APLIKASI BERBASIS WEB UNTUK PENGAMANAN DATA DENGAN METODE STEGANOGRAFI LSB DAN KRIPTOGRAFI DES

Fiqih Putra Pratama[1)], Wahyu Pramusinto[2]

[1), 2)] Teknik Informatika Fakultas Teknologi Informasi Universitas Budi Luhur
Jl Ciledug Raya, Petukangan Utara, Jakarta Selatan 12250
Email : fiqihputrapratama@gmail.com [1)], wahyu.pramusinto@budiluhur.ac.id [2)]

Abstrak

Pertukaran data untuk kepentingan bisnis seringkali dilakukan melalui email. Akan tetapi karena maraknya kasus pencurian data elektronik, data penting tersebut bias jatuh ke tangan orang lain yang tidak berhak mendapatkannya. Oleh karena ini sebelum dikirimkan via email, data tersebut harus dirahasiakan agar tidak diketahui oleh pihak lain selain pengiriman dan penerima. Ada beberapa cara untuk mengamankan data elektronik, diantaranya adalah dengan metode kriptografi dan steganografi. Kriptografi adalah metode mengubah data asli menjadi bentuk lain yang tidak dapat dibaca. Sementara steganografi adalah suatu metode keamanan data dengan cara menyisipkan data rahasia ke dalam media gambar.

Metode yang digunakan adalah Steganografi LSB (Least Significant Bit Insertion) dan Kriptografi DES (Data Encryption Standard). Sistem kerja kedua metode ini yaitu dengan cara mengenkripsi data terlebih dahulu (kriptografi), lalu setelah itu menyisipkan data rahasia tersebut ke dalam media gambar (steganografi). Setelah menyisipkan data rahasia tersebut yang tidak bisa dilihat oleh kasat mata maka tidak akan ada yang mengetahui isi data rahasia tersebut.

Pada penelitian ini dibuat sebuah aplikasi berbasis web dengan bahasa pemrograman PHP dan database MySQL. Aplikasi ini bisa digunakan oleh pengirim untuk mengamankan file rahasia dan digunakan oleh penerima untuk membuka file rahasia yang sudah diamankan. Aplikasi ini mampu menyisipkan file rahasia ke dalam gambar tanpa diketahui oleh orang lain sehingga file rahasia menjadi lebih aman.

Kata kunci: *steganografi, kriptografi, aplikasi php*

1. Pendahuluan

1.1 Latar Belakang

Seiring berkembangnya zaman, penggunaan internet semakin marak dan sangat membantu manusia dalam berbagai hal. Pengiriman data sering kali dilakukan oleh seseorang untuk kepentingan bisnis dan mereka mengirimnya melalui media *e-mail.* Akan tetapi banyaknya kasus pencurian data elektronik bisa menimbulkan kerugian bagi pemilik data. Solusi dalam mengatasi hal seperti ini adalah dengan mengamankan data yang akan dikirim dengan cara menggunakan metode Steganografi dan Kriptografi

Steganografi adalah suatu metode keamanan data dengan cara menyisipkan data rahasia ke dalam media gambar tanpa ada orang lain yang tahu kecuali oleh si pengirim dan penerima. Kriptografi adalah metode mengubah data asli menjadi bentuk lain yang tidak dapat dibaca. Sehingga kombinasi dari steganografi dan kriptografi sangat dapat meningkatkan keamanan data.

Cara kerja kedua metode ini yaitu dengan cara mengenkripsi pesan terlebih dahulu (Kriptografi), lalu setelah itu menyisipkan pesan rahasia tersebut kedalam sebuah media yaitu salah satunya media gambar (Steganografi). Gambar inilah yang nantinya akan dikirimkan melalui email kepada si penerima

1.2 Rumusan Masalah

a. Bagaimana menjaga kerahasiaan data sehingga tidak diketahui oleh orang lain ?
b. Bagaimana mengimplementasikan metode *Least Significant Bit Insertion* (LSB) dan algoritma *Data Encryption Standard* (DES) untuk mengamankan data elektronik?

1.3 Tujuan Penulisan

Penelitian ini bertujuan untuk mengimplementasikan algoritma Stegnografi *Least Significant Bit Insertion* (LSB) dan algoritma *Data Encryption Standard* (DES) untuk keamanan data.

1.4 Batasan Masalah

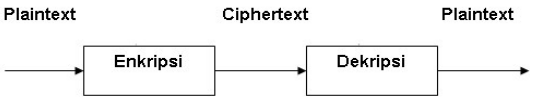
a. Bahasa pemrograman yang digunakan adalah PHP.
b. Panjang kunci yang digunakan pada algoritma DES adalah 56 bit.
c. Algoritma yang digunakan untuk Steganografi adalah LSB.
d. Format gambar yang dapat digunakan adalah png, jpg.
e. Ukuran file rahasia yang dapat disembunyikan maksimum

1.5 Landasan Teori

1.5.1 Kriptografi

Kriptografi (*cryptography*) berasal dari Bahasa Yunani: "*cryptós*" artinya "*secret*" (rahasia), sedangkan "*gráphein*" artinya "*writing*" (tulisan). Jadi, kriptografi berarti "*secret writing"* (tulisan rahasia). Menurut Request for Comments (RFC), kriptografi merupakan ilmu matematika yang berhubungan dengan transformasi data untuk membuat artinya tidak dapat dipahami (untuk menyembunyikan maknanya), mencegahnya dari perubahan tanpa izin atau mencegahnya dari penggunaan yang tidak sah. Jika transformasinya dapat dikembalikan, kriptografi juga bisa diartikan sebagai proses mengubah kembali data yang terenkripsi menjadi bentuk yang dapat dipahami. Kriptografi adalah ilmu yang mempelajari bagaimana menjaga kerahasiaan suatu pesan, agar isi pesan yang disampaikan tersebut aman sampai ke penerima pesan [1].

Secara umum kriptografi merupakan teknik pengamanan informasi yang dilakukan dengan cara mengolah informasi awal (plainteks) dengan suatu kunci tertentu menggunakan suatu metode enkripsi tertentu sehingga menghasilkan suatu informasi baru (chipertext) yang tidak dapat dibaca secara langsung. Chipertext tersebut dapat dikembalikan menjadi informasi awal (plainteks) melalui proses dekripsi [2].
Berikut ini adalah gambaran proses kriptografi secara umum :



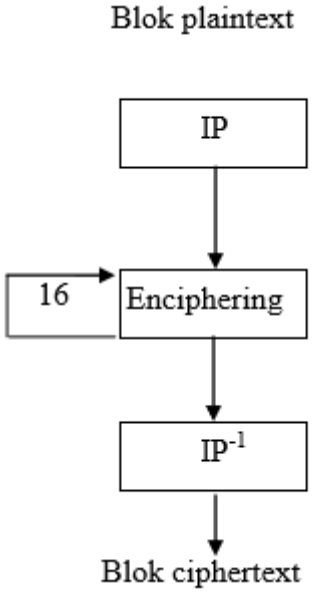
Gambar 1.*Proses Kriptografi [2]*

1.5.2 Algoritma DES

DES (*Data Encryption Standard*) adalah algoritma *Cipher* blok yang populer karena dijadikan standard algoritma enkripsi kunci simetri, yang berarti menggunakan kunci yang sama untuk melakukan enkripsi dan dekripsi.

DES merupakan salah satu chiper block penyandian/kriptografi data yang populer dan telah dijadikan standard enkripsi kunci simetri sejak tahun 1976 dengan ukuran blok 64 bit dan ukuran kuncinya 56 bit. Algoritma DES dibuat di IBM, dan merupakan modifikasi dari algoritma terdahulu yang bernama Lucifer. Lucifer merupakan algoritma cipher block yang beroperasi pada blok masukan 64 bit dan kuncinya berukuran 128 bit [3]

DES termasuk ke dalam sistem kriptografi simetri dan tergolong jenis cipher blok. DES peroperasi pada ukuran blok 64 bit. DES mengenkripsikan 64 bit plainteks menjadi 64 bit chipherteks dengan menggunkan 56 kunci internal atau upa-kunci. Kunci internal dibangkitkan dari kunci eksternal yang panjangnya 64 bit.
Berikut ini adalah skema global algoritma DES.



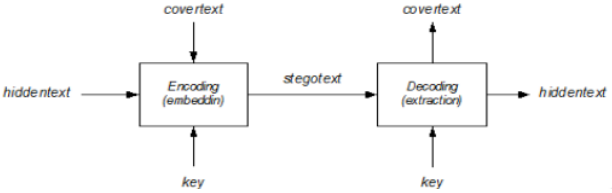
Gambar 2. *Skema Algoritma DES [4]*

Penjelasan gambar di atas :
a. Blok plainteks di permutasikan dengan *metric* permutasi awal (*initial permutation* atau IP).
b. Hasil permutasian awal kemudian di-*enciphering*-sebanyak 16 kali (16 putaran). Setiap putaran menggunakan kunci internal yang berbeda.
c. Hasil enciphering kemudian di-permutasikan dengan matriks permutasi balikan (*invers initial permutation atau ip-1*) menjadi blok *cipherteks*.

1.5.2 Steganografi

Steganografi merupakan suatu ilmu atau seni dalam menyembunyikan informasi dengan memasukkan informasi tersebut ke dalam pesan lain. Dengan demikian keberadaan informasi tersebut tidak diketahui oleh orang lain [4].

Steganografi berasal dari bahasa Yunani, yang berarti tulisan yang tertutup/tersamar ("covered letter"). Dalam arti lain dapat dikatakan steganografi sebagai cara komunikasi yang menyembunyikan pesan. Data/pesan yang akan dikirim disembunyikan ke media lain [5]. Pada penelitian ini media yang digunakan untuk menyembunyikan pesan adalah media gambar.



Gambar 3. *Proses Steganografi [5]*

1.5.3 Metode LSB

Teknik Steganografi modifikasi LSB dilakukan dengan memodifikasi bit-bit yang termasuk bit LSB pada setiap byte warna pada sebuah pixel. Bit-bit LSB ini akan dimodifikasi dengan menggantikan setiap LSB yang ada dengan bit-bit informasi lain yang ingin disembunyikan. Setelah semua bit informasi lain menggantikan bit LSB didalam file tersebut, maka informasi telah berhasil disembunyikan.

Ketika informasi rahasia tersebut ingin kembali dibuka, maka bit-bit LSB yang sekarang ada, diambil satu per satu kemudian disatukan kembali menjadi sebuah informasi yang utuh seperti semula. Penentuan bit-bit LSB dilakukan secara berurutan. Melalui dari *byte* awal sampai *byte* terakhir sesuai panjang dari data rahasia yang akan disembunyikan. LSB hanya mengubah nilai *byte* satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya tidak berpengaruh terhadap persepsi visual/auditori.

2. Pembahasan

2.1 Perancangan Aplikasi

Aplikasi yang akan dibuat terdiri dari beberapa form, yaitu form login, form menu utama, form enkrip, form dekrip, form embed, form retrieve, form admin. Untuk dapat menggunakan sistem ini user harus melakukan login terlebih dahulu. Setelah itu, user bisa melakukan proses enkripsi file dangan memilih form enkrip, bisa melakukan proses dekripsi file dangan memilih form dekrip. Selain itu user juga dapat melakukan proseses steganografi yang terdiri dari proses embed file dan retrieve file. Pada form ini, user diharuskan memilih file dokumen, selanjutnya memilih gambar dan kunci publik yang telah dibuat dan memilih direktori untuk menyimpan file dokumen hasil embed. Sedangkan untuk mengembalikan file yang sudah di embed menjadi file asli, user juga dapat memilih form retrieve. Serta terdapat form bantuan untuk membantu user dalam menggunakan aplikasi ini.

2.2 Studi Literatur

M. Fairuzabadi membuat penelitian untuk mengimplementasikan kriptografi klasik menggunakan Borland Delphi. Metode kriptografi klasik yang digunakan adalah tehnik substitusi, shift chiper, monoalphabetic chiper, polyalphabetic chiper dan tehnik transposisi. Penelitian dilakukan pada plain teks, bukan pada file dokumen. Secara teknik kriptograsi klasik mudah diimplementasikan menggunakan Borland Delphi [6].

Budi Prasetyo melakukan penelitian untuk mengkombinasikan stenografi bit matching dan kriptografi DES untuk mengamankan data. Data yang diamankan pada penelitian tersebut adalah data teks saja, bukan berupa file dokumen. Hasil dari penelitian tersebut menyimpulkan kombinasi kedua metode tersebut dapat digunakan untuk mengamankan data. Citta tidak mengalami perubahan kualitas dan kapasitas pesan yang disimpan dapat lebih besar dari citra [7].

Penelitian lain mengenai kriptografi dan steganografi juga dilakukan oleh Muhammad Fajar Alamsyah, Pada penelitiannya digunakan metode steganografi LSB dengan algoritma RSA. Penggunaan steganografi LSB tanpa dilengkapi sistem keamanan akan terpengaruh dari upaya untuk menghilangkan pesan yang disisipkan sehingga perlu dikembangkan dengan kombinasi algoritma RSA [8].

2.3 Komponen yang digunakan

2.3.1 Perangkat Lunak (*Software*)

Perangkat lunak yang dipakai yaitu perangkat lunak untuk mengembangkan aplikasi. Spesifikasi perangkat lunak yang dipakai yaitu sebagai berikut:

1) Sistem Operasi *Windows* 7
2) *Sublime Text* Versi 3
3) *Notepad++*
4) Google Chrome
5) *Xampp* 5.5.19
6) *MySQL-Front*

2.3.2 Perangkat Keras (*Hardware*)

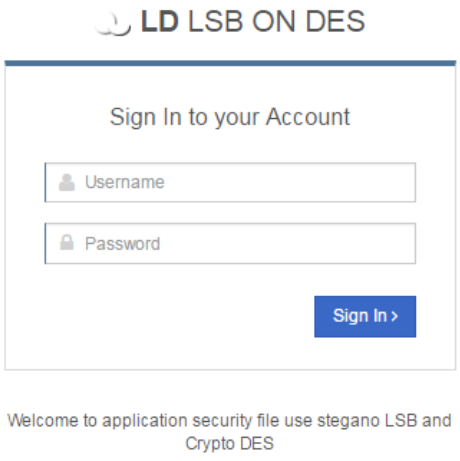
Perangkat keras yang digunakan untuk membuat aplikasi ini agar dapat berjalan dengan baik yaitu sebagai berikut:

1) *Processor* Intel Core i3-2330M *Processor* 2.20 *GHz*
2) *Memory* 6 GB RAM
3) VGA Intel(R) HD *Graphics Family*
4) *Harddisk space* 500 GB

2.4 Tampilan Layar

2.4.1 Tampilan Layar Login

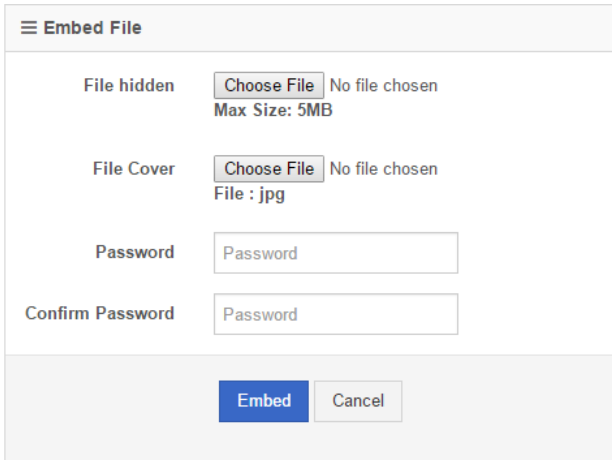
Sebelum menggunakan aplikasi, user diharuskan login dengan cara mengisi username dan password. Jika login berhasil, user bisa menggunakan aplikasi ini.



Gambar 4. *Tampilan Layar Form Login*

2.4.2 Tampilan Layar Form Embed

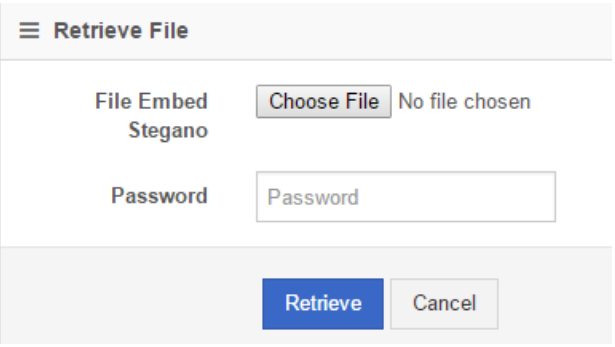
Form ini dipakai untuk menyembunyikan file rahasia ke dalam gambar. Pada kolom file hidden dipilih file dokumen yang akan disembunyikan, ukuran maks file ini adalah 5MB. Pada kolom file cover adalah gambar yang akan digunakan untuk menyembunyikan file rahasia, jenis gambar yang bisa digunakan adalah jpg. Pada kolom *password* diisi *password* untuk kunci kriptografi DES dan kolom *confirm password* dipakai untuk validasi password yang diinputkan sudah benar. Jika tombol *embed* diklik, maka proses enkripsi kriptografi dan *embed* steganografi dijalankan.



Gambar 5. *Tampilan Layar Form Embed*

2.4.3 Tampilan Layar Form Retrieve

Form ini dipakai untuk mengembalikan file rahasia yang sudah disembunyikan ke dalam gambar. Pada kolom file embed stegano dipilih file gambar yang di dalamnya terdapat file rahasia. Pada kolom *password* diisi password yang sebelumnya dimasukkan pada form *embed* file. Jika password salah, proses retrieve file tidak akan berhasil. Jika tombol *retrieve* diklik, maka proses dekripsi kriptografi dan *retrieve* steganografi dijalankan



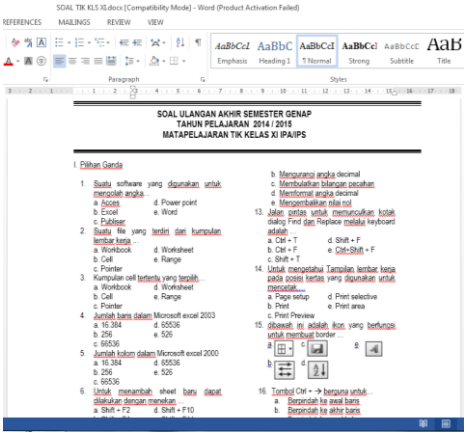
Gambar 6. *Tampilan Layar Form Retrieve*

2.5 Uji Coba Aplikasi

Setelah kebutuhan terpenuhi baik *software* maupun *hardware,* maka proses selanjutnya adalah menguji coba aplikasi yang telah dibuat. Pada bagian ini dapat di urakian mengenai hasil pengujian enkripsi dan dekripsi *file*, *embed* dan *retrieve*. Pengujian tersebut nantinya akan mendapatkan hasil perbandingan *file* asli dan hasil proses dari aplikasi *file* yang sudah di proses.

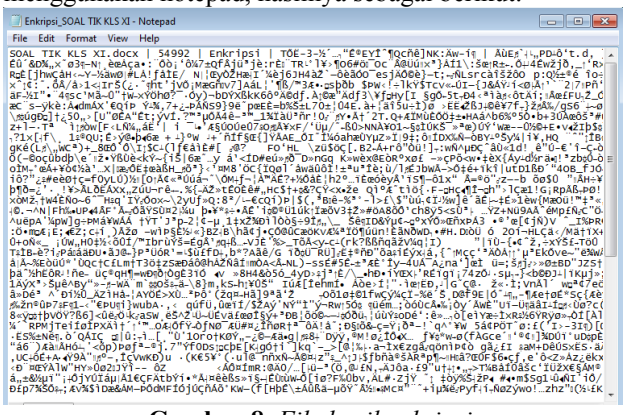
2.5.1 Uji Coba Enkripsi dan Dekripsi

Untuk melakukan proses *encrypt* file, admin bisa melakukan proses tersebut pada menu *Encrypt*, admin akan memilih file yang mana file tersebut akan diproses enkripsi dan jika proses berhasil file akan ter-download secara otomatis, Dibawah ini adalah tampilan file docx sebelum di proses enkripsi.



Gambar 7. *File awal sebelum dienkripsi*

Setelah dienkripsi, file tersebut tidak bisa dibuka menggunakan aplikasi Microsoft Word. Jika dibuka menggunakan notepad, hasilnya sebagai berikut:

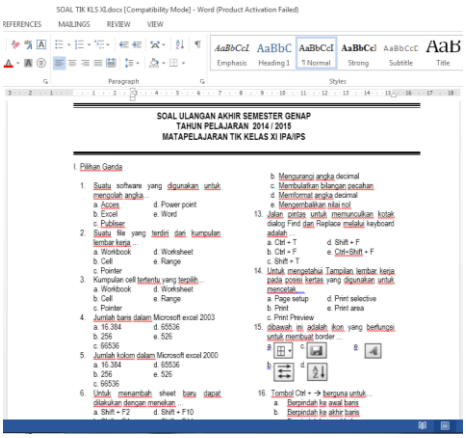


Gambar 8. *File hasil enkripsi*

Saat file hasil enkripsi tersebut di-dekripsi, file tersebut akan kembali menjadi seperti semula.

2.5.2 Uji Coba Embed dan Retrieve

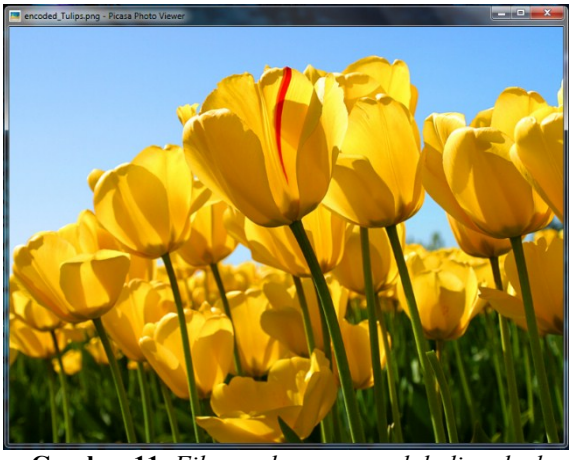
Dalam uji coba *Embed* dan *Retrieve user* dapat melakukan proses tersebut dalam menu *Embed* yang dimana *form Embed* berfungsi untuk menyembunyikan *file* kedalam sebuah image. Pada uji coba ini sebuah file docx akan disembunyikan ke dalam sebuah file jpg.



Gambar 9. *File awal sebelum di-embed*

Gambar 10. *File gambar yang akan menjadi cover*

Sedangkan di bawah ini adalah tampilan file *image* yang sudah berhasil disisipkan file rahasia, mungkin secara kesat mata tidak melihat perubahan dari file image asilnya akan tetapi file *image* yang sudah disisipkan mengalami perubahan dari kualitas gambar serta ukuran file *image* yang semakin besar. Pada contoh ini, ukuran file image awal adalah 606 KB, file rahasia yang akan disembunyikan 53.7 KB dan hasil file embed adalah 1,02 MB.



Gambar 11. *File gambar yang sudah di-embed*

2.6 Evaluasi Aplikasi

a. Kelebihan Program

1) Aplikasi ini hanya memproses file dan ketika selesai file langsung ter-*Download* membuat aplikasi tidak menampung *file* hasil proses enkripsi maupun *embed*.
2) *File* hasil dekripsi sama sekali tidak mengalami perubahan dari *file* aslinya.

b. Kekurangan Program

1) Aplikasi ini hanya dapat menerima *file* dengan ukuran maksimal 5MB.
2) Waktu yang dibutuhkan untuk proses enkripsi tergolong tidak sebentar.
3) Waktu proses enkripsi-dekrip berbanding lurus dengan ukuran *file*, semakin besar ukuran *file* semakin lama waktu yang dibutuhkan untuk proses enkripsi-dekrip.
4) Tidak adanya Database dan *directory file* hasil proses enkrip dan embed.

Proses *encrypt, decrypt, embed* dan *retrieve* dilakukan secara terpisah.

3. Kesimpulan

Kesimpulan yang dapat diambil pada penelitian ini adalah sebagai berikut :

a. Aplikasi ini dapat menyisipkan *(embed)* pesan kedalam sebuah gambar tanpa merusak kualitas gambar.
b. Aplikasi ini mampu mengambil *(retrieve)* pesan maupun data file pada berkas file stego.
c. Aplikasi ini mampu menyisipkan data rahasia ke dalam sebuah file gambar tanpa ada yang mengetahuinya.

Aplikasi steganografi ini mempunyai sifat mudah rusak *(fragile)* apabila mengalami rotasi, *cropping* dan gangguan, sehingga menyebabkan pesan hilang, untuk itu aplikasi ini dapat dikembangkan sehingga dapat mengatasi kerusakan file yang sudah di-encode meskipun sudah mengalami rotasi, *cropping* dan perubahan lainnya.

Daftar Pustaka

[1] D. Ariyus, "Pengantar Imu Kriptografi", Yogyakarta : Andi, 20
[2] M.Y. Andri, "Implementasi Algoritma Kriptografi DES, RSA dan Algoritma Kompresi LZW Pada Berkas Digital", 2009
[3] R. Munir, "Kriptografi", Bandung : Informatika Bandung, 2006
[4] Febriansyah, "Analisis dan Perancangan Keamanan Data Meggunakan Algoritma Kriptografi DES (Data Encryption Standard), 2012
[5] Z. Niswati, "Steganografi Berbasis Least Significant Bit (LSB) Untuk Menyisipkan Gambar Ke Dalam Citra Gambar", Jurnal Factor Exacta Vol 5 No. 2 : 181-191, ISSN : 1979276X, 2012
[6] M. Fairuzabadi, "Implementasi Kriptografi Klasik Menggunakan Borland Delphi", Jurnal Dinamika Informatika Vol 4 No 2, September 2010.
[7] B. Prasetiyo, "Kombinasi Steganografi Bit Matching dan Kriptografi DES Untuk Pengamanan Data", 2013.
[8] M. F. Alamsyah, "Implementasi Metode Steganografi Least Significant Bit Dengan Algoritma RSA Pada Citra BMP", 2012

Biodata Penulis

Fiqih Putra Pratama, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Teknik Informatika Universitas Budi Luhur, lulus tahun 2016.

Wahyu Pramusinto, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Teknik Informatika Universitas Budi Luhur, lulus tahun 2007. Memperoleh gelar Magister Komputer (M.Kom) Program Pasca Sarjana Magister Ilmu Komputer Universitas Budi Luhur, lulus tahun 2010. Saat ini menjadi Dosen di Universitas Budi Luhur.