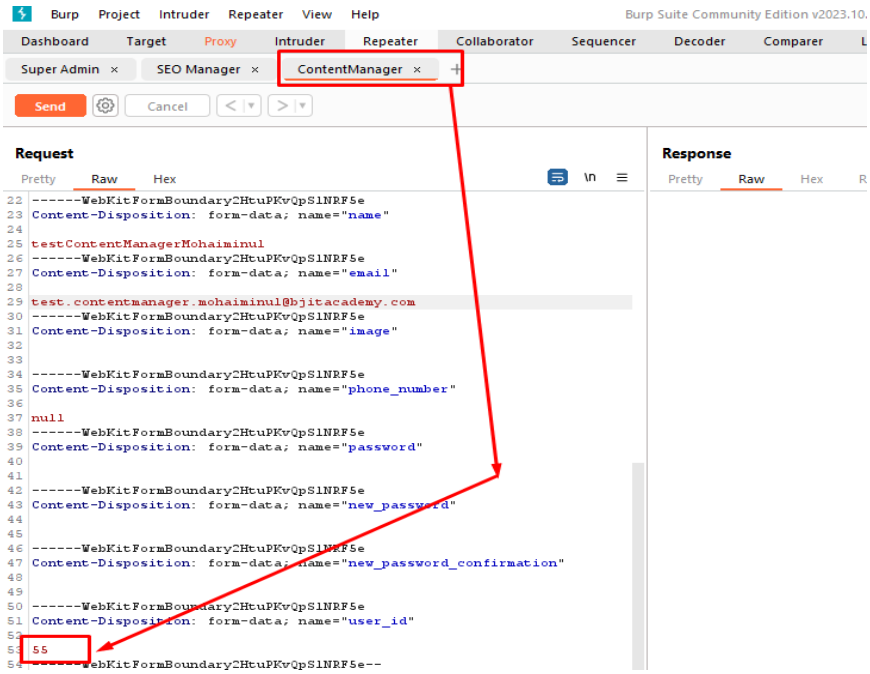
User_id: 55

Login→ dashboard → home→ add client→ capture request to the repeater

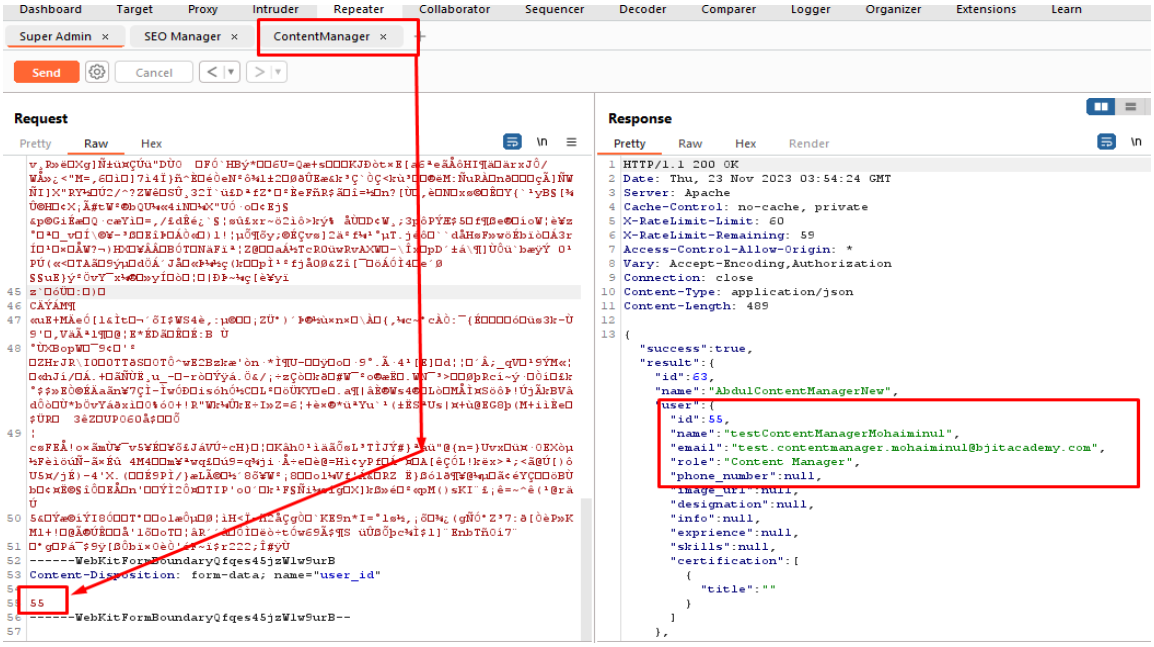


Now Collect the Content Manager id from,

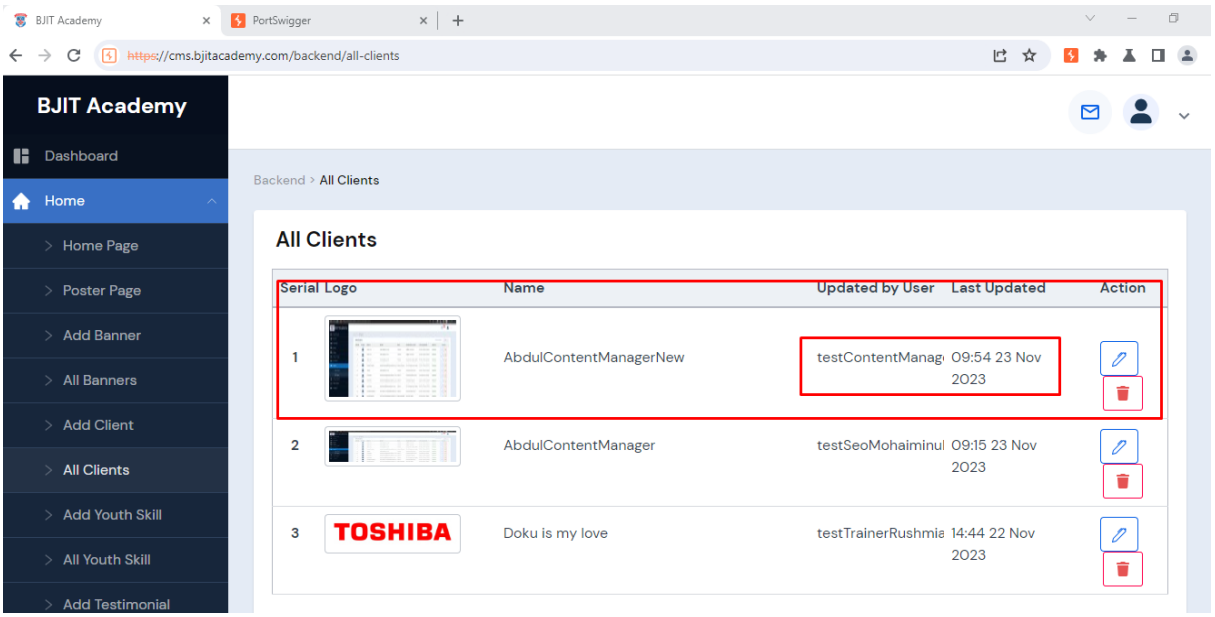




Now replace the superadmin user_id with a content manager id in the burp repeater,



Output from the frontend,



Expected Output: client can be added with only super admin credential.

Actual Output: Clients information can be added by replacing their user_id.

Similar affected API:

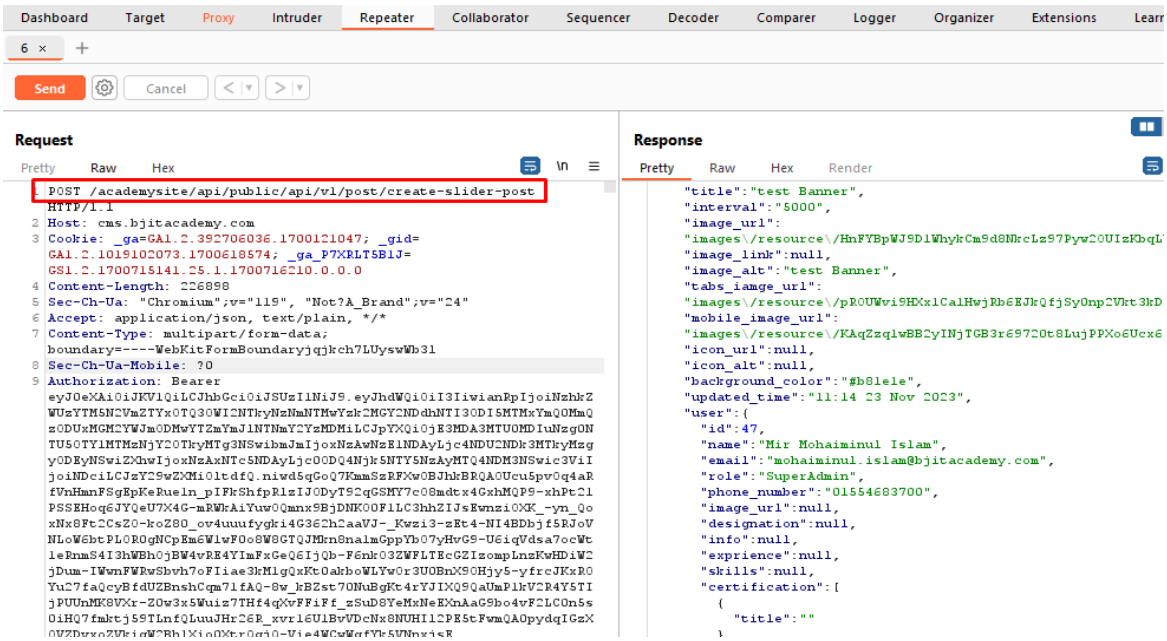
i) POST /academysite/api/public/api/v1/post/create-slider-post (for Add Banner)

POC:

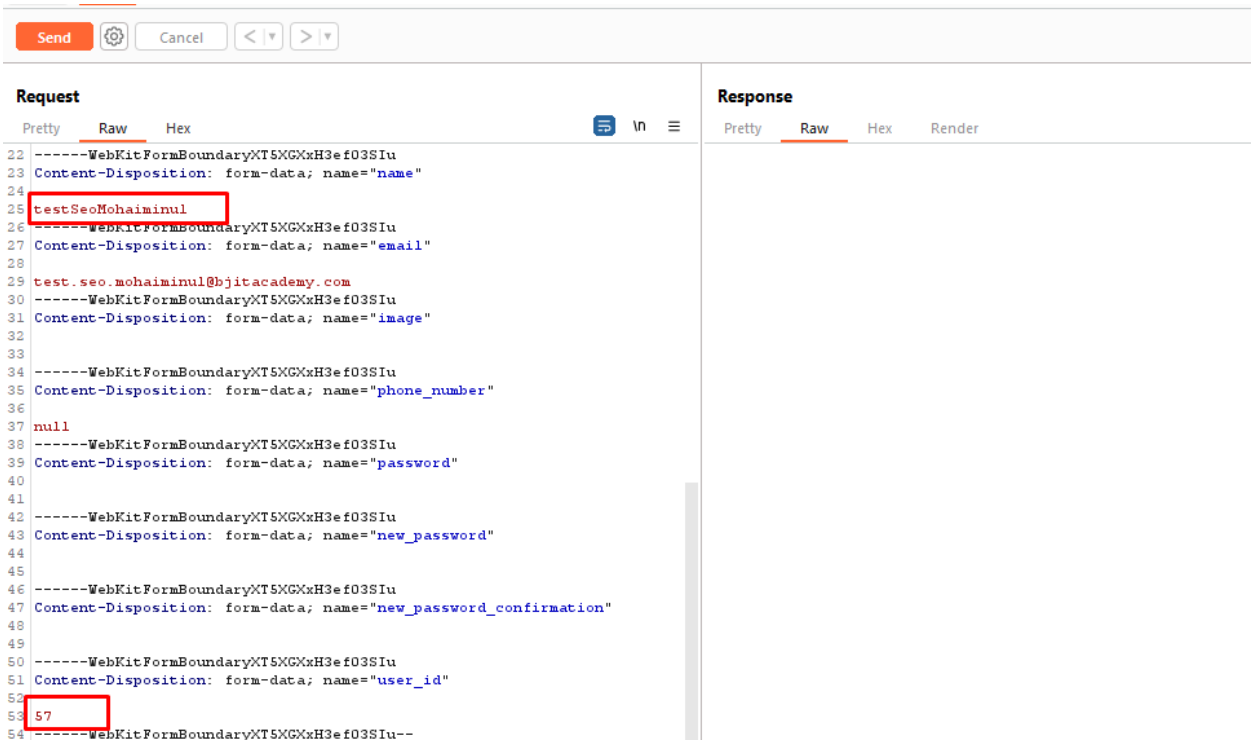
Login as super admin → go to home → add banner → capture request and send it to repeater.

Super admin account: mohaiminul.islam@bjitacademy.com

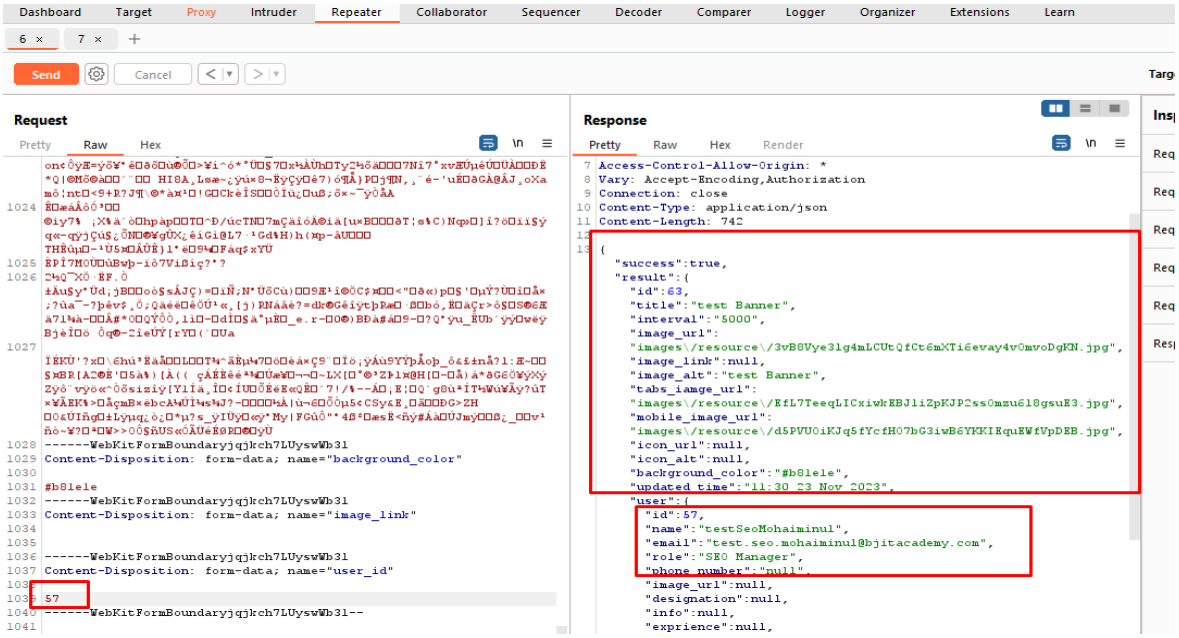
Super admin user_id: 47



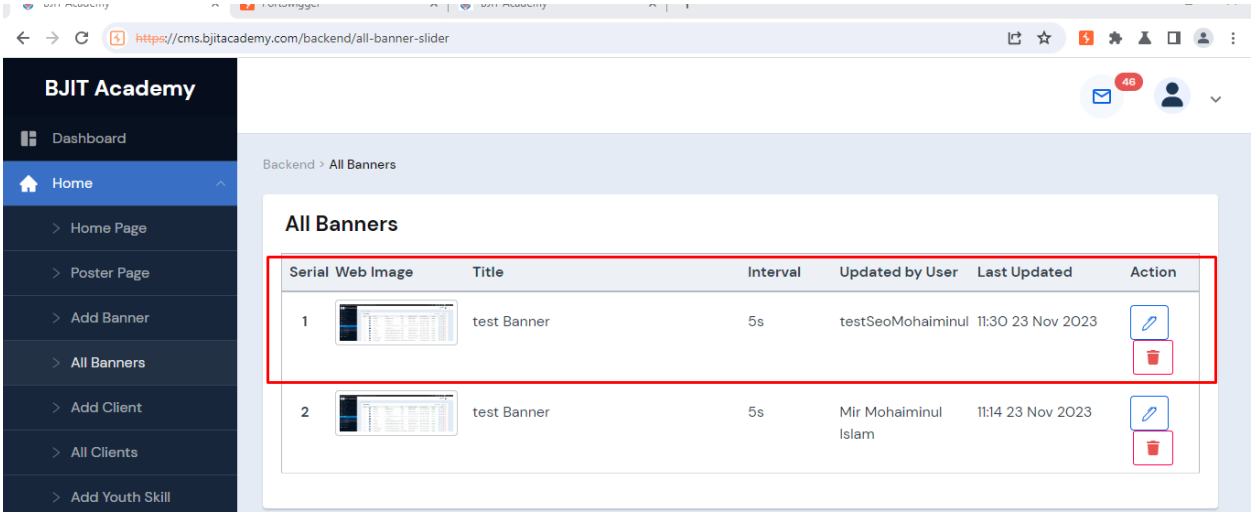
Login as a seo manager and get the user_id from **POST /academysite/api /public/api/v1/user/update-user** and replace it with the super admin authorization token.



After replacing the user_id , **SEO Manager** can create a banner.



From the frontend,

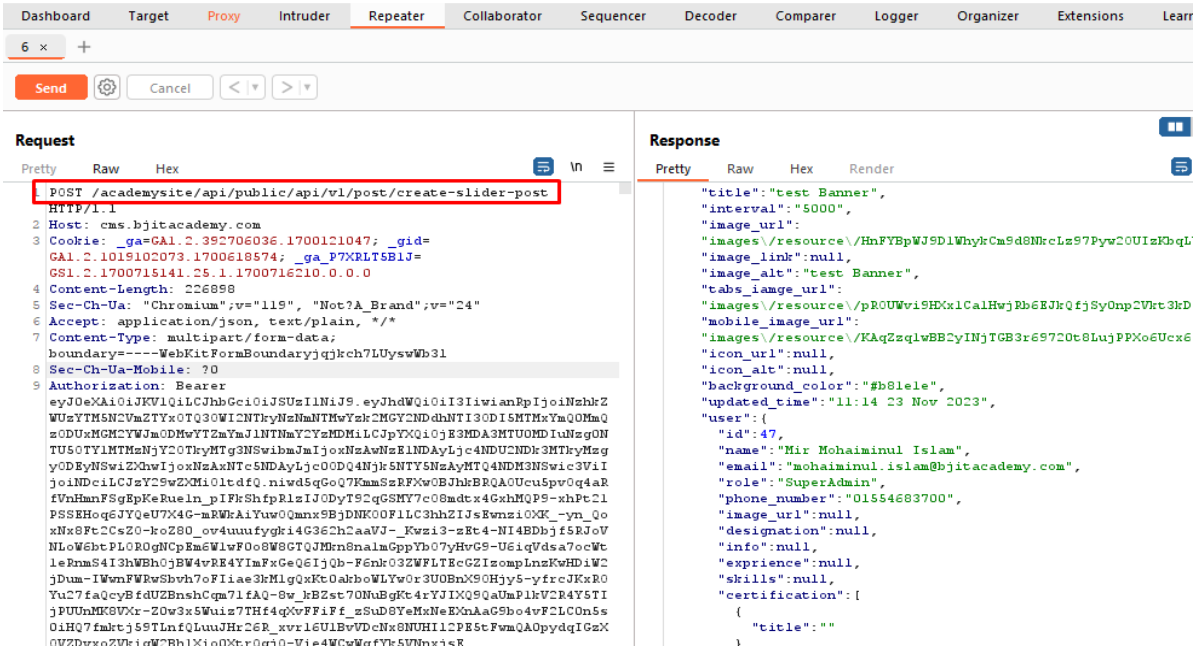


From Content Manager:

Login as super admin → go to home → add banner → capture request and send it to repeater.

Super admin account: mohaiminul.islam@bjitacademy.com

Super admin user_id: 47

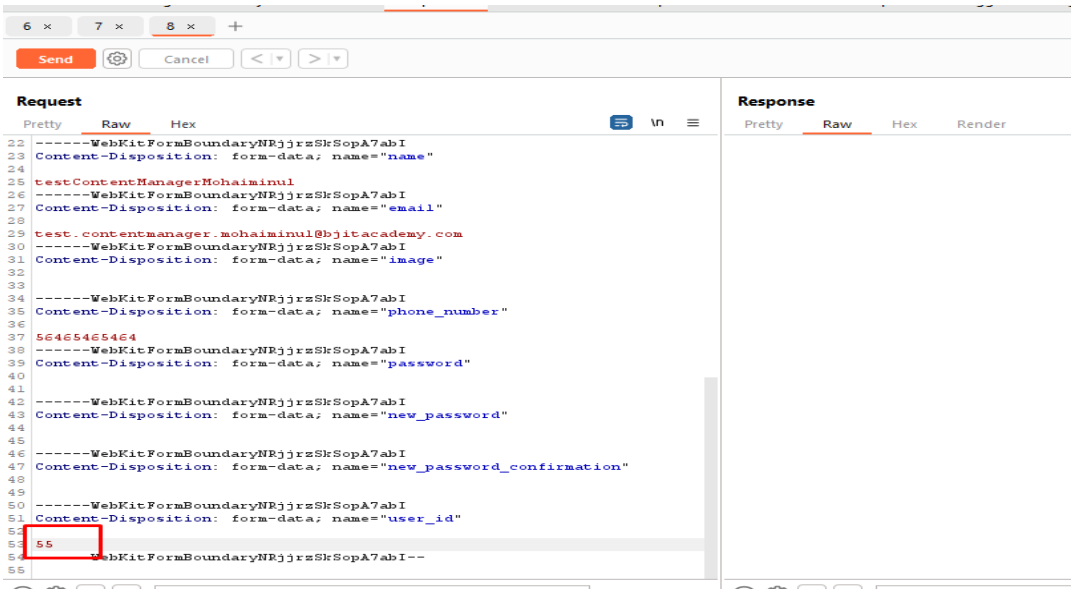


Login as a content manager and get the user_id and replace it with the super admin authorization token.

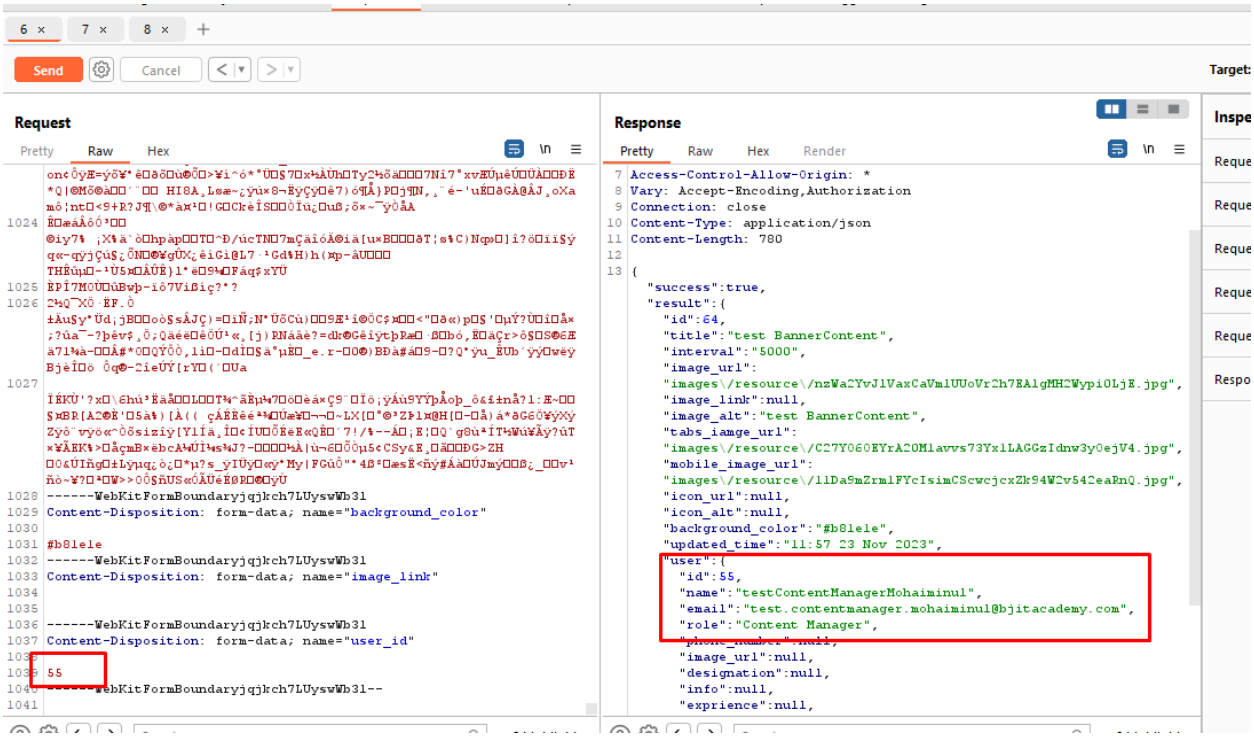
Content Manager account: test.contentmanager.mohaiminul@bjitacademy.com

User_id: 55

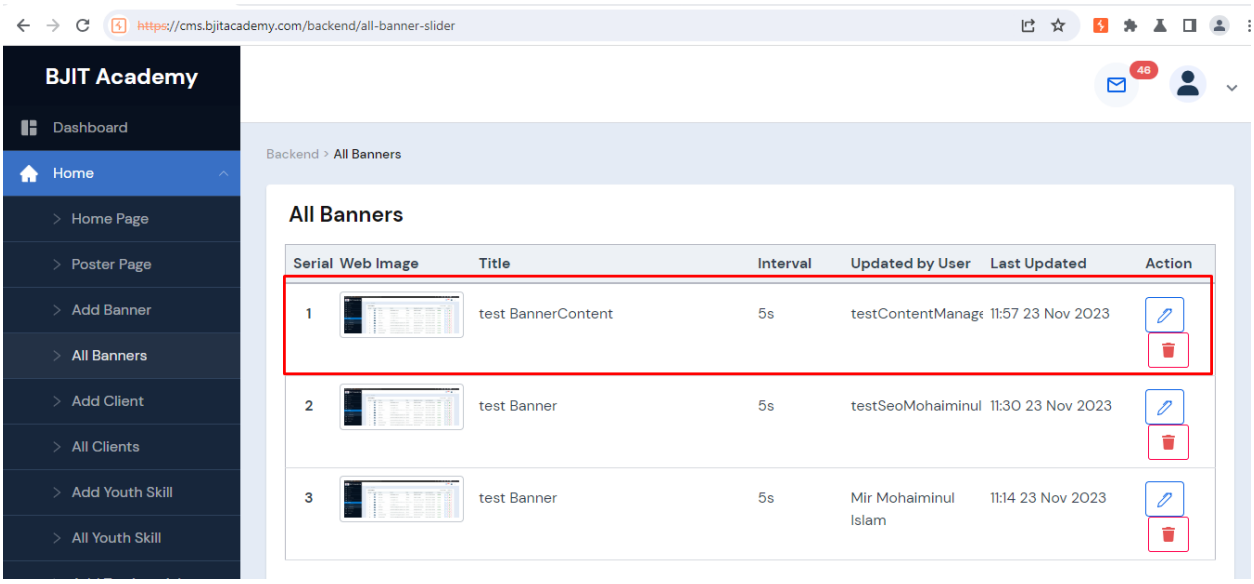
Content Manager Id found From: **POST /academysite/api /public/api/v1/user/update-user**



After Replacing the user_id I observe that new banner is created by content manager credential.



From the frontend,

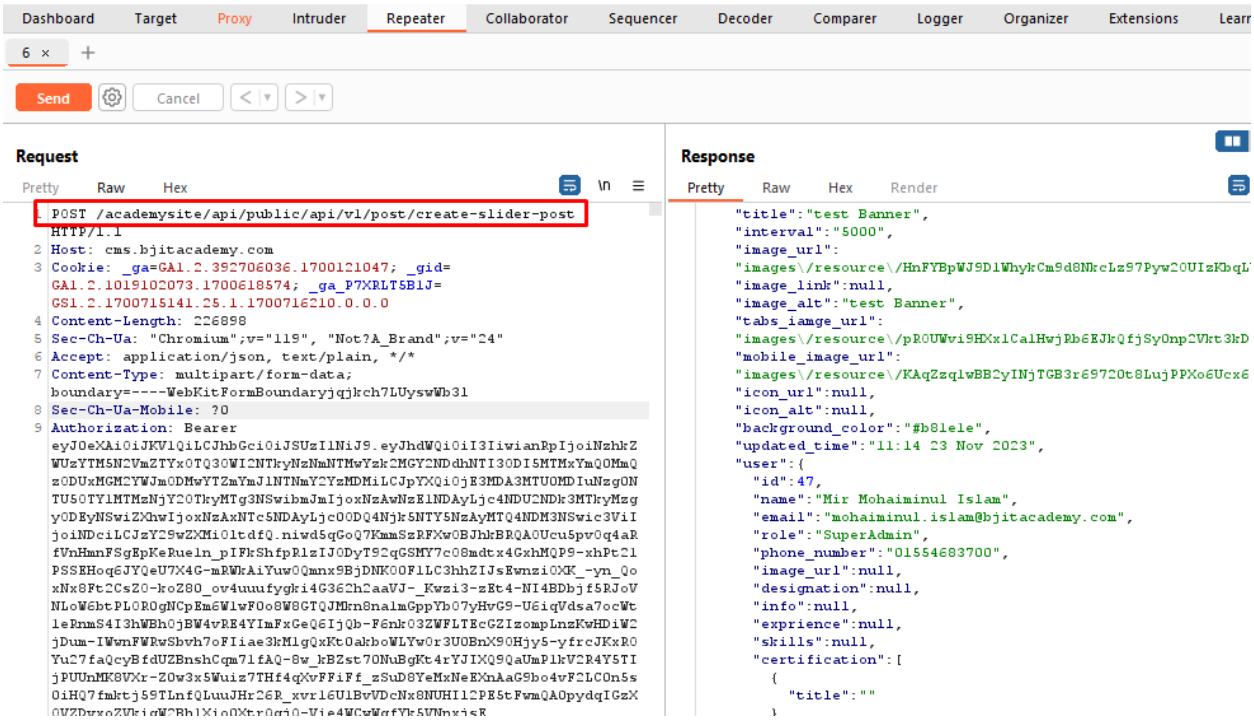


From trainer:

Login as super admin → go to home → add banner → capture request and send it to repeater.

Super admin account: mohaiminul.islam@bjitacademy.com

Super admin user_id: 47

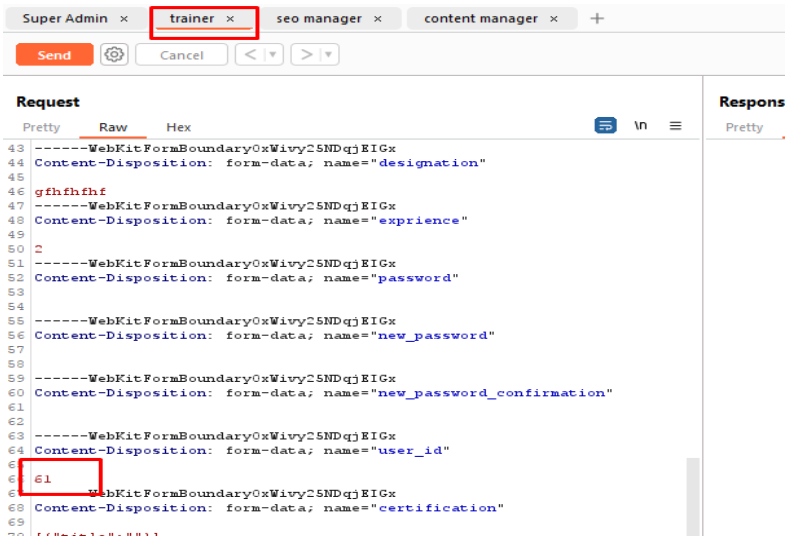


Login as a trainer and get the user_id and replace it with the super admin authorization token.

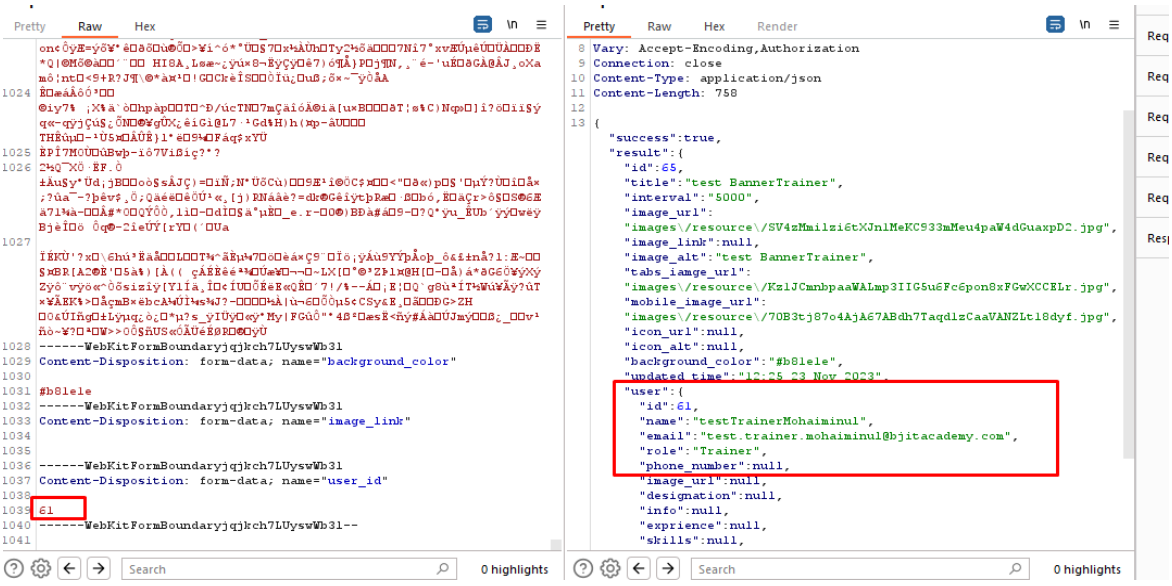
Trainer account: test.trainer.mohaiminul@bjitacademy.com

User_id: 61

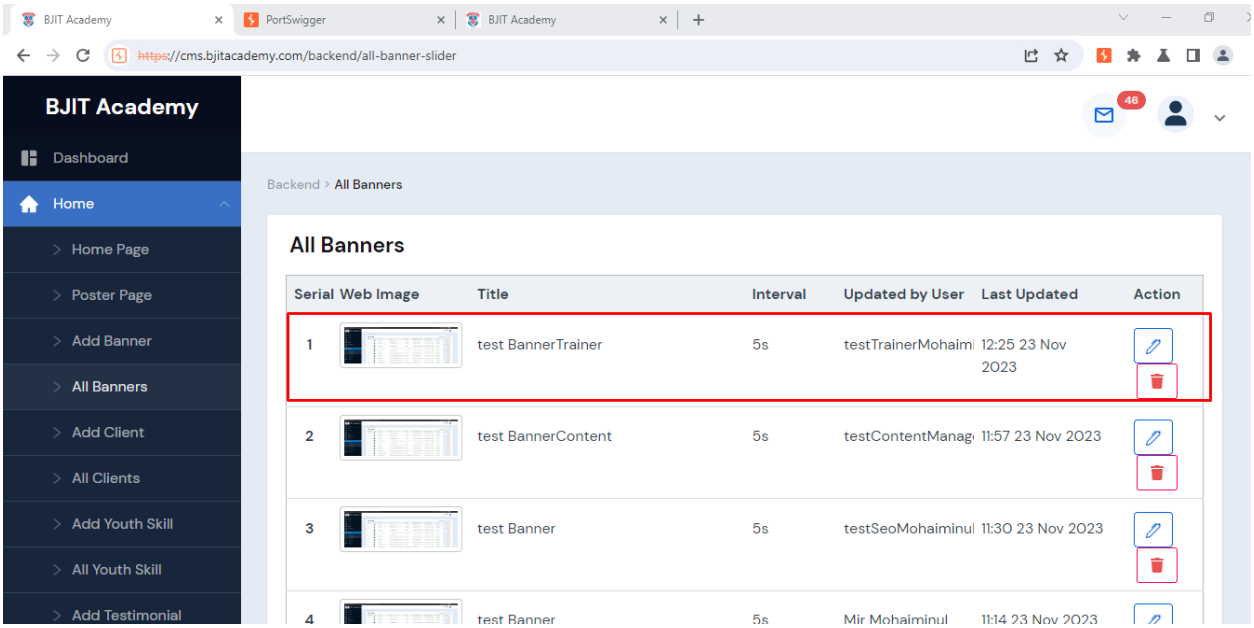
Trainer Id found From: **POST /academysite/api /public/api/v1/user/update-user**



Now replace the trainer user_id with the super admin id



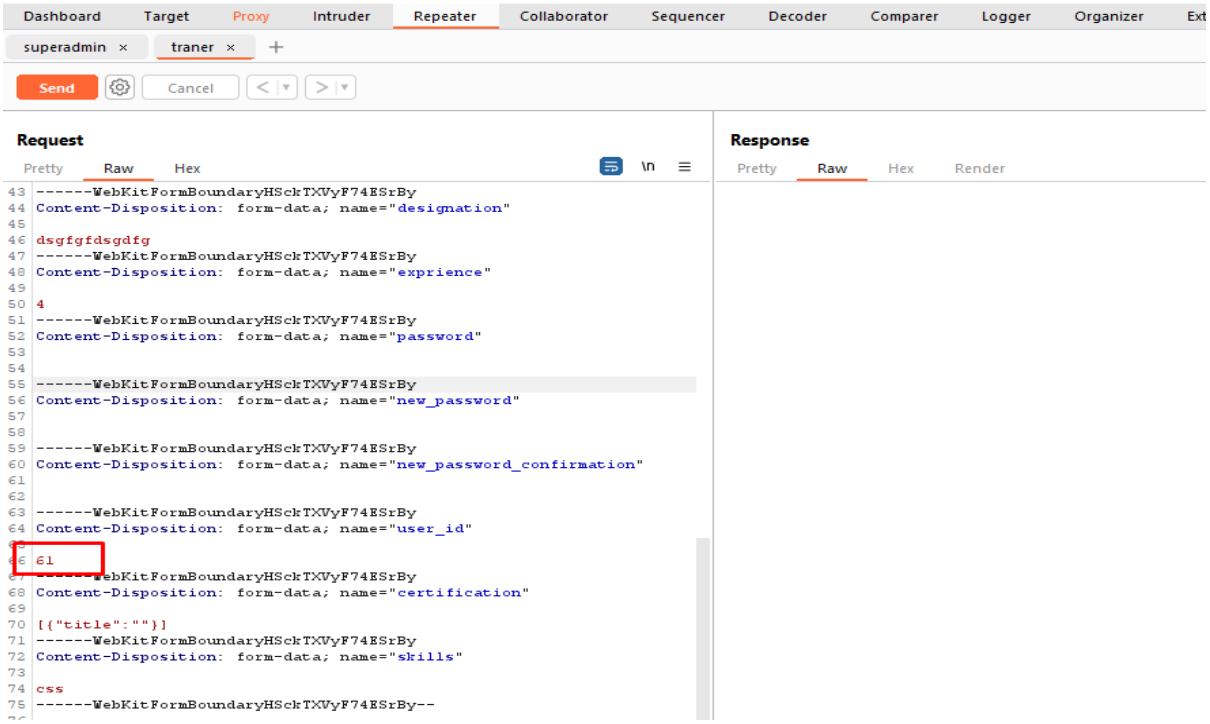
Output from frontend,



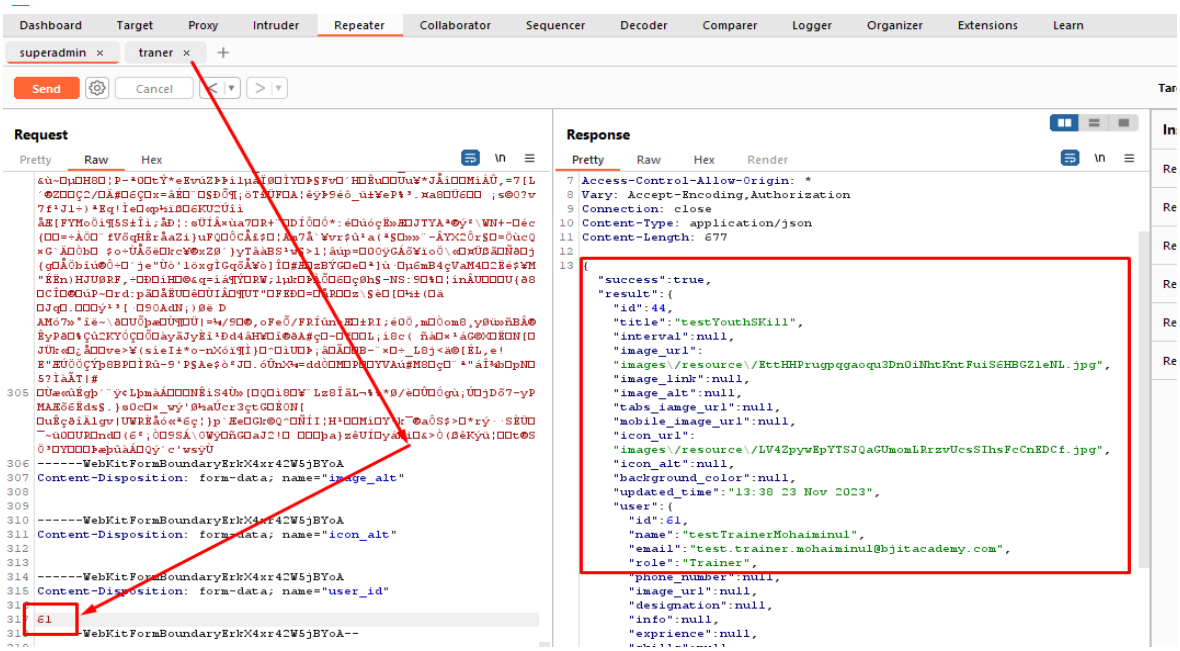
ii) POST /academysite/api/public/api/v1/post/create-fresh-talent-scope

POC:

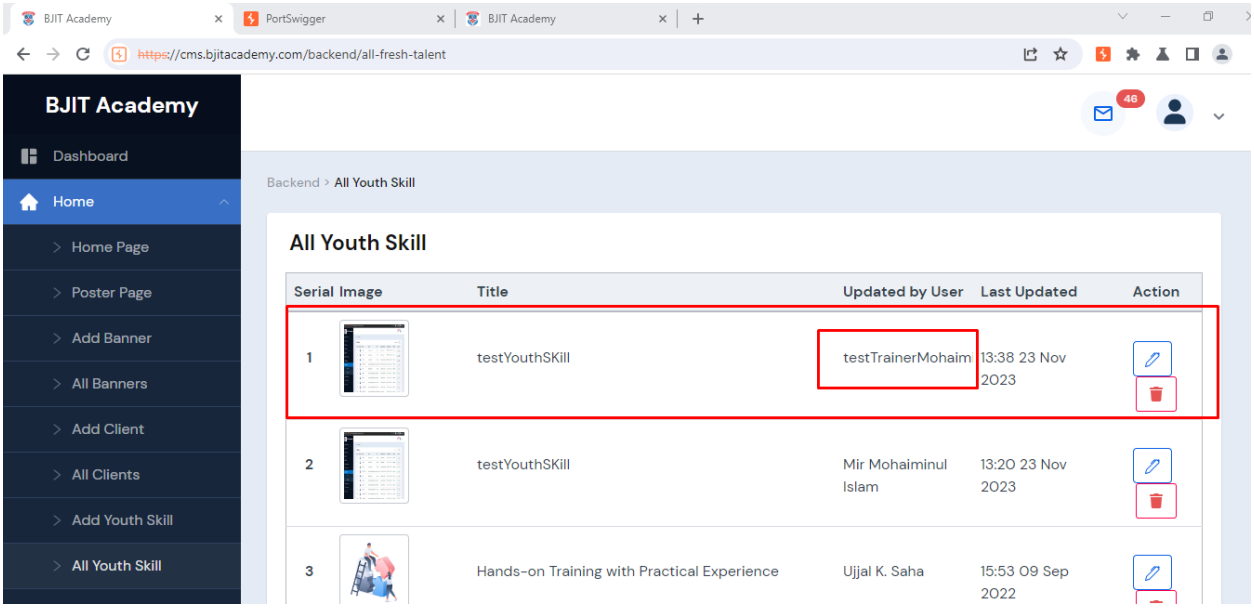
I login with trainer (test.trainer.mohaiminul@bjitacademy.com) → went to profile and try to update profile → capture the request and send to repeater → capture the user_id.



Replace the super admin user_id with the trainer user_id.

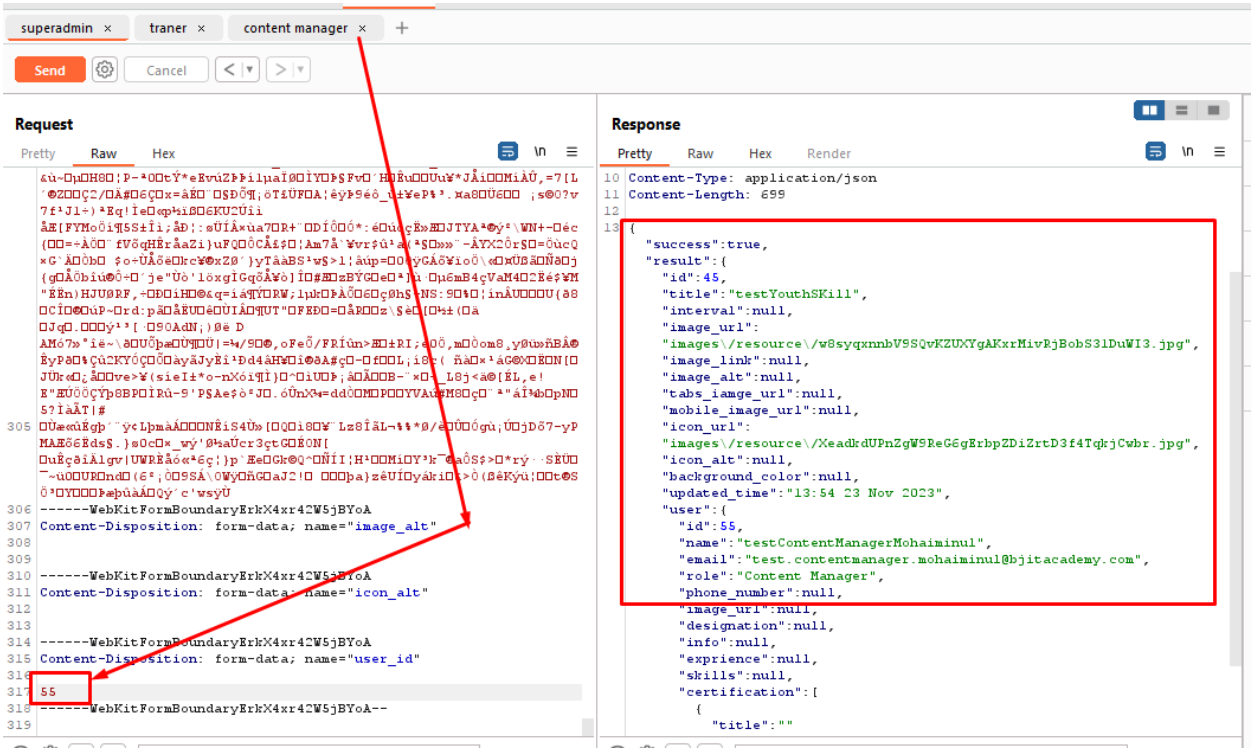


From the frontend,

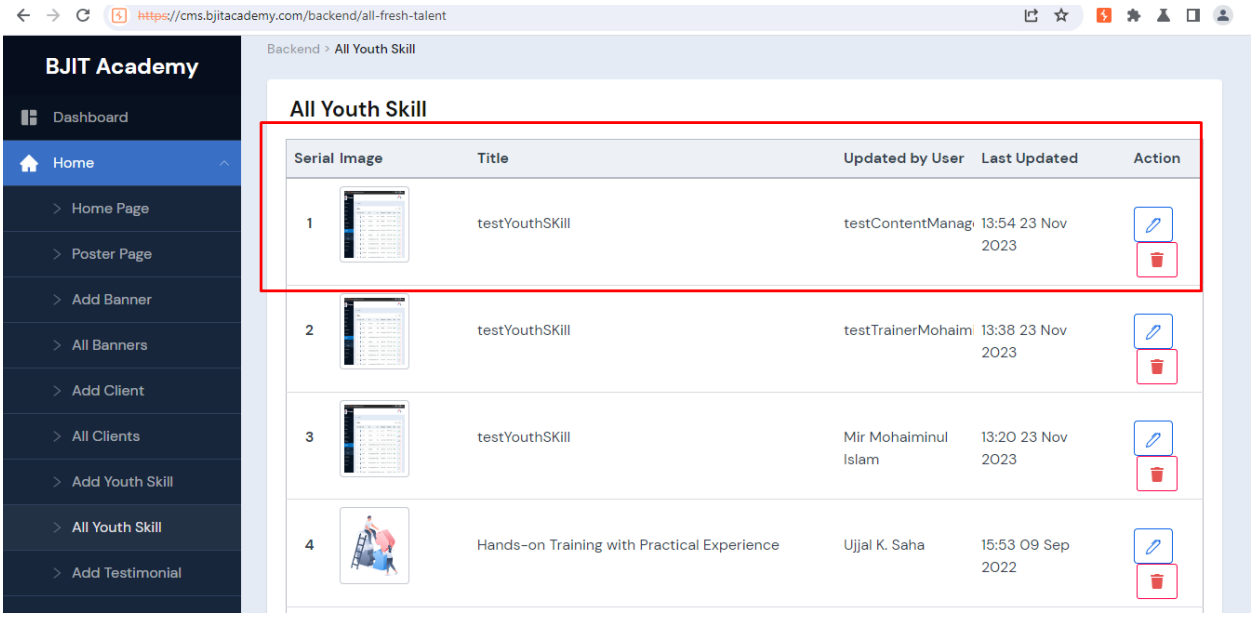


SImilarly for replacing content manager user_id,

```
test.contentmanager.mohaiminul@bjitacademy.com
------WebKitFormBoundaryAfT2uPeZY4tGHxjw
Content-Disposition: form-data; name="image"


------WebKitFormBoundaryAfT2uPeZY4tGHxjw
Content-Disposition: form-data; name="phone_number"

null
------WebKitFormBoundaryAfT2uPeZY4tGHxjw
Content-Disposition: form-data; name="password"


------WebKitFormBoundaryAfT2uPeZY4tGHxjw
Content-Disposition: form-data; name="new_password"


------WebKitFormBoundaryAfT2uPeZY4tGHxjw
Content-Disposition: form-data; name="new_password_confirmation"


------WebKitFormBoundaryAfT2uPeZY4tGHxjw
Content-Disposition: form-data; name="user_id"

55
------WebKitFormBoundaryAfT2uPeZY4tGHxjw--
```



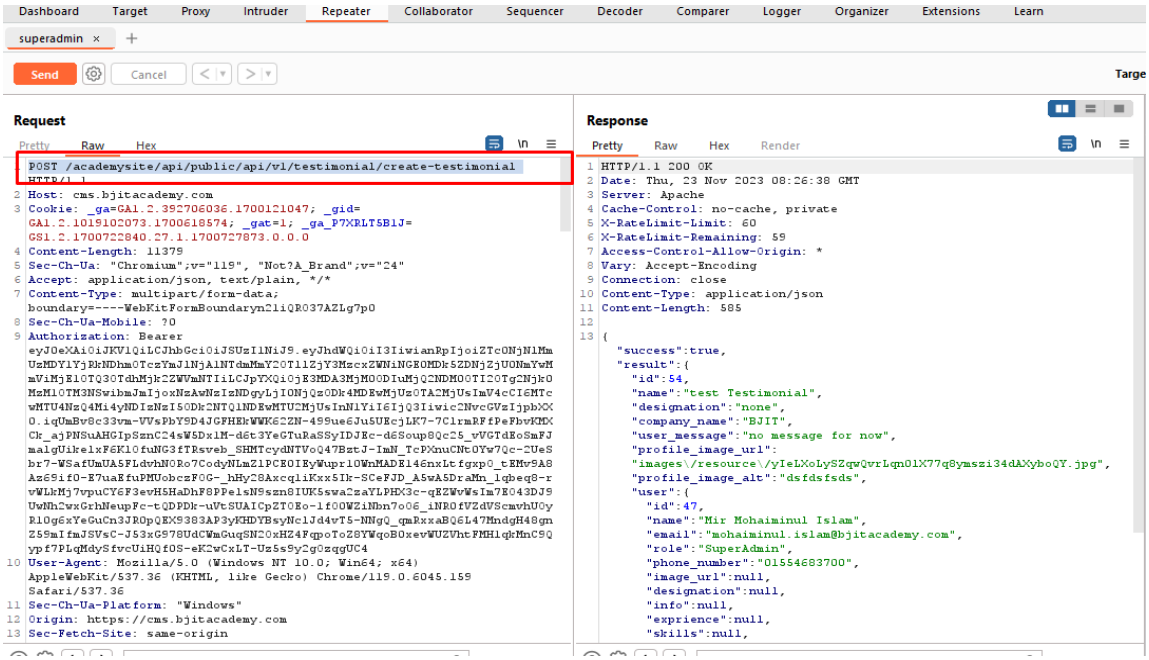
Now see the output from the frontend,



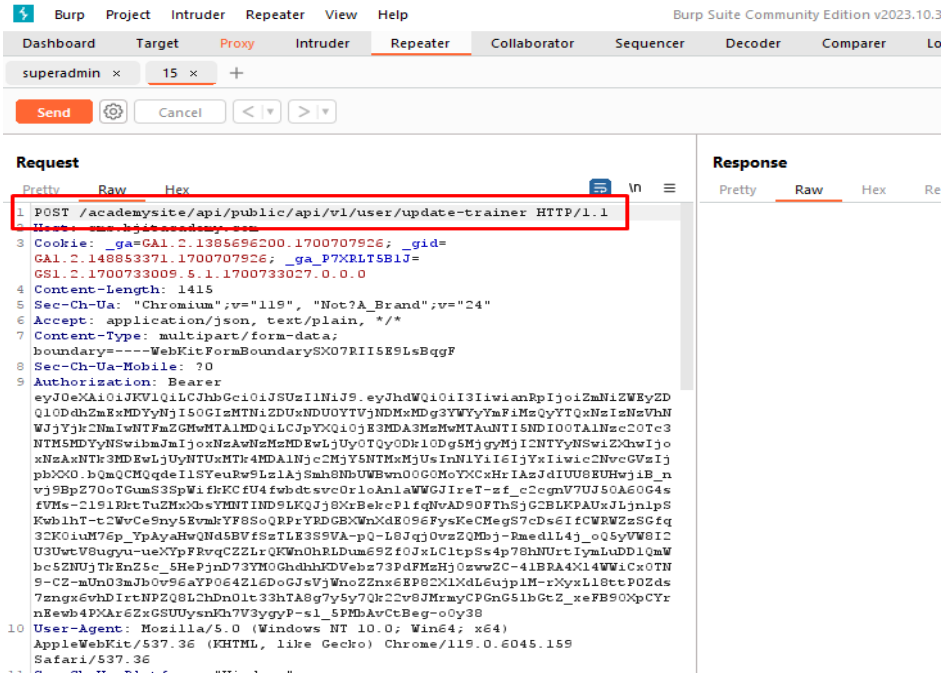
iii) POST /academysite/api/public/api/v1/testimonial/create-testimonial

POC:

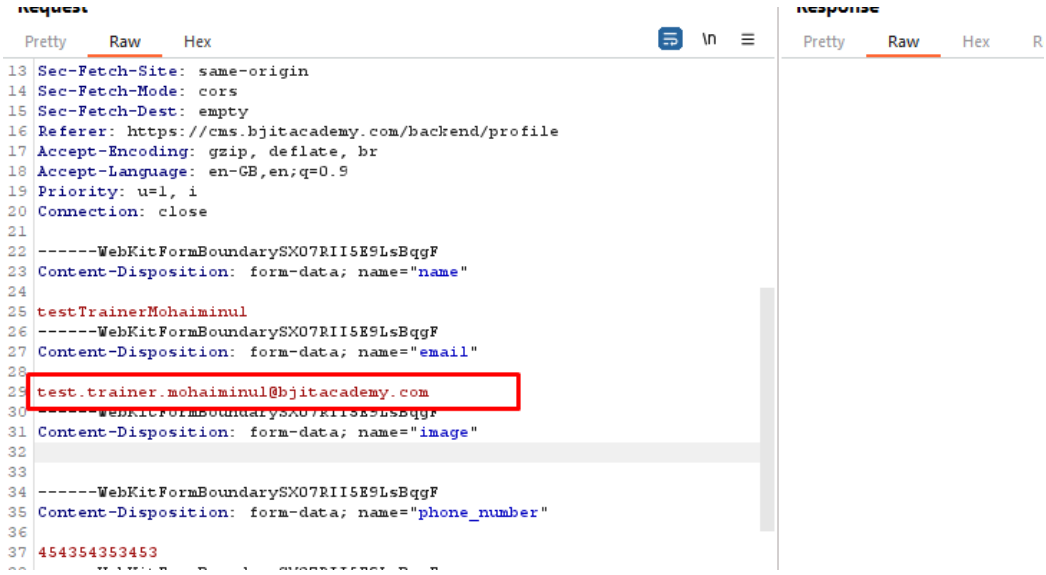
First login as super admin → go to dashboard → try to add testimonial → capture the request and send it to burp Repeater



Now replace the super admin user_id with trainer user_id,



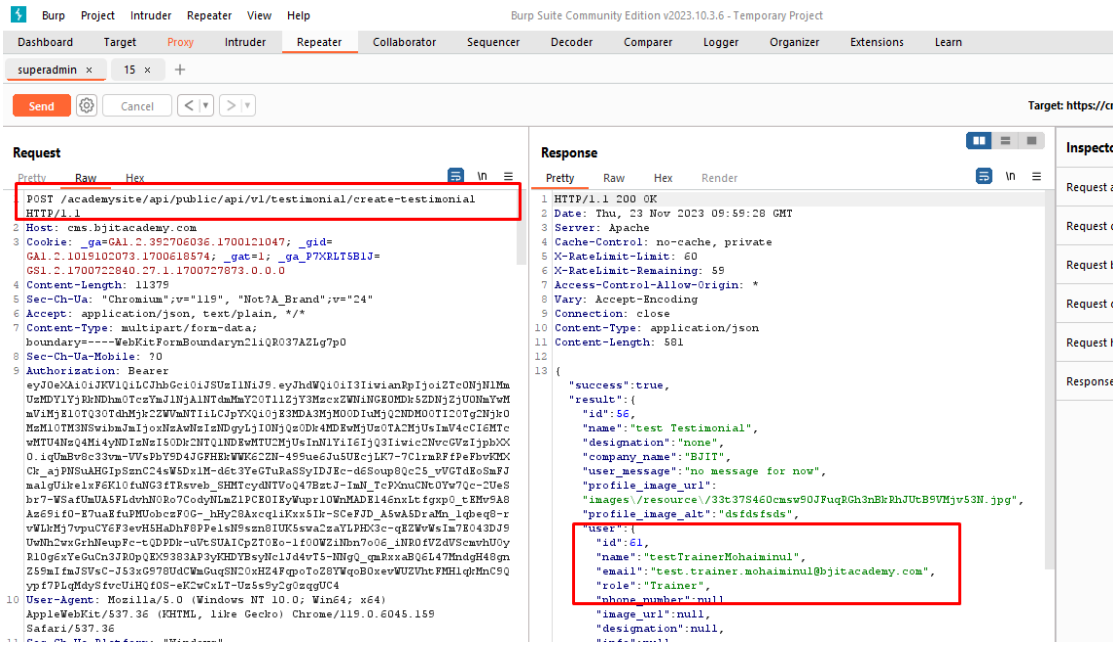
Here the trainer account: test.trainer.mohaiminul@bjitacademy.com



And the id of the trainer is:61.

```
Content-Disposition: form-data; name="designation"

dfgfdgdg
------WebKitFormBoundarySX07RII5E9LsBqgF
Content-Disposition: form-data; name="exprience"

3
------WebKitFormBoundarySX07RII5E9LsBqgF
Content-Disposition: form-data; name="password"


------WebKitFormBoundarySX07RII5E9LsBqgF
Content-Disposition: form-data; name="new_password"


------WebKitFormBoundarySX07RII5E9LsBqgF
Content-Disposition: form-data; name="new_password_confirmation"


------WebKitFormBoundarySX07RII5E9LsBqgF
Content-Disposition: form-data; name="user_id"

61
------WebKitFormBoundarySX07RII5E9LsBqgF
Content-Disposition: form-data; name="certification"
```

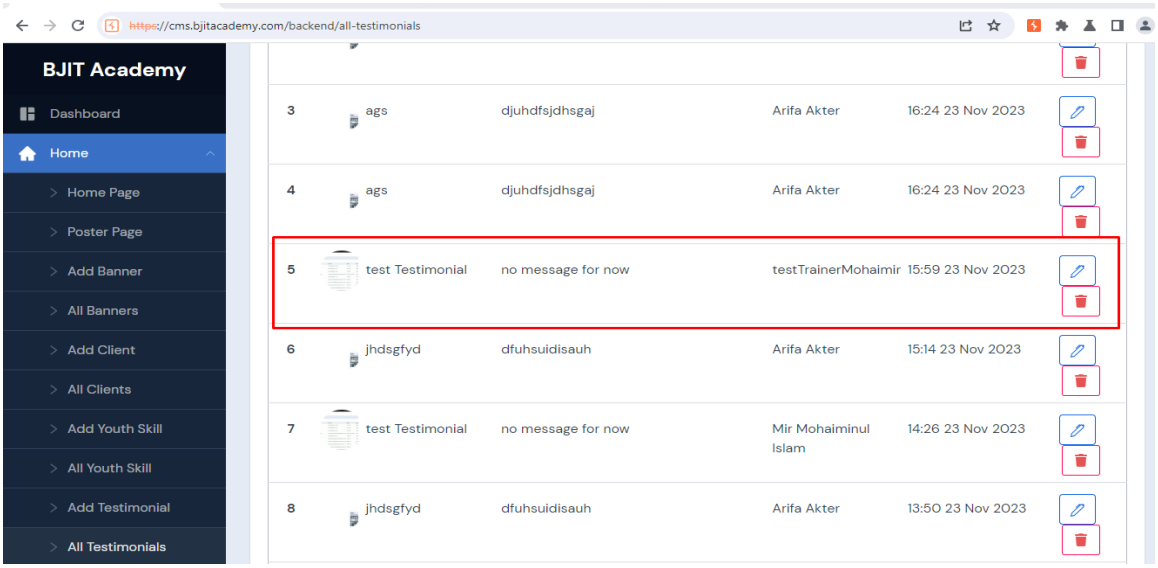
Now replace the super admin id with the trainer id in the POST/academysite/api/public/api/v1/testimonial/create-testimonial



```
------WebKitFormBoundaryn2liQR037AZLg7p0
Content-Disposition: form-data; name="profile_image_alt"

 dsfdsfsds
------WebKitFormBoundaryn2liQR037AZLg7p0
Content-Disposition: form-data; name="user_id"

61
------WebKitFormBoundaryn2liQR037AZLg7p0--
```

```
HTTP/1.1 200 OK
Date: Thu, 23 Nov 2023 09:59:28 GMT
Server: Apache
Cache-Control: no-cache, private
X-RateLimit-Limit: 60
X-RateLimit-Remaining: 59
Access-Control-Allow-Origin: *
Vary: Accept-Encoding
Connection: close
Content-Type: application/json
Content-Length: 581

{
  "success":true,
  "result":{
    "id":56,
    "name":"test Testimonial",
    "designation":"none",
    "company_name":"BJIT",
    "user_message":"no message for now",
    "profile_image_url":
    "images\/resource\/33t37S46Ocmsw90JFuqRGh3nBkRhJUtB9VMjv53N.jpg",
    "profile_image_alt":"dsfdsfsds",
    "user":{
      "id":61,
      "name":"testTrainerMohaiminul",
      "email":"test.trainer.mohaiminul@bjitacademy.com",
      "role":"Trainer",
      "phone_number":null,
      "image_url":null,
      "designation":null,
      "info":null,
      "exprience":null,
```

From the frontend we see that,



So here trainer's user id is manipulated and a testimonial is created.

4. **Title**: Normal User's (Admin,Trainer, SEO Manager,Content Manager) user_id can be manipulated to make a new user.

   Target: cms@bjitacademy.com

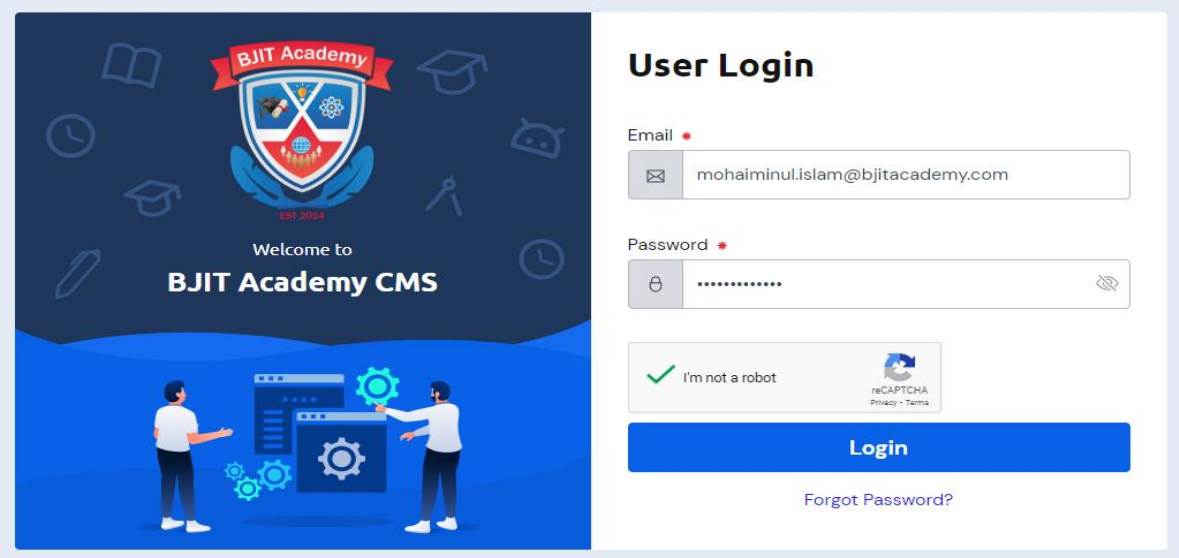
   URL/API: https://cms.bjitacademy.com/backend/register

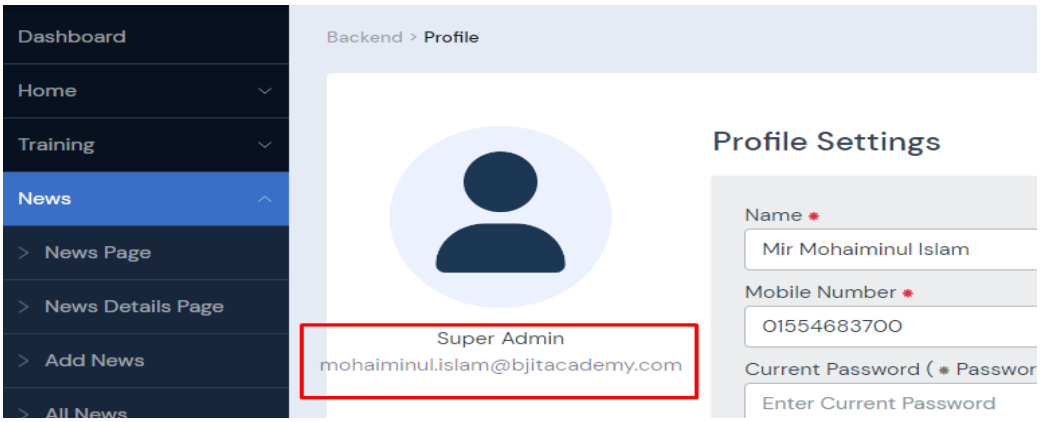
   Summary: In **cms.bjitacademy.com**, only **Super Admin** has the permission to create a new user. But after login with super admin credentials when I take the request of creating a new user in burp suite and change the id of the Super Admin with **Admin/Trainer/SEO Manager/Content Manager** then a new user is created. In that case they are unauthorized and it's a vulnerability.

POC:

For the proof of concept At first I have to login with Super Admin login credentials with the build in chromium browser of burp suite.

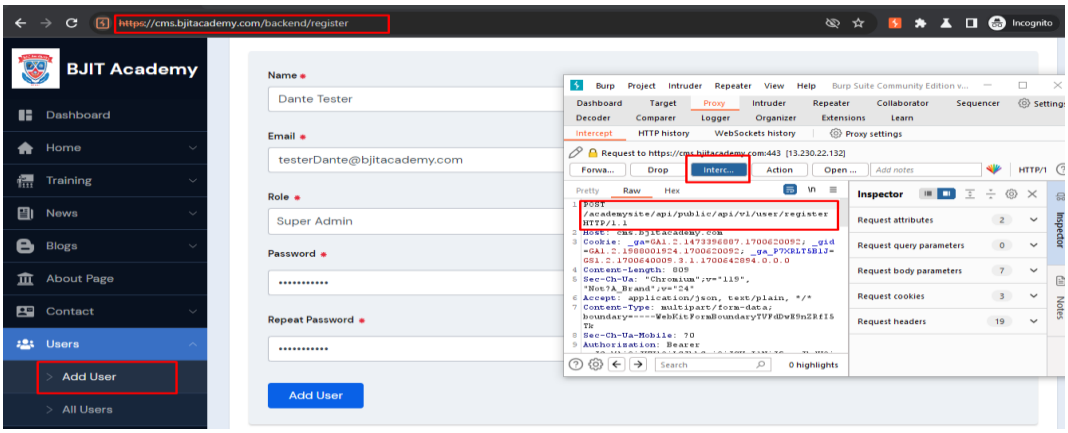
Here the Super-Admin User: mohaiminul.islam@bjitacademy.com





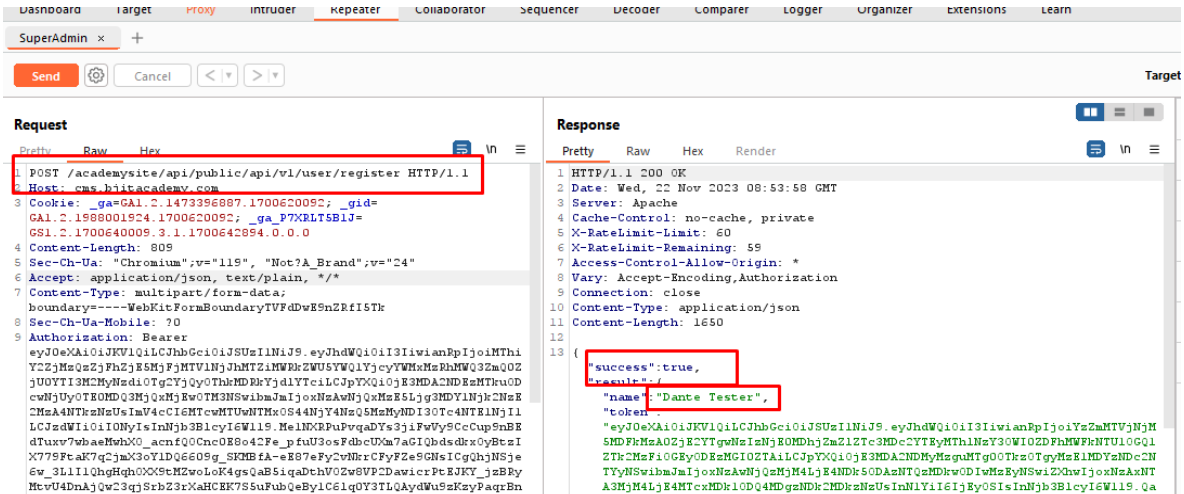
Now I went to Dashboard→Users→Add User page with **/backend/register** Url.

After fillup the form I click the Add User button.

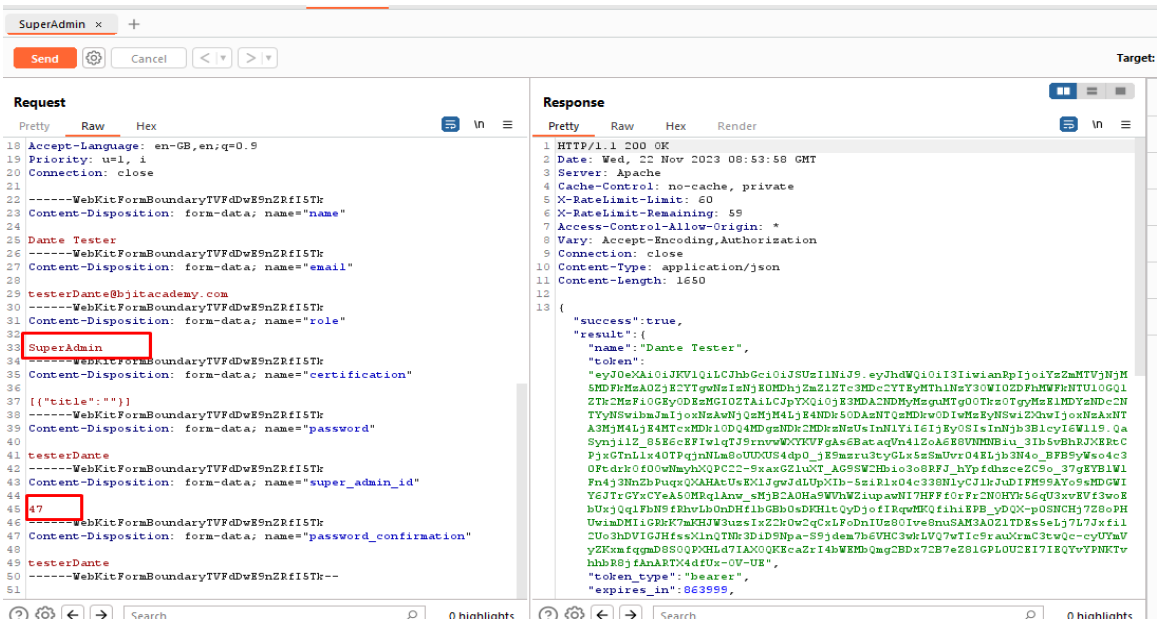


Here I capture the request with proxy intercept on. After that I send the request to the burp Repeater.

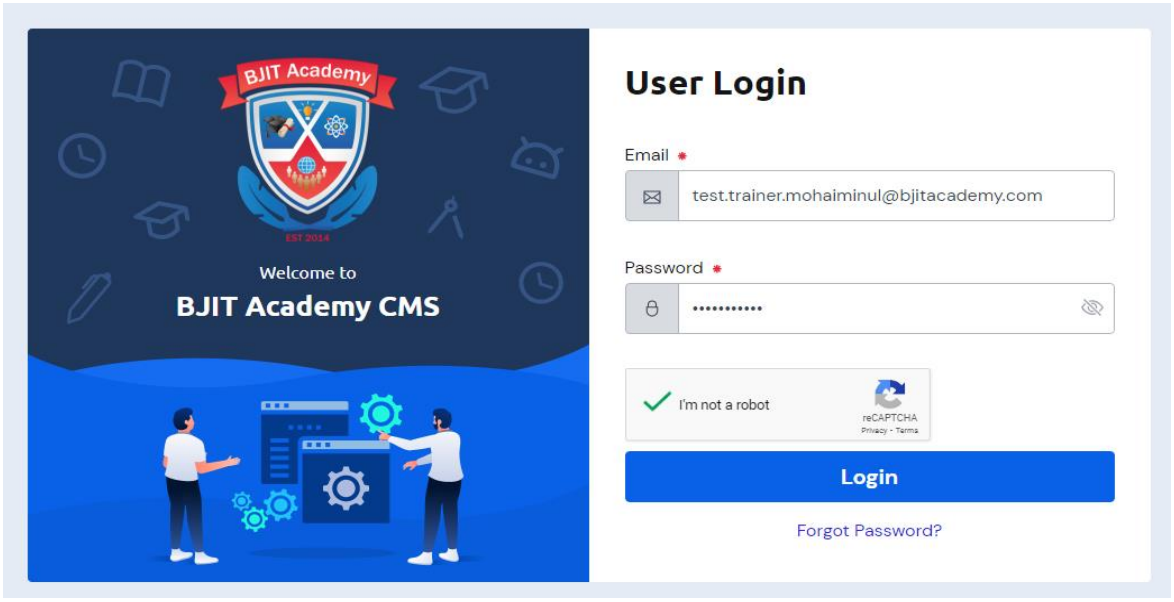
Then I clicked the send button and checked the request.

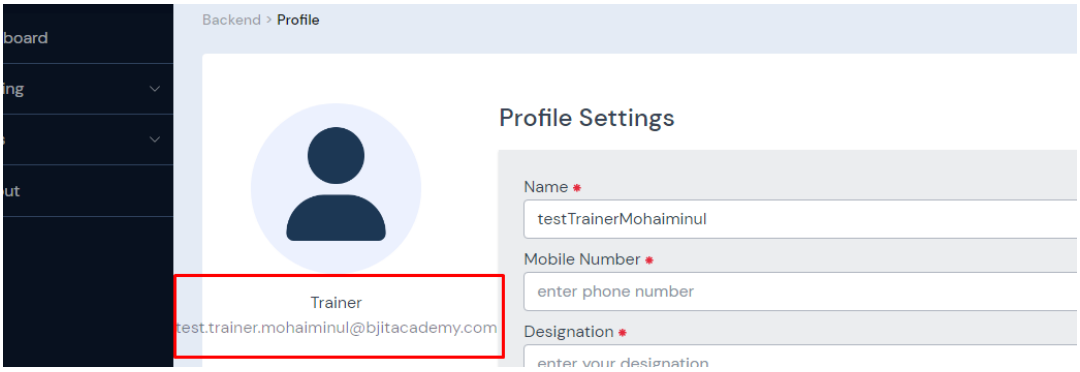


Here the user role is Super Admin and id of that Super Admin is 47.



Now I will login with Trainer's credentials to demonstrate that trainer level users also has the access to create users.

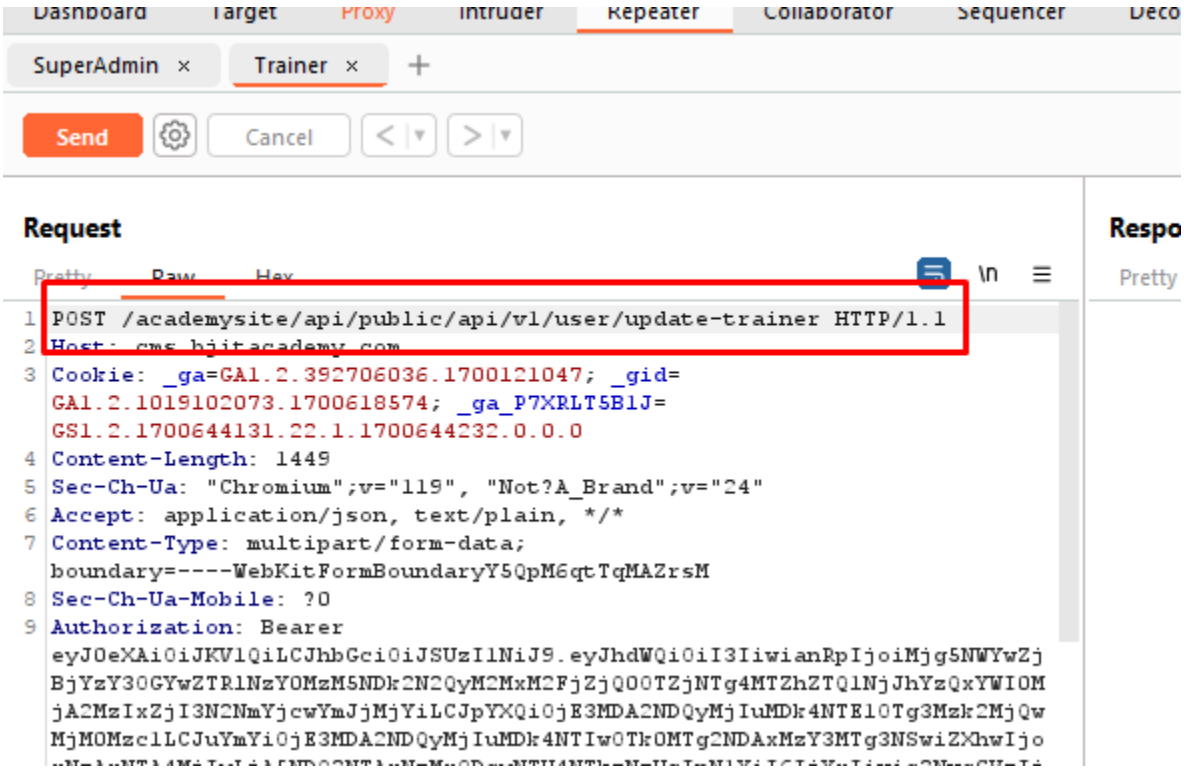




Here the trainer username: test.trainer.mohaiminul@bjitacademy.com

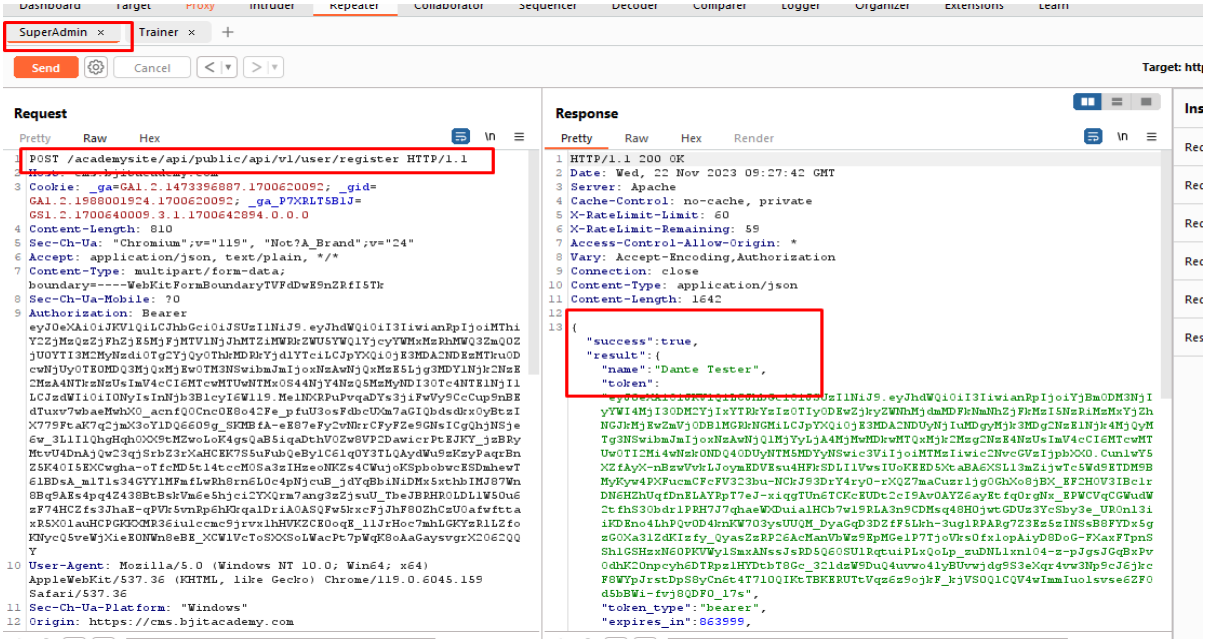
Now I will go to profile Dashboard → profile page from the top bar

Then capture the request of update profile. Here the actual purpose is to get the id of a trainer. Then I send request to the burp Repeater.

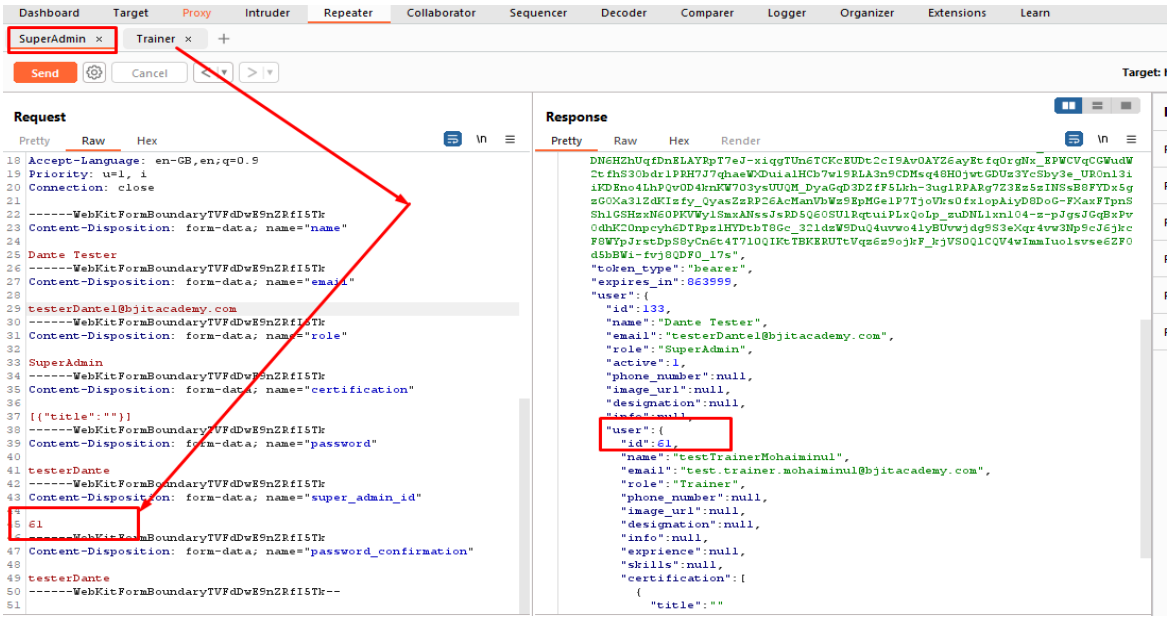


Here the trainer id is 61

Now Replace the Super Admin user_id with the trainer user_id .

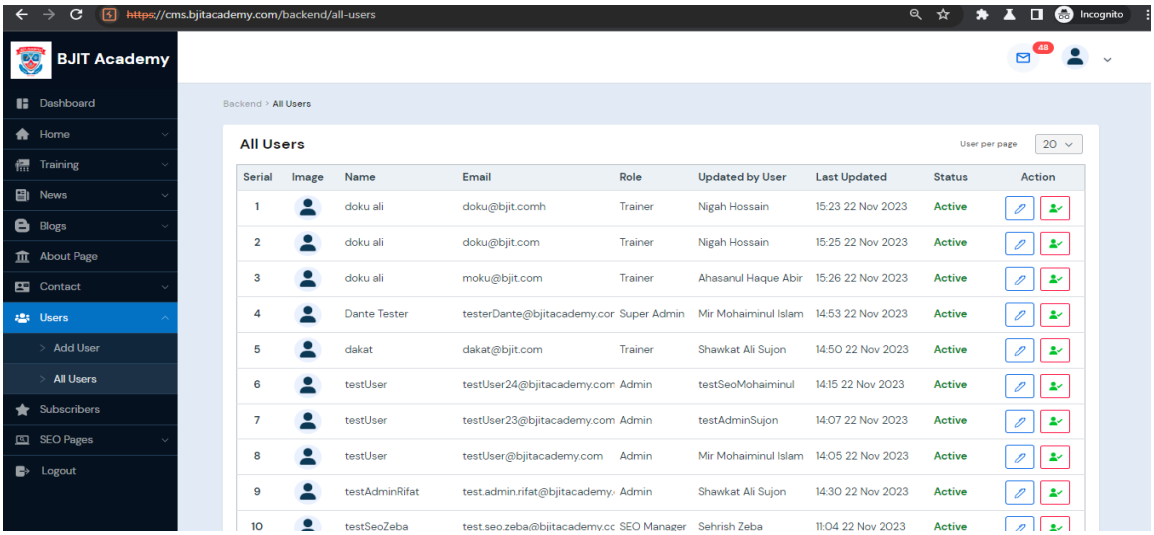


Here in the super Admin request I replaced the super admin's user_id with the trainer user_id which is 61

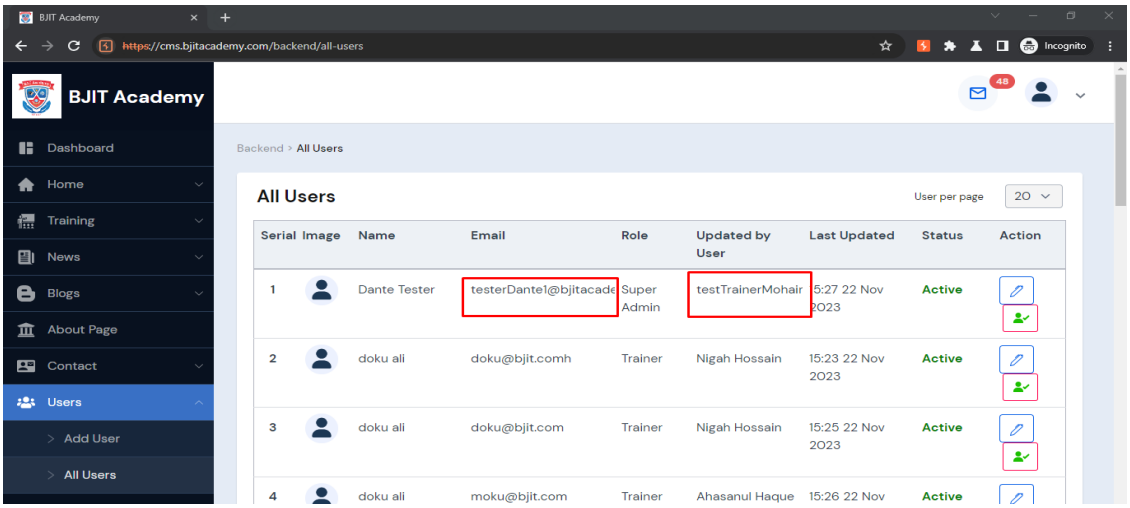


Now from the above we can see that new user is created by the trainer "testTrainerMohaiminul". Now I will show you from the frontend that the user is created and who created the user.

The first picture will show that before adding the user the frontend interface



After creating the user by Trainer the frontend.



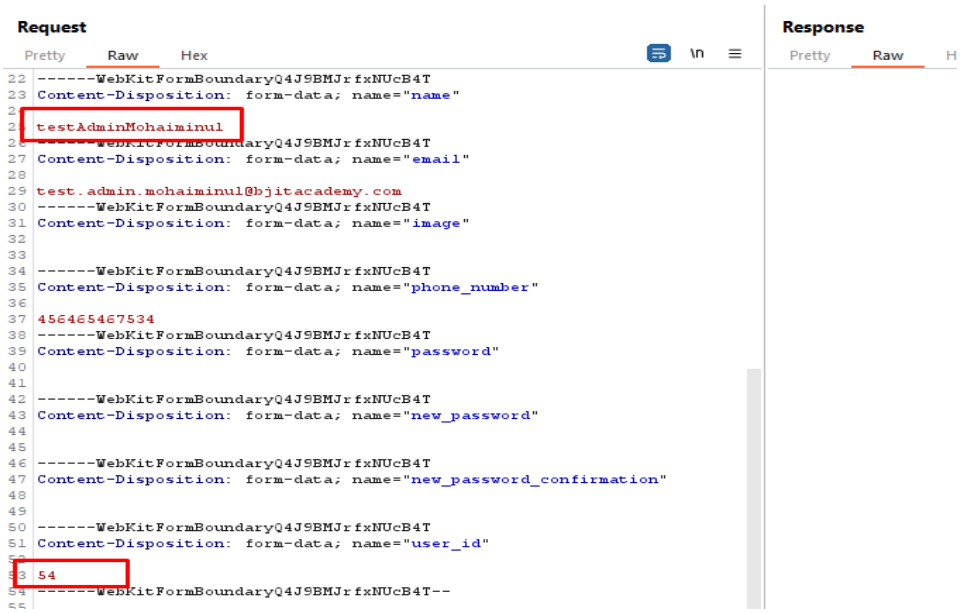
Similarly Admin, SEO Manager, Content Manager can also has unauthorized access to create a new User.

For Admin:

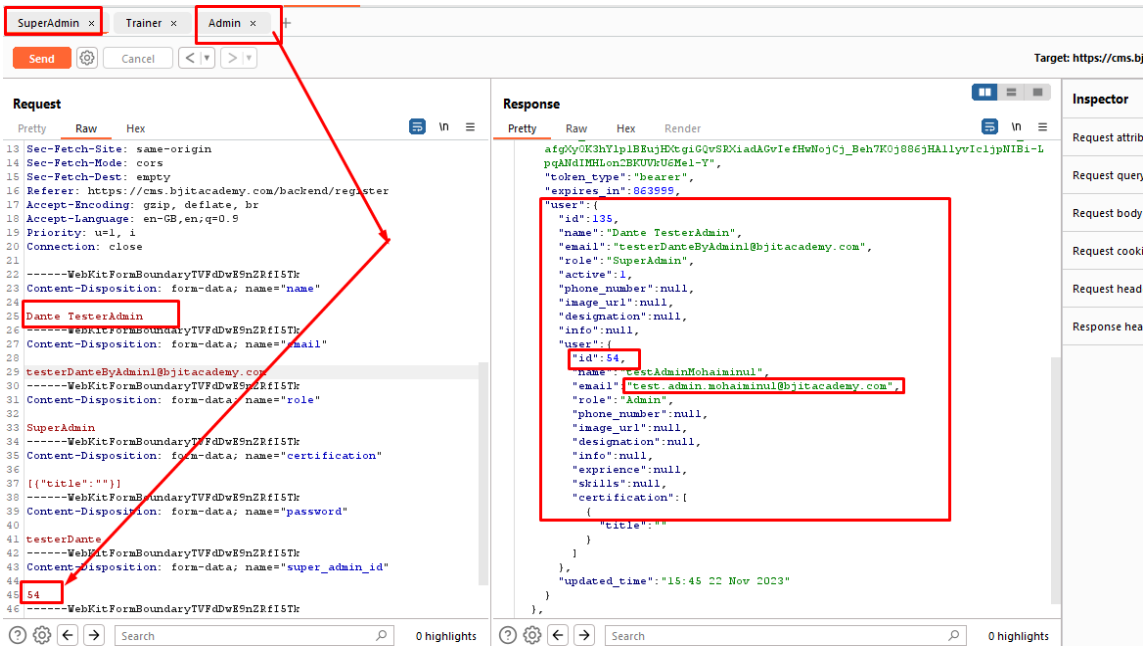
Here the user email: test.admin.mohaiminul@bjitacademy.com

Username: testAdminMohaiminul

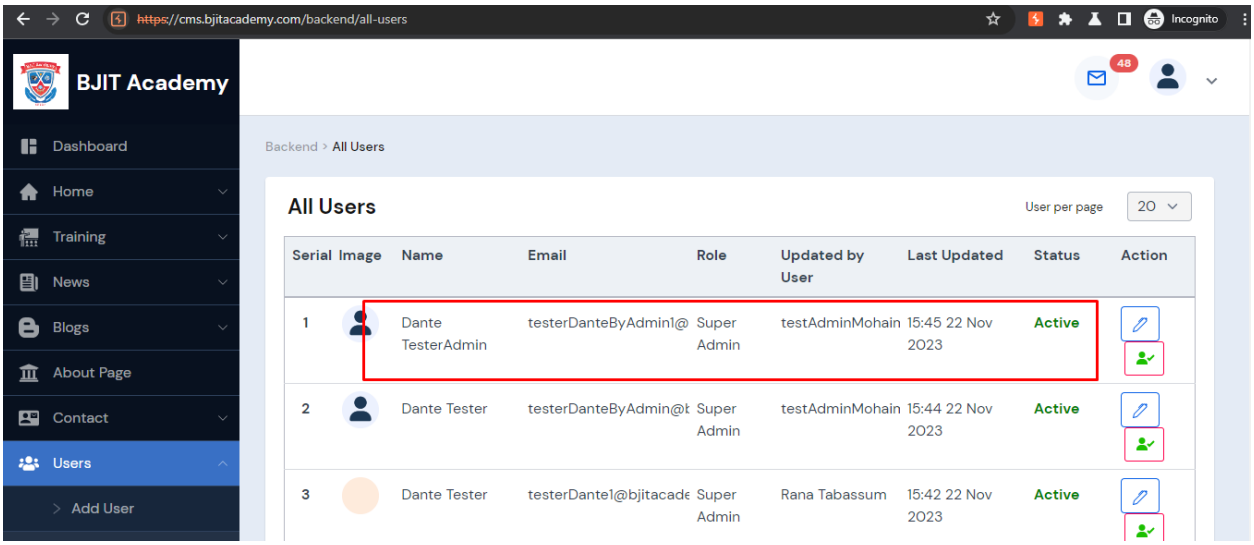
User_id : 54



Now Replace this superadmin user_id with the Admin user_id.



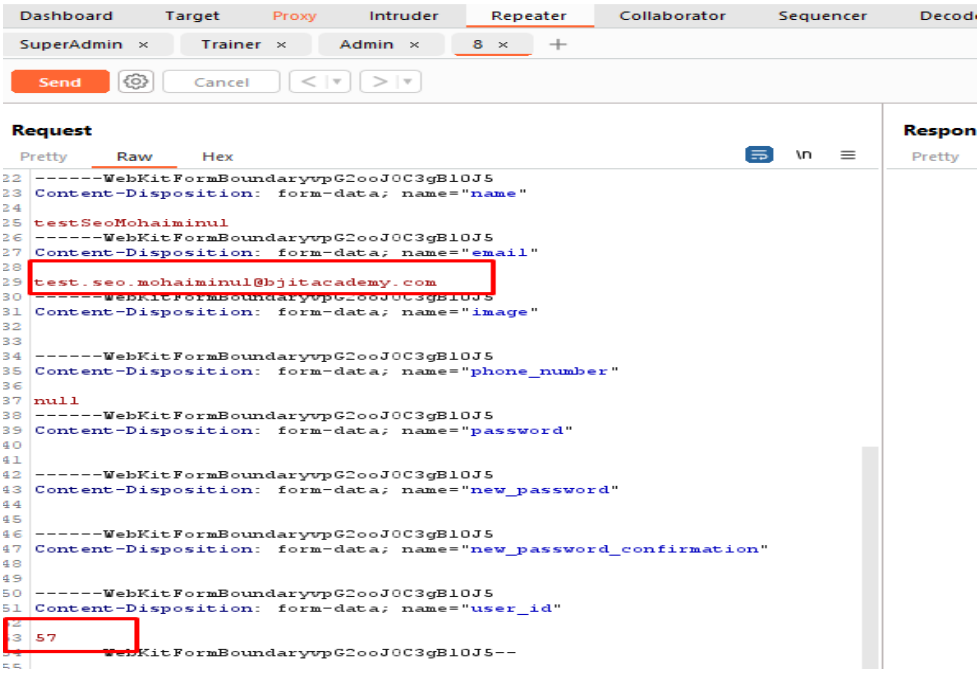
Here Admin also can create a user lets see it from frontend.



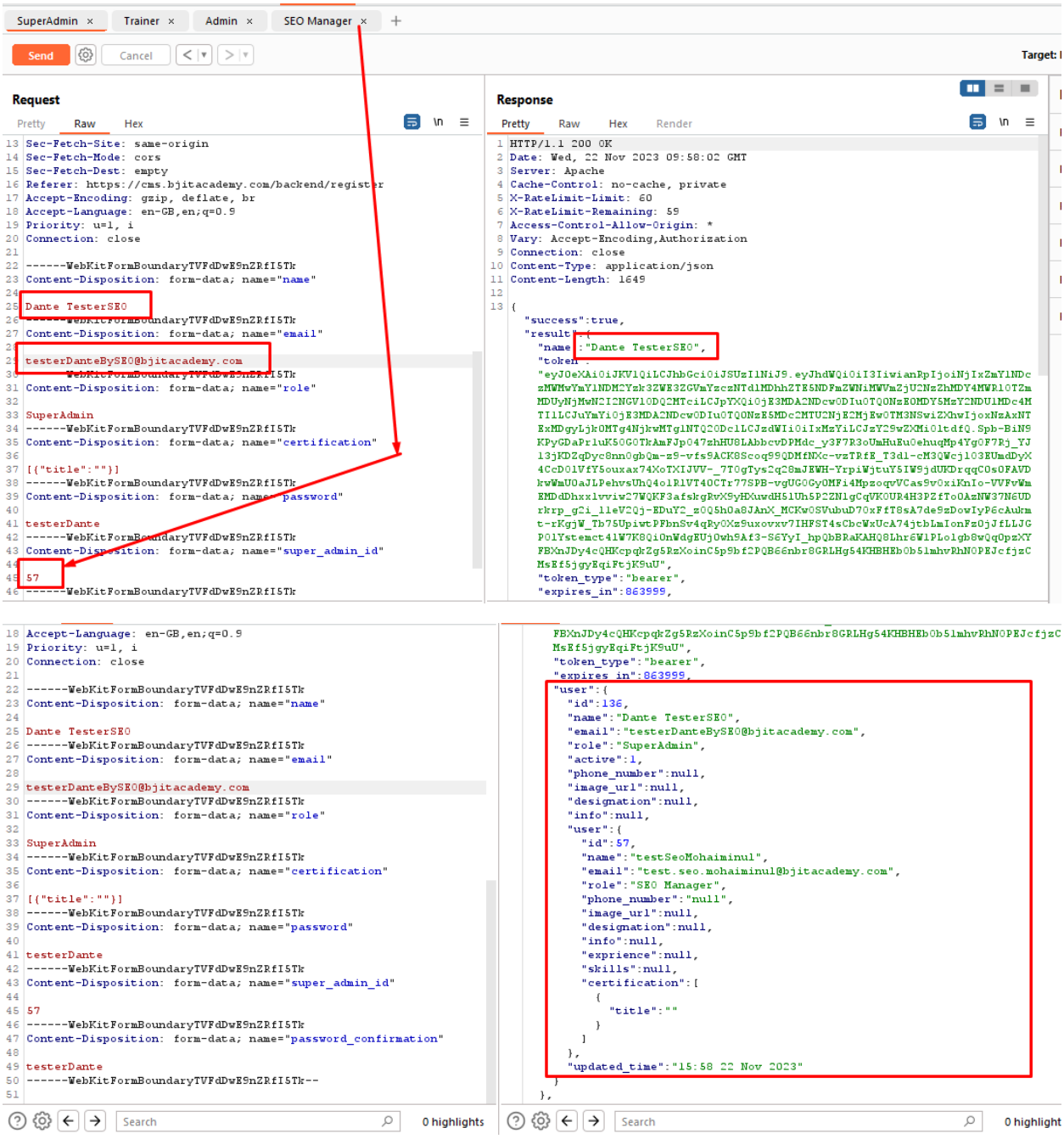
From SEO Manager:

Here the useremail: test.seo.mohaiminul@bjitacademy.com

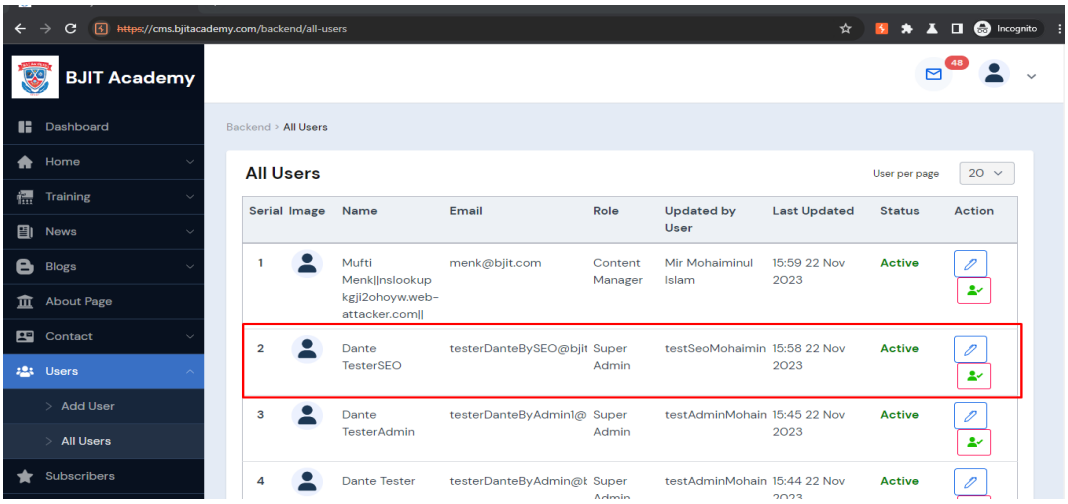
User_id: 57



Now replace the super admin user_id with the seo manager user_id.



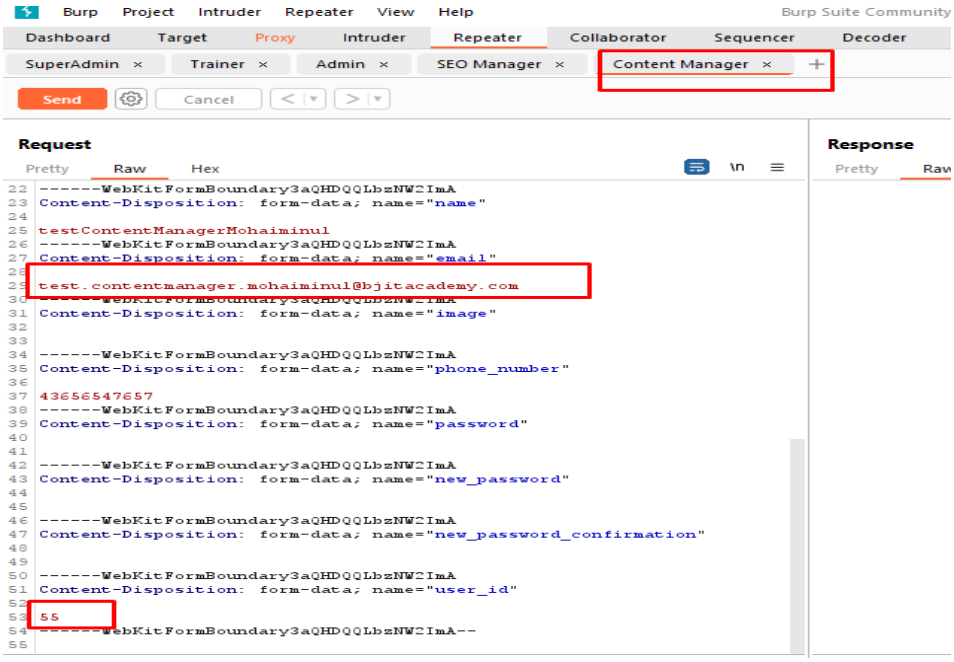
Now see the output from the frontend.



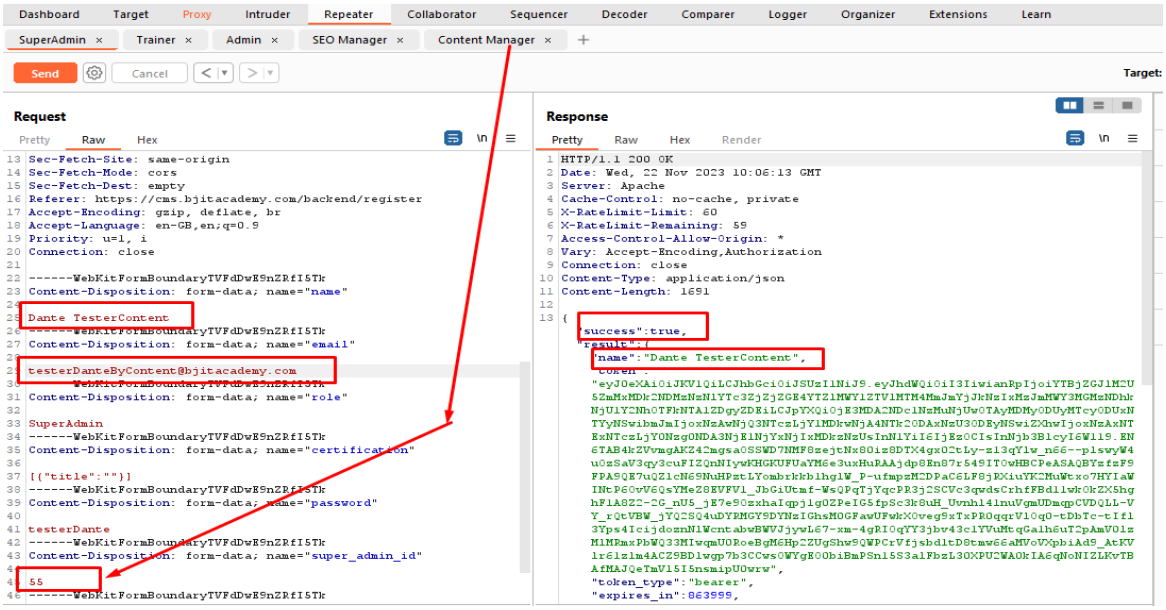
From Content Manager:

Here the content manager email_id : test.contentmanager.mohaiminul@bjitacademy.com

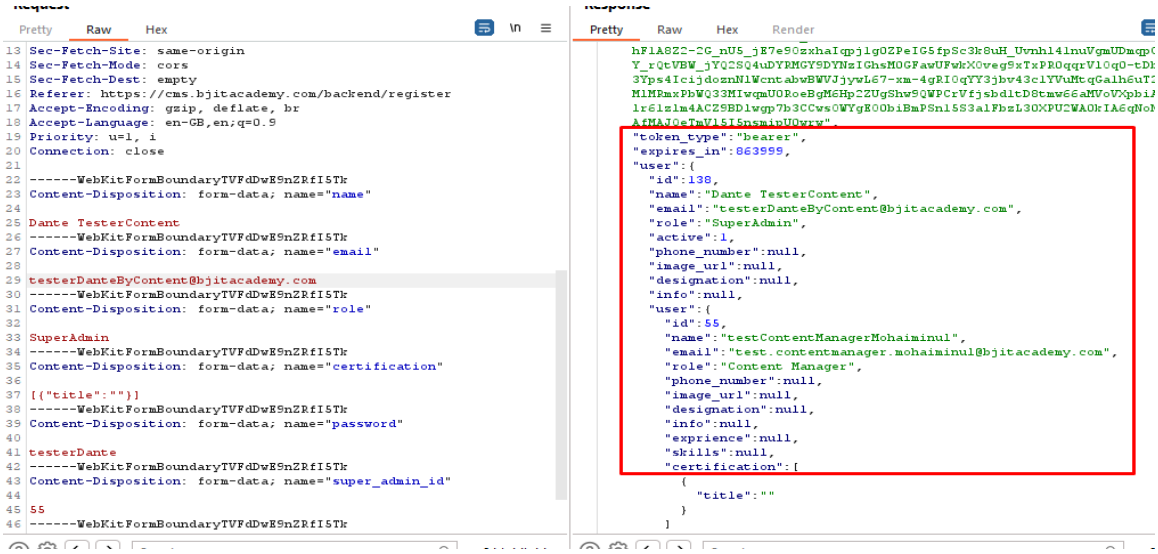
User_id: 55



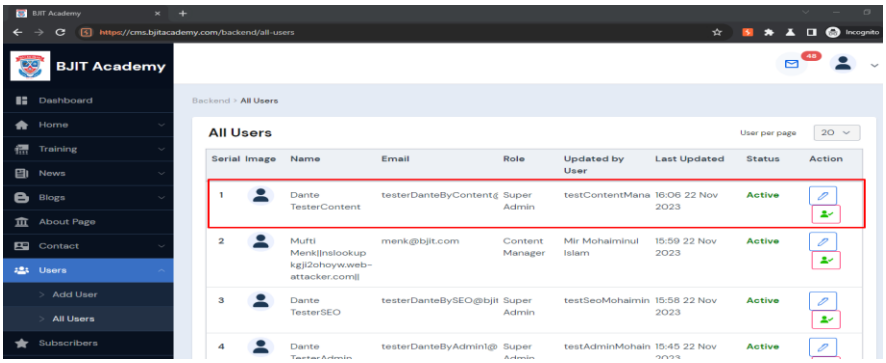
Now replace the super admin user_id with the content manager user_id.



After creation of the user we can see that content manager id can be used to create a new user and it shows that content manager create the account in reality it is created by a other user.



Now from the frontend we see that,



Here ,

Expected Result: Only super admin can create an user and one user can't create another user by manipulating the user_id.

Actual Result: User_id can be manipulated and by replacing a super admin user_id with another user's user_id , a new user can be created where the normal user is unauthorized to perform this task.

5. **Title: Unauthorized role assignment (Horizontal Privilege Escalation) to the new users(Add User) and the existing users(Edit User).**

   Target: cms.bjitacademy.com

   Affected Url/API:

   POST /academysite/api/public/api/v1/user/register (For Add a user with unauthorized role)

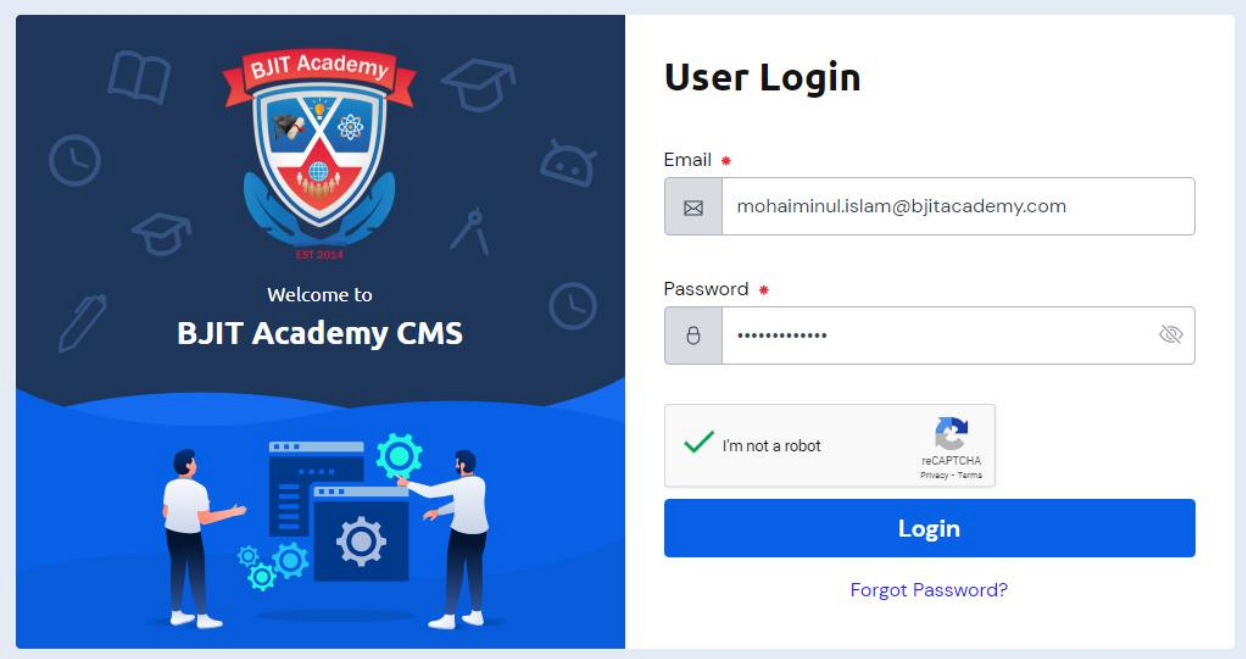
   POST /academysite/api/public/api/v1/user/update-user-by-super-admin (For existing user update who has unauthorized user role)

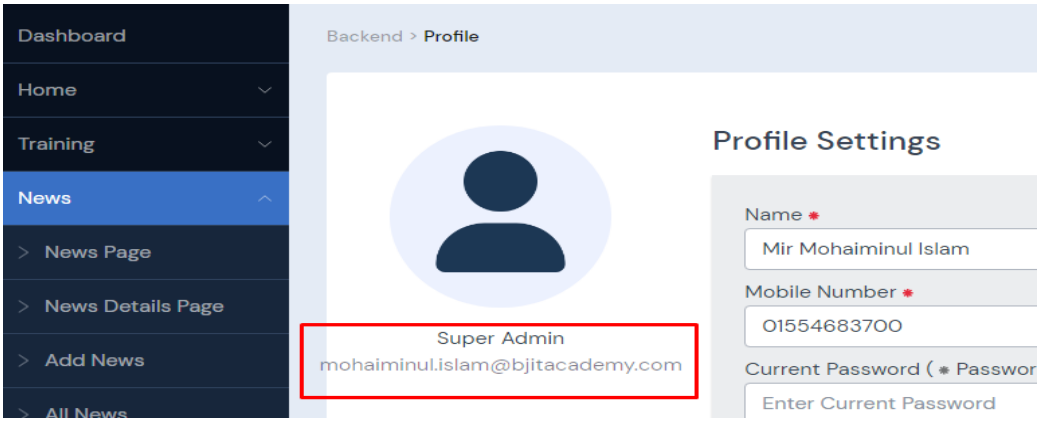
   Summary: At the time of creating new user any kinds of user role can be assigned which is not exist in the website which is created horizontal privilege escalation.

   Proof of Concept:

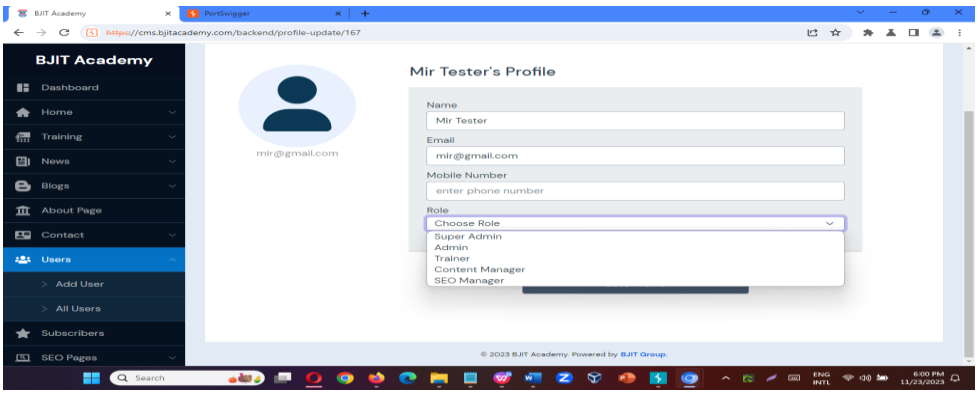
   For the proof of concept at first, I have to login with Super Admin login credentials with the build in chromium browser of burp suite.

   Here the Super-Admin User: mohaiminul.islam@bjitacademy.com



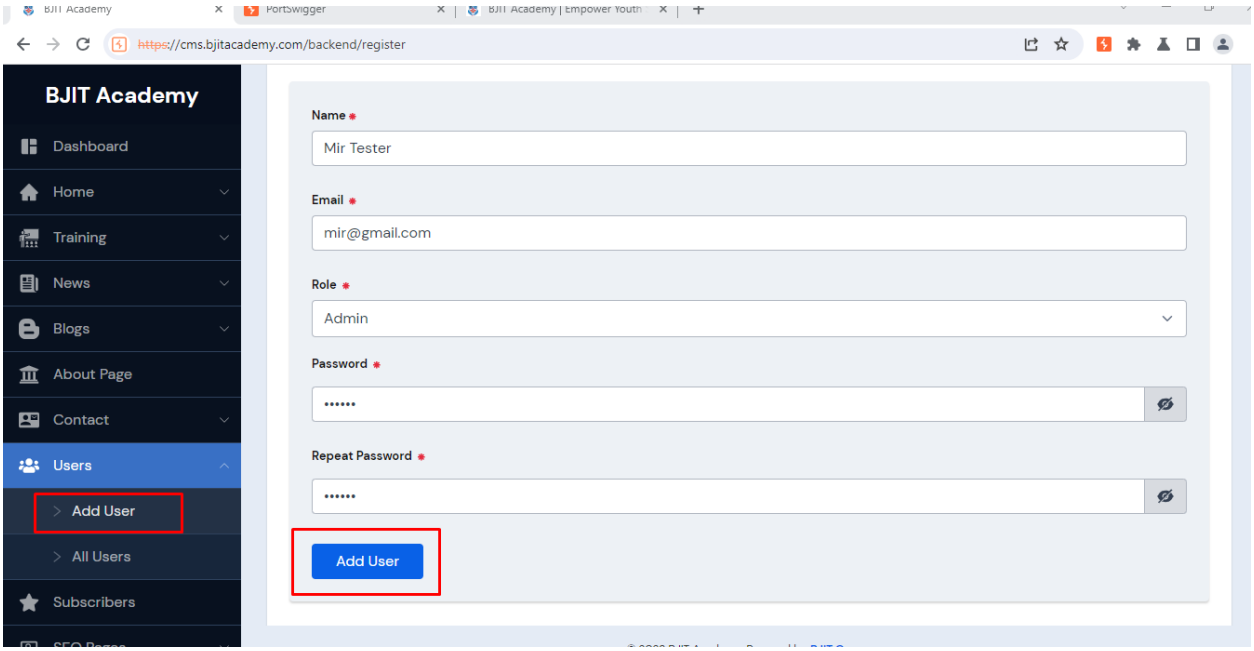


The website has several roles : Super Admin, Admin, Content Manager, SEO Manager, Trainer

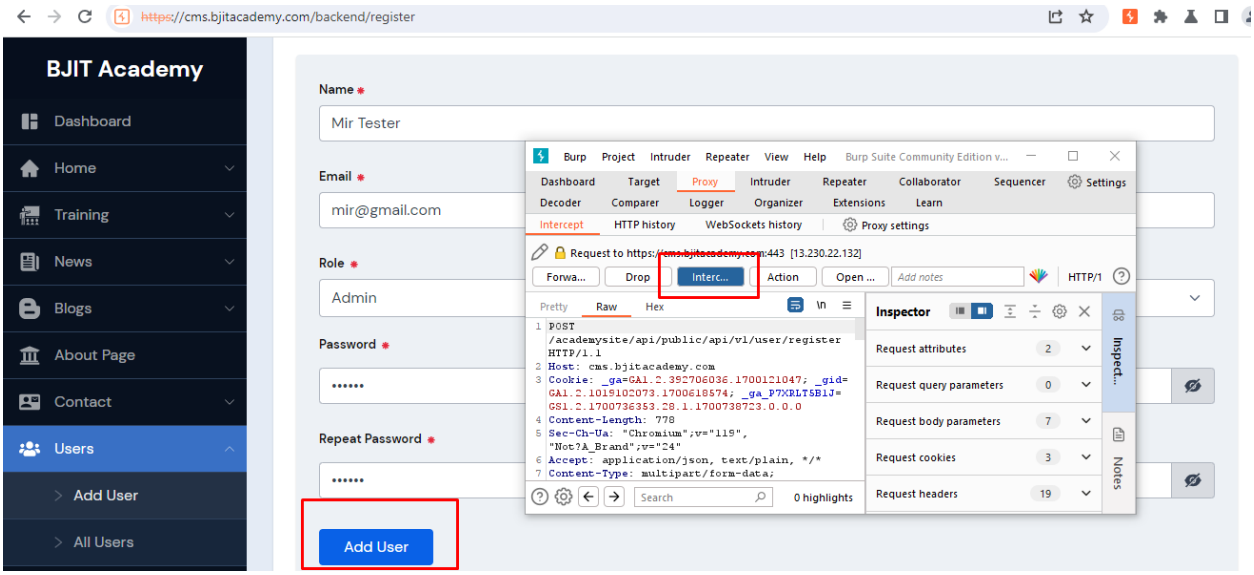


Now I went to Dashboard→Users→Add User page with **/backend/register** Url.

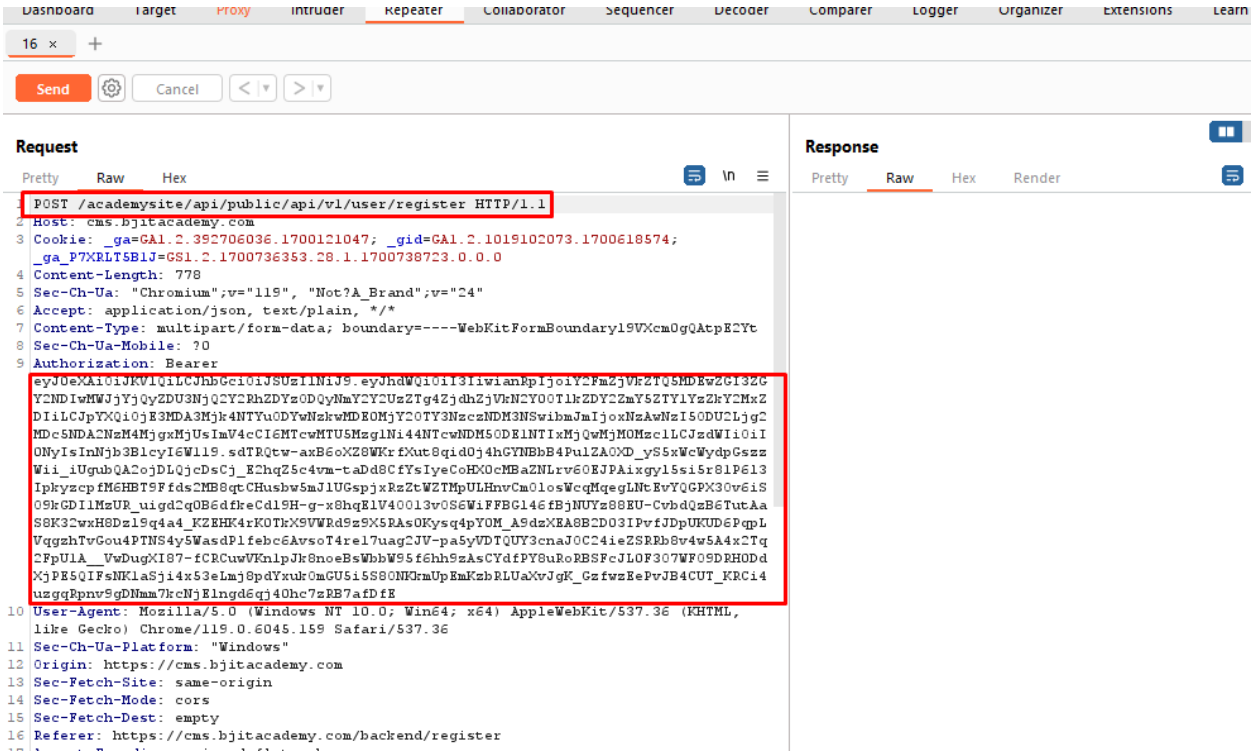
After fillup the form I click the Add User button.



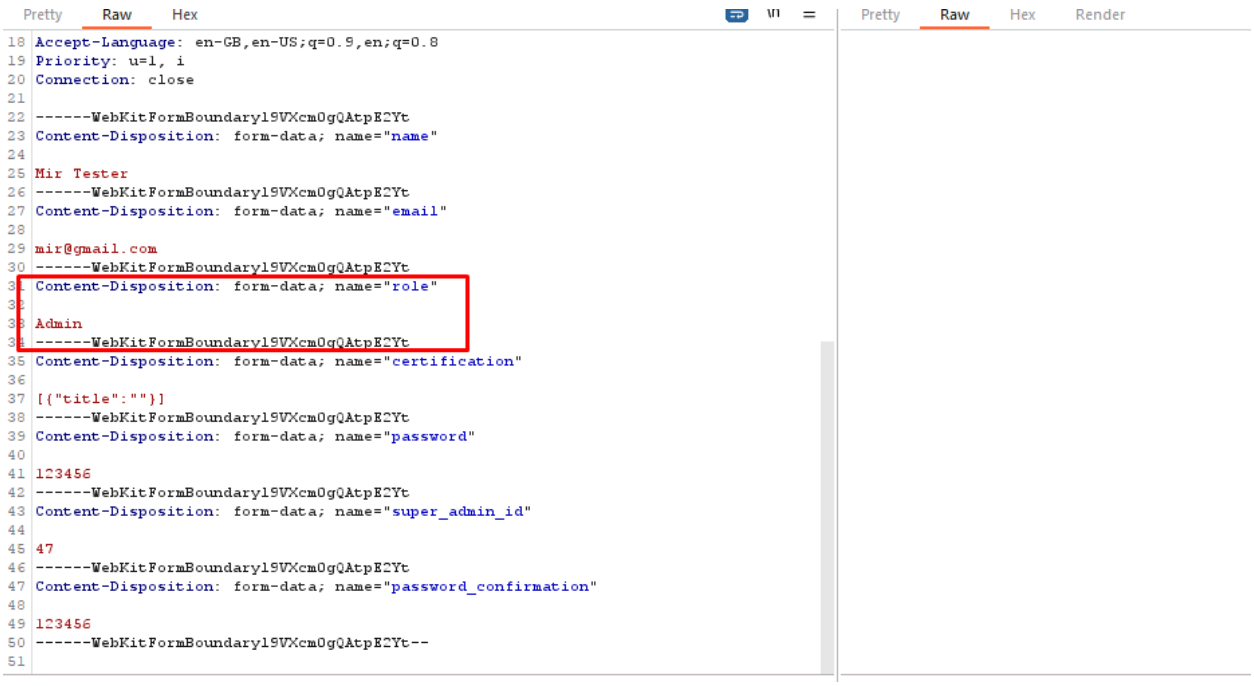
Here I capture the request with proxy intercept on. After that I send the request to the burp Repeater.



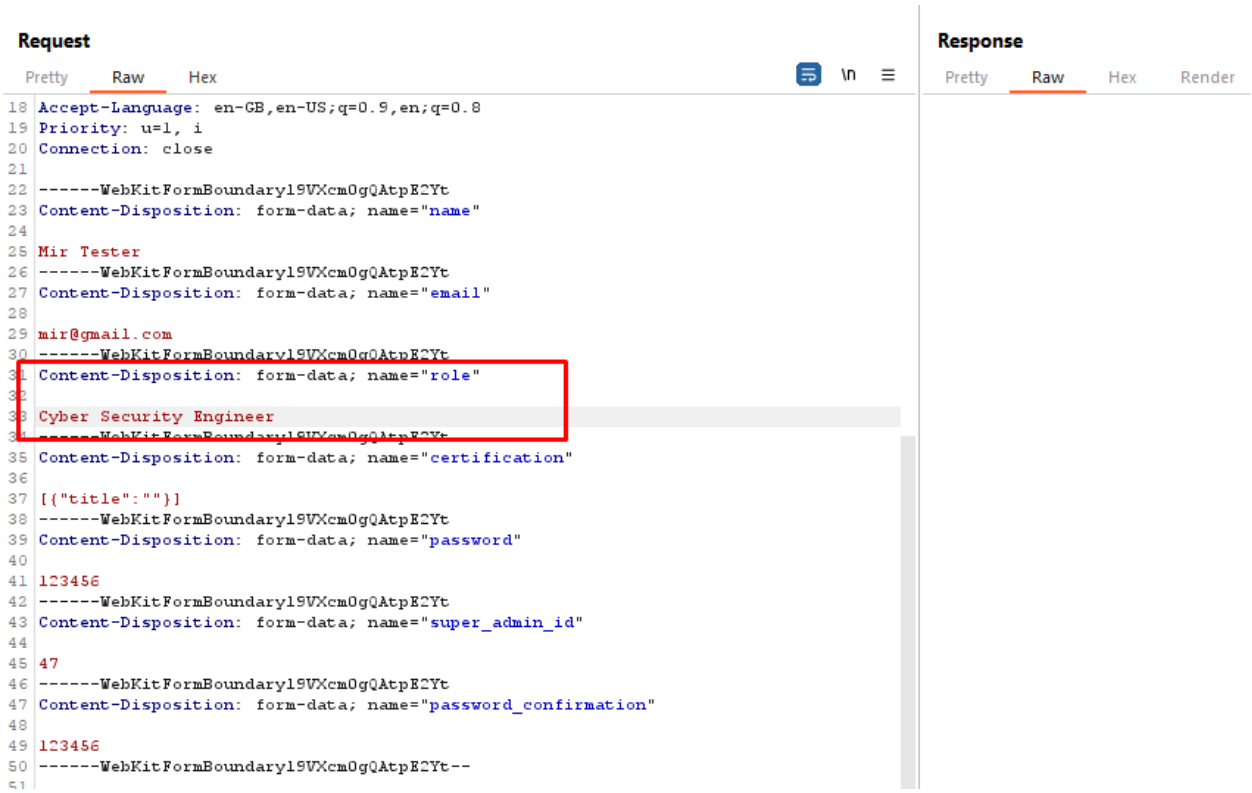
Here the API I am working with : **POST /academysite/api/public/api/v1/user/ register.**



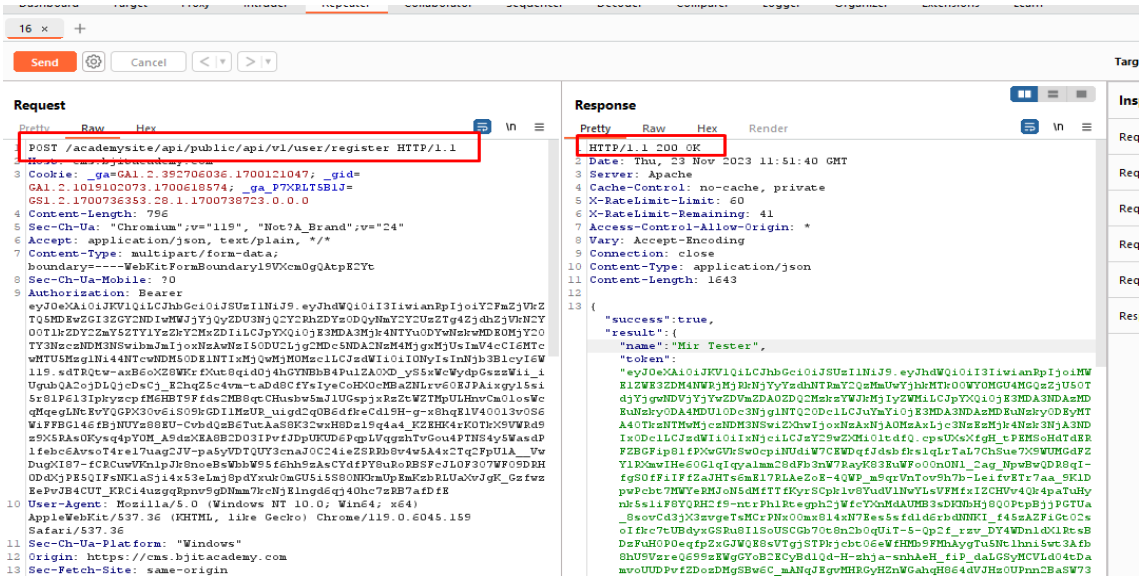
The User role is selected here is admin. The new user's role should not be manipulated by any kinds of role name. Lets try to change the role parameter.



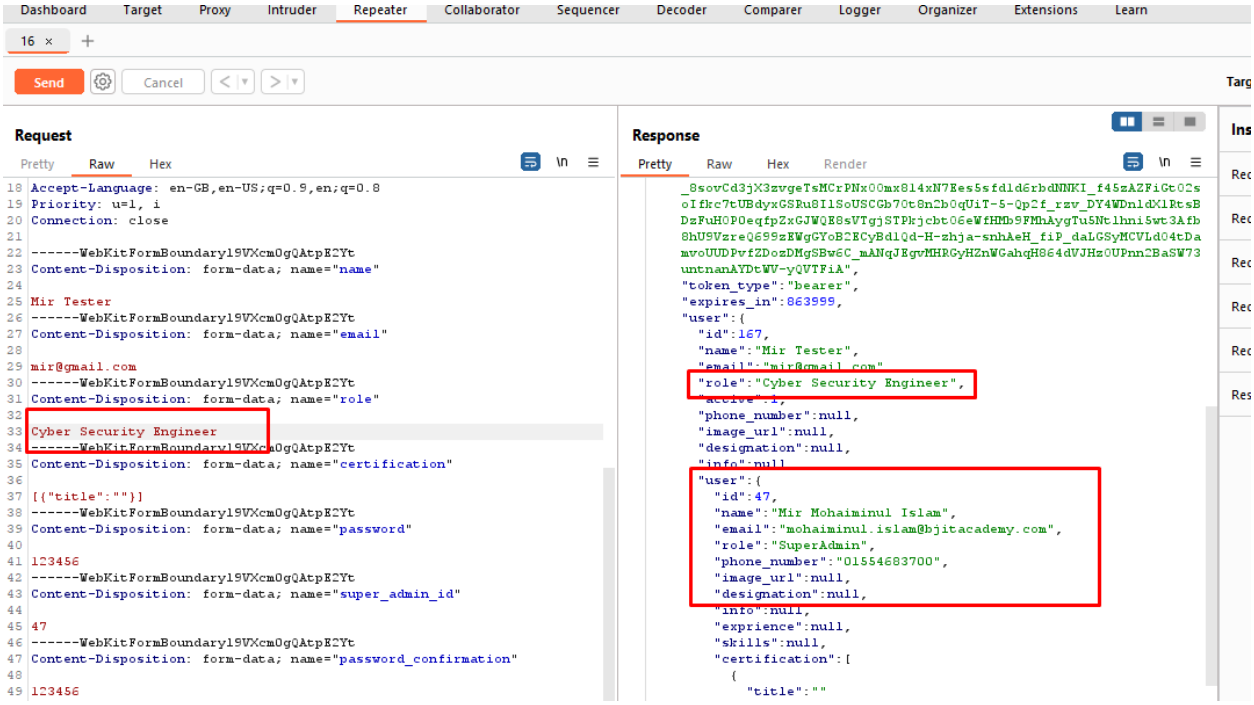
Here i changed the user role from 'Admin' to '**Cyber Security Engineer**'



Now I will click the send button to see the response.

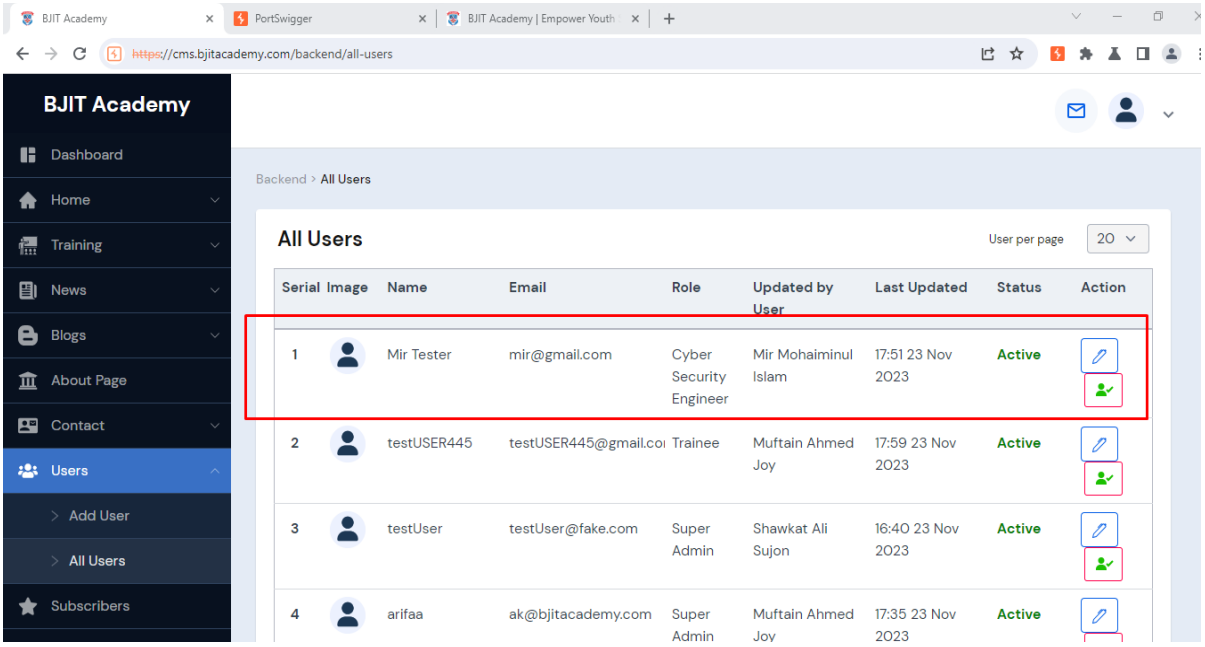


Here I found that the response is 200 OK. And It shows that the user is added successfully with the unauthorized role **'Cyber Security Engineer'.** In the website there is no such kinds of role found in the role list when a Super Admin take attempt to create a user.

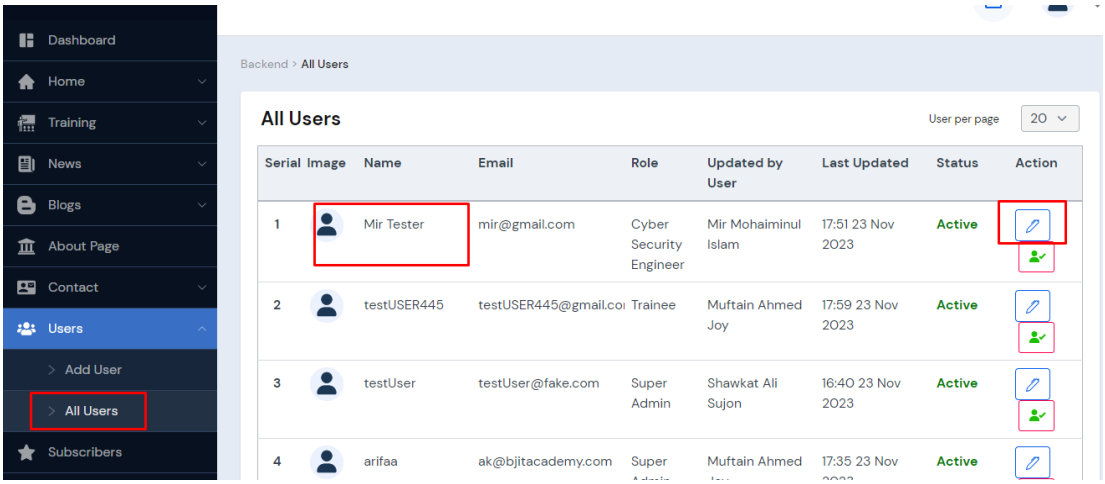


Here a super admin manage to create an unauthorized role based user. Here I changed the role to ‘Cyber Security Engineer’ and after clicking send the response shows that the user is created with the unauthorized user role.

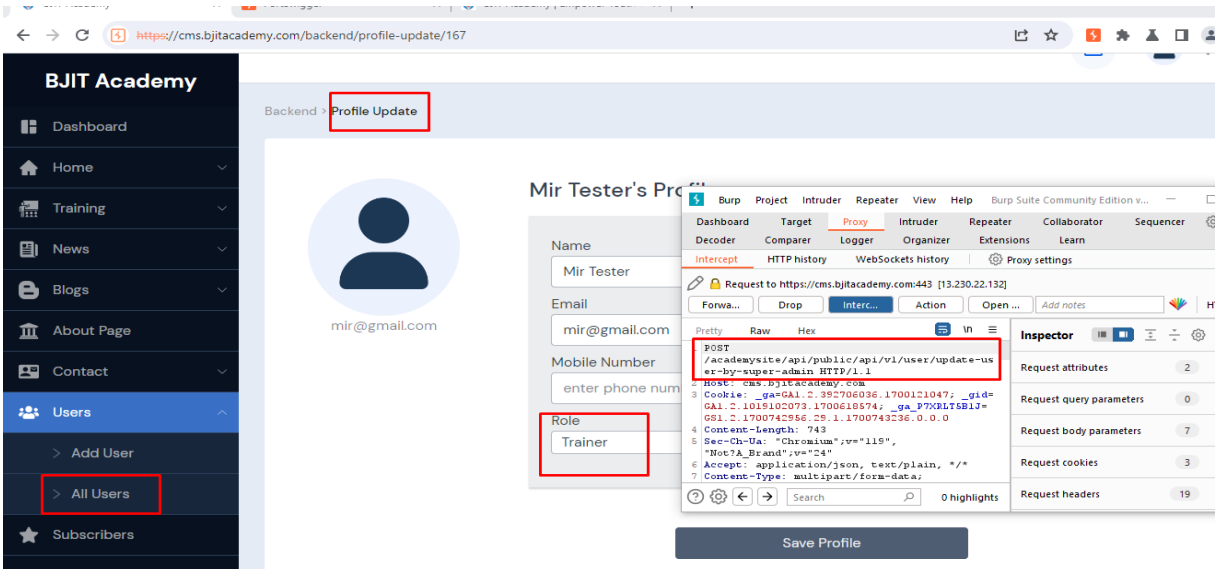
From the frontend We can see the output,



Now what will happen if I try to manipulate a user role which already exists. For that I will go to Dashboard → Users→ All User → edit a user by clicking the pen symbol→ update profile.

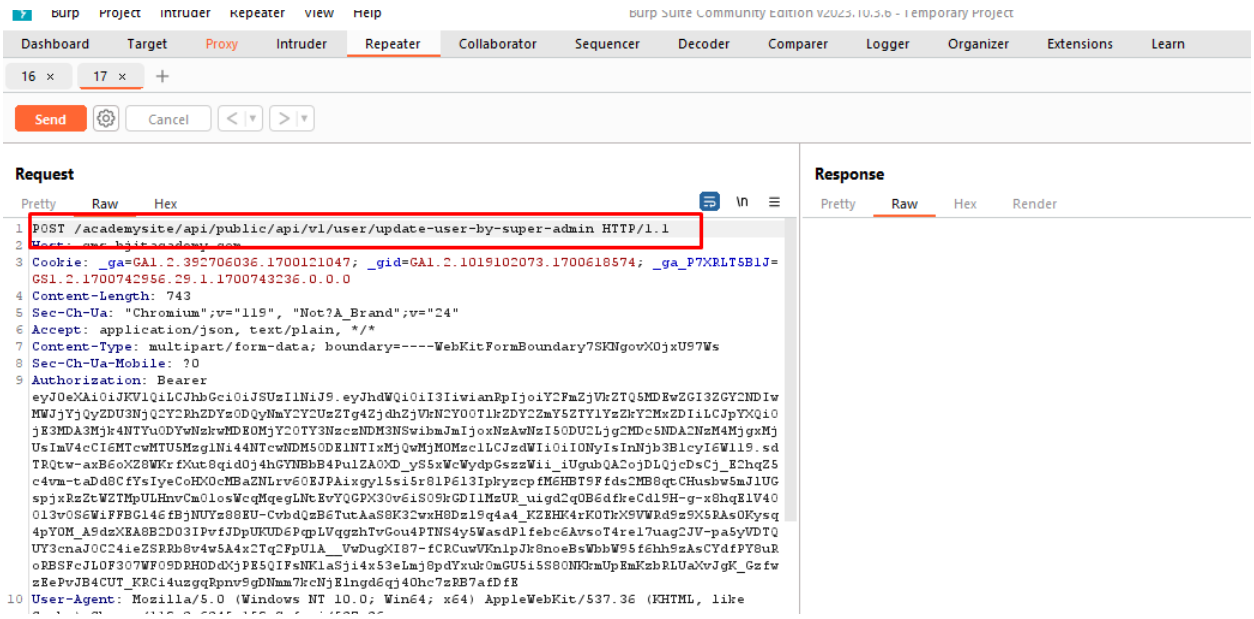


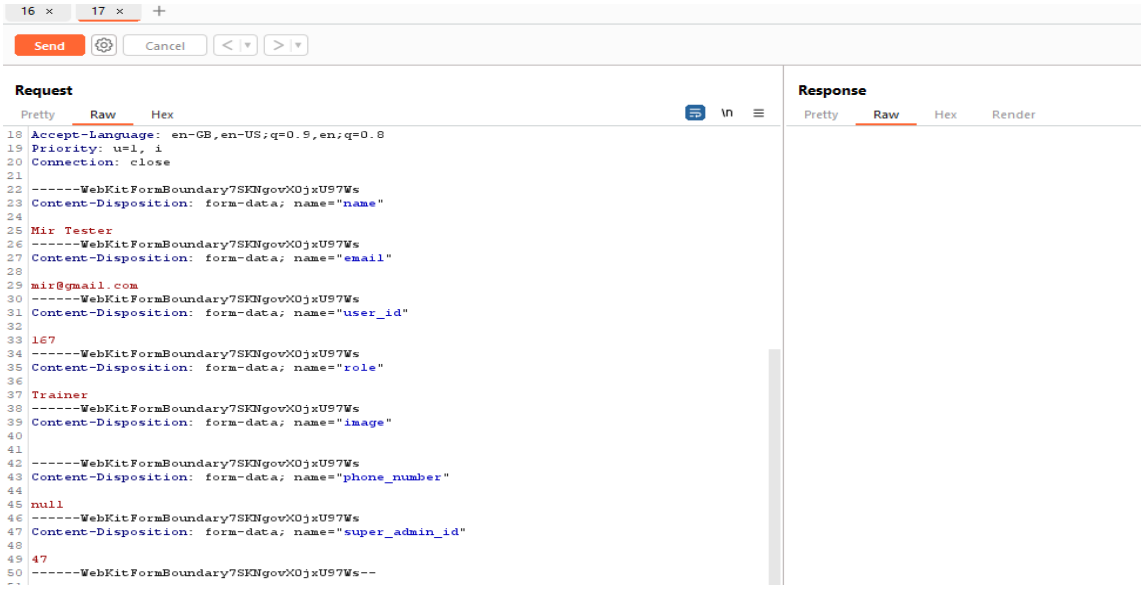
Here the existing user is Mir Tester and his current role is 'Cyber Security Engineer'. Let's try to change the role and capture the request.



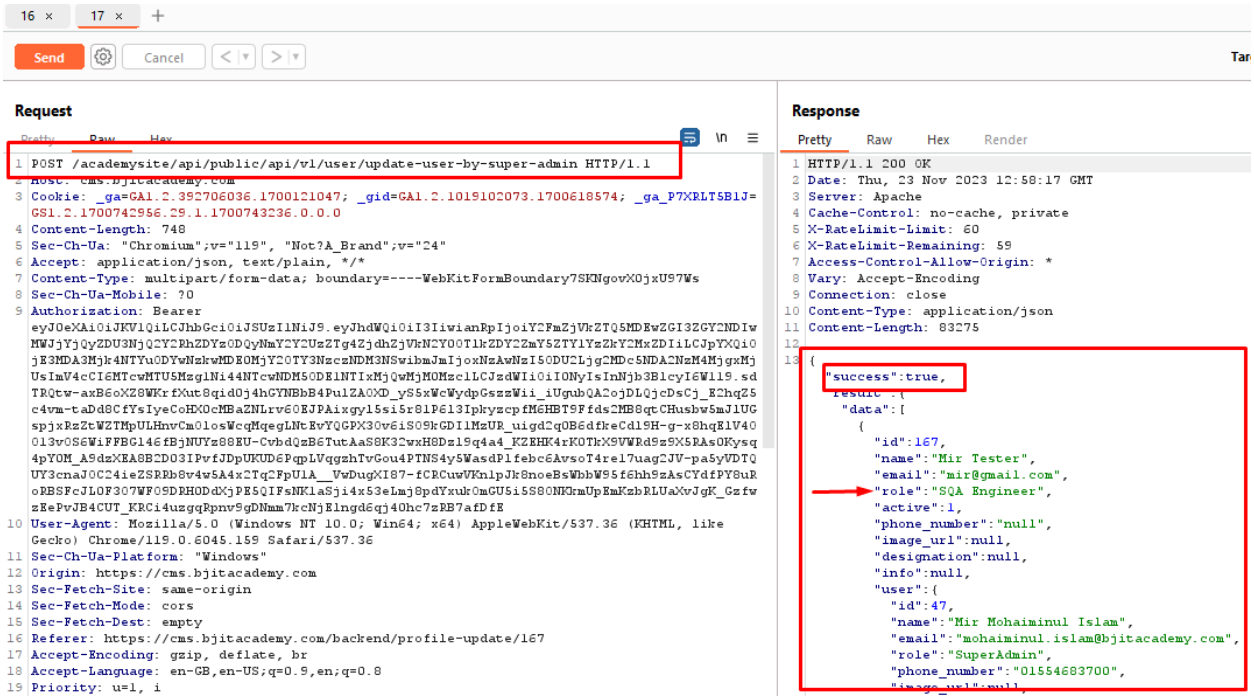
Let's take the request to the burp Repeater here the api is for update:

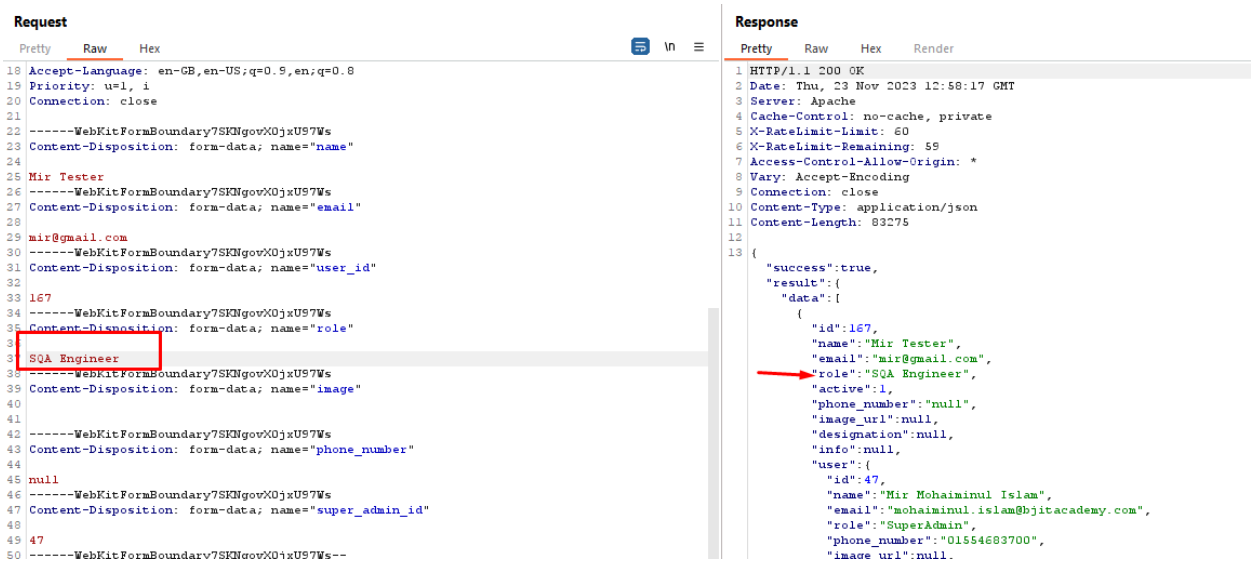
POST /academysite/api/public/api/v1/user/update-user-by-super-admin





Now I will change the role of this existing user to **''SQA Engineer''** which is also not in the list of the website user role and click the send button.





Here the response shows 200 OK messages and also shows the role of the existing user "**SQA Engineer**" whose user role was **'Cyber Security Engineer'** before.

6. **Title :** Non-Admin users (Content Manager, SEO Manager, Trainer) can update some unauthorized information such as clients information.

   Target: cms.bjitacademy.com

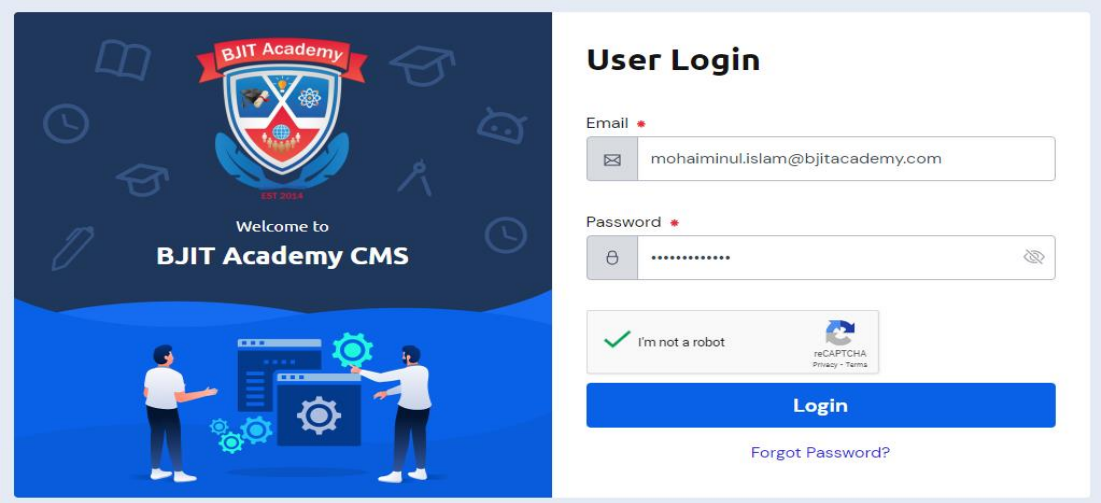
   Affected URL/API: POST /academysite/api/public/api/v1/client/edit-client/

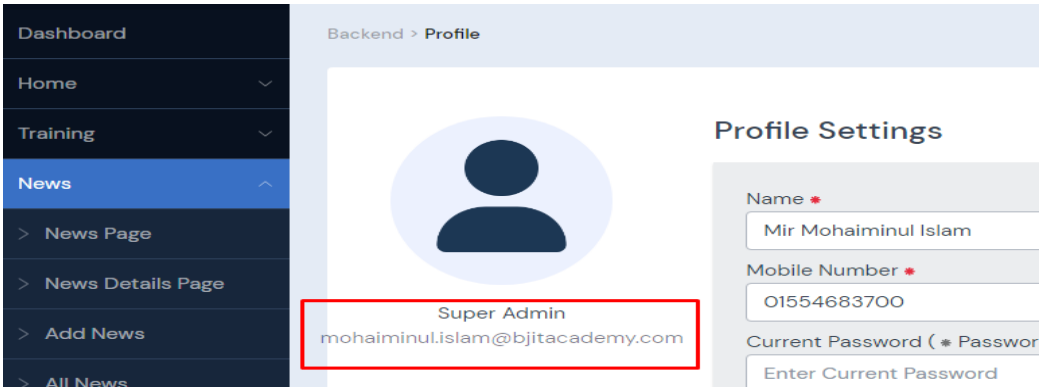
   Summary: Content Manager, SEO Manager, Trainer don't have access to update the information of client from the frontend but from backend API they can update the client information.

   Proof of Concept:

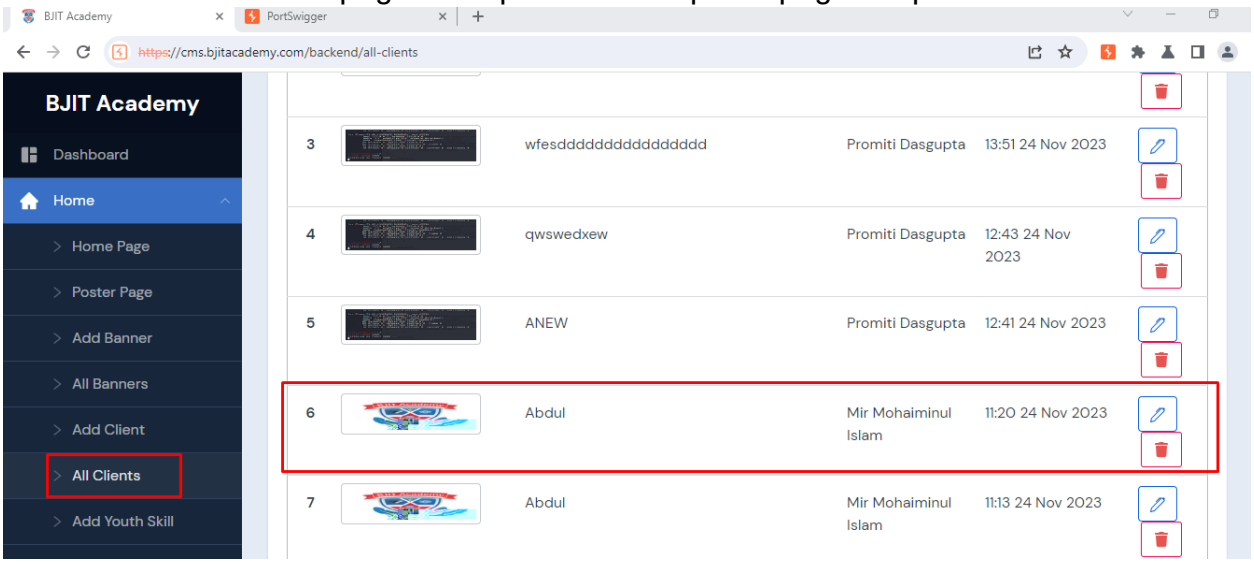
   At first go to http://cms.bjitacademy.com/login and login with super admin credential with built in chromium browser in the burpsuite.

   Here the Super-Admin User: Mir Mohaiminul Islam (mohaiminul.islam@bjitacademy.com)

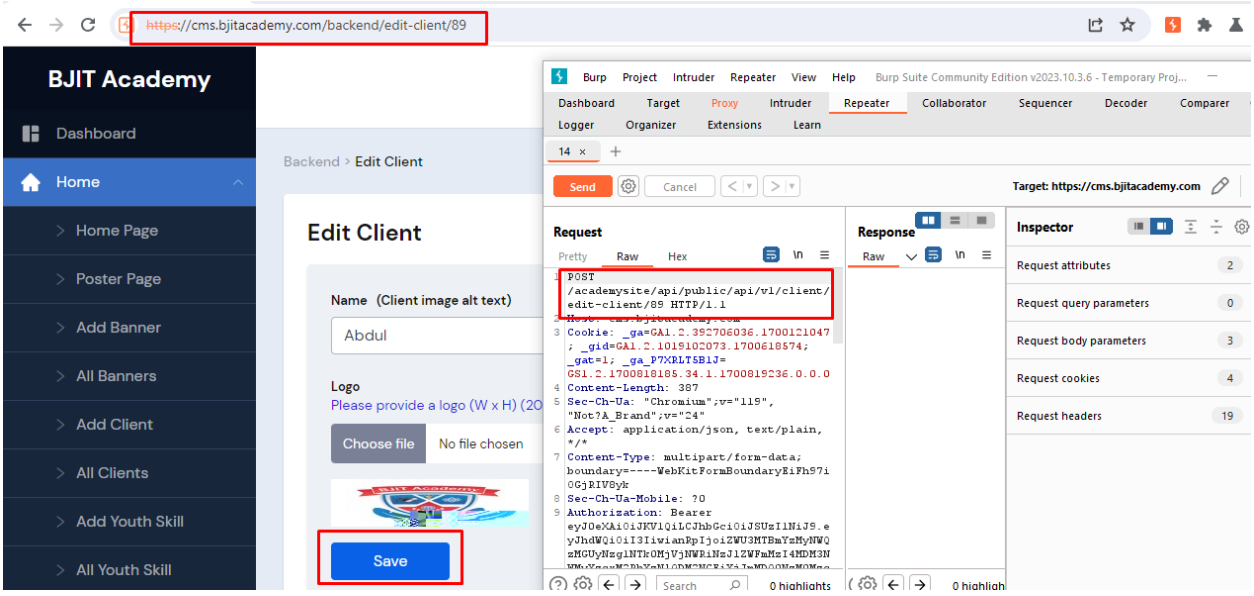




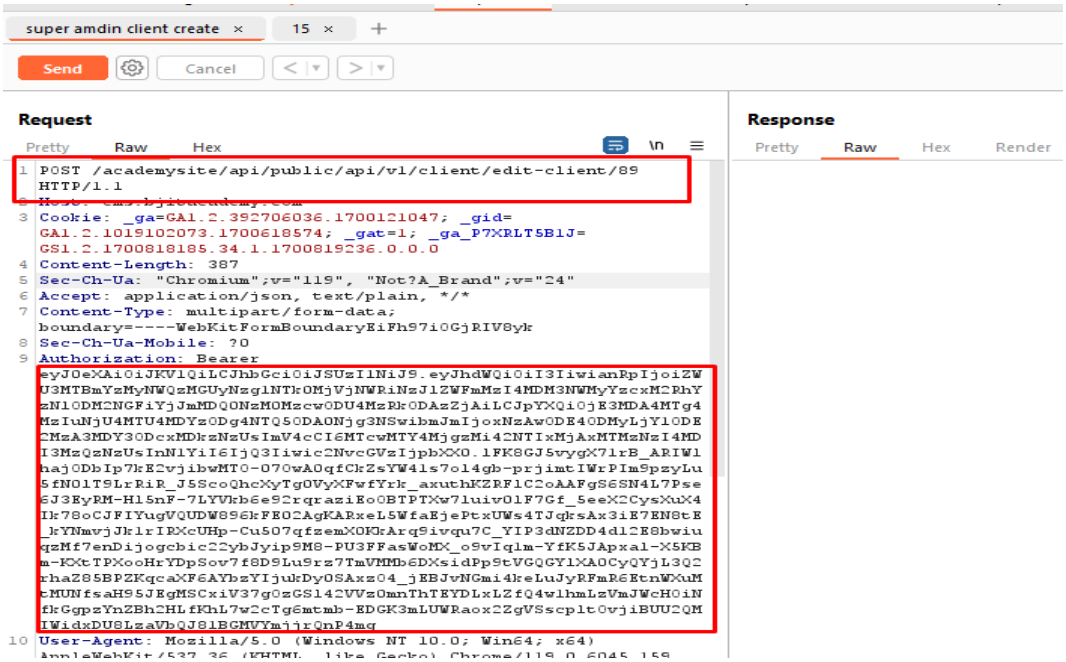
Then I went to all client page and open a client update page to update the client



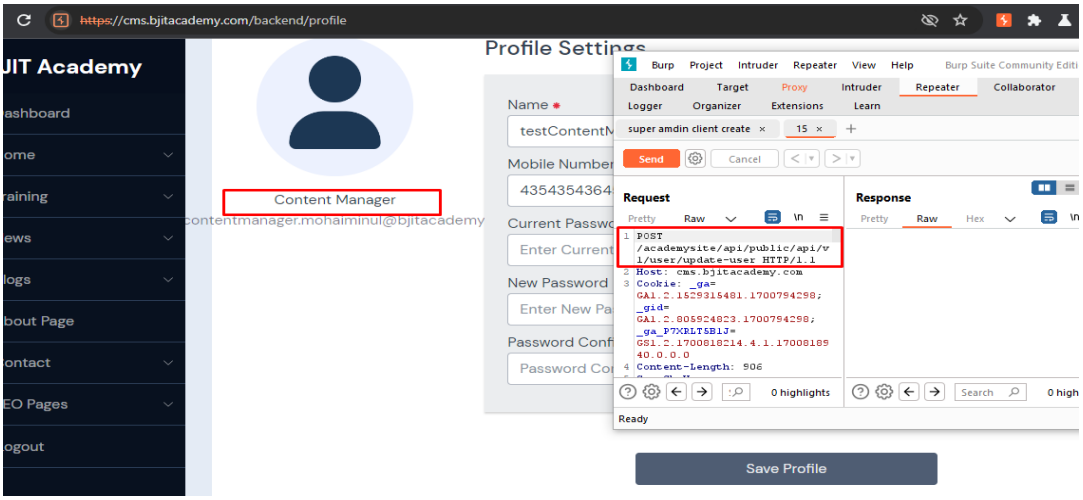
Now I clicked on the pen icon to edit the client information. After that I save the form by clicking **Save** button and send it to repeater by capture it.



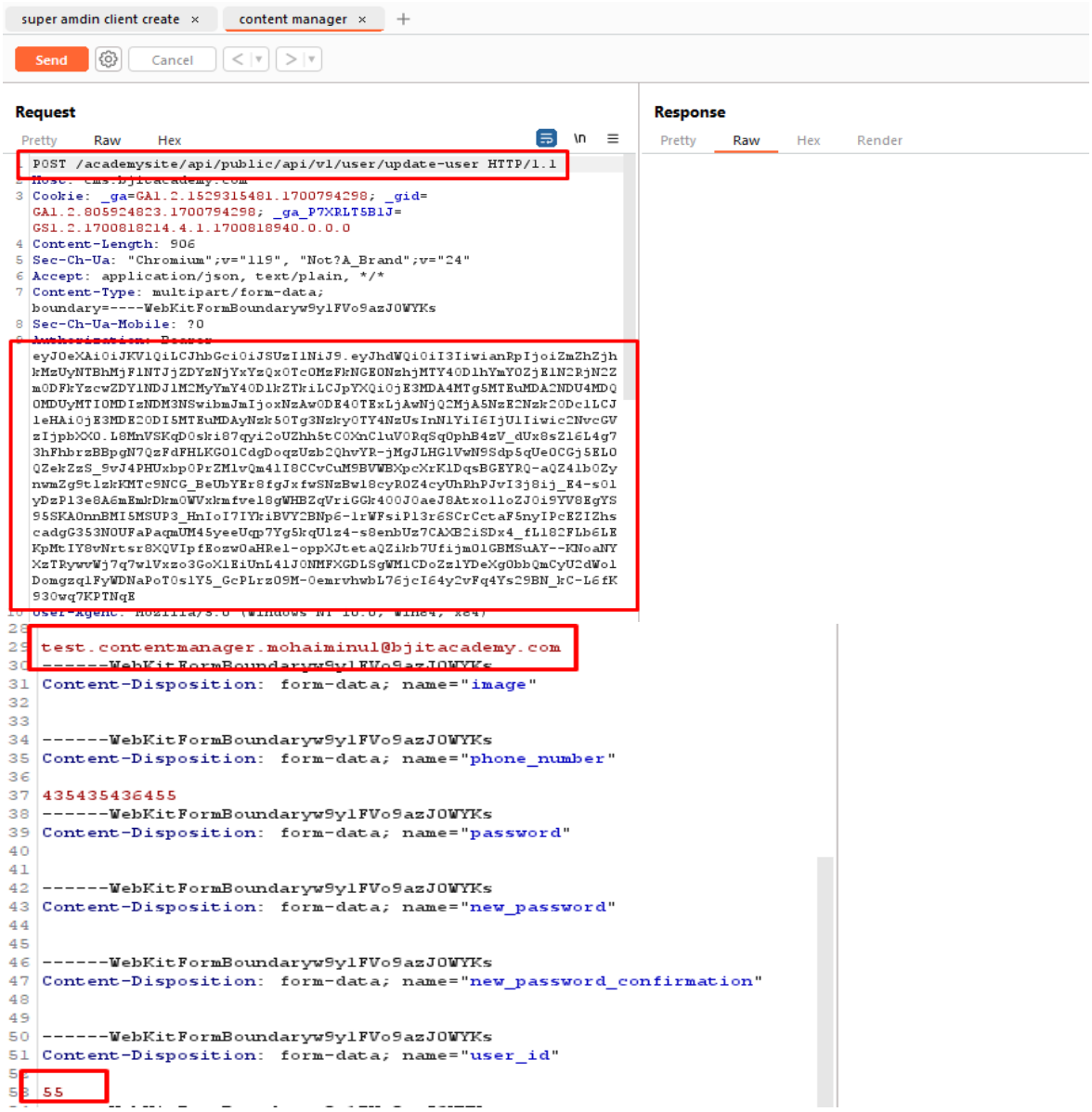
The below page shows the authorization token of the super admin.
Super admin's id: 47



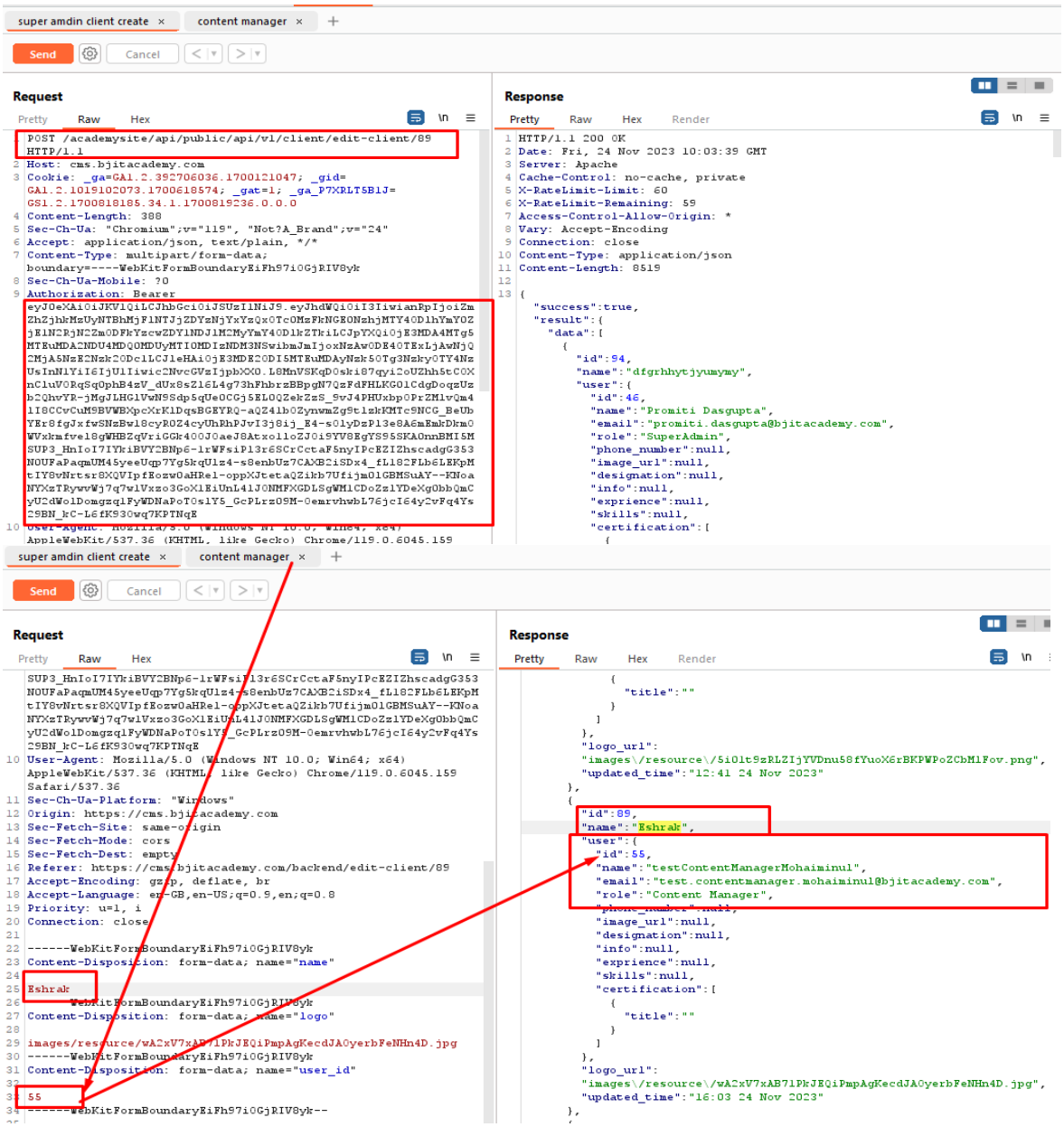
Now I need to collect content manager authorization token and user_id by Login→profile of content manager→save the profile and capture then send it to repeater.



Here the content manager authorization token and the user_is is shown in the below picture:

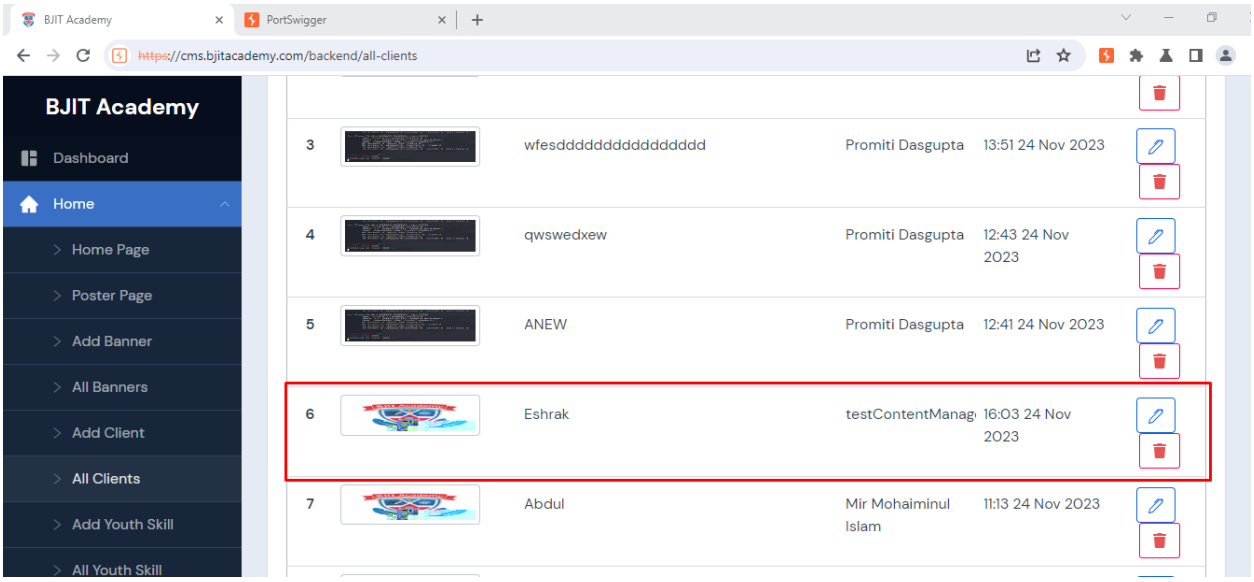


Now Replace the authorization token and user_id of a super admin with the authorization token of Content Manager.



Here client is successfully updated by manipulating the super admin's authorization token and his use_id with the content manager's credentials.

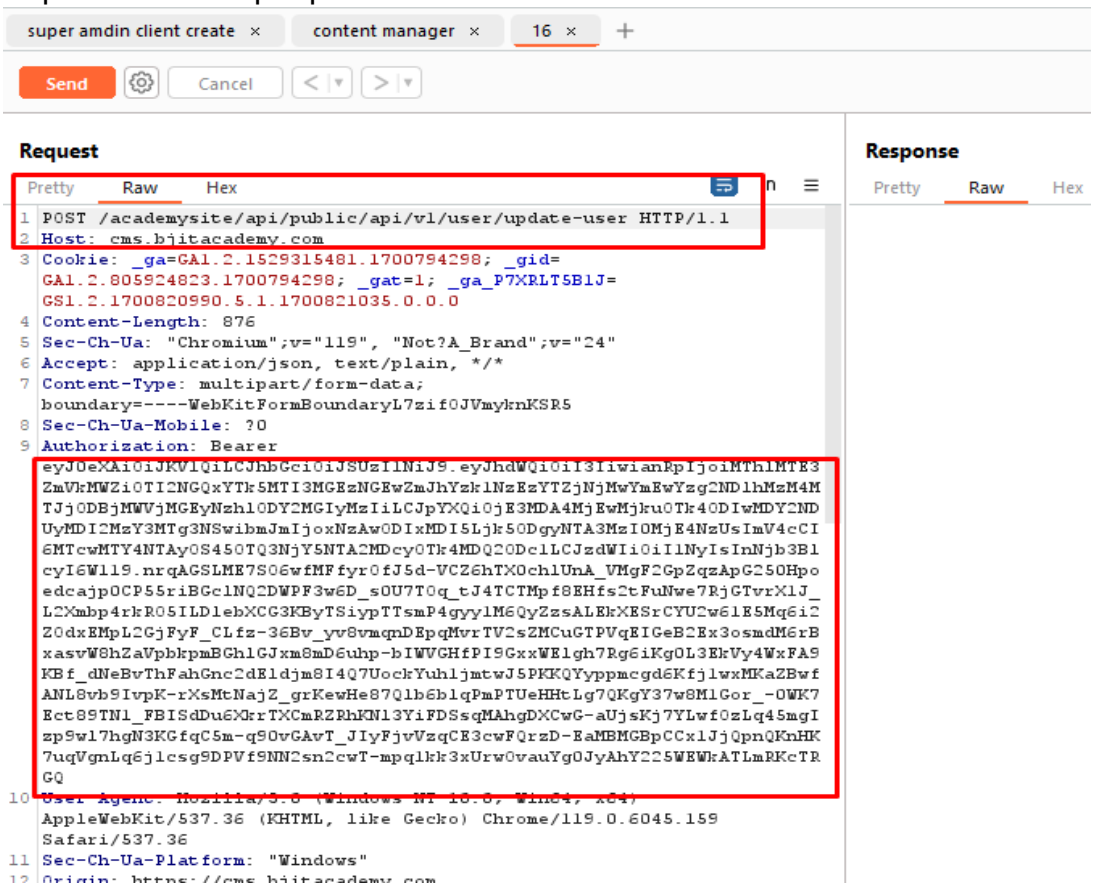
Now from frontend we can see that:



Similarly for SEO Manager:

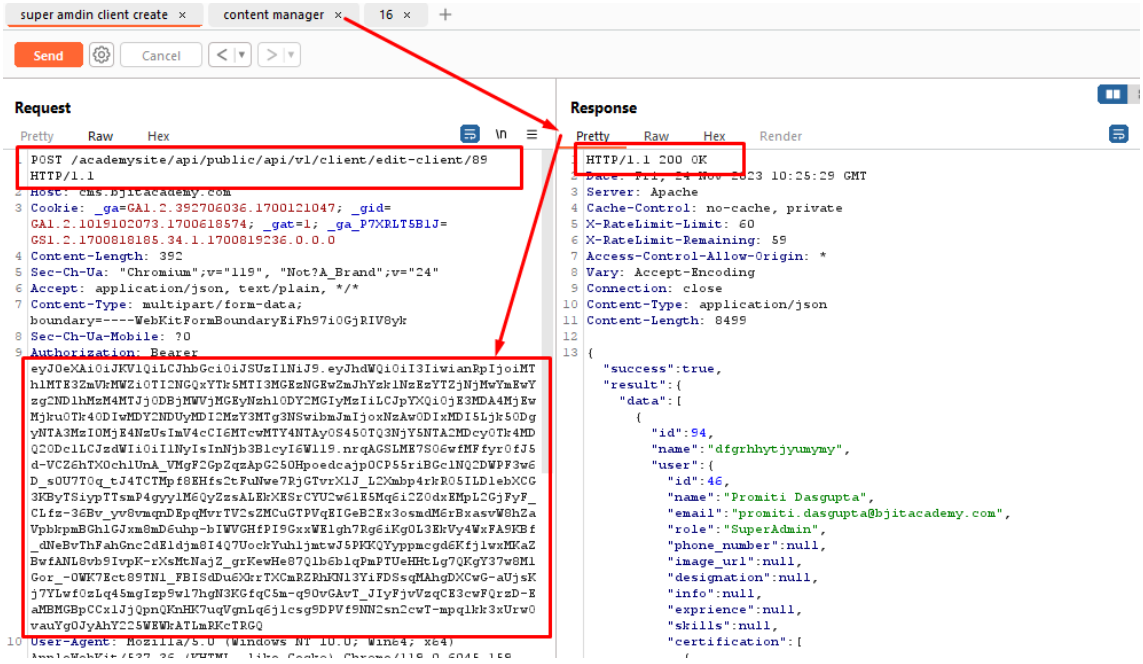
Getting SEO Manager's Authorization Token and user_id

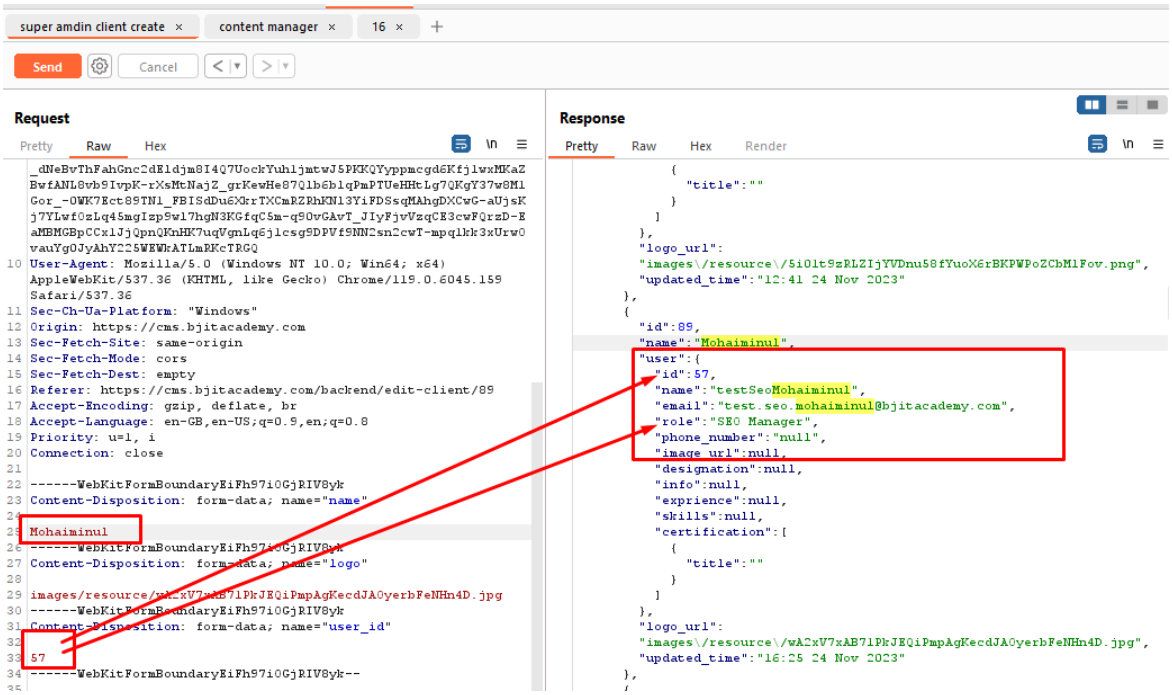
Login → go to profile of SEO Manager → update the page and capture the request to the burp repeater.



```
test.seo.mohaiminul@bjitacademy.com
------WebKitFormBoundaryL7zif0JVmyknKSR5
Content-Disposition: form-data; name="image"


------WebKitFormBoundaryL7zif0JVmyknKSR5
Content-Disposition: form-data; name="phone_number"

null
------WebKitFormBoundaryL7zif0JVmyknKSR5
Content-Disposition: form-data; name="password"


------WebKitFormBoundaryL7zif0JVmyknKSR5
Content-Disposition: form-data; name="new_password"


------WebKitFormBoundaryL7zif0JVmyknKSR5
Content-Disposition: form-data; name="new_password_confirmation"


------WebKitFormBoundaryL7zif0JVmyknKSR5
Content-Disposition: form-data; name="user_id"

57
------WebKitFormBoundaryL7zif0JVmyknKSR5--
```

Here the SEO Manager_id is 57.
Now replace the Super Admins authorization token and user_id with SEO Manager's authorization token and user_id.



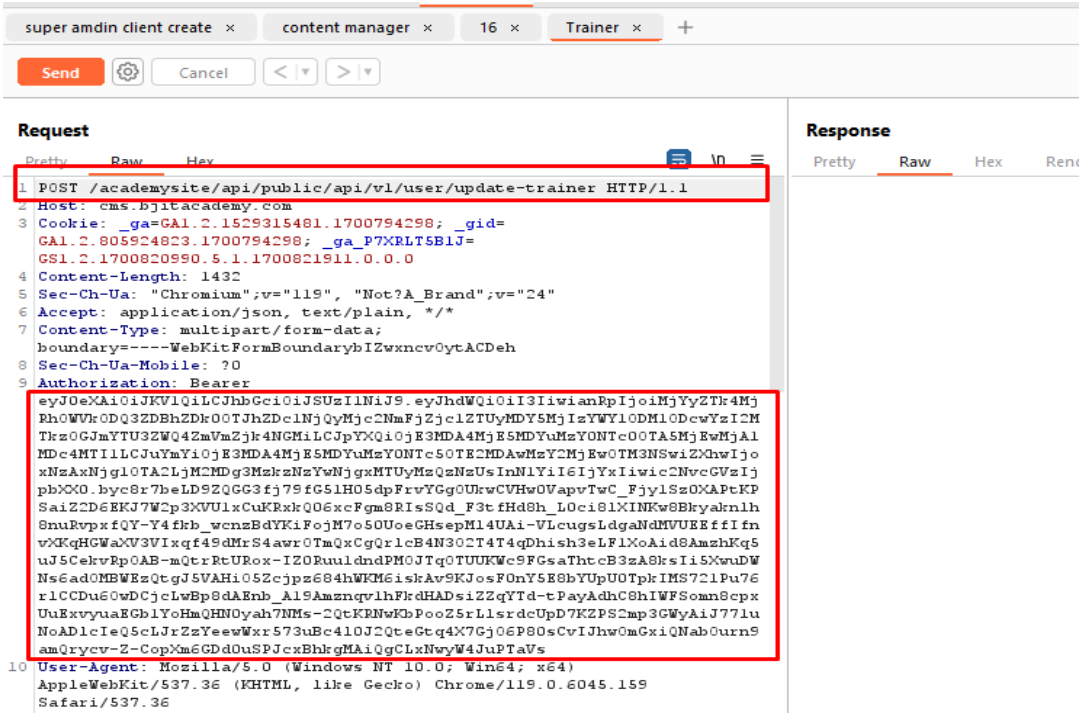


So SEO Manager can also update client information by manipulating the authorization token and user_id of super admin.

Similarly for Trainer:

Getting Trainer Authorization Token and user_id

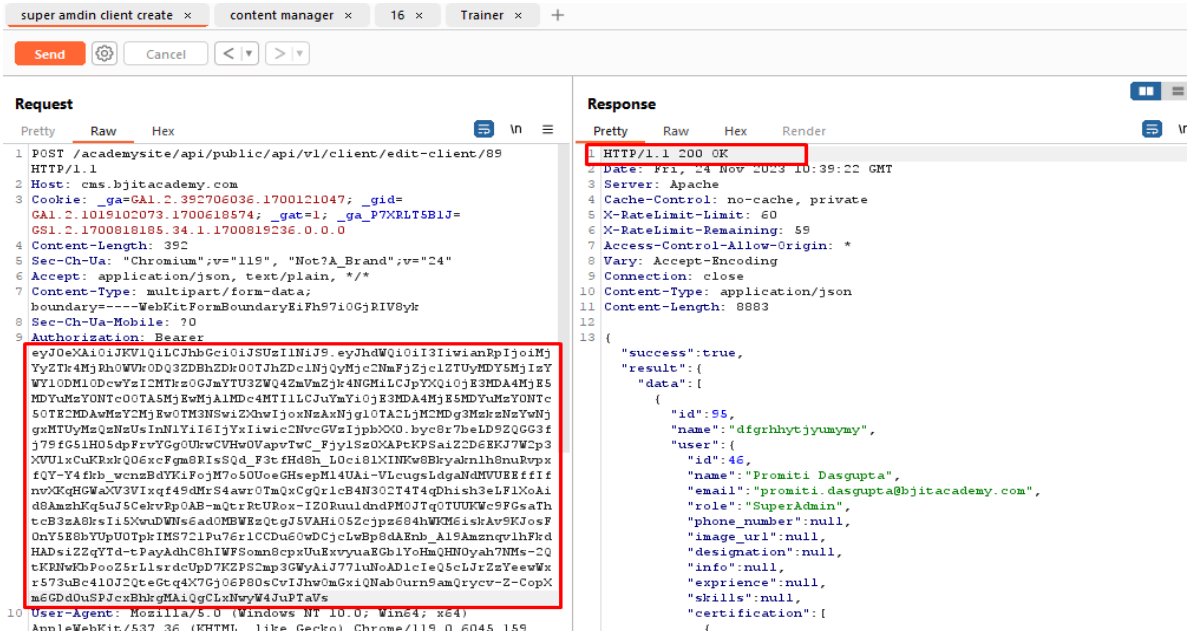
Login → go to profile of Trainer → update the page and capture the request to the burp repeater.



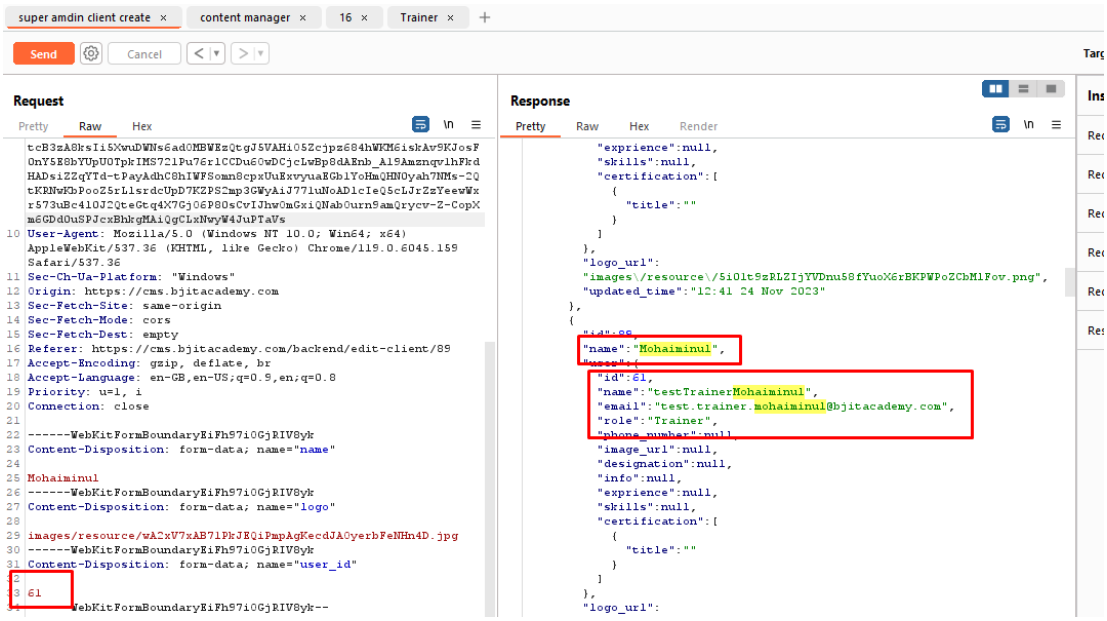
Here the id of trainer: 61

```
Content-Disposition: form-data; name="password"


------WebKitFormBoundarybIZwxncv0ytACDeh
Content-Disposition: form-data; name="new_password"


------WebKitFormBoundarybIZwxncv0ytACDeh
Content-Disposition: form-data; name="new_password_confirmation"


------WebKitFormBoundarybIZwxncv0ytACDeh
Content-Disposition: form-data; name="user_id"

61
------WebKitFormBoundarybIZwxncv0ytACDeh
Content-Disposition: form-data; name="certification"

[{"title":""}]
------WebKitFormBoundarybIZwxncv0ytACDeh
Content-Disposition: form-data; name="skills"
```

Now Replace the super admin's credential with trainer credential.



Here I got 200 OK for after replacing the credential of super admin with the trainer credentials.
Also the client information is successfully uploaded which is shown in the below picture



Expected Output: SEO Manager, Content Manager, Trainer can't update the client information.

Actual Output: SEO Manager, Content Manager ,Trainer can update the client information by manipulating the authorization token and user_id of a Super Admin.