# BAB V KESIMPULAN DAN SARAN

## 5.1 Kesimpulan

RC6 melakukan proses enkripsi/dekripsi dengan kunci yang berbeda pada setiap putarannya, secara garis besar pembentukan kunci pada algoritma rc6 dilakukan dengan mengambil nilai b byte kunci yang dimasukan oleh user dimana nilai b ini terletak anatara 0 sampai 255. Kunci ini kemudian dimasukan kedalam array sepanjang c dimana tiap array diisi dengan satu byte dari kunci, dari kunci ini kemudian diambil sejumlah 2r + 4 kata dengan panjang 32 bit tiap kata dan disimpan dalam array S[0..2 r+ 3] atau S[0] hingga S[43]. Adapun ‘konstanta ajaib’ yang digunakan pada proses pembentukan kunci ini adalah P32 = B7E15163 yang didapat dari perluasan bilangan biner e-2 dimana e sebuah fungsi logaritma  dan Q32 =9E3779B9 yang didapat dari biner ∅-1, dimana ∅ dapat dikatakan sebagai “golden ratio” (rasio emas). Implementasi kunci pada perangkat lunak ini adalah merupakan hasil hash dari inputan admin, pada proses pembangkitan kunci ini tidak dilakukan proses padding, hal ini dikarenakan hasil hash dari inputan admin panjangnya sudah tetap yaitu 32 byte atau 32 karakter, sehingga tidak perlu lagi dilakukan proses panambahan padding.

Dalam proses enkripsi dekripsinya algoritma rc6 menggunkan empat buah register A, B, C dan D sepanjang 32 bit. Dimana untuk plainteks atau ciphertext pertama akan ditempatkan pada register A sedangkan plainteks atau ciphertext

terakhir akan ditempatkan pada register D. Proses enkripsi/dekripsi pada algoritma RC6 dimulai dan diakhiri dengan proses *whitening*. Pada proses whitening awal, nilai B dijumlahkan dengan S[0], dan nilai D dijumlahkan dengan S[i]. sedangkan untuk masing – masing iterasinya menggunakan 2 buah sub kunci. Sub kunci pada iterasi pertama menggunakan S[2] dan S[3], sedangkan iterasi-iterasi berikutnya menggunakan sub-sub kunci selanjutannya. Setelah iterasi ke-20 selesai, dilakukan proses whitening akhir dimana nilai A dijumlahkan dengan S[42], dan nilai C dijumlahkan dengan S[43]. Setiap iterasi pada algoritma RC6 mengikuti aturan sebagai berikut, nilai B dimasukan ke dalam fungsi f, yang didefinisikan sebagai f(x) =x(2x+1) , kemudian diputar kekiri sejauh lg-w atau 5 bit. Hasil yang didapat pada proses ini dimisalkan sebagai t dan u. Nilai t dan u kemudian di XOR dengan A dan C dan hasilnya menjadi nilai C dan A. Nilai t juga digunakan sebagai acuan bagi C untuk memutar nilainya kekiri. Begitu pula dengan nilai u, juga digunakan sebagai acuan bagi nilai A untuk melakukan proses pemutaran kekiri. Kemudian sub kunci S[2i] pada iterasi dijumlahkan dengan A, dan sub kunci S[2i+1] dijumlahkan dengan C. keempat bagian dari blok kemudian akan dipertukarkan dengan mengikuti aturan, bahwa nilai A ditempakan pada D, nilai B ditempatkan pada A, nilai C ditempatkan apda B, dan nilai (asli) D ditempatkan pad C. demikian iterasi tersebut akan terus berlangsung hingga 20 kali.

Untuk proses dekripsi dari algoritma RC6 merupakan pembalikan dari proses enkripsi. Untuk proses whitening, bila pada proses enkripsi menggunakan operasi

penjumlahan, maka pada proses dekripsi menggunakan operasi pengurangan. Sub kunci yang digunakan pada proses whiteneing setelah iterasi terakhir digunakan sebelum iteasi pertama, begitu juga sebaliknya sub kunci yang diterapkan pada proses whitening sebelum iterasi pertama digunakan pada whitening setelah iterasi terkahir. Akibatnya, untuk melakukan dekripsi, hal yang harus dilakukan semata-mata hanyalah menerapkan algoritma yang sama dengan enkripsi, dengan tiap iterasi menggunakan sub kunci yang sama dengan yang digunakan pada saat enkripsi, hanya saja urutan sub kunci yang digunakan terbalik.

Dalam algoritma enkripsi panjang kunci juga menentukan kekuatan dari enkripsi. Kunci yang lebih panjang biasanya lebih aman daripada kunci yang pendek. Pada implementasinya nilai kunci yang diambil pada proses pembentukan kunci ini adalah hasil hash dari inputan yang diberikan oleh admin yang panjangnya adalah 32 byte. Panjang kunci tersebut sudah cukup untuk mengamankan data yang disimpan di database. Selain itu dengan adanya konstanta P32 dan Q32 yang dikenal sebagai konstantan ajaib dapat mencegah agar tidak terjadi '*weak key*' yaitu terjadinya nilai sama pada dua entri dari s-box. Keamanan suatu pesan tidak tergantung pada sulitnya algoritma tetapi pada kunci yang digunakan. Pada RC6 dengan adanya fungsi f(x) = x(2x + 1) yang diikuti pergeseran lima bit kekiri dapat member tingkat keamanan data yang tinggi, sedangkan untuk jumlah iterasi yang berjumlah 20 round/iterasi memberikan asas kesimbangan pada algoritma RC6, karena jika round terlalu banyak

akan menyebabkan kecepatan proses enkripsi dekripsi menjadi lambat, namun jika jumlah roundnya sedikit dapat menyebabkan ciphertext mudah untuk dipecahkan.

## 5.2 Saran

Untuk penentuan jumlah rotasi atau round, berdasarkan security margin yang ditetapkan oleh pencipta RC6 adalah minimal 12 round atau rotasi, hal ini dilakukan karena serangan terbaik terhadap RC6 ditemukan ketika jumlah round atau rotasinya sama dengan 8. Pada proses enkripsi, agar panjang hasil enkripsi atau cipherteks sama dengan panjang plainteks maka sebaiknya digunakan algoritma kompresi.

Adapun algoritma RC6 yang telah diimplementasikan dalam sistem ujian online ini, merupakan salah satu metode yang dianggap cocok untuk melindungi data yang ditransmisikan dalam jaringan sehingga data tersebut tidak dapat dicuri atau dimodifikasi oleh pihak yang tidak bertanggung jawab. Semoga apa yang telah dipaparkan penulis dalam skripsi ini dapat menginspirasi berbagai pihak khususnya yang tertarik pada bidang kriptografi sehingga dapat melakukan penelitian lebih lanjut lagi.