APLIKASI KEAMANAN EMAIL MEMANFAATKAN SPAM DAN ALGORITMA VIGENERE

Bambang Sugiantoro[1]
[1]Prorgram Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sunan Kalijaga Yogyakarta
Jl. Marsda Adisucipto Yogyakarta
Email: bambang.sugiantoro@uin-suka.ac.id

Abstrak

Aktivitas penggunaan internet semakin meningkat dilingkungan UIN Sunan Kalijaga Yogyakarta. Pertukaran informasi pun sudah banyak menggunakan media internet, salah satunya adalah menggunakan media e-mail (electronic mail). Namun seiring dengan berjalannya waktu, muncul berbagai macam masalah yang kerap dijumpai dalam aplikasi berkirim surat ini terutama dalam hal keamanan informasi. Permasalahan keamanan informasi dalam e-mail yang sering dijumpai antara lain, penyadapan pasif, penyadapan aktif, penipuan, dan lain-lain.Atas dasar permasalahan tersebut peneliti membuat suatu aplikasi yang berguna untuk mengubah pesan yang memerlukan pengamanan agar terhindar dari penyadapan aktif seorang cracker. Pengamanan yang akan penulis tawarkan menggunakan dua model pengamanan yaitu enkripsi dan steganografi.Penyembunyian pesan akan dilakukan dengan menggunakan media e-mail spam. Jadi, logika sederhananya ketika penyadap membaca pesan yang sudah disembunyikan, dia akan menganggapnya sebagai spam biasa dan akan mengabaikannya. Sehingga, pesan akan sampai dengan aman tanpa mengalami perubahan isi sedikitpun dari sang penyadap.

Kata kunci : ***e-mail spam; aplikasi; cracker***

Pendahuluan

Aktivitas penggunaan sistem informasi akademik (SIA) dan Internet di UIN Sunankalijaga khususnya fakultas sains dan teknologi semakin meningkat dengan pesat akhir-akhir ini. Pertukaran informasi pun sudah banyak menggunakan media internet, salah satunya adalah menggunakan media *e-mail* (*electronic mail*). Dengan menggunakan *e-mail*, pesan menjadi lebih cepat tersampaikan bahkan hanya dalam hitungan detik serta tidak memakan banyak biaya. Berbeda dengan pengiriman pesan atau surat menggunakan metode konvensional yang segala sesuatunya harus diurus secara fisik dan tentunya banyak mengeluarkan biaya.

Namun seiring dengan berjalannya waktu, muncul berbagai macam masalah yang kerap dijumpai dalam aplikasi SIA dan internet terutama dalam hal keamanan informasi. Permasalahan keamanan informasi dalam *e-mail* yang sering dijumpai antara lain, penyadapan pasif, penyadapan aktif, penipuan, dan lain-lain. Penyadapan pasif dan aktif ini sering dilakukan jika pengguna *SIA* (Dosen , mahasiswa maupun tata usaha) menggunakan akses seperti warnet, area *hotspot*, dan lain-lain. Pesan yang disadap biasanya merupakan pesan-pesan yang "menarik" (bersifat rahasia, memiliki nilai jual). Untuk mendapatkan pesan-pesan yang "menarik" tersebut, seorang penyadap akan memantau tiap aktivitas.

Jika terdapat surat yang dia anggap tidak "menarik", tentu akan dilewatkannya. Salah satu jenis surat yang tidak "menarik" ini biasanya berjenis spam atau junk. Spam juga merupakan salah satu masalah serius saat ini yang sering dijumpai dalam pertukaran informasi melalui media e-mail. Spam merupakan suatu pesan yang isinya mengarah ke penjerumusan, penipuan, iklan, dan lain-lain. Spam biasanya dikirim dari seseorang yang tak dikenal ataupun melalui robot/tool otomasi. Jadi, seorang penyadap jika dalam pemantauannya menemukan pesan spam, penyadap akan mengabaikannya.

Atas dasar permasalahan tersebut akan dilakukan penelitian untuk membuat suatu *website* yang berguna untuk mengubah pesan yang memerlukan pengamanan agar terhindar dari penyadapan aktif seorang *cracker*. Pengamanan yang akan penulis tawarkan menggunakan dua model pengamanan yaitu enkripsi dan steganografi. Dengan enkripsi, pesan teks biasa (*plain text*) akan diubah menjadi suatu pesan yang tidak mudah dibaca (*cipher text*). Dengan steganografi, pesan yang tidak terbaca tadi akan disembunyikan di dalam media tertentu seperti gambar, teks dan suara.

Penyembunyian pesan akan dilakukan dengan menggunakan media *e-mail spam*. logika sederhananya ketika penyadap membaca pesan yang sudah disembunyikan, dia akan menganggapnya sebagai *spam* biasa dan akan mengabaikannya. Sehingga, pesan akan sampai dengan aman tanpa mengalami perubahan isi sedikitpun dari sang

penyadap. Dan penerima pesan cukup mengubah kembali pesan *spam* yang sudah diterima menjadi pesan asli (*plaintext*) melalui *website*.

Karena pada penelitian ini menggunakan dua macam pengamanan pesan yaitu enkripsi dan steganografi, tentu tidak bisa melupakan salah satu sifat yang terdapat pada teks hasil enkripsi (*ciphertext*) yaitu terlihat (*visible*) dan acak. Orang akan dengan mudah menebak bahwa teks tersebut merupakan teks hasil enkripsi. Hal ini terjadi, karena biasanya teks hasil enkripsi mengandung karakter-karakter tak lazim dan susunannya sangat acak atau tidak bisa dieja, meskipun pada akhirnya tetap susah untuk memecahkan isi pesan yang terkandung di dalam *ciphertext* tersebut. Untuk meminimalisasi munculnya karakter-karakter tak lazim inilah penulis memilih algoritma kriptografi *vigènere cipher*. Selain itu algoritma ini memiliki kelebihan lain yaitu pengguna bisa memilih sendiri karakter-karakter apa saja yang boleh muncul pada *ciphertext* nanti melalui bujur sangkar *vigènere* /*tabula recta*.

Landasan Teori

Kriptografi

Kriptografi berasal dari bahasa Yunani, menurut bahasa berasal dari dua kata yaitu *kripto* dan *graphia*. *Kripto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Menurut terminologinya, kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat yang lain (Ariyus, 2006).
Definisi yang dipakai dalam buku-buku yang lama (sebelum tahun 1980-an) menyatakan bahwa kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya (Munir, 2006). Schneier menyatakan bahwa kriptografi adalah seni untuk menjaga keamanan pesan (Schneier, 1996). Sedangkan Menezes dan kawan-kawan menyatakan bahwa kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritas data, serta otentikasi (Menezes, 1996).

Vigènere cipher

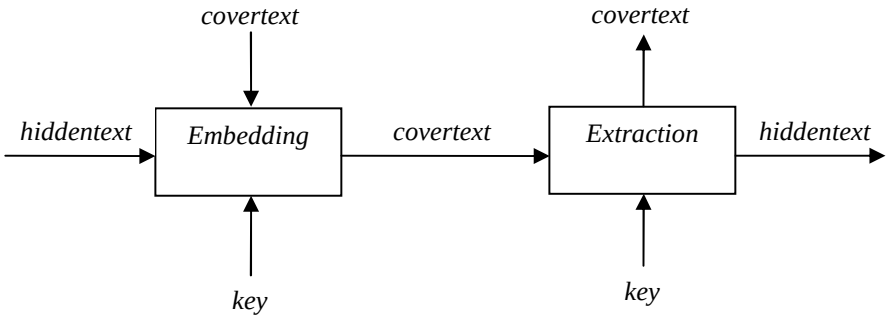
Kode *vigènere* termasuk kode abjad-majemuk (*polyalphabetic substitution cipher*). Dipublikasikan oleh diplomat (sekaligus seorang kriptologis) Perancis, Blaise de Vigènere pada abad 16, tahun 1586. Sebenarnya Giovan Batista Belaso telah menggambarkannya untuk pertama kali pada tahun 1533 seperti ditulis di dalam buku *La Cifra del Sig*. Algoritma ini baru dikenal luas 200 tahun kemudian dan dinamakan kode *vigènere*. *Vigènere* merupakan pemicu perang sipil di Amerika dan kode *vigènere* digunakan oleh Tentara Konfederasi (*Confederate Army*) pada perang sipil Amerika (*American Civil War*). Kode *vigènere* berhasil dipecahkan oleh Babbage dan Kasiski pada pertengahan abad 19. (Ariyus, 2008).
Algoritma enkripsi jenis ini sangat dikenal karena mudah dipahami dan diimplementasikan. Teknik untuk menghasilkan *ciphertext* bisa dilakukan menggunakan substitusi angka maupun bujursangkar *vigènere*. Teknik susbtitusi *vigènere* dengan menggunakan angka dilakukan dengan menukarkan huruf dengan angka, hampir sama dengan kode geser.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

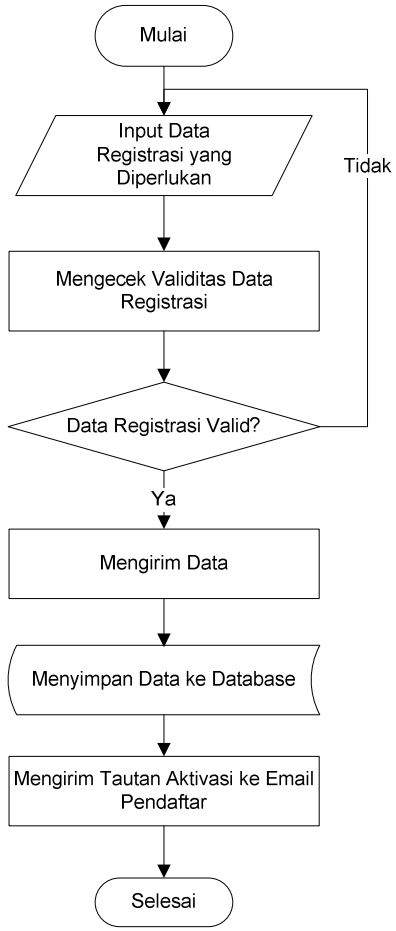
Gambar 1. Plaintext : PLAINTEXT

Steganografi

Steganografi (*steganography*) adalah ilmu dan seni menyembunyikan pesan rahasia di dalam pesan lain sehingga keberadaan pesan rahasia tersebut tidak dapat diketahui. Steganografi sangat kontras dengan kriptografi, jika kriptografi merahasiakan makna pesan sementara eksistensi pesan tetap ada, maka steganografi menutupi keberadaan pesan. (Munir, 2006). Steganografi adalah teknik atau seni menyimpan pesan atau file rahasia ke dalam media lain. (S'to, 2009). Sedangkan Mangarae menjelaskan bahwa steganografi adalah usaha menyembunyikan informasi privasi atau sensitif ke dalam sesuatu namun terlihat tidak berbeda dari biasanya (Mangarae, 2006).
Steganografi memiliki dukungan yang sangat luas terhadap metode berkomunikasi rahasia untuk menyembunyikan keberadaan pesan. Metode-metode yang termasuk di dalamnya antara lain, tinta tak terlihat (*invisible ink*), *microdots*, penyusunan karakter (*character arrangement*), tanda tangan digital (*digital signature*), *covert channel*, dan *spread spectrum communications* (Johnson dan Jajodia, 1998).



Gambar 2. Diagram penyisipan dan ekstraksi pesan



Gambar 3 *Flowchart* Proses Registrasi Pengguna

Penyisipan pesan ke dalam media *covertext* dinamakan *encoding*, sedangkan ekstraksi pesan dari *stegotext* dinamakan *decoding*. Kedua proses ini mungkin memerlukan kunci rahasia yang dinamakan *stegokey* agar hanya pihak yang berhak saja yang dapat melakukan penyisipan pesan dan ekstraksi pesan. Gambar 2.1 memperlihatkan diagram penyisipan dan ekstraksi pesan. Diagram ini mirip dengan diagram enkripsi-dekripsi pada kriptografi. (Munir, 2006).

Perancangan Sistem

Website ini dibangun menggunakan pendekatan algoritma, sehingga perancangan dibuat menggunakan diagram alir atau *flowchart*. Karena sistem ini juga menggunakan database, maka juga dilakukan perancangan desain database berdasarkan analisis fungsional dan nonfungsional, serta berdasarkan rancangan diagram alir. Karena hasil penelitian berupa *website*, maka tampilan antar muka tidak bisa diabaikan. Oleh karena itu, antar muka *website* juga perlu didesain dengan baik.

Diagram Alir Proses Registrasi

Langkah pertama agar bisa menggunakan *website* ini yaitu dengan terlebih dahulu mendaftar sebagai *member*. Pendaftaran dilakukan dengan mengisi *form* pendaftaran yang disediakan di menu **Register**. Setelah *form* diisi pengguna menekan tombol *Submit* untuk *register*, maka *website* melakukan pengecekan terhadap isian *form*, apakah sudah terisi dengan baik atau belum. Jika sudah terisi dengan baik, maka *website* melanjutkan proses *submit* dan memasukkan data registrasi ke dalam *database*. Lalu *website* akan mengirim *link* aktivasi ke *e-mail* pendaftar dan harus diakses oleh pendaftar agar *account*-nya menjadi aktif, sehingga pendaftar bisa *login* ke *website*. Diagram alir untuk proses registrasi ini penulis sajikan pada gambar 4.1.

Rancangan *Database*

Rancangan *database* untuk *website* ini dibuat berdasarkan analisis fungsional dan nonfungsional, serta berdasarkan rancangan diagram alir (*flowchart*).

Tabel sys_user

Tabel sys_user merupakan tabel yang berisi tentang informasi pengguna *website*. Tabel ini juga mengelompokkan pengguna yang dapat melakukan akses terhadap *website* berdasarkan hak aksesnya.

Tabel 2 sys_user

Field	Type	Constraint
id	int(5)	Primary Key
email	varchar(128)	
password	char(32)	
name	varchar(50)	
birthday	date	
level	enum(‘1’, ‘2’, ‘3’)	
active	enum('Yes', 'No')	
date	datetime	

Tabel master_key

Tabel master_key merupakan tabel yang berisi kunci-kunci yang digunakan oleh pengguna untuk melakukan proses enkripsi pesan menggunakan algoritma *vigènere cipher*.

Tabel 2 master_key

Field	Type	Constraint
id	int(5)	Primary Key
user_id	int(5)	Foreign Key to sys_user.id
key	varchar(12)	

Tabel master_table

Tabel master_table merupakan tabel yang berisi *tabula recta* atau kombinasi karakter-karakter yang diperbolehkan untuk melakukan proses enkripsi menggunakan algoritma *vigènere cipher.*

Tabel 4.4 master_table

Field	Type	Constraint
id	int(5)	Primary Key
user_id	int(5)	Foreign Key to user_id.id
table	char(65)	

Tabel sys_guestbook

Tabel sys_guestbook merupakan tabel yang berisi pesan-pesan dari pengguna maupun dari pengunjung *website* yang ditujukan langsung kepada administrator.

Tabel .5 sys_guestbook

Field	Type	Constraint
id	int(5)	Primary Key
name	varchar(50)	
sender_id	int(5	Foreign Key to sys_user.id
date	datetime	
email	varchar(128)	
message	varchar(512)	
status	enum(‘unread’, ‘read’, ‘replied’)	

Tabel sys_reply_guestbook

Tabel sys_reply_guestbook merupakan tabel yang berisi jawaban administrator atas pesan-pesan yang dikirim melalui menu **Contact Us**.

Tabel.6 sys_reply_guestbook

Field	Type	Constraint
id	int(5)	Primary Key
guestbook_id	int(5)	Foreign Key to sys_guestbook.id
sent_date	datetime	
destination	varchar(128)	
subject	varchar(128)	
message	text	

Tabel sys_menu

Tabel sys_menu merupakan tabel yang berisi menu-menu yang disediakan di *website* dan dikelompokkan berdasarkan level akses pengguna.

Tabel 7 sys_menu

Field	Type	Constraint
id	int(2)	Primary Key
menu	varchar(28)	
level	enum('1', '2', '3')	
url	varchar(64)	

Tabel sys_message

Tabel sys_message merupakan tabel yang berisi pesan-pesan yang sudah dienkripsi dan di-*embed* ke pesan *spam* yang sudah dikirimkan kepada alamat *e-mail* tujuan.

Tabel 8 sys_message

Field	Type	Constraint
id	int(10)	Primary Key
sent_date	datetime	
sender_id	int(5)	Foreign Key to sys_user.id
plain_text	text	
cipher_text	varchar(50)	
destination	varchar(128)	
subject	varchar(128)	
key	int(5)	Foreign Key to master_key.id
table	int(5)	Foreign Key to master_table.id

Tabel sys_login

Tabel sys_login merupakan tabel yang berisi informasi mengenai pengguna ketika pengguna melakukan proses *login* ke dalam *website*.

Tabel 9 sys_login

Field	Type	Constraint
id	int(8)	Primary Key
user_id	int(5)	Foreign Key to sys_user.id
date	datetime	
ip_address	varchar(15)	
web_browser	varchar(24)	
operating_system	varchar(24)	
full_user_agent	varchar(128)	
session_id	char(32)	

Tabel sys_logout

Tabel sys_logout merupakan tabel yang berisi informasi mengenai pengguna ketika pengguna melakukan proses *logout* dari *website*.

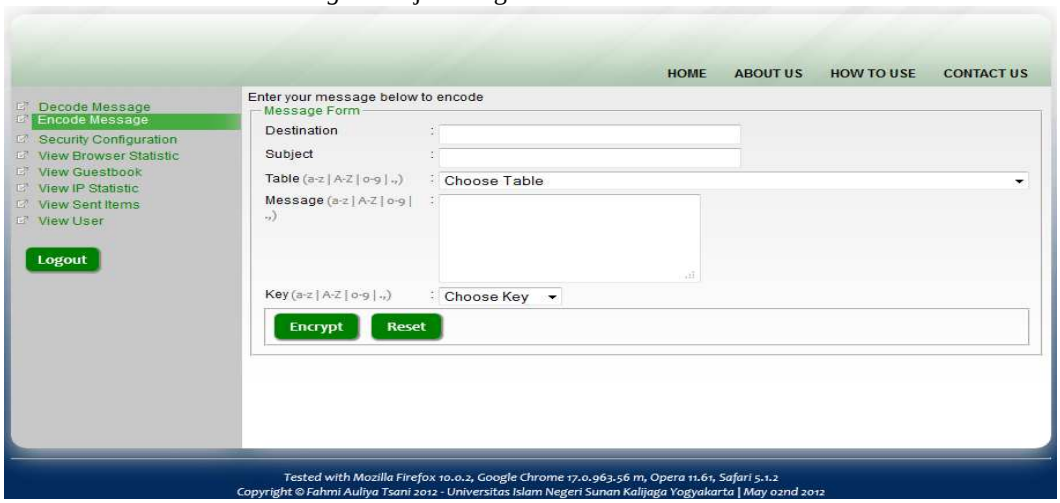
Tabel 10 sys_logout

Field	Type	Constraint
id	int(8)	Primary Key
user_id	int(5)	Foreign Key to sys_user.id
session_id	char(32)	
date	datetime	

Implementasi

Tampilan *Interface* Halaman *Encode Message*

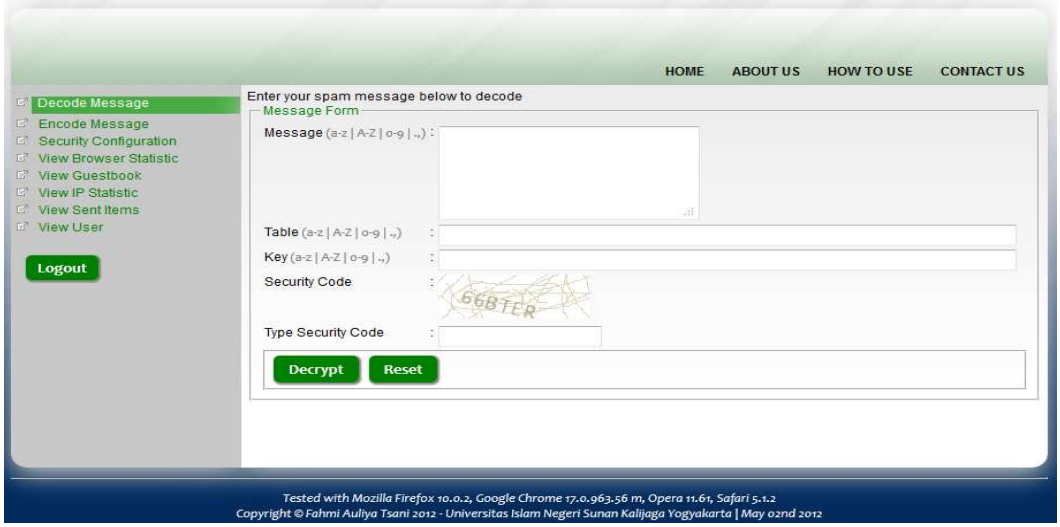
Setelah pengguna melakukan proses registrasi dan aktivasi akun, maka pengguna bisa *login* ke *website* dan menikmati fitur utama yang disediakan yaitu *Encode Message*. Pada halaman ini, pengguna bisa memasukkan pesan penting, rahasia, maupun pesan lainnya untuk diubah menjadi pesan tersandi dan diubah lagi menjadi pesan *spam*. Hasil pesan *spam* inilah yang akan dikirimkan ke alamat *e-mail* tujuan yang dimasukkan. Implementasi desain *interface* untuk halaman *Encode Message* ditunjukkan gambar 5.4.



Gambar 5.4 *Interface* Menu *Encode Message*

Tampilan *Interface* Halaman *Docode Message*

Selain bisa mengubah pesan biasa menjadi pesan *spam*, pengguna juga bisa mengubah kembali pesan *spam* tersebut menjadi pesan biasa dengan terlebih dahulu mengetahui pola *tabula recta* dan kombinasi kata kunci yang digunakan. Fitur ini bisa diakses melalui menu *Docode Message*. Pengguna cukup menyalin pesan *spam* ke *textbox* yang sudah disediakan dan memasukkan kombinasi *tabula recta* dan kata kunci yang digunakan. Proses untuk mengubah pesan *spam* menjadi pesan asli (*decoding*) ini berjalan di *background* menggunakan teknologi AJAX. Sehingga, pengguna tidak perlu me-*refresh* halaman untuk bisa melihat hasil proses *decoding*. Implementasi desain *interface* halaman *Docode Message* ini ditunjukkan pada gambar 5.5.



Gambar 5.5 *Interface* Menu *Docode Message*

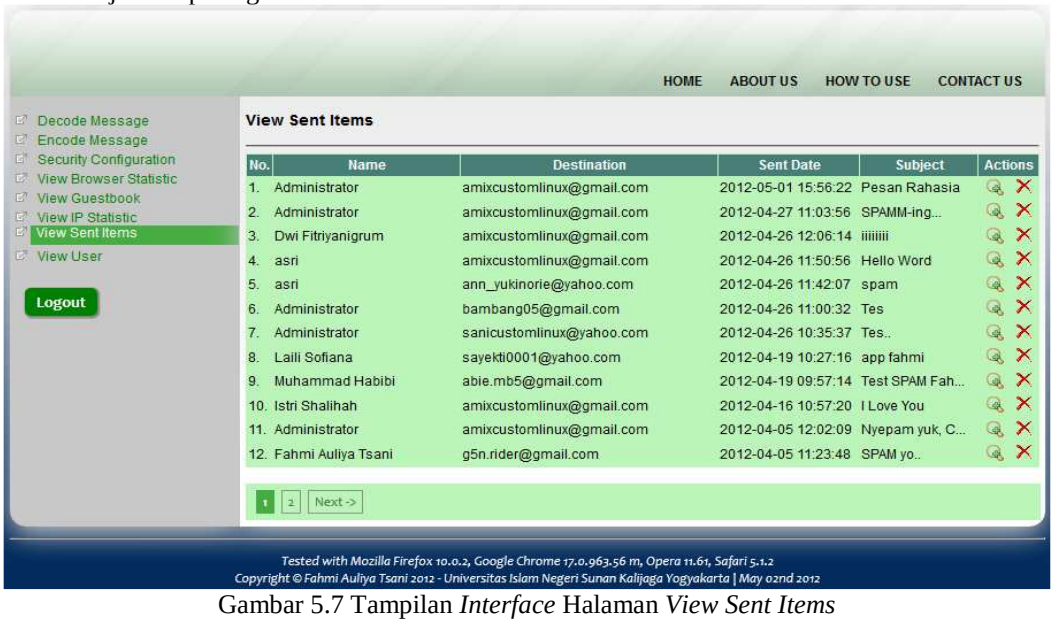
Tampilan *Interface* Halaman *View Users* untuk Admin

Pengguna *website* ini dibedakan menjadi 3 jenis, yaitu administrator, *user*, dan *public user*. Pengguna dengan jenis administrator memiliki menu-menu khusus yang bisa diakses setelah login ke *website*. Salah satu menu khusus untuk administrator adalah *View User*. Melalui menu ini, administrator dapat melihat siapa saja pengguna dari *website*. Administrator juga memiliki wewenang untuk menghapus pengguna dari *database* jika memang diperlukan. Implementasi desain *interface* untuk halaman *View User* diperlihatkan pada gambar 5.6.

Gambar 5.6 *Interface View Users*

Tampilan *Interface* Halaman *View Sent Item*

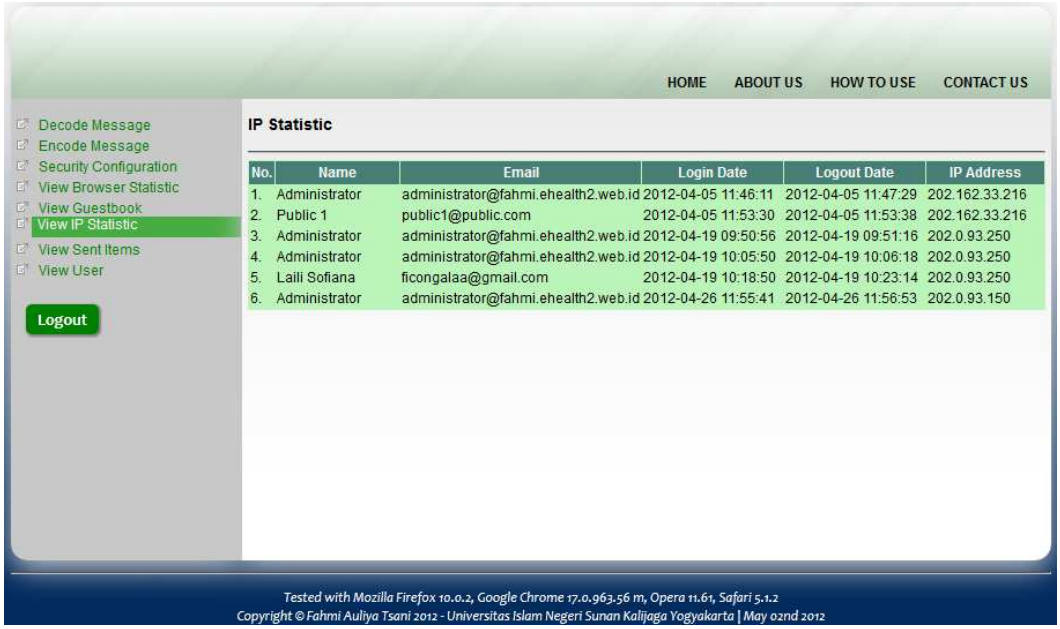
Baik administrator maupun *user* biasa, masing-masing bisa melihat pesan apa saja yang pernah dikirimkan. Untuk *user* jenis administrator, dapat melihat semua pesan terkirim dari semua pengguna *website* ini. Fitur ini bisa diakses melalui menu *View Sent Items*. Pengguna dapat membaca lagi pesan yang sudah dikirimkan. Hanya administrator yang memiliki kewenangan untuk menghapus pesan terkirim. Implementasi desain *interface* untuk halaman *View Sent Items* ini ditunjukkan pada gambar 5.7



Gambar 5.7 Tampilan *Interface* Halaman *View Sent Items*

Tampilan *Interface* Halaman *View IP Statistic* untuk Admin

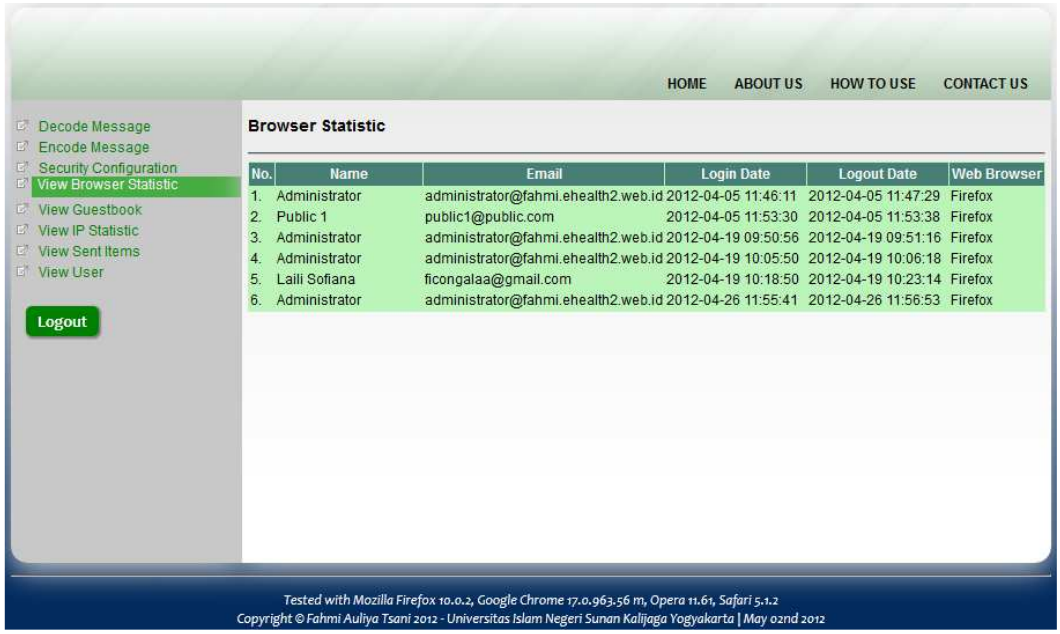
Fitur *View IP Statistic* ini hanya dapat diakses oleh administrator melalui menu *View IP Statistic*. Halaman ini berisi tabel *log* pengaksesan oleh semua jenis pengguna *website* ini. Terdapat beberapa informasi yang ditampilkan pada tabel *log IP Statistic* antara lain nama pengguna, alamat e-mail pengguna, waktu *login*, waktu *logout*, serta alamat IP yang digunakan oleh pengguna untuk mengakses *website* ini. Implementasi desain *interface* untuk halaman *View IP Statistic* ini dapat dilihat pada gambar 5.8.



Gambar 5.8 *Interface* Menu *View IP Statistic* untuk Admin

Tampilan *Interface* Halaman *View Browser Statistic* untuk Admin

Halaman ini memiliki desain tampilan yang mirip dengan halaman *View IP Statistic*. Halaman ini juga hanya dapat diakses oleh administrator. Informasi-informasi yang ditampilkan juga mirip dengan informasi yang ditampilkan pada halaman *View IP Address*. Perbedaannya hanya terletak pada kolom *IP Address* pada menu *View IP Statistic* diganti dengan kolom *Web Browser*. Menu ini disediakan untuk administrator dengan harapan administrator dapat melakukan studi lanjut mengenai hasil *render website* jika diakses menggunakan *web browser* tertentu dengan melihat persentase jumlah pengguna yang mengakses *website* ini. Implementasi desain *interface*-nya dapat dilihat pada gambar 5.9.



Gambar 5.9 *Interface* Menu *View Browser Statistic* untuk Admin

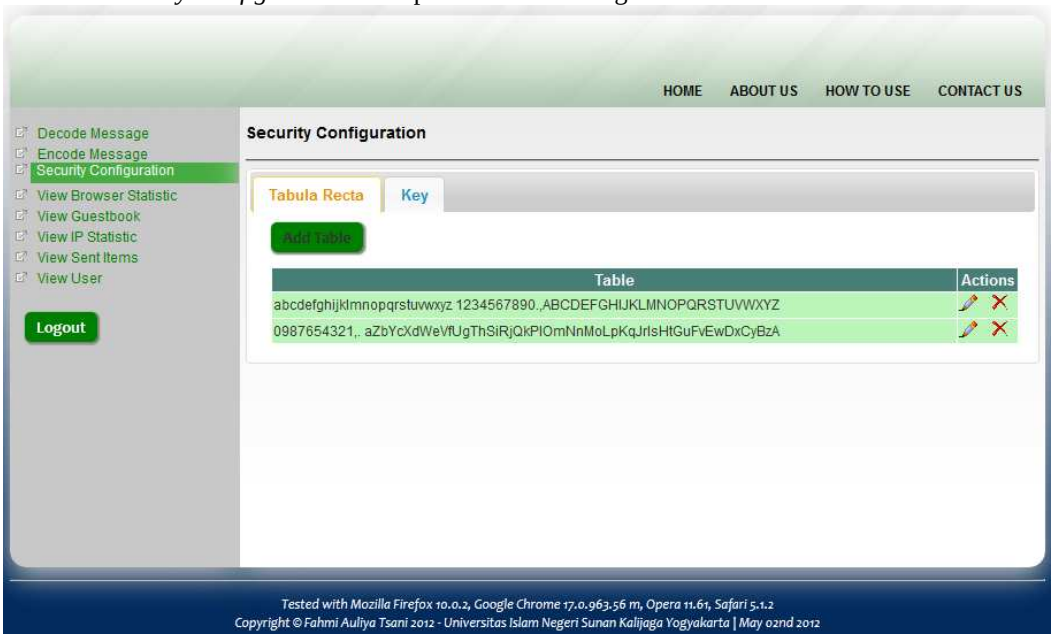
ampilan *Interface* Halaman *View Guestbook* untuk Admin

Terdapat satu lagi menu yang khusus disediakan untuk administrator yaitu *View Guestbook*. Melalui halaman ini, administrator dapat melihat pesan-pesan yang ditujukan khusus untuknya yang dikirim oleh pengguna melalui menu *Contact Us*. Melalui halaman ini, administrator dapat membalas pesan tersebut dengan dikirim langsung melalui alamat *e-mail* pengguna. Administrator juga memiliki kewenangan untuk menghapus pesan-pesan ini. Implementasi desain *interface* untuk halaman *View Guestbook* ini ditunjukkan oleh gambar 5.10.

Gambar 5.10 *Interface* Menu *View Guestbook* untuk Admin

Tampilan *Interface* Halaman *Security Configuration*

Halaman *Security Configuration* ini dapat diakses oleh pengguna biasa dan administrator. Melalui menu ini, pengguna dapat mengonfigurasi kombinasi *tabula recta* dan kata kunci yang dapat ia gunakan untuk proses *Encoding-Decoding*. Pengguna dibatasi hanya dapat memiliki masing-masing lima kombinasi. Implementasi desain *interface* halaman *Security Configuration* ini dapat dilihat melalui gambar 5.11.



Gambar 5.11 *Interface* Menu *Security Configuration*

Kesimpulan

Berdasarkan penelitian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. *Website* dapat melakukan proses penyandian (enkripsi) menggunakan algoritma *vigènere cipher*.
2. *Website* dapat menyembunyikan pesan yang sudah disandikan ke dalam pesan berpola *spam*.
3. *Website* dapat mengubah kembali pesan *spam* menjadi pesan sandi dan kembali menjadi pesan teks asli.
4. *Website* mampu mengirim *e-mail* berisi pesan *spam*.

DAFTAR PUSTAKA

Ariyus, Dony. (2006). *Kriptografi, Keamanan Data, dan Komunikasi*. Yogyakarta: Graha Ilmu.
Ariyus, Dony. (2008). *Pengantar Ilmu Kriptografi: Teori, Analisis, dan Implementasi*. Yogyakarta: Penerbit Andi.

Azdy, Rezania Agramisti. (2009). *Implementasi Tanda Tangan Digital untuk Meningkatkan Keamanan Email Client pada Perangkat Mobile*. Skripsi Universitas Gadjah Mada Yogyakarta.

Burnett, Steve dan Paine, Stephen. 2004. *RSA Security's Official Guide to Cryptography*. McGraw-Hill, Inc.

Gollmann, Dieter. (2006). *Computer Security, Second Edition*. John Wiley & Sons, Ltd.

Gracia, Kusuma. (2005). *Implementasi Steganography dengan Metode Low Bit Encoding untuk Penyisipan Data dalam File Audio WAV*. Skripsi Universitas Gadjah Mada Yogyakarta

Hidayat, Sri. (2007). *Implementasi Steganography pada Media File Audio MP3 dengan Polyalphabetic Substitution Cipher*. Skripsi Universitas Gadjah Mada Yogyakarta

Johnson, Neil F. dan Jajodia, Sushil. (1998). *Exploring Steganography: Seeing The Unseen*. George Mason University.

Kadir, Abdul. (2008). *Dasar Pemrograman WEB Dinamis Menggunakan PHP*. Yogyakarta: Penerbit Andi.

Manaf, Abdul. (2006). *Penyandian Data Pesan Menggunakan Algoritma Kriptografi AES*. Skripsi Universitas Gadjah Mada.

Mangarae, Aelphaeis. (2006). *Steganography FAQ*. http://www.infosecwriters.com/text_resources/pdf/Steganography_AMangarae.pdf, diakses tanggal 28 Juni 2011.

Menezes, dkk. (1996). *Handbook of Applied Cryptography*. CRC Press.

Munir, Rinaldi. (2006). *Kriptografi*. Bandung: Penerbit Informatika.

Oetomo, B.S.D. 2002. *Perencanaan dan Pembangunan Sistem Informasi*. Yogyakarta: Penerbit Andi.

Pressman, Roger. (1997). *Software Engineering: A Practitioner's Approach*. McGraw-Hill Companies, Inc.

Rafiudin, Rahmat. (2009). *Internet Forensic*.Yogyakarta: Penerbit Andi.

S'to. (2009). *Certified Ethical Hacker 200% Illegal*. Jakarta: Jasakom.

S'to. (2011). *Certified Ethical Hacker 400% Illegal*. Jakarta: Jasakom.

Schneier, Bruce. (1996). *Applied Cryptography 2nd*. John Wiley & Sons, Ltd.

Sommerville, Ian. (2000). *Software Engineering, Rekayasa Perangkat Lunak Edisi 6 Jilid 1*. Jakarta: Penerbit Erlangga.

Sukrisno dan Utami. (2007). *Implementasi Steganografi Teknik EOF dengan Gabungan Enkripsi Rijndael, Shift Cipher, Dan Fungsi Hash MD5*. Yogyakarta: Proceeding Seminar Nasional Teknologi 2007.

Wijaya, dkk. (2004). *Konsep Hidden Message Menggunakan Teknik Steganografi Dynamic Cell Spreading*. Yogyakarta: Jurnal Media Informatika Vol. 2.