# BAB III

# ANALISIS SISTEM YANG BERJALAN

## Gambaran Umum Objek Penelitian

Banyak cara untuk mengamankan data, salah satunya dengan menggunakan algoritma MD5. MD5 merupakan salah satu algoritma enkripsi satu arah yang telah dimanfaatkan secara bermacam-macam pada aplikasi keamanan, dan MD5 juga umum digunakan untuk melakukan pengujian integritas sebuah [berkas](https://id.wikipedia.org/wiki/Berkas%22%20%5Co%20%22Berkas).

Gambar 3.1

Virus Scanner Jotti MD5

Sumber: https://virusscan.jotti.org/

Gambar 3.1 Menunjukan hasil scan virus dari file “Simple CSGO External v2.4.0.zip” dan menunjukan hasil MD5 “140604a5f76ae34fb024bed5d0b23fb6”.

Sumber: Aplikasi MD5Checker

Gambar 3.2

MD5 Checker

Gambar 3.2 Menunjukan hasil MD5Checker dari file “*Simple CSGO External v.2.4.0.zip*” dan hasil *Hash MD5* dari file “*Simple CSGO External v.2.4.0.zip*“ adalah “140604A5F76AE34FB024BED5D0B23FB6” Maka dari itu, MD5 digunakan sebagai alat pembanding validitas data yang masih digunakan hingga sekarang.

## Analisis Data

### Aplikasi Enkripsi dan Dekripsi data.

Telah dibahas pada Gambaran Umum Objek Penelitian Metodologi Penelitian algoritma MD5 sangat berguna selain sebagai salah satu cara untuk mengamankan data. Tetapi algoritma MD5 memiliki beberapa kekurangan seperti:

#### Hanya dapat menenkripsi data satu arah.

#### Dekripsi harus menggunakan *Dictionary* atau *Database MD5 Hash Result.*

#### Tidak memiliki kata sandi sebagai alat pengamanan.

Sedangkan *International Data Encryption Algoritm* memiliki kelebihan yang dapat menyaingi algoritma MD5 seperti menggunakan password dalam enkripsi file maupun teks. Dan menggunakan password yang sama untuk dapat mendekripsi file atau teks yang telah di enkrip. Hal ini membuat *International Data Encryption Algorithm* lebih aman untuk mengamankan data.

### Teknik Pengumpulan Data

#### Data yang dibutuhkan oleh penulis

##### Bagaimana cara kerja Algoritma *International Data Encryption Algorithm.*

##### Pengaruh format file yang digunakan terhadap hasil dari enkripsi *International Data Encryption Algorithm*.

##### Hasil Enkripsi dan Dekripsi Algoritma *International Data Encryption Algorithm*.

##### Hasil validitas data dengan *Hash MD5* yang telah di enkripsi dan di dekripsi menggunakan Aplikasi Algoritma *International Data Encryption Algorithm.*

#### Teknik pengumpulan data yang digunakan:

##### Metode Studi Pustaka

Metode pengumpulan data yang digunakan dalam penelitian ini adalah Metode Studi Kepustakaan dari buku dan jurnal yang berada di internet. Untuk melakukan studi kepustakaan, perpustakaan merupakan suatu tempat yang tepat guna memperoleh bahan-bahan dan informasi yang relevan untuk dikumpulkan, dibaca dan dikaji, dicatat dan dimanfaatkan. Dalam hal ini, penulis menggunakan buku yang telah didapat dari internet yang telah dipublikasikan serta jurnal-jurnal maupun dokumen pada suatu website.

##### Observasi

Teknik pengumpulan data yang dilakukan selanjutnya yaitu observasi tidak langsung dari hasil yang dikeluarkan dari program Enkripsi dan Dekripsi algoritma *International Data Encryption Algoritm* yang mana ketentuanya,data tersebut dapat di enkripsi atau dekripsi, Data tersebut apakah dapat dikembalikan seperti semula dengan *password* yang berbeda Apakah data yang telah di enkripsi dapat di edit oleh aplikasi lain atau teks editor, dan validasi data yang telah dilakukan enkripsi maupun dekripsi dengan menggunakan algoritma *Hash MD5*.

### Teknik Analisis Data

Penulis dalam menganalisia data yang digunakan dalam penelitian ini adalah Teknik analisis data deskriptif yang merupakan suatu cara dalam meneliti fenomena yang terjadi akibat perlakuan suatu hal dan menjabarkanya secara deskriptif. Disini penulis menerapkan algoritma *International Data Encryption Algorithm*. Dengan memperhatkan hasil output program, penulis akan menganalisais apakah data yang telah dienkripsi maupun dekripsi dapat terjaga keaslianya (validitas).

### Teknik Pengukuran Data

Teknik pengukuran data yang dilakukan oleh penulis yaitu dengan cara melakukan *experiment* terhadap file yang telah dilakukan enkripsi sebanyak tiga kali dengan menggunakan password enkripsi yang sama, beserta pengecekan *Hash MD5*. Lalu data yang telah di hasilkan di dekripsi sebanyak tiga kali dengan password yang sama pada saat melakukan enkripsi, beserta pengecekan *Hash MD5* pada file. Pengukuran data yang dinilai yaitu apakah data yang telah di enkripsi sebanyak tiga kali maupun dekripsi tiga kali mengalami perubahan setiap enkripsi, hubungan tahapan enkripsi dan dekripsi (enkripsi pertama apakah sama dengan hasil dekripsi yang ketiga), dan data enkripsi maupun dekripsi dibandingkan apakah dapat kembali seperti data asli sebelum di enkripsi atau tidak dapat kembali.

Table 2.1

Perbandingan Algoritma

|  |  |  |  |
| --- | --- | --- | --- |
| Pembanding | MD5 | DES | IDEA |
| Jenis Enkripsi | *One-Way Hash* | Kunci Simetris | Kunci Simetris |
| Panjang hasil Dekripsi | 512-*Bit* | 64-*bit* | 64-*bit*  |
| Menggunakan *private key*? | Tidak | Ya | Ya |
| Kelebihan | Digunakan sebagai Validasi Data/File | - | Algoritma Kunci Simetris pengembangan algoritma dari PES |
| Kekurangan | Tidak Dapat Melakukan Dekripsi  | Telah ditinggalkan karena ditemukan kekurangan dan kelemahan algoritma | Belum ditemukan Kekurangan  |
| Panjang Kunci | - | 56-*bit* | 128-*bit* |
| Operasi Aritmatika yang digunakan | *OR, XOR, NOR, AND* | *XOR* | Pertambahan, Perkalian, dan *XOR* |

Tabel 2.6 menunjukan tabel perbandingan antara Algoritma MD5, DES, dan IDEA yang didapat dari teori yang diusung penulis. Dapat dilihat bahwa algoritma IDEA memiliki kelebihan dan dapat menggantikan algoritma DES dikarenakan algoritma DES memiliki kelemahan algoritma.

## C:\Users\Leonardo\Desktop\e-gpg2-01.gifMetode International Data Encryption Algorithm

Gambar 3.3

Diagram Kriptografi IDEA

Keterangan simbol pada gambar 3.3:

### Exclusive OR(XOR) ATAU (dilambangkan dengan lingkaran plus ⊕).

### Penambahan modulo 216 (dilambangkan dengan kotak plus ⊞).

### Modul perkalian 216 + 1, di mana kata semua-nol (0 × 0000) dalam *input* ditafsirkan sebagai 216 dan 216 dalam *output* ditafsirkan sebagai kata nol (0 × 0000) (dilambangkan dengan titik lingkaran ⊙).

### PK1 sampai dengan PK52 merupakan *Sub* Kunci pertama sampai dengan Sub Kunci ke lima puluh dua

### Txt1 sampai dengan Txt 4 merupakan *Sub* *Plaintext* Pertama hingga ke empat

Dengan menggunakan algoritma IDEA, penulis akan melakukan enkripsi dengan 9 tahapan dengan 8 tahapan perulangan yang dijabarkan sebagai berikut

1. Tahapan Pertama:

#### Gandakan *sub plaintext* pertama dan *sub* kunci pertama.

#### Tambahkan *sub plaintext* kedua dan *sub* kunci kedua.

#### Tambahkan *sub paintext* ketiga dan *sub* kunci ketiga

#### Gandakan *sub plaintext* keempat dan *sub* kunci keempat

1. Tahapan Kedua:

#### Lakukan *eXclusive OR* hasil langkah (a) dan (c) dari tahapan pertama.

#### Lakukan *eXclusive OR* hasil langkah (b) dan (d) dari tahapan pertama.

1. Tahapan Ketiga:

#### Gandakan hasil langkah (a) dari tahapan kedua dengan *sub kunci* kelima.

#### Tambahkan hasil langkah (b) dari tahapan Kedua dan (a) dari Tahapan ketiga.

#### Gandakan hasil langkah (b) dari tahapan ketiga dengan sub kunci keenam.

#### Tambahkan hasil langkah (a) dari tahapan Ketiga dan (c) dari tahapan ketiga.

1. Tahapan Keempat:

#### Lakukan *eXclusive OR* hasil dari langkah (a) tahapan pertama dan (c) dari tahapan ketiga.

#### Lakukan *eXclusive OR* hasil dari langkah (c) tahapan pertama dan (c) dari tahapan ketiga.

#### Lakukan eXclusive OR hasil dari langkah (b) tahapan pertama dan (d) dari tahapan ketiga.

#### Lakukan *eXclusive OR* hasil dari langkah (d) tahapan pertama dan (d) dari tahapan ketiga.

Setelah dilakukan delapan putaran, selanjutnya dilakukan setengah putaran. Hasil dari setengah putaran merupakan output. Dilakukan dengan cara sebagai berikut (asumsi *sub plaintext* pertama, *sub plaintext* kedua, *sub plaintext* ketiga, *sub plaintext* keempat merupakan hasil dari putaran kedelapan dan *sub* kunci pertama, *sub* kunci kedua, *sub* kunci ketiga, *sub* kunci keempat merupakan sub kunci yang terakhir):

1. Gandakan *sub plaintext* pertama dan *sub* kunci pertama.
2. Tambahkan *sub plaintext* kedua dan *sub* kunci kedua.
3. Tambahkan *sub paintext* ketiga dan *sub* kunci ketiga
4. Gandakan *sub plaintext* keempat dan *sub* kunci keempat