BAB I

PENDAHULUAN

1.1 Latar Belakang

Aplikasi *web* merupakan suatu aplikasi dengan konsep *client server* yang dapat membentuk halaman-halaman *website* berdasarkan permintaan pengguna. Salah satu teknologi *open source* yang sangat banyak digunakan untuk membangun *website* adalah PHP. Hanya dalam beberapa tahun PHP telah cepat berkembang dari bahasa pemrograman kecil menjadi sebuah bahasa pengembangan web yang populer. Sekarang PHP sudah digunakan oleh jutaan *website* yang ada di seluruh dunia, dan saat ini PHP semakin stabil dan lebih dikembangkan (Rahmatulloh & Munir, 2015).

Sayangnya aplikasi *website* dengan teknologi *open source* berbasis PHP yang digunakan dalam pengembangan aplikasi website memiliki celah keamanan. Hal tersebut disebabkan karena aplikasi dengan teknologi PHP harus didistribusikan dalam bentuk *source code*, yang menyebabkan *source code* mudah diambil dan dimodifikasi. Hal ini juga akan membahayakan hak cipta pembuat *website* tersebut (Aprianto & Winarno, 2016).

Website yang dibangun dengan teknologi PHP dapat dilindungi dengan menggunakan teknik *obfuscation. Obfuscation* merupakan teknik yang digunakan untuk melindungi hak cipta properti intelektual yang berada dalam program dengan cara mengenkripsi dan mengacak *source code* asli dari program tersebut agar tidak dapat dipahami oleh manusia secara langsung sehingga dapat mencegah atau mempersulit proses *cracking* atau *reverse engineering* (*decompile* program).

Pada implementasi *obfuscation* salah satu algoritma kriptografi yang dapat digunakan sebagai metode *obfuscation source code* ini adalah algoritma *Rivest Cipher* 4 atau biasa disebut RC4. Algoritma RC4 merupakan algoritma dengan kunci simetris yang dibuat oleh *RSA Data Security Inc (RSADSI)*. Algoritma ini

bekerja dengan kunci enkripsi yang didapat dari 256 bit *state array* yang diinisialisasi dengan sebuah *key* tersendiri dengan panjang 1-256 bit. Setelah itu, *state array* yang didapatkan diacak kembali dan diproses untuk menghasilkan sebuah kunci enkripsi yang akan di-XOR dengan *plaintext* ataupun *ciphertext* sehingga didapatkan hasil dari enkripsi ataupun dekripsi (Sholeh dkk, 2019).

Pada penelitian yang dilakukan oleh Jumrin dkk (2016) yang berjudul "Aplikasi Sistem Keamanan Basis Data dengan Teknik Kriptografi RC4 *Stream Cipher*" dijelaskan bahwa hasil penelitian tersebut menunjukkan bahwa data pada tabel basis data dapat terenkripsi dengan menggunakan algoritma RC4 yang bersifat s*tream cipher* yakni didekripsikan secara *byte per byte* dan proses enkripsi dan dekripsi jauh lebih cepat karena menggunakan kunci yang sama serta memiliki tingkat keamanan yang tinggi sesuai panjang kunci.

Selanjutnya pada penelitian yang dilakukan oleh Rifai dkk (2016) yang berjudul "Implementasi Algoritma Kriptografi Rivest Code 4, Rivest Shamir Adleman, dan Metode Steganografi Untuk Pengamanan Pesan Rahasia Pada Berkas Teks Digital", dijelaskan bahwa hasil uji waktu enkripsi dan dekripsi dari algoritma RC4 lebih cepat dibandingkan dengan waktu enkripsi dan dekripsi algoritma RSA. Sehingga sesuai dasar teori bahwa kecepatan operasi dari algoritma simetris lebih tinggi bila dibandingkan dengan algoritma asimetris.

Pada Penelitian yang dilakukan oleh Watrianthos (2015) yang berjudul "Perbandingan Teknik Kriptografi Metode Sapphire II dan RC4" menyatakan bahwa, hasil uji coba secara umum didapat metode RC4 mempunyai waktu kumputasi yang cepat yang disebabkan oleh algoritma RC4 lebih sederhana dibandingkan dengan Sapphire II.

Oleh karena itu, dalam penelitian ini akan dirancang sebuah aplikasi yang dapat digunakan untuk melakukan *obfuscation* pada *source code* aplikasi *website* berbasis PHP dalam tugas akhir yang berjudul "**Implementasi Teknik *Obfuscation* pada *Source Code* PHP dengan Algoritma *Rivest Cipher* 4**".

1.2 Rumusan Masalah

Rumusan masalah dalam tugas akhir ini adalah sebagai berikut :

1. Bagaimana mengimplementasikan algoritma RC4 sebagai metode *obfuscation* pada *source code* PHP?
2. Bagaimana menghasilkan *source code* PHP terenkripsi namun masih dapat dieksekusi oleh *server*?
3. Apa kelebihan dan kelemahan dari penerapan algoritma RC4 sebagai metode *obfuscation* pada *source code* PHP?

1.3 Tujuan Penelitian

Adapun tujuan dalam penelitian ini adalah sebagai berikut :

1. Menerapkan teknik *obfuscation* pada *source code* PHP yang ditulis dengan struktur *procedural* maupun *object oriented*.
2. Membuat aplikasi yang dapat mengimplementasikan algoritma RC4 sebagai metode *obfuscation* pada *source code* PHP.
3. Menghasilkan *source code* PHP *obfuscated* yang telah teracak, namun masih dapat dieksekusi oleh *server*.
4. Mengetahui perbedaan kecepatan eksekusi dan ukuran antara file asli *source code* dan *source code obfuscated.*

1.4 Manfaat Penelitian

Manfaat penelitian dapat dirumuskan sebagai berikut :

1. Dapat menerapkan *obfuscation* pada *source code* PHP yang ditulis dengan struktur *procedural* maupun *object oriented.*
2. Dapat dikembangkan untuk mengamankan *source code* PHP sehingga dapat melindungi hak cipta dari aplikasi tersebut.
3. Sebagai referensi ilmiah penggunaan algoritma RC4 pada teknik *obfuscation source code* PHP.

1.5 Batasan Masalah

Batasan masalah pada tugas akhir ini adalah sebagai berikut :

1. Aplikasi dibangun dengan bahasa pemrograman PHP 7.
2. *Source code* PHP hasil enkripsi ditujukan untuk *web server* dengan versi PHP 5 keatas serta terhubung ke internet.
3. Hanya mengenkripsi *source code* PHP murni.

1.6 Sistematika Penulisan

Sistematika penulisan proposal Tugas Akhir ini adalah sebagai berikut:

BAB I PENDAHULUAN

Bab ini menjelaskan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, sistematika penulisan dan tinjauan pustaka.

BAB II LANDASAN TEORI

Bab ini memuat pengertian-pengertian dan teori-teori yang menjadi acuan dalam pembuatan analisa dan pemecahan dari permasalahan yang dibahas meliputi *Stream Cipher*, RC4, PHP, Database, MySQL, Code Igniter dan pendukung lain.

BAB III METODOLOGI PENELITIAN

Bab ini berisi metode penelitian yang digunakan. Langkah – langkah pengumpulan data, prosedur pengembangan perangkat lunak dan perangkat keras yang dilakukan dalam penelitian.

BAB IV ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi tentang gambaran umum sistem, desain perangkat keras dan perancangan sistem yang telah dibuat.

BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini membahas mengenai implementasi dan pengujian sistem yang telah dibuat.

BAB VI PENUTUP

Bab ini berisi kesimpulan yang diambil dari hasil pembuatan sistem serta saran-saran untuk pengembangan dari penulis.

1.7 Tinjauan Pustaka

Pada tahun 2016, Ali Latiful Aprianto melakukan penelitian dengan judul Rancang Bangun PHP 5 Encoder. Pada penelitian ini *source code* PHP dienkripsi dengan metode *Blowfish*. Hasil penelitian ini menunjukan bahwa *source code* yang telah teracak dapat dieksekusi oleh *server* yang telah dipasang *loader* dengan cepat karena telah dimodifikasi dengan penambahan sistem *cache* pada *loader*.

Penelitian yang dilakukan oleh Jumrin dkk (2016) dalam membuat sebuah Aplikasi Sistem Keamanan Basis Data dengan Teknik Kriptografi RC4 *Stream Cipher*, menyimpulkan bahwa Metode untuk mengamankan basis data dengan enkripsi *Stream Cipher* RC4 memiliki kelebihan dalam kecepatan pemrosesan dan tingkat keamanan yang cukup tinggi. Dengan penggunaan metode enkripsi *Stream Cipher* untuk menjaga keamanan basis data, informasi yang terdapat dalam basis data tersebut hanya dapat dilihat oleh orang yang memiliki kepentingan dengan informasi tersebut. Metode RC4 (*Rivest Code*) *Stream Cipher* merupakan salah satu algoritma kunci simetris berbentuk *stream chipper* yang memproses unit atau input data, pesan ataupun informasi. Hasil penelitian ini menunjukan bahwa data pada tabel basis data dapat terenkripsi atau *ciphertext*, serta proses enkripsi dan dekripsi yang jauh lebih cepat dan memiliki tingkat keamanan yang tinggi.

Ardiansyah Wardana Waluyo (2013) melakukan penelitian dengan judul Aplikasi Enkripsi Halaman *Web* Berbasis HTML dengan Menggunakan Metode *Shift Cipher* Dan Metode Javascript *Obfuscator*. Dari penelitian tersebut, aplikasi

atau sistem yang dibangun dapat berjalan dengan baik mengaplikasikan algoritma kriptografi *Shift Cipher* dan algoritma JavaScript *Ofuscator*. Berkas halaman web HTML dapat diamankan isinya. Algoritma kriptografi metode *Shift Cipher* dapat dikombinasikan dengan algoritma metode Obfuscator menghasilkan keamanan ganda untuk berkas HTML.