

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Dari hasil perancangan sistem kemudian dilanjutkan dengan pengujian dan analisa, maka dapat disimpulkan sebagai berikut :

1. Pada tahapan pengujian disediakan *query* SQL- Injection yang telah diracik sebanyak 45 *query*. Pada form *login* biasa, didapat hasil 12 *query* sukses masuk ke dalam sistem. Akan tetapi setelah algoritma Rijndael di implementasikan pada form *login*, tidak ada satupun *query* yang berhasil masuk kedalam sistem.
2. Algoritma Rijndael dapat digunakan dalam memproteksi sistem dari serangan SQL Injection. Ketika suatu sistem telah diserang oleh SQL Injection maka penyerang dapat mengakses *database* sistem tersebut. *Database* merupakan tempat penyimpanan data yang harus dijaga dan dilindungi dari pihak yang tidak diberi hak akses.

#### **5.2 Saran**

Berikut beberapa saran dalam penelitian ini untuk diteliti lebih lanjut :

1. Algoritma Rijndael dapat digantikan/digabungkan dengan metode pengamanan lainnya untuk melindungi sistem *login* suatu *web*.
2. Penerapan Algoritma Rijndael pada penelitian ini melindungi *database* suatu sistem dari *SQL Injection*. Algoritma Rijndael juga dapat diimplementasikan untuk mencegah serangan *hacking* lainnya.