REPOSITOR, Vol. 2, No. 6, Juni 2020, Pp. 781-786
ISSN : 2714-7975
E-ISSN : 2716-1382

Plug-In Pada Microsoft Office Word Untuk Enkripsi Dokumen Dengan Menggunakan Algoritma Kunci Publik RSA

Zamah Sari[*1], Ali Sofyan Kholimi[2], Miftah Faisal Hamdani[3]
[1,2,3]Teknik Informatika/Universitas Muhammadiyah Malang
zamahsari@umm.ac.id[*1], kholimi@umm.ac.id[2], faisalexcellent59@gmail.com[3]

Abstrak

Ms office ialah salah satu aplikasi dekstop yang sering digunakan oleh banyak orang diberbagai belahan dunia. Dengan adanya aplikasi ini, pengguna lebih dimudahkan untuk mengetik dokumen yang akan digunakan dalam kehidupan sehari-hari baik itu dokumen biasa ataupun dokumen yang bersifat rahasia. Kriptografi ialah suatu metode keamanan untuk melindungi informasi dengan menggunakan sandi tertentu yang hanya diketahui oleh orang yang berhak mengakses informasi tersebut. Untuk membuat dokumen agar tidak terbaca oleh orang yang tidak berhak, maka salah satu solusi yang dapat digunakan ialah menggunakan perangkat lunak (Plug-in) yang dapat diakses dalam Ms. Office Word yang mana dalam plug-in tersebut telah diimplementasikan sebuah algoritma kriptografi kunci publik Rivest Shamir Adleman (RSA).

Kata Kunci: *Algoritma RSA, Kriptografi, Ms. Office Word, Plug-in*

Abstract

Ms office is one of the desktop applications that are often used by many people in different parts of the world. With this application, users are more easy to type documents that will be used in everyday life whether ordinary documents or documents that are confidential. Cryptography is a security method for protecting information by using a specific password that only those who are eligible to access the information it self. To make the document unreadable by an unauthorized person, one solution that can be used is to use a software (Plug-in) accessible in Microsoft Office Word which in the plug-in has implemented a public key cryptography algorithm Rivest Shamir Adleman (RSA).

Keywords: *RSA Algorithm, Cryptography, Microsoft Office Word, Plug-in*

1. Pendahuluan

Perkembangan teknologi yang semakin pesat telah memudahkan banyak pekerjaan yang dahulu sulit dan membutuhkan waktu yang lama kini menjadi lebih praktis dan cepat. Salah satu contoh ialah aplikasi buatan Microsoft Corporation yaitu Microsoft Office. Dalam hal ini banyak pengguna di berbagai belahan dunia yang telah menggunakan aplikasi Ms. Office, terlebih lagi aplikasi pengolah kata yaitu Ms. Office Word. Yang mana, dahulu banyak orang diharuskan untuk mengetik menggunakan mesin ketik dan menekan setiap huruf dengan keras serta hati-hati karena ketika salah satu huruf saja harus mengulang kembali dari awal. Adapun kini dengan menggunakan aplikasi Ms. Office Word, pengguna lebih dimudahkan. Karena ketika terdapat kesalahan dalam pengetikan pengguna hanya perlu menekan tombol *backspace* di *keyboard* PC ataupun laptop yang pengguna miliki.

Akan tetapi, dengan adanya kemudahan tersebut menyebabkan data yang telah dibuat rentan terhadap suatu perubahan. Dan memberikan suatu kesempatan kepada pihak lain untuk melakukan duplikasi ataupun modifikasi tanpa izin dari pemilik asli untuk berbagai kepentingan. Sehingga diperlukan suatu sistem keamanan yang dapat mengamankan informasi dari pihak–pihak yang tidak berkepentingan. Adapun dari pihak Microsoft sendiri telah memberikan suatu sistem pengaman dalam bentuk *password* untuk membuka ataupun modifikasi data yang ada dalam Ms. Office Word. Namun, sistem tersebut masih bisa diretas oleh aplikasi pihak ketiga yang bisa diunduh secara bebas di internet. Maka dari itu, dalam tugas akhir ini akan diajukan suatu aplikasi *plug-in* berupa kriptografi untuk melindungi data pengguna dengan lebih baik lagi.

Kriptografi ialah suatu cara untuk memastikan bahwa *confidentiality* (kerahasiaan), *authentication* (autentikasi), *integrity* (integritas), *availability* (ketersediaan) dan *identification*

(identifikasi) pengguna data dapat dipertahankan serta keamanan dan privasi data dapat disediakan untuk pengguna [1].

Untuk menentukan suatu algoritma kriptografi yang akan digunakan dalam sistem keamanan data selain mempertimbangkan kekuatan terhadap serangan *cryptanalist* dan *bruteforce* ialah mempertimbangkan kecepatan algoritma tersebut, penggunaan memori ketika algoritma tersebut dijalankan dan ukuran file yang telah dienkripsi maupun didekripsi. Pada saat ini terdapat berbagai macam algoritma kriptografi simetri maupun asimetri. Jika suatu algoritma kriptografi dipercaya kuat namun diketahui lambat dalam proses enkripsi maupun dekripsinya, maka algoritma tersebut tidak akan dijadikan suatu pilihan oleh pengguna. Dalam tugas akhir ini, penulis akan menggunakan algoritma kunci publik RSA. sebagaimana tercantum dalam beberapa jurnal [2][3][4][5] algoritma tersebut layak untuk diimplementasikan. Selain itu, algoritma RSA sendiri merupakan contoh yang baik dan aman dari macam-macam kriptografi kunci publik [6].

2. Metode Penelitian

2.1 Pembangkitan Kunci RSA

Adapun langkah–langkah pembangkitan kunci algoritma kunci publik RSA ialah sebagai berikut:

1. Memilih dua buah bilangan prima, dalam hal ini kita menyebutnya dengan variabel *p* dan *q* (rahasia)
2. Menghitung variabel *n* dengan rumus $n = p \,.\, q$
3. Menghitung variabel ϕ*n* dengan rumus $\phi n = (p\text{-}1) \,.\, (q\text{-}1)$
4. Selanjutnya kita memilih sebuah bilangan bulat yang kita beri nama variabel *e* dengan syarat, variabel *e* tersebut harus relatif prima terhadap ϕ*n* yang telah kita ketahui hasilnya tadi.
5. Kemudian kita mencari variabel *d* dengan rumus persamaan $e.d \equiv 1 \pmod{\phi(n)}$ atau $d \equiv e^{-1} \bmod (\phi(n))$
6. Setelah semua variabel diatas telah diketahui, maka dapat dihasilkan dua buah kunci. Untuk kunci publik ialah pasangan (*e, n*) dan kunci privat ialah pasangan (*d, n*)

2.2 Metode Enkripsi RSA

Adapun langkah – langkah untuk mengenkripsi pesan:

1. Pesan diubah dari kode ASCII (*char*) menjadi kode ASCII (*decimal*) terlebih dahulu
2. Kemudian kode tersebut dipecah menjadi beberapa bagian blok *plaintext* (m_1, m_2, m_3, ... m_i) dengan syarat besaran dari setiap blok ialah antara 0 sampai *n*-1 atau ($0 < m_i < n\text{-}1$)
3. Hitung setiap blok *plaintext* dengan persamaan $c_i = m_i^{\,e} \bmod n$
4. Kemudian setiap blok c_i digabungkan, maka jadilah *chipertext* dari algoritma kunci publik RSA.

2.3 Metode Dekripsi RSA

Adapun proses untuk mendekripsi pesan:

1. Hitung setiap blok c_i dengan persamaan $m_i = c_i^{\,d} \bmod n$
2. Kemudian gabungkan setiap blok m_i
3. Lalu mengubah setiap kode ASCII (*desimal*) kembali menjadi pesan yang bisa dibaca oleh manusia (*char*).

3. Hasil Penelitian dan Pembahasan

3.1 Rancangan Algoritma RSA

Algoritma RSA adalah kriptografi asimetrik yang menghasilkan dua pasang kunci publik dan kunci privat dengan cara membangkitkan kunci yang sesuai, algoritma RSA sendiri saat ini merupakan salah satu standar de facto kriptografi asimetrik [5]. RSA juga menggunakan dua buah kunci yaitu kunci publik dan kunci privat dalam membuat kunci tersebut.

Proses pembangkitan algoritma RSA untuk menghasilkan kunci publik dan kunci privat untuk mengamankan data yang dibuat dengan memenuhi $n = p < q < 2p$. Langkah-langkah pembangkitan kunci sebagai berikut :

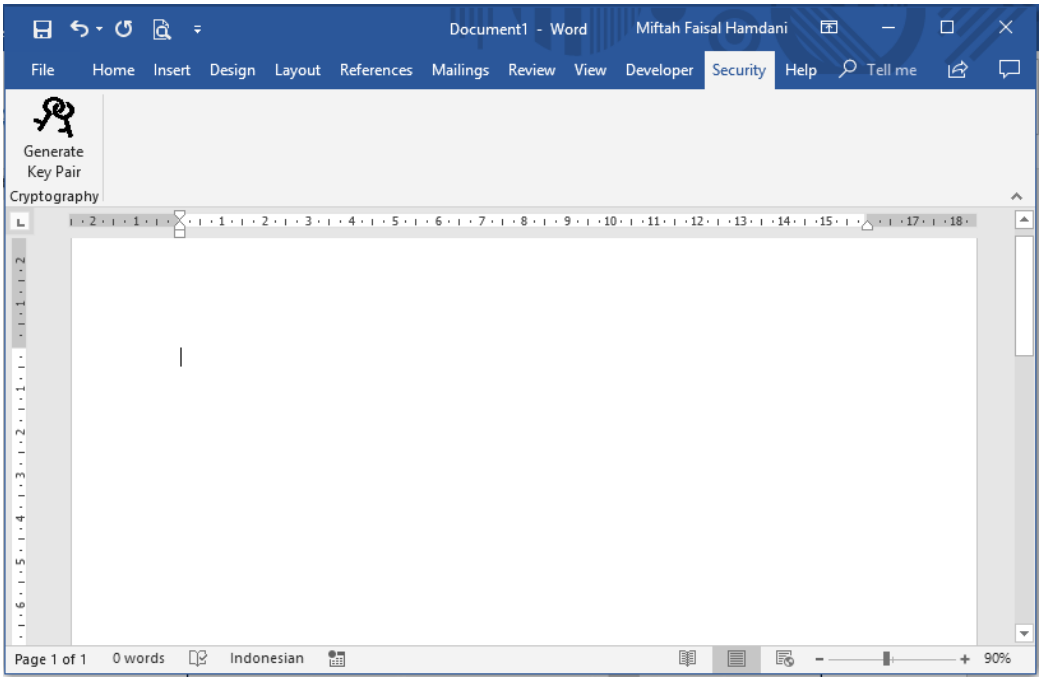
1. Pilih dua bilangan prima p dan q dengan nilai yang berbeda.
2. $n = p * q$ (1)
3. $\phi n = (p - 1)(q - 1)$ (2)
4. FPB (e, ϕn) = 1 (3)
5. $e * d \equiv 1 \pmod{m}$ (4)

Variabel n pada persamaan (1) merupakan kunci publik yang akan di dibangkitkan dengan persamaan bilangan prima *p* dan bilangan prima *q* untuk memenuhi pembangkitan kunci yang diinginkan. Variabel ϕ*n* pada persamaan (2) merupakan nilai phi-*n* yang dicari dengan mengitung dari persamaan (2). Variabel FPB (*e*, ϕ*n*) pada persamaan (3) merupakan persamaan untuk menentukan kunci publik. Variabel *e* * *d* dari persamaan (4) adalah untuk mencari kunci privat.

3.2 Implementasi

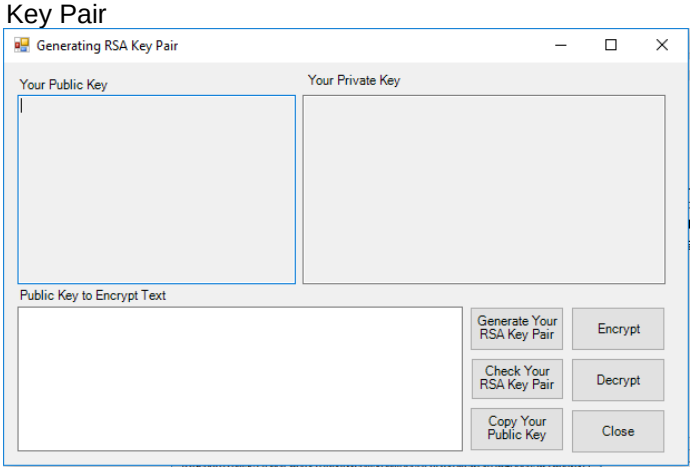
Setelah mengimplementasikan algoritma RSA kedalam aplikasi *plug-in* pada Ms. Word dengan menggunakan pemrograman berbasis C#, maka didapatkan hasil interface seperti pada Gambar 1 dan Gambar 2.

1. Tampilan Utama *Plug-in*



Gambar 1. Tampilan Utama Plug-in Kriptografi Kunci Publik RSA yang Telah Terintegrasi Dengan Microsoft Office Word.

2. Form Generate Key Pair



Gambar 2. Tampilan Ketika User meng-klik Button Generate Key Pair

Adapun kegunaan dari beberapa tombol pada gambar diatas ialah sebagai berikut:

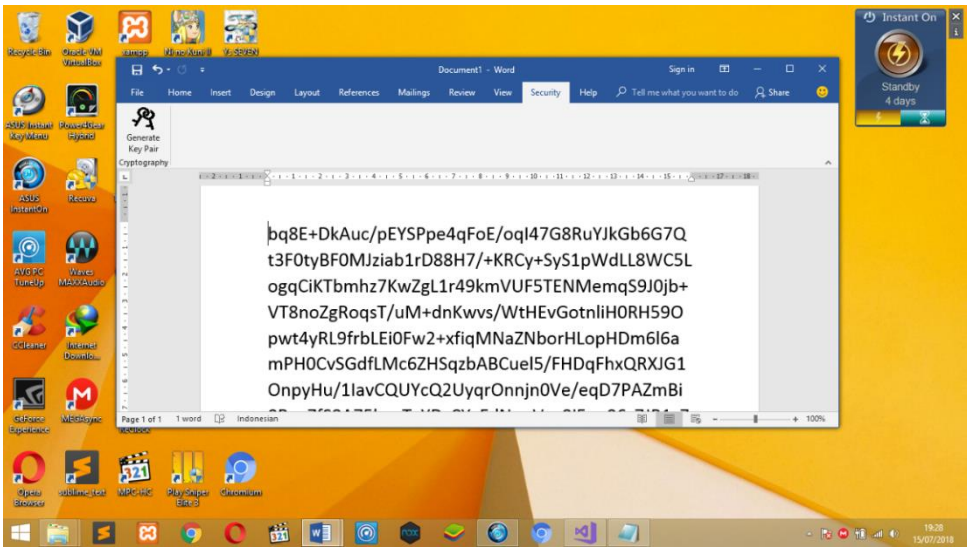
1. *Button Generate Your RSA Key Pair* ialah tombol untuk memperbaharui *Key Pair* yang digunakan untuk enkripsi dan dekripsi teks.

2. *Button Check Your RSA Key Pair* ialah tombol untuk mengecek *Key Pair* yang telah tersimpan dalam komputer *User*.
3. *Button Copy Your Public Key* ialah tombol untuk meng-*copy Texbox Your Public Key* kedalam *Textbox Public Key to Encrypt Text.*
4. *Button Encrypt* ialah tombol untuk mengenkripsi teks dengan menggunakan *Public Key* yang telah ada.
5. *Button Decrypt* ialah tombol untuk mendekripsi teks dengan menggunakan *Private Key* yang telah ada.
6. *Button Close* ialah tombol untuk keluar dari *Form Generate Key Pair*.

3.3 Skenario Pengujian

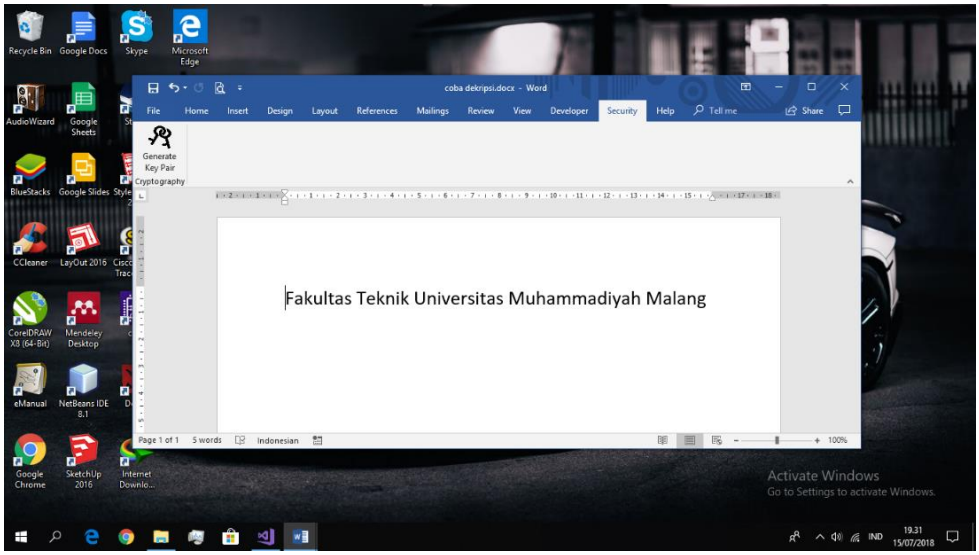
Pengujian dilakukan dengan cara mengirimkan file dokumen Ms. Word yang telah dienkripsi kepada user lain baik dengan melalui *email*, *port* ataupun lainnya, dan membandingkan antara *plaintext* sebelum di enkripsi dan *plaintext* setelah di dekripsi sama atau tidak, seperti pada Gambar 3 dan Gambar 4.

1. Percobaan Enkripsi Teks pada *User1*



Gambar 3. Teks Telah Berhasil di Enkripsi Pada Komputer User2 Dengan Menggunakan Kunci Publik User1

2. Percobaan Dekripsi Teks pada *User1*



Gambar 4. Teks Telah Berhasil di Dekripsi Dengan Menggunakan Kunci Private User1

4. Kesimpulan

Kesimpulan yang dapat diambil dari serangkaian pengujian yang telah dilakukan dapat disimpulkan bahwa:

1. Algoritma kunci publik RSA bisa diimplementasikan pada Ms. Word dengan diimplementasikan kedalam aplikasi berbasis *plug-in*.
2. Karena file dokumen biasanya terdiri dari ratusan bahkan ribuan karakter, maka hal tersebut akan sangat berpengaruh terhadap proses enkripsi dan dekripsi dari algoritma RSA.

Sampai saat ini penulis masih hanya sampai di enkripsi dan dekripsi teks file. Mungkin diperlukan pengembangan lebih lanjut yang bertujuan untuk memaksimalkan proses enkripsi dan dekripsi teks dan gambar pada *microsoft office word*.

Referensi

[1] Bhanot, R., & Hans, R. (2015). A Review and Comparative Analysis of Various Encryption Algorithms. *International Journal of Security and Its Applications, 9*, 289-306.

[2] Caroline, M. L. (2010/2011). Perbandingan Algoritma Kriptografi Kunci Publik RSA, Rabin dan Elgamal. *Intitut Teknologi Bandung*.

[3] Shetty, A., K, S. S., & K, K. (2014). A Review on Asymetric Cryptography - RSA and ElGamal Algorithm. *International Journal of Innovative Research in Computer and Communication Engineering*, 98-105.

[4] Mulya, M. (2013). Perbandingan Kecepatan Algoritma Kriptografi Asimetri. *Journal of Research in Computer Science and Aplications*, 7-12.

[5] Farah, S., Javed, M. Y., Shamim, A., & Nawaz, T. (2012). An Experimental Study on Performance Evaluation of Asymetric Encryptions Algorithms. *Recent Advances in Information Science*, 121-124.

[6] Listiyono, H. (2009). Implementasi Algoritma Kunci Publik Pada Algoritma RSA. *Dinamika Informatika, I*, 95-99.

ISSN: 2714-7975; E-ISSN: 2716-1382