Security Vulnerabilities

1. SQL Injection Vulnerability in Login Endpoint

Severity: Critical (CVSS Score: 9.8)
Location: /client_login endpoint
Description: Direct string concatenation is used to build SQL queries with user input:

Code

```
qMail = 'select privillage from users where email = "' + email +'" and
password = "' + password + '"'
```

Impact: Attackers can inject malicious SQL code to:

- Bypass authentication
- Extract sensitive data
- Modify database contents Recommendation: Use parameterized queries with bind variables.

2. SQL Injection in Registration Endpoint

Severity: Critical (CVSS Score: 9.8)
Location: /client_registeration endpoint
Description: Similar SQL injection vulnerability in email check:

Code

```
q = 'select userName from users where email = "' + email + '"'
```

Impact: Similar to above.

Recommendation:

- Use parameterized queries.

3. Weak JWT Implementation

Severity: High (CVSS Score: 8.6)
Location: generateJWT() function
Description: Multiple critical issues:

- Hardcoded secret key ('123456')
- Very weak secret key

- No token expiration Impact: Attackers can:
- Forge valid tokens
- Gain unauthorized access
- Tokens remain valid indefinitely

Recommendation:

- Use strong, environment-based secrets
- Implement token expiration
- Add proper key rotation mechanism

4. Plain Text Password Storage

Severity: High (CVSS Score: 7.5)
Location: /client_registeration endpoint
Description: Passwords are stored in plain text in the database
Impact: If database is compromised, all user passwords are exposed
Recommendation:

- Use strong password hashing (e.g., bcrypt, Argon2)

Logical Vulnerabilities

5. Weak Input Validation

Severity: Medium (CVSS Score: 6.5)
Location: /client_registeration endpoint
Description: Only checks if fields are non-empty:

Code

```
if fullName != '' and userName != '' and email != '' and password != '' and phone != ''
```

Impact:

- No validation of email format
- No password complexity requirements
- No phone number format validation

Recommendation:

- Implement proper validation for all fields

6. Insufficient Authentication Logic

Severity: Medium (CVSS Score: 6.0)
Location: /client_login endpoint
Description:Allows login with either email or username without proper validation
No rate limiting on login attempts Impact: Makes brute force attacks easier

Recommendation:

- Implement consistent authentication flow
- Add rate limiting
- Add account lockout after failed attempts

7. Missing Session Management

Severity: Medium (CVSS Score: 5.5)
Location: Both endpoints
Description: No proper session management or token invalidation mechanism
Impact:

- No way to invalidate compromised tokens
- No session timeout

Recommendation:

- Implement proper session management with token blacklisting

8. Privilege Level Assignment

Severity: Medium (CVSS Score: 5.0)
Location: /client_registeration endpoint
Description: Hardcoded privilege level (2) for all new users
Impact: No flexibility in user role assignment
Recommendation: Implement proper role management system

9. Error Handling

Severity: Low (CVSS Score: 3.5)
Location: Both endpoints
Description: Generic error messages that could leak information
Impact: Could help attackers in enumeration attacks
Recommendation:

- Implement consistent
- secure error handling