Security Assessment Report
Prepared For

Report Issued: March 6, 2022

1 – SQL Injection

HIGH RISK (8/10)	
Exploitation Likelihood	**Possible**
Business Impact	**Severe**
Remediation Difficulty	**Meduim**

Security Implications

SQL Injection is a code injection technique used to modify or retrieve data from SQL databases. By inserting specialized SQL statements into an entry field, an attacker is able to execute commands that allow for the retrieval of data from the database, the destruction of sensitive data, or other manipulative behaviors.

With the proper SQL command execution, the unauthorized user is able to spoof the identity of a more privileged user, make themselves or others database administrators, tamper with existing data, modify transactions and balances, and retrieve and/or destroy all server data.

Analysis

The username parameter appears to be vulnerable to SQL injection attacks. The payload '+(select*from(select(sleep(20)))a)+' was submitted in the username parameter. The application took 20191 milliseconds to respond to the request, compared with 915 milliseconds for the original request, indicating that the injected SQL command caused a time delay.

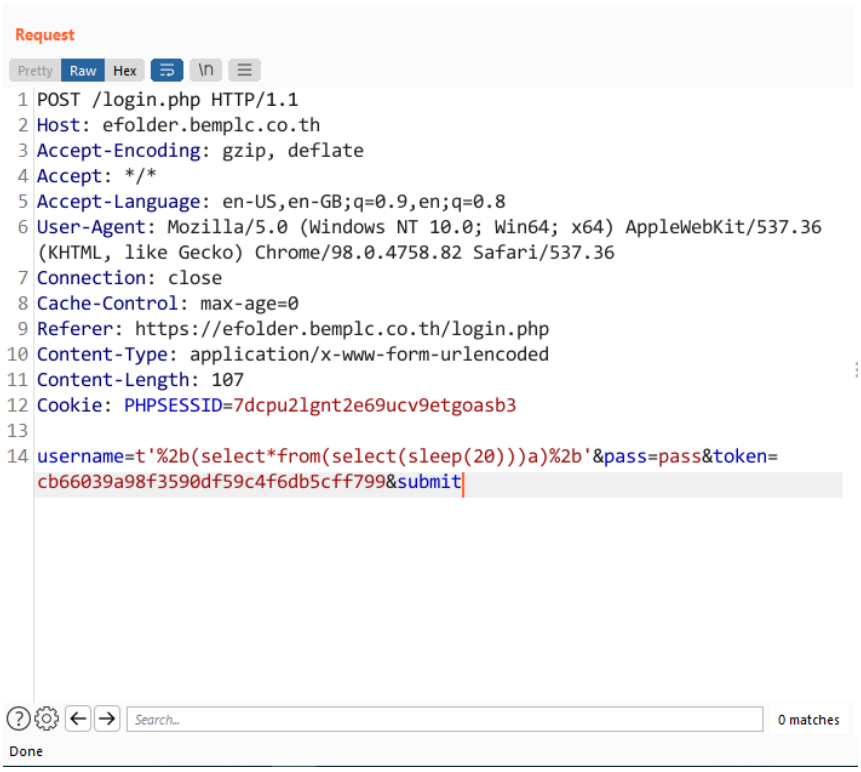

***Figure 1**: The HTTP post method inject time delay via username parameter*

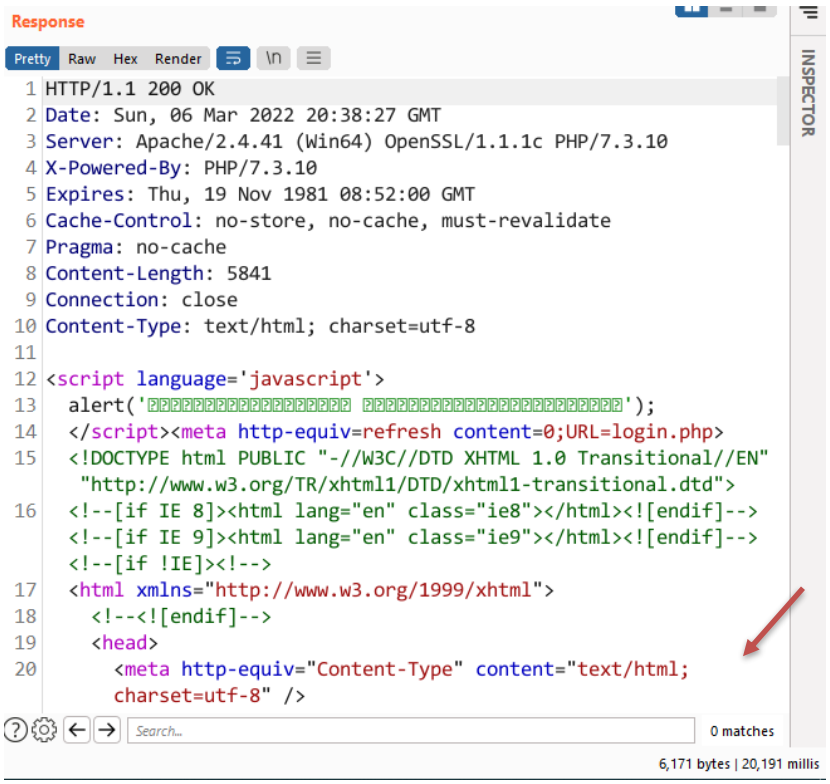

***Figure 2**: The HTTP response show time delay* 20191 milliseconds

Recommendations

- Check out MySQL queries username parameter in the login.php.
- Use DNS proxy such as Cloudflare DNS to prevent malicious of HTTP request. If you need it.
- Don’t use the Client side to reject input special character (with JavaScript code).

References (opt)

- https://owasp.org/www-community/attacks/SQL_Injection