Implementasi *AES* 256 Untuk Pencegahan *SQL Injection* di Parameter Alamat URL di tahap *Prepare Statement*

Oleh Sabilillah Fauzal Addim

ABSTRAK

Dengan pesatnya kemajuan akan teknologi web membuat sebagian kegiatan individu dilakukan di sana. Sehingga aplikasi web menjadi hal penting sebagai pendukung dalam aktivitas individu di internet dalam kesehariannya. Bahaya kebocoran data pada situs web akibat injeksi *SQL* sering ditemui kasusnya dan bahkan menurut organisasi OWASP yang merupakan komunitas yang membahas aplikasi web menetapkan bahwa ancaman injeksi *SQL* berada di urutan ke 3 dari 10 urutan teratasnya. Cara kerja dari ancaman ini akibat hasil *input* pengguna terhadap aplikasi ditampilkan secara jelas dan tidak validasi, ini berpotensi digunakan sebagai area injeksi untuk pernyataan *SQL* berbahaya yang berfungsi mengekstraksi data dari basis data. Sehingga ancaman dari serangan injeksi *SQL* memiliki dampak yang besar sebab penyerang dapat mendapatkan hak akses ke basis data situs web dan informasi pada situs web layaknya pengguna sah. Makalah ini menyajikan pendekatan dalam mengidentifikasi dan mengamankan nilai atau isi di parameter yang di*temp*atkan pada alamat *URL* yang menyorot pada 2 aspek: yang pertama membuat nilai atau isi parameter terjaga kerahasiaannya dengan enkripsi dan langkah kedua melakukan penyiapan pernyataan *SQL* untuk menjaga agar kueri *SQL* tidak menjadi kueri yang berbahaya serta memastikan nilai parameter yang diikat dalam kueri memiliki tipe data yang sesuai dan dimodifikasi dalam bentuk aman sehingga layak untuk dieksekusi.

Kata kunci : *SQL Injection Attack*, Keamanan parameter alamat *URL*, *AES-256*, *Prepare Statement.*

ABSTRACT

With the rapid advancement of web technology, some individual activities are carried out there. So that web applications become important as a support for individual activities on the internet in their daily lives. The danger of data leakage on websites due to *SQL injection* is often encountered cases and even according to the OWASP organization which is a community that discusses web applications determines that the threat of *SQL injection* is in 3rd place out of the top 10. The way this threat works is because the results of user *input* to the application are displayed in a clear and invalidated manner, this can potentially be used as an *injection* area for malicious *SQL state*ments that function to extract data from the *database*. Therefore, the threat of *SQL injection* attacks has a great impact because the attacker can gain access rights to the website *database* and information on the website like a legitimate user. This paper presents an approach to identify and secure the values or contents in the parameters placed in the URL address highlighting 2 aspects: the first is to make the parameter values or contents confidential by encryption and the second step is to *prepare SQL state*ments to keep the *SQL* query from becoming a malicious query and ensure that the parameter values bound in the query have appropriate data types and are modified in a safe form so that they are eligible for execution.

***Keywords*: *SQL Injection Attack, Security URL address parameters, AES-256, Prepare Statement*.**