

## IMPLEMENTASI PENGAMANAN EMAIL MENGGUNAKAN METODE BLOWFISH BERBASIS WEB PADA UPT PUSKESMAS PONDOK KACANG TIMUR

Erick Erwin Nylis<sup>1)</sup>, Purwanto<sup>2)</sup>

Program studi, Fakultas Teknologi Informasi, Universitas Budi Luhur

Jl. Raya Ciledug, Petukangan Utara, Jakarta Selatan 12260

E-mail : 1511511030@student.budiluhur.ac.id<sup>1)</sup>, purwanto@ budiluhur.ac.id<sup>2)</sup>

### Abstrak

Seiring perkembangan pola pikir manusia saat ini, keamanan data menjadi sangat penting untuk menghindari pencurian informasi secara rahasia oleh pihak-pihak yang tidak bertanggung jawab. Berbagai cara dilakukan untuk menjaga keamanan data, Pembuatan e-mail enkripsi pada Unit Pelayanan Teknis Puskesmas ini yang ingin penulis lakukan, dikarenakan Unit Pelayanan Teknis merupakan pelayanan umum yang melayani masyarakat di dalam bidang kesehatan Puskesmas selalu mendapati data-data penting yang perlu dijaga kerahasiaannya data informasi data informasi pegawai data informasi penting Puskesmas yang lainnya. Dengan membuat suatu penelitian dari sebuah media komunikasi agar bertujuan informasi tetap menjadi aman. Salah satu cara yang digunakan yaitu dengan mengenkrip e-mail melalui aplikasi kriptografi dalam menangani masalah akan bocornya informasi yang akan dikirim, maka diperlukan aplikasi pengamanan data yang dapat mengamankan e-mail yang dimiliki dengan menerapkan algoritma kriptografi blowfish, aplikasi ini dibuat dengan bahasa pemrograman PHP berbasis web. Penulis membuat rancangan dengan menggunakan editor dengan beberapa form agar memudahkan dari pengguna aplikasi ini. Email pengguna dari aplikasi ini harus terdaftar pada informasi pada Unit Teknis Pelayanan Puskesmas hasil dari output enkripsi dan mengembalikannya kembali dekripsi, proses enkripsi dan dekripsi dapat mengurangi kebocoran informasi atau pencurian data, serta membantu dari komunikasi dalam menggunakan media surat elektronik yaitu e-mail

Kata kunci : Blowfish, Enkripsi, Dekripsi, E-mail

### 1. PENDAHULUAN

Bagian administrasi yang bergerak dibidang pelayanan seperti (UPT) Unit Pelayanan Teknis Puskesmas informasi yang diterima atau dikirimkan, yaitu melalui media komunikasi surat elektronik e-mail merupakan salah satu media yang serign digunakan dalam pelayanan, maka dibutuhkan metode dapat menjaga rahasia informasi tersebut.

Metode dimaksud adalah kriptografi merupakan ilmu penyandian yang pada mula dipakai pada zaman perang dunia ke dua dengan tujuan menjaga agar komunikasi tetap aman dan terjaga kerahasiaannya, berbeda media yang dipakai zaman perang ke dua zaman dahulu dengan media komunikasi zaman sekarang, berdasarkan kenyataan diatas, perlu ada suatu sistem pengamanan informasi baik saat pengiriman maupun penerima e-mail.

Untuk melakukan yang disebut penyandian data, penulis mencoba implementasikan cabang ilmu matematika dengan kriptografi yaitu komunikasi text menjadi sandi-sandi yang kurang dipahami oleh sembarang orang serta mengembalikannya ke bentuk semula, proses ini disebut enkripsi dekripsi, algoritma enkripsi banyak dan beragam metode.

Dalam laporan penelitian ini akan menggunakan metode blowfish untuk enkripsi dan dekripsi text pada e-mail.

#### 1.1 Batasan Masalah

Pada batasan masalah merupakan bagian utama dari penyusunan masalah, sehingga dalam menyelesaikannya secara sistematis agar tetap dalam pembahasan dari materi berikut hal-hal menjadi perhatian bagi penulis:

- Metode algoritma kriptografi yang digunakan adalah *blowfish*.
- Data yang di enkripsi dan dekripsi adalah data berbentuk *text*.
- Bahasa pemrograman yang digunakan adalah PHP.
- Aplikasi yang dibuat berbasis *web*, dijalankan dengan menggunakan *browser*.

#### 1.2 Tujuan Penulisan

Tujuan dari penulisan laporan tugas akhir ini adalah:

- Mengembangkan suatu aplikasi keamanan yang dapat mengamankan data dan informasi yang

berbentuk teks yang dikirim atau diterima melalui media email.

- b. Menggunakan metode Blowfish untuk mengenkripsi dan dekripsi data dan informasi berbentuk teks.

## 2. METODE PENELITIAN

Dalam hal metode penelitian, penulis melakukan beberapa tahapan dalam proses penelitian dan penulisan pada tugas akhir ini.

### (1) Peninjauan Tempat Riset

Pada tahapan ini penulis melakukan kunjungan langsung ketempat Riset, bertempat di jalan Gelora 1 No 1 A Pondok Kacang Timur Kota Tangerang Selatan.

### (2) Analisis Permasalahan

Puskesmas Pondok Kacang Timur adalah Unit Pelayanan Teknis dibawah naungan Dinas Kesehatan Kota Tangerang Selatan, pelayanan kesehatan khusus nya untuk warga atau masyarakat sekitar wilayah Pondok Kacang Timur. Dalam hal ini penulis dengan melakukan peninjauan langsung ke UPT Puskesmas Pondok Kacang Timur mendapati permasalahan pada bagian administrasi.

Permasalahan komunikasi yang digunakan melalui media internet, kesulitan dalam hal untuk menyampaikan pesan secara rahasia terhadap sesama UPT Puskesmas wilayah sekitar Pondok Kacang Timur. Maka penulis memberikan solusi dengan membuat implementasi berbasis Web untuk membantu komunikasi dalam bentuk pesan text, diharapkan dapat membantu dalam hal menjaga informasi agar aman dan cepat disampaikan.

### (3) Analisa Pembuatan

Penulis merancang sebuah tampilan aplikasi berbasis web dengan penerapan *user* dan *administrator* agar mempermudah dalam penggunaan.

### (4) Rancangan Sistem

Pembuatan bentuk dan tampilan dengan menggunakan *PHP personal home page 5.6* relasi menggunakan *database* dari *xampp* editor penulis memakai *sublime text3* alat server pengirimian *SMTP elastic email* dan *browser* dari *google chrome*.

## 3. HASIL DAN PEMBAHASAN

### 3.1 Penyelesaian Masalah

Pembahasan penyelesaian masalah data dari sebuah email text sehingga isi dari data tersebut tidak bisa dibaca atau tidak bisa diketahui oleh pihak lain yang tidak berhak atas email text tersebut. Aplikasi tersebut nantinya akan mengubah sebuah email text asli menjadi email text yang isinya tidak bisa dibaca

dan tidak bisa diketahui, agar isi dari data tersebut terjaga kerahasiaannya.

Kemudian mengembalikan email text tersebut menjadi seperti semula tanpa mengalami perubahan sedikitpun. Pada aplikasi ini, penulis menggunakan algoritma Blowfish sebagai metode kriptografi. Algoritma Blowfish merupakan algoritma *block cipher* yang termasuk kedalam jenis sistem kriptografi kunci simetris. Agar email text bisa dienkripsi menjadi email *text ciphertext* dan mengembalikannya menjadi email *text* semula, *email text* akan terenkripsi pada saat akan melakukan proses enkripsi dan akan menjadi email *text plaintext* saat setelah melakukan dekripsi. Untuk membangun aplikasi yang terkomputerisasi ini, saya selaku penulis menggunakan bahasa pemrograman *PHP* dengan *sublime text 3* sebagai pendukungnya

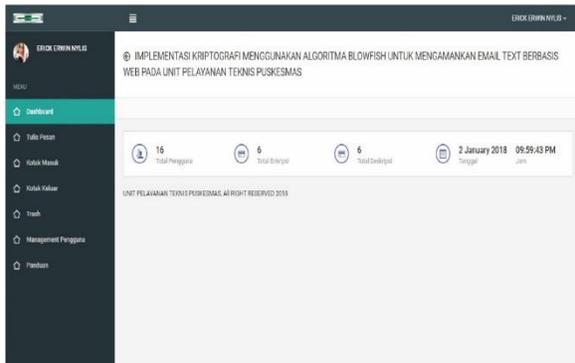
Bab ini membahas implementasi dan evaluasi sistem pengamanan email text algoritma Blowfish, pada bagian ini, diuraikan mengenai tampilan layar aplikasi enkripsi dan dekripsi email *text* mulai dari pertama kali aplikasi ini dijalankan sampai selesai dijalankan.

Gambar 1: Tampilan Layar Form Log In

Tampilan layar menu ini merupakan tampilan *form* awal muncul dua kotak dan tombol *log in* untuk melakukan proses masuk ke dalam aplikasi dijalankan atau diakses. Pada *form log in* terdapat *email* dan *password* yang terdaftar agar dapat masuk menggunakan aplikasi ini.

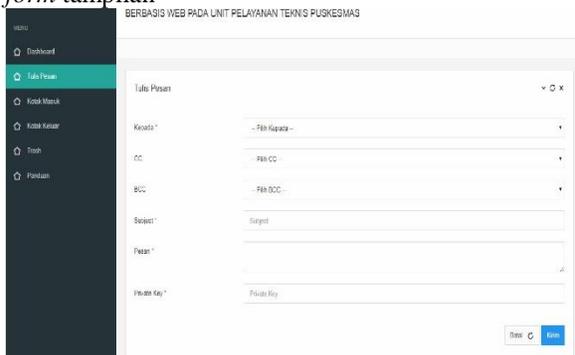
Gambar 2: Tampilan Layar Form salah Log In

Tampilan Layar dari menu *log in* pada gambar 2 ini merupakan *password* atau salah *email*. Terdapat peringatan *email* dan *password* salah mohon dicek kembali. *User* harus memasukkan *email* dan *password* yang terdaftar agar dapat masuk dan menggunakan Aplikasi ini.



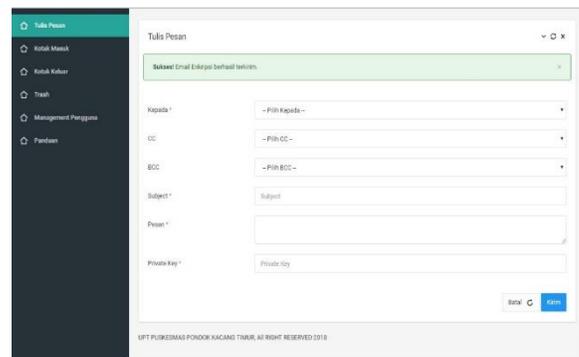
Gambar3: Tampilan Layar Menu Dashboard

Tampilan layar dari *menu* utama dari gambar 3. *form* ini akan muncul pada saat user telah berhasil melakukan *login*. Didalam tampilan *dashboard* terdapat *submenu* yaitu *my profil* dan *logout*, dan *menu form* tampilan



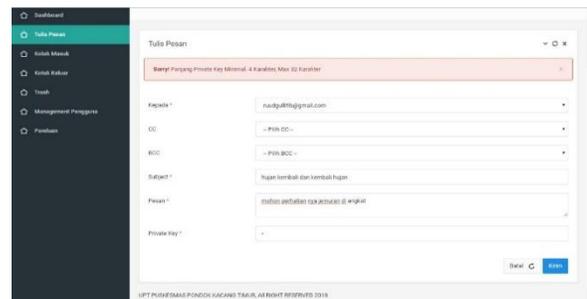
Gambar 4: Tampilan Layar Form Tulis Pesan

*Form* ini untuk melakukan proses enkripsi *email text*, *user* terlebih dahulu memilih *menu* tulis pesan, setelah itu isi *form* kata kunci, isi *subject*, dan isi kepada. Jika *user* pengirim ingin menambahkan tujuan alamat penerima diharuskan untuk memasukkan alamat *email user* penerima baru yang terdaftar pada *cc*, dan *bcc* setelah itu *user* bisa menuliskan *email text* pada *form* pesan. Setelah itu *user* pengirim bisa langsung mengirimkan *email text* yang terenkripsi pada *user* penerima.



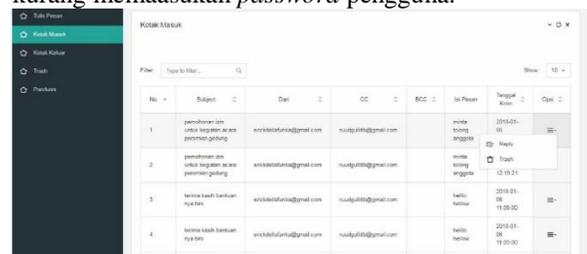
Gambar 5: Tampilan Layar Form Tulis Pesan Berhasil Terkirim

Sesudah melakukan proses enkripsi *email text*, kotak kepada, *cc*, *bcc*, *subject*, pesan, *private key* kembali kosong dan muncul pesan bahwa pesan sudah terkirim sukses *email* enkripsi berhasil terkirim.



Gambar 6: Tampilan Layar Form Tulis Pesan Tidak Terkirim

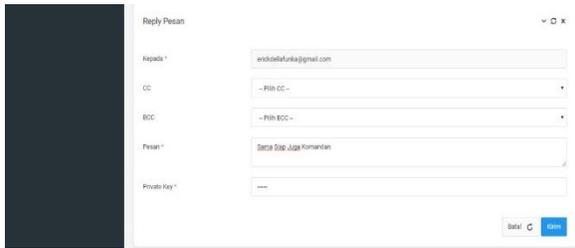
Tampilan pesan gagal terkirim dikarenakan salah atau kurang memasukkan *password* pengguna.



Gambar 7: Tampilan Layar Form Kotak Masuk

Tampilan kotak masuk di fungsikan sebagai tempat penyimpanan *email user* penerima yang sudah terenkripsi dari *user* lain atau pengirim. Selain berfungsi sebagai tempat penyimpanan *email* masuk yang sudah terenkripsi, *form* ini juga dapat difungsikan sebagai tempat dekripsi *email text* yang masuk pada *user* penerima. Untuk mendekripsi *email text* yang masuk *user* penerima dapat memilih perintah *decrypt* pada tabel opsi maka *emailtext* yang terenkripsi tadi sudah terdekrips, dan untuk melihat

langsung pada *email* utama yang didaftarkan pada Aplikasi ini.



Gambar 8: Tampilan Layar Form Reply

*Form* ini difungsikan sebagai *form* balas pesan *text* yang masih terenkripsi, didalam *form reply* ini pengguna harus mendekrip *text* terlebih dahulu agar pengguna mengetahui isi pesan *text* enkrip yang diterima pengguna, pengguna yang menerima bisa membalas pesa *text* dari *form* pengguna penerima yang telah di dekrip.



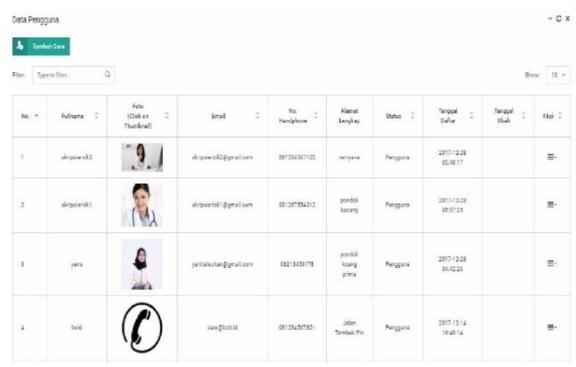
Gambar 9: Tampilan Layar Form Reply Berhasil Kirim

Tampilan *layar form reply* berhasil pengirim tidak bisa melihat tampilan isi tampilan *text* dari *form reply* akan tetapi pengirim dapat melihat pesan *text* di *menu form trash*.



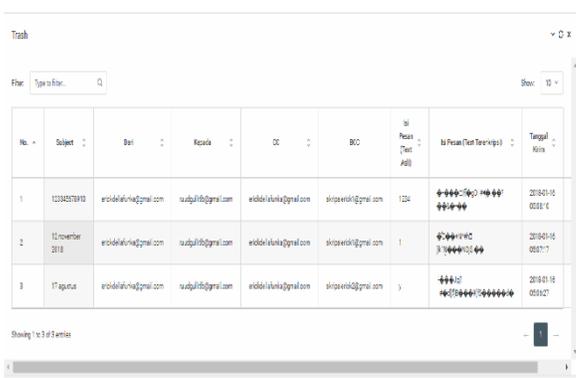
Gambar 10: Tampilan Layar Form Kotak Keluar

*Form* ini sebagai tempat penyimpanan riwayat keluarnya *email user* pengirim yang sudah terenkripsi oleh Aplikasi ini. Selain sebagai tempat penyimpanan riwayat *email* keluar *user* pengirim. Terdapat opsi yang terdiri dari pilihan hapus, *reply* fungsi dari pilihan hapus sendiri membuang riwayat *email* kedalam *form trash*, sedangkan pilihan *reply* untuk membalas pesan.



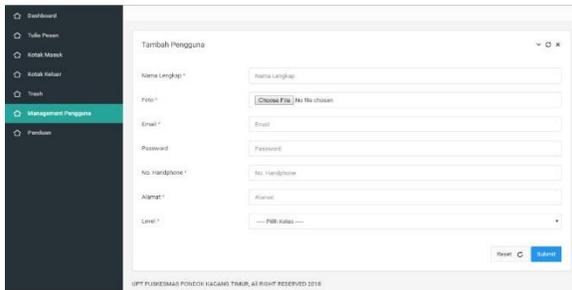
Gambar 11: Tampilan Layar Form Management Pengguna

Pada *management* pengguna terdapat tabel pengguna yang telah daftar data pengguna yang telah memakai Aplikasi ini. Pada *management* pengguna *user* yang telah diberikan hak *administrator* dapat menambah, mengubah, atau menghapus pengguna dari Aplikasi ini. Terdapat *button* tambah *data* dapat menambahkan, jika *user* ingin menambahkan pengguna akan muncul *form* penambah pengguna, jika ingin mengubah dan melihat data pengguna, *user* dapat mengklik tombol *button* tabel aksi dan memilih, lihat, atau ubah.



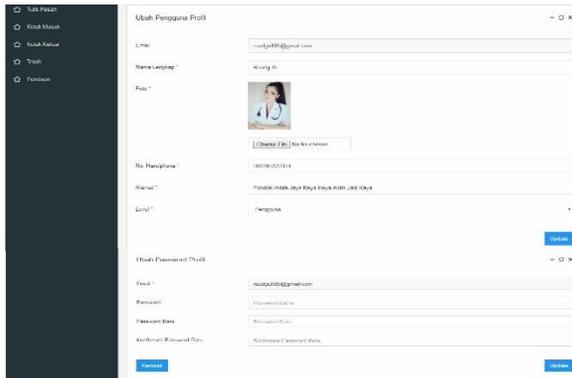
Gambar 12: Tampilan Layar Form Trash

Menampilkan riwayat hasil penghapusan *email text* yang berasal dari kotak masuk, dan kotak keluar dalam bentuk tabel. Pada *table* tersebut terdapat identitas *table-table* diantaranya nomor, *subject email*, tujuan *email*. Didalam *menu* ini tidak terdapat *restore* atau pengembalian data yang telah dihapus.



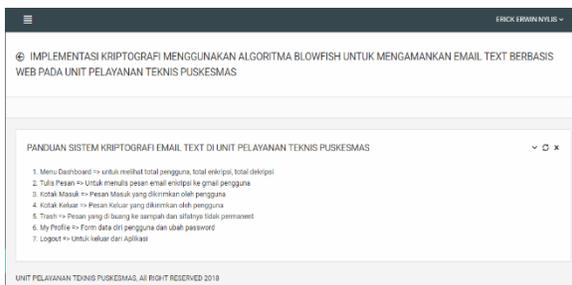
Gambar 13: Tampilan Layar Form Tambah Pengguna

Tambah pengguna yang terdapat pada *form management* pengguna, pada *form* ini *admin* harus mengisi data pengguna, *password* pengguna, *no handphone* pengguna, *level* didalam *form* ini ada 2 pilihan terdiri *administrator*, pengguna.



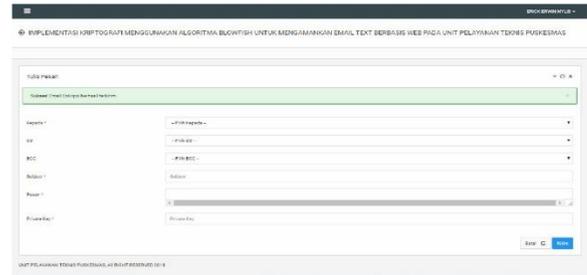
Gambar 14: Tampilan Layar Form Edit User

Berfungsi mengubah data *profil* pengguna dan ubah *password profil* pengguna. Ubah *profil* harus mengisi setiap kotak isi *form* terdiri dari *email* pengguna, nama lengkap (dapat diisi dan merubah nama), foto pengguna, *no handphone*, alamat, Level dijadikan sebagai *admin* atau pengguna.



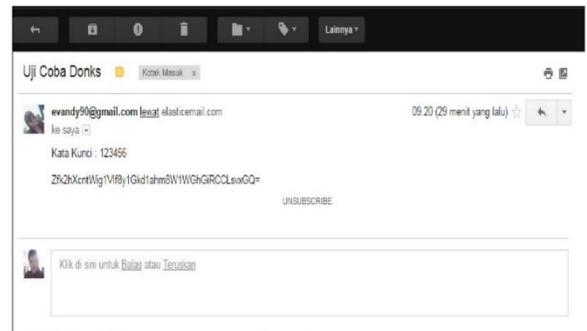
Gambar 15: Tampilan Layar Form Panduan

*Form* ini berfungsi hanya uncutuk memberikan bantuan dalam pengarahannya, bagaimana cara menggunakan Aplikasi ini, agar *user* mudah dalam pengoperasian nya.



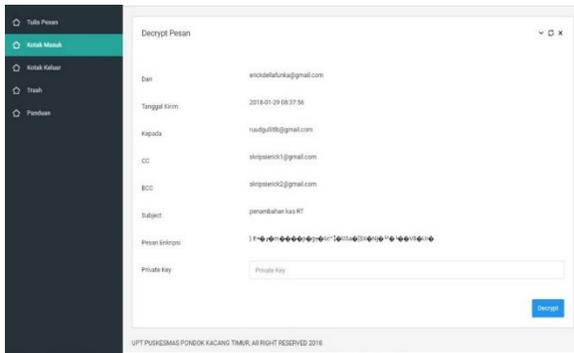
Gambar 16: Tampilan Layar Kotak Masuk Dekripsi Berhasil

Untuk melakukan enkrip *email text*, *user* pengirim terlebih dahulu memilih *menu* tulis pesan, kemudian mengisi beberapa *form* wajib, seperti *form private key* yang wajib untuk tidak dikosongkan, Selanjutnya *user* wajib mengisi kotak kepada, jika *user* ingin menambahkan *cc*, *bcc* dapat menambahkan pada *cc*, *bcc*. kemudian menambahkan *user* mengisi kotak kosong pesan berfungsi meyisipkan *text* batas minimal satu paragraf atau 900 karakter, selanjutnya ada *button* kirim dan *button* batal fungsi dari *button* kirim untuk mengirim isi *text* dan dilanjutkan dengan proses enkripsi, dikirimkan isi pesan *text* ke *email* penerima. *Button* batal berfungsi untuk mengosongkan seluruh isi dari *form* menu isi pesan, jika berhasil akan muncul notifikasi pada Tampilan Layar Kotak Masuk Dekripsi Berhasil.



Gambar 17: Tampilan Layar Kotak Masuk Email

Setelah muncul notifikasi *email text* berhasil terenkripsi oleh Aplikasi ini, maka *email text* yang bersisi pesan dalam ciphertext akan diterima oleh akun *email user* penerima yang terdaftar pada Aplikasi ini.



Gambar 18: Tampilan Layar Proses Dekripsi Email Text Dari Kotak Masuk

Jika ingin mengembalikan *email text* enkrip menjadi *email text user* terlebih dahulu ke kotak masuk pada *form* kotak masuk dan memilih pada pilihan opsi dekrip untuk pilihan dekrip akan muncul *form* dekrip pesan untuk mendekripsi *email text* berbentuk  *ciphertext*.

Tabel 1: Hasil Enkripsi Email Text

| Subject         | Key   | Ukuran Size Byte | (Milidetik) | Isi Pesan Text Enkripsi   | Ukuran Size Byte |
|-----------------|-------|------------------|-------------|---------------------------|------------------|
| Percobaan Tes 1 | 12345 | 23               | 11.000      | 1 h c b - ; i qzIJ        | 19               |
| Percobaan Tes 2 | 12345 | 28               | 11.000      | ] I D ' F                 | 24               |
| Percobaan Tes 3 | 12345 | 22               | 9.000       | 9 1 S p V                 | 19               |
| Percobaan Tes 4 | 12345 | 24               | 10.000      | 1 h _ k t X - # X h k     | 20               |
| Percobaan tes 5 | 12345 | 23               | 11.000      | # r. J b M [ x r L - e \$ | 19               |

| Subject          | Key   | Ukuran Size Byte | (Milidetik) | Isi Pesan Text Enkripsi | Ukuran Size Byte |
|------------------|-------|------------------|-------------|-------------------------|------------------|
| Percobaan Tes 6  | 12345 | 25               | 9.000       | K F k 2 H J n p V A     | 20               |
| Percobaan Tes 7  | 12345 | 25               | 9.000       | 1 h . ? } 7 B L y u 4 & | 20               |
| Percobaan Tes 8  | 12345 | 23               | 10.000      | 1 h . _ k t X - # W _ Z | 19               |
| Percobaan Tes 9  | 12345 | 24               | 10.000      | 1 h . { 0 p - i         | 20               |
| Percobaan Tes 10 | 12345 | 30               | 13.000      | + [ 8 P } . \$          | 25               |
|                  |       | Jumlah           | 230         | 92.000                  | 198              |
|                  |       | Rata - Rata      | 23          | 9,200                   | 19.8             |

Tabel 2: Hasil Dekripsi Email Text

| Subject          | Key   | Ukuran Size Byte | Waktu Dekripsi (Mili detik) | Email Text Dekripsi         | Ukuran Size Byte |
|------------------|-------|------------------|-----------------------------|-----------------------------|------------------|
| Percobaan Test 1 | 12345 | 19               | 1.8600                      | percobaan test 1            | 23               |
| Percobaan Tes 2  | 12345 | 24               | 1.9000                      | percobaan lagi ulang test 2 | 28               |
| Percobaan Tes 3  | 12345 | 19               | 1.3800                      | Kembali Perocobaan Test 3   | 22               |
| Percobaan Tes 4  | 12345 | 20               | 1.4500                      | percobaan tes kembali untuk | 24               |

| Subject          | Key       | Ukuran Size Byte | Waktu Dekripsi(Mili detik) | Email Text Dekripsi                                 | Ukuran Size Byte |
|------------------|-----------|------------------|----------------------------|---|------------------|
|                  |           |                  |                            | keempat kalinya                                     |                  |
| Percobaan Tes 5  | 12345     | 19               | 1.31000                    | coba coba lagi                                      | 23               |
| Percobaan Tes 6  | 12345     | 20               | 1.32000                    | selamat malam percobaan tes ke 6                    | 25               |
| Percobaan Tes 7  | 12345     | 20               | 1.26000                    | percobaan tes untuk urutan ke tujuh                 | 25               |
| Percobaan Tes 8  | 12345     | 19               | 1.31000                    | percobaan tes ke urutan ke delapan                  | 23               |
| Percobaan Tes 9  | 12345     | 20               | 1.21000                    | percobaan tes urutan sembilan mohon untuk di terima | 24               |
| Percobaan Tes 10 | 12345     | 22               | 9300                       | Percobaan tahap akhir urutan sepuluh                | 26               |
|                  | Jumlah    | 198              | 1.017.300                  |   | 230              |
|                  | Rata-Rata | 19.8             | 101.730                    |   | 23               |

#### 4. KESIMPULAN

Sebuah Aplikasi yang mengimplementasikan algoritma kriptografi Blowfish untuk enkripsi email text telah berhasil diciptakan, hasil dari 10 uji coba text

enkripsi rata-rata size nya 20.5 byte dan hasil dari dekrip text rata-rata size nya 22.2 byte dengan selisih dari rata-rata hasil yang di ujikan 6 byte.

#### 5. DAFTAR PUSTAKA

- [1] Blom h.T dan Lengstorf J, 2014, PHP for Absolute Beginners, Apress Media.
- [2] Firmansyah, Ade 2015. Penerapan Algoritma DES (Data Encryption Standart) dan Blowfish untuk Keamanan Data Pada Rancangan Aplikasi Not Diary Berbasis Android.
- [3] Mardiana, Ardi 2013, Rancang Bangun Aplikasi Enkripsi SMS Berbasis Android Menggunakan
- [4] Nasution, Naliana et al. 2015, Aplikasi keamanan File Algoritma Blowfish Pada Universitas Lancang Kuning.
- [5] Panjaitan, Gani Yonathan et al. 2017, Enkriptor Dekriptor Isi Berbasis android Dengan Algoritma Blowfish.
- [6] Rahani, Anah 2014. Implementasi Algoritma Kriptografi blowfish Untuk Keamanan File
- [7] Susanto, 2017. Implementasi Keadaan Data Menggunakan Algoritma Blowfish Pada Sistem Informasi Koperasi Rias.
- [8] Yuliana, Cucu Tri Eka 2013, FASILKOM UDINUS: Implementasi Algoritma Kriptografi Blowfish dan Metode Steganografi End Of File (EOF) untuk Keamanan Data Semarang.
- [9] Wardoyo Siswo et al. 2016, Enkripsi dan Dekripsi File Dengan Algoritma Blowfish Pada Perangkat Mobile Berbasis Android.
- [10] Wijaya, Denaldy 2014. Implementasi Kriptografi Dengan Metode AES dan Blowfish Untuk Keamanan File pdf, doc, txt.
- [11] Wibowo dan Suprayogo 2014, Aplikasi Enkripsi Email dengan Menggunakan Metode Blowfish berbasis J2SE (Java 2 Standart Edition).
- [12] Widayan, Sidiq Nur 2014, Konversi Bilangan Biner, Oktal, Desimal, dan Hexadesimal,