e-ISBN 978-623-93343-0-7
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 28 Januari 2020

PENGAMANAN DATA PELAUT MENGGUNAKAN ALGORITMA RIVEST SHAMIR ADLEMAN (RSA) PADA PT. BSM CSC INDONESIA

Mohammad Dimas Arya Wicaksono
Program Studi Informatika
Jl. RS. Fatmawati Raya, Pd. Labu, Kec. Cilandak, Kota Depok, Jawa Barat 12450
dimaszarya@gmail.com

Abstrak. Penelitian ini dilakukan untuk mengamankan data dari pencurian data oleh orang yang tidak bertanggung jawab. Pada hal ini terjadi di lingkungan perusahaan yang bergerak pada bidang pelayaran. Banyak data pelaut yang di ambil oleh orang yang tidak berhak atas data tersebut. Penelitian ini bertujuan untuk melakukan pengamanan pada data pelaut menggunakan algoritma RSA yang berstudi kasus di perusahaan pelayaran bernama PT. BSM CSC Indonesia. Penelitian ini menggunakan metode pengembangan RAD (Rapid Application Development) yang bertujuan untuk menekankan kecepatan pengembangan. Pembuatan aplikasi pengamanan data tersebut menggunakan bahasa pemrograman Java dengan editor Netbeans. Proses penelitian ini terbagi kedalam beberapa tahapan mulai dari pengumpulan data, studi literatur dalam mendukung teori penyelesaian masalah serta Unified Modelling Language (UML) untuk merancang dan pengujian dari aplikasi ini. Hasil yang diperoleh dari penelitian ini adalah dengan adanya pengamanan data akan menghilangkan pencurian terhadap data pelaut didalam ruang lingkup perusahaan oleh pihak yang tidak bertanggung jawab dengan menerapkan proses Kriptografi menggunakan Algoritma RSA.

Kata Kunci: Data Pelaut, Unified Modelling Language, Java, Netbeans dan Algoritma RSA

1 Pendahuluan

1.1 Latar Belakang

Pesatnya perkembangan kemajuan teknologi saat ini memiliki suatu dampak negatif adalah pencurian data. Dari kurangnya kesadaran dan kewaspadaan akan keamanan data dapat menimbulkan celah-celah pencurian data melalui media elektronik sehingga sangat rentan sekali untuk disalahgunakan oleh pihak yang tidak bertanggung jawab. Dengan adanya pencurian data tersebut, maka aspek keamanan data dalam melakukan aktivitas pertukaran dan penyimpanan data informasi amatlah penting agar sifat keaslian dan kerahasiaan pada data tersebut akan tetap terjaga.

Dalam hal teknik pengamanan data, banyak metoda kriptografi yang dapat digunakan. Metode –metode kriptografi tersebut mempunyai teknik dan cara tersendiri. Langkah –langkah pengerjaan setiap metode pun berbeda –beda, baik dari segi panjang maupun kerumitan. RSA (Rivest Shamir Adleman) adalah algoritma kunci publik yang paling populer. Algoritma RSA dibuat oleh tiga orang peneliti dari MIT (Massachussets Institute of Technology), yaitu Ron Rivest, Adi Shamir, dan Len Adleman pada tahun 1976. Jenis keamanan algoritma RSA terletak pada sulitnya memfaktorkan bilangan yang besar menjadi faktor-faktor prima. Sistem keamanan data yang akan dibuat dengan bahasa pemrograman java menggunakan metode kriptografi dengan algoritma RSA sebagai proses enkripsi dan dekripsi file. Metode kriptografi merupakan metode untuk mengamankan sebuah data, baik data tersebut berupa bentuk teks, berupa bentuk angka, maupun berupa bentuk gambar.

1.2 Tujuan Penelitian

Tujuan pada penelitian ini adalah:

a. Membangun aplikasi kriptografi pada data digital untuk mengurangi pencurian isi data secara ilegal.

e-ISBN 978-623-93343-0-7
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 28 Januari 2020

b. Mengimplementasikan algoritma RSA pada isi data digital dengan menggunakan bahasa pemrograman Java dengan editor Netbeans.

1.3 Batasan Masalah

Untuk menghindari terjadinya penyimpangan pada pokok masalah dalam penyusunan penelitian ini maka peneliti memberikan batasan masalah, yaitu:
a. Metode yang digunakan untuk melakukan proses enkripsi dan dekripsi menggunakan algoritma RSA. b.
File yang digunakan untuk aplikasi ini berekstensi *.doc.

2 Landasan Teori

2.1 Kriptografi

Kriptografi menurut Maulana (2012) dapat diartikan sebagai ilmu atau seni untuk menjaga keamanan pesan. Ketika suatu pesan dikirim dari suatu tempat ke tempat lain, isi pesan tersebut mungkin disadap oleh pihak lain yang tidak berhak untuk mengetahui isi pesan tersebut. Untuk menjaga pesan, maka pesan tersebut dapat diubah menjadi sebuah kode yang tidak dimengerti oleh pihak lain. [1]

2.2 Algoritma RSA

Algoritma RSA, ditemukan oleh tiga orang peneliti dari MIT (Massachussets Institute of Technology) pada tahun 1976, yaitu: Ron Rivest, Adi Shamir, dan Leonard Adleman. RSA adalah salah satu teknik kriptografi dimana kunci untuk melakukan proses enkripsi berbeda dengan kunci untuk melakukan proses dekripsi. Kunci untuk melakukan proses enkripsi disebut kunci umum, sedangkan kunci yang digunakan untuk melakukan proses dekripsi disebut kunci rahasia. Kunci umum dapat dimiliki oleh sembarang orang akan tetapi kunci rahasia hanya dimiliki oleh orang yang memiliki hak.

2.3 Pemrograman Java

Menurut Rickyanto (2003) Java adalah suatu teknologi di dunia perangkat lunak komputer. Selain merupakan sebuah bahasa pemrograman, Java juga merupakan suatu platform. Java merupakan teknologi dimana teknologi tersebut mencakup java sebagai bahasa pemrograman yang memiliki sintaks dan aturan pemrograman sendiri. Menurut M. Shalahudin dan Rosa A.S (2010), Java adalah nama sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer yang berdiri sendiri ataupun pada lingkungan jaringan.

3 Hasil dan Pembahasan

Pengujian program bertujuan untuk mengetahui apakah program dapat berjalan dengan baik setelah kebutuhan software maupun hardware sudah terpenuhi untuk di uji coba. File yang akan di uji coba adalah *.doc.
Pengujian dilakukan berdasarkan spesifikasi sistem. Pengujian ini diuraikan menjadi empat faktor pengujian sebagai berikut
a. Kesesuaian Proses
Pengujian terhadap proses dilakukan sesuai dengan use case yang dirancang untuk mengetahui apakah sistem dapat melakukan proses enkripsi dan dekripsi. Cara pengujiannya adalah menjalankan aplikasi dan melihat proses enkripsi dan dekripsi berjalan sesuai use case dan hasilnya adalah sebagai berikut :

Tabel 1 Kesesuaian Proses Generate Key

No	Rancangan Proses	Hasil yang Diharapkan	Hasil
1	Membuka Aplikasi	Menampilkan halaman menu *generate key*	Menu utama *generate key* tampil
2	Memasukkan nama kunci yang akan dibuat	Aplikasi akan membuat kunci sesuai dengan nama kunci	Menghasilkan nama kunci yang sesuai

e-ISBN 978-623-93343-0-7
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 28 Januari 2020

3	*Generate key*	Tampilan *message box* bahwa kunci telah berhasil dibuat	Kunci berhasil dibuat
4	Simpan kunci	Kunci berhasil disimpan	Menampilkan kunci *public* dan *private*
5	Membuka kunci	Kunci hasil dari *generate key* terlihat perbedaannya	Menghasilkan kunci *public* dan *private* yang berbeda

Tabel 2 Kesesuaian Proses Enkripsi

No	Rancangan Proses	Hasil yang Diharapkan	Hasil
1	Membuka Aplikasi	Menampilkan halaman menu enkripsi	Halaman menu enkripsi tampil
2	Memasukkan dokumen (*plaintext*)	Aplikasi dapat mengidentifikasi dokumen	Dokumen berhasil dimasukkan
3	Memasukkan kunci *public*	Dapat mengenkripsi dokumen dengan kunci *public* yang telah dimasukkan	Dokumen akan dienkripsi menggunakan kunci *public*
4	Enkripsi dokumen	Tampilan *message box* bahwa dokumen berhasil dienkripsi	Dokumen telah berhasil dienkripsi
5	Simpan dokumen (*ciphertext*)	Dokumen (*ciphertext*) berhasil disimpan	Menampilkan dokumen yang telah dienkripsi (*chipertext)*
6	Membuka dokumen	Dokumen hasil enkripsi menampilkan angka acak	Isi dokumen yang berhasil dienkripsi berubah menjadi angka acak

Tabel 3 Kesesuaian Proses Dekripsi

No	Rancangan Proses	Hasil yang Diharapkan	Hasil
1	Membuka Aplikasi	Menampilkan halaman menu dekripsi	Halaman menu dekripsi tampil
2	Memasukkan dokumen (*ciphertext*)	Aplikasi dapat mengidentifikasi dokumen	Dokumen berhasil dimasukkan
3	Memasukkan kunci *private*	Dapat mendekripsi dokumen dengan kunci	Dokumen akan didekripsi

e-ISBN 978-623-93343-0-7
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 28 Januari 2020

		private yang telah dimasukkan	menggunakan kunci *private*
4	Dekripsi dokumen	Tampilan *message box* bahwa dokumen berhasil didekripsi	Dokumen telah berhasil didekripsi
5	Simpan dokumen (*plaintext*)	Dokumen (*plaintext*) berhasil disimpan	Menampilkan dokumen yang telah didekripsi (*plaintext)*
6	Membuka dokumen	Dokumen hasil dekripsi menampilkan dokumen awal (asli)	Isi dokumen yang berhasil didekripsi kembali seperti semula (asli)

1. Kesesuaian Data

Pengujian terhadap kesesuaian data dilakukan untuk mengetahui apakah dokumen berhasil di enkripsi dan di dekripsi. Kriteria pengujian adalah proses enkripsi dan dan proses dekripsi berjalan dan terdapat kesesuaian antara dokumen sebelum di enkripsi dengan dokumen setelah di dekripsi.

Tabel 4 Hasil Enkripsi

Nama	Ukuran	Halaman	Waktu Enkripsi	Hasil Enkripsi
Data Capt	2012KB	7 lembar	19,01 s	Isi dokumen berhasil menjadi data acak
Data Chief Engineer	251KB	4 lembar	4,232 s	Isi dokumen berhasil menjadi data acak
Data Pumpman	74KB	2 lembar	0,459 s	Isi dokumen berhasil menjadi data acak

Tabel 5 Hasil Dekripsi

Nama	Ukuran	Halaman	Waktu Dekripsi	Hasil Dekripsi
en.Data Capt	8874KB	2199 lembar	673,644 s	Isi dokumen berhasil menjadi data awal
en.Data Chief Engineer	1100KB	273 lembar	81,393 s	Isi dokumen berhasil menjadi data awal
en.Data Pumpman	318KB	79 lembar	22,521 s	Isi dokumen berhasil menjadi data awal

Berdasarkan tabel hasil proses kriptografi, proses enkripsi dan proses dekripsi dapat berjalan. Dokumen sebelum di enkripsi dengan dokumen setelah di dekripsi hasilnya sesuai. Ukuran dan banyak halaman dokumen mempengaruhi lamanya proses enkripsi maupun proses dekripsi. Semakin besar ukuran dan banyak halaman dokumen, maka semakin lama proses enkripsi maupun proses dekripsi. Proses enkripsi dan proses dekripsi akan

e-ISBN 978-623-93343-0-7
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 28 Januari 2020

semakin cepat apabila menggunakan komputer dengan spesifikasi yang lebih tinggi. Semua sampel yang digunakan memerlukan waktu lebih lama untuk proses dekripsi dibanding proses enkripsinya.

2. Ukuran File

Pengujian terhadap ukuran dokumen dilakukan untuk mengetahui perbedaan ukuran dokumen sebelum di enkripsi dan setelah di dekripsi. Apabila ukuran dokumen sebelum di enkripsi dan setelah di dekripsi tidak membesar, maka pengujian berhasil.

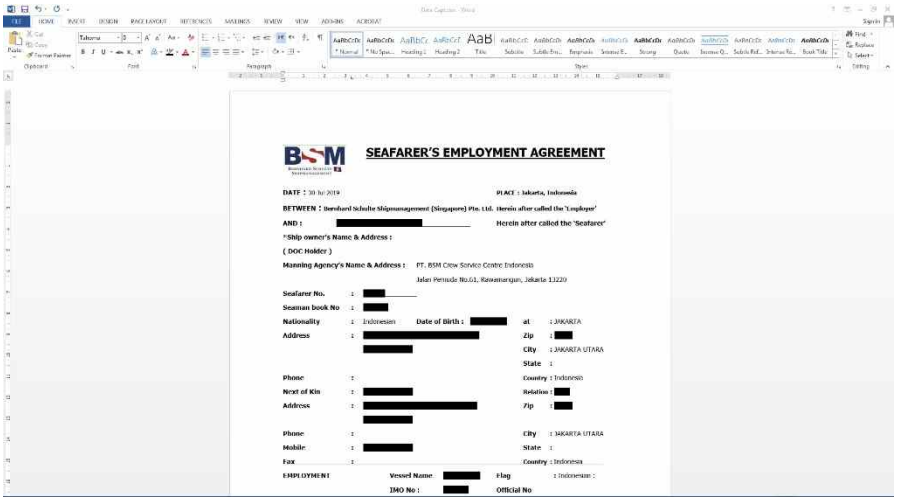
Tabel 6 Pengujian Ukuran dan Halaman Dokumen

	Sebelum Enkripsi		Setelah Enkripsi		Setelah Dekripsi	
Nama File	Ukuran	Halaman	Ukuran	Halaman	Ukuran	Halaman
Data Capt	2012KB	9 lembar	8.874KB	2199	2012KB	9 lembar
Data Chief Engineer	251KB	4 lembar	1100KB	273	251KB	4 lembar
Data Pumpman	74KB	2 lembar	318KB	79	74KB	2 lembar

Berdasarkan tabel pengujian ukuran dan halaman dokumen, semua sampel yang digunakan tidak mengalami perbesaran ukuran maupun jumlah halaman dokumen setelah melalui proses enkripsi kemudian dekripsi.

3. Kualitas Data

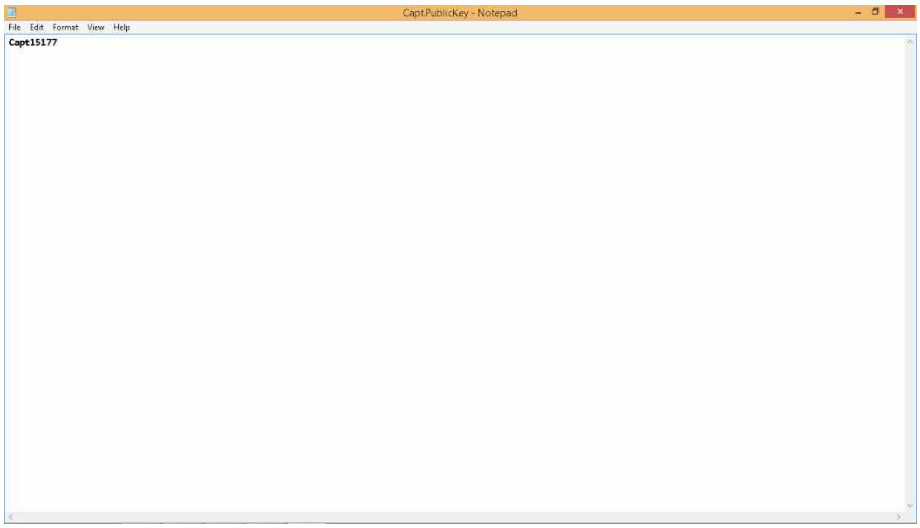
Pengujian kualitas data dilakukan untuk mengukur kualitas data sebelum dan sesudah enkripsi. Kualitas data dengan ekstensi .doc tercermin dari isi tulisan di lembar kerja. Berikut ini tampilan dokumen sebelum enkripsi. (*plaintext)*



Gambar 1 Tampilan Lembar Kerja Dokumen sebelum Enkripsi

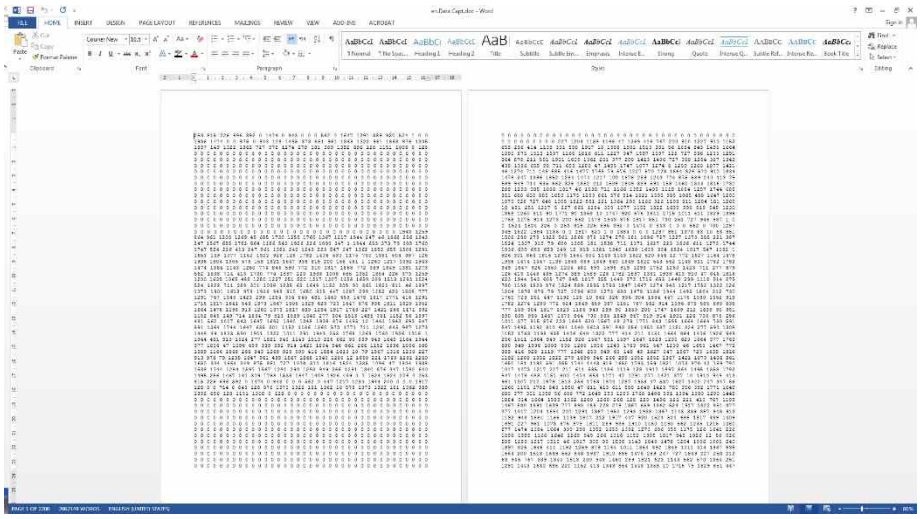
Untuk melakukan proses enkripsi, diperlukan kunci *public* dengan tipe *keyfile* yang didapat dari proses *Generate Key*. Berikut merupakan kunci *public*.

e-ISBN 978-623-93343-0-7
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 28 Januari 2020



Gambar 2 Tampilan Public Key

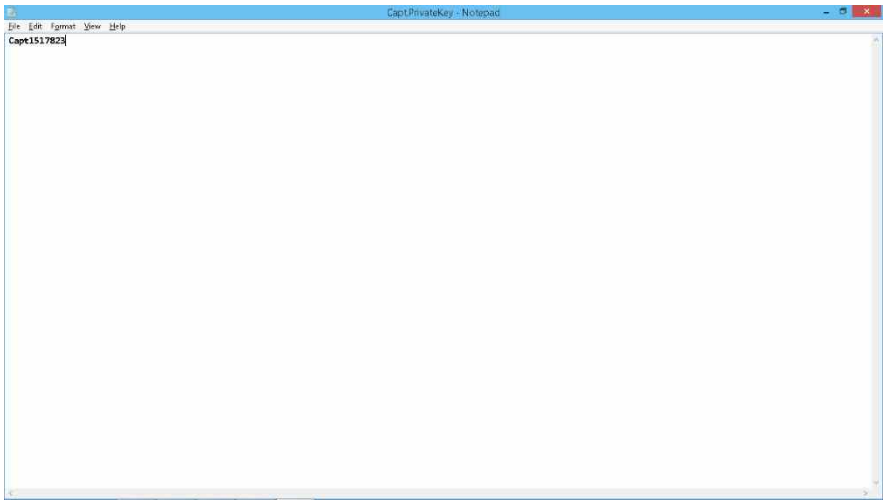
Setelah melakukan enkripsi dokumen, tampilan lembar kerja dokumen berubah menjadi angka bilangan prima. Berikut ini tampilan dokumen setelah melakukan enkripsi. (*chipertext)*



Gambar 3 Tampilan Lembar Kerja Dokumen Setelah Enkripsi

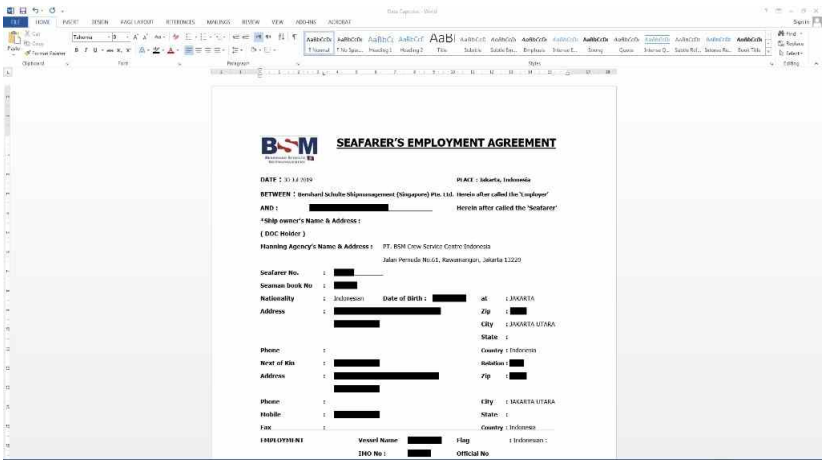
Data yang telah melalui proses enkripsi dilakukan proses dekripsi. Untuk melakukan proses dekripsi dibutuhkan kunci *private* yang didapat dari proses *generate key.*

e-ISBN 978-623-93343-0-7
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 28 Januari 2020



Gambar 4 Tampilan Private Key

Setelah melakukan dekripsi dokumen, tampilan lembar kerja dokumen berubah menjadi kembali seperti awal (asli). Berikut merupakan tampilan dokumen setelah melakukan dekripsi. (*plaintext)*



Gambar 5 Tampilan Lembar Kerja Dokumen Setelah Dekrips

e-ISBN 978-623-93343-0-7
Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)
Jakarta-Indonesia, 28 Januari 2020

4 Penutup

4.1 Kesimpulan

Setelah melakukan pembahasan secara teoritis, implementasi, dan pengujian aplikasi, penelitian ini berhasil menerapkan algoritma RSA untuk melakukan proses enkripsi dan dekripsi pada file format .doc dengan kesimpulan sebagai berikut :

a. Dengan adanya pengamanan data akan menghilangkan pencurian terhadap informasi dari data pelaut didalam ruang lingkup perusahaan oleh pihak yang tidak bertanggung jawab dengan menerapkan proses Kriptografi menggunakan Algoritma RSA.
b. Berkas yang telah dienkripsi (cipherteks) berubah menjadi berkas yang tidak dapat diterjemahkan secara langsung dengan ukuran lebih besar dari ukuran berkas orisinil. Hal ini dikarenakan RSA menggunakan pasangan kunci asimetris.
c. Dokumen digital dengan format .doc di enkripsi menggunakan algoritma RSA sehingga menghasilkan file yang terenkripsi dan dapat di dekripsi kembali melalui tahapan yang sama seperti proses enkripsi. Berkas tulisan dan gambar cipherteks dapat dikembalikan menjadi berkas orisinil menggunakan kunci yang telah disesuaikan. Pengamanan Data pelaut dengan menggunakan algoritma RSA cocok digunakan pada PT BSM CSC Indonesia.
d. File Dokumen .doc yang telah melalui proses enkripsi tidak dapat dibuka, namun apabila dapat dibuka hanya berupa bilangan prima dan ketika telah dilakukan proses dekripsi menghasilkan isi yang sama dengan file dokumen yang asli.

Referensi

1. Aji Supriyanto (2005). Pengantar Teknologi Informasi. Jakarta: Salemba Empat.
2. Dony Ariyus (2008). Pengantar Ilmu Kriptografi Teori Analisis dan Implementasi. Yogyakarta: Andi.
3. Eko Arryawan (2010). Anti Forensik Mengatasi Investigasi Komputer Forensik. Jakarta: Elex Media
4. Komputindo.
5. Fowler Martin (2005). UML *Distilled* Panduan Singkat Bahasa Pemodelan Objek Standar.Yogyakarta: Andi. Frans Thamura (2004). Netbeans *Open Source* Java IDE berbasiskan Swing. Jakarta: Media Elex Komputindo. Hartono (2013). Sistem Informasi Manajemen Berbasis Komputer. Jakarta: Rineka Cipta.
6. Maulana (2012). Penerapan Algoritma WAKE Pada Aplikasi Chatting & Internet Monitor Berbasis Lan. Yogyakarta: STMIK Amikom Yogyakarta.
7. Mollin (2007). *An Introduction to Cryptography*. Newyork: Taylor and Francis Group. Munawar (2005). Pemodelan Visual dengan UML.Yogyakarta: Graha Ilmu.
8. Munir, Rinaldi (2006). Kriptografi. Bandung: Informatika.
9. Prayudi (2005). Studi dan Analisis Algoritma Rivest Code 6 Dalam Enkripsi dan Dekripsi Data. Yogyakarta: Universitas Islam Indonesia.
10. Pressman (2010). Software Engineering. Newyork: McGraw-Hill.
11. Rickyanto (2003). Dasar Pemrograman Berorientasi Objek dengan Java 2. Yogyakarta: Andi. Schneider (2006). *Practical Cryptography.* Indiana Polis: Wiley Publishing.
12. Shalahudin, M. (2010). Rekayasa Perangkat Lunak. Bandung: Modula.
13. Sugiyono (2012). Metode Penelitian Kuantitatif Kualitatif dan R&D. Bandung: Alfabeta. Sutabri. (2012). Analisis Sistem Informasi. Yogyakarta: Andi.
14. Wahyudi. (2003). Pengantar Struktur Data & Algoritma. Yogyakarta: Andi