Makalah UAS Kriptografi – IF 3054

Studi dan Implementasi Algoritma Kunci Publik RSA pada Aplikasi Web Messenger Berbasis Ajax

Oleh : Hadyan Ghaziani Fadli – 13505005

Abstrak

Makalah ini membahas tentang studi dan implementasi Algoritma kunci publik RSA pada Sebuah perangkat lunak / aplikasi web berbasis yang dibuat dengan menggunakan AJAX. Seperti yang dapat kita perhatikan, penggunaan aplikasi messenger tidak lagi dilakukan dengan menggunakan aplikasi desktop, tetapi telah merambah ke berbagai macam aplikasi baik yang berdiri sendiri ataupun yang menyatu dengan aplikasi lain seperti aplikasi messenger pada sebuah aplikasi web yang menggunakan AJAX. AJAX merupakan teknologi pemrograman aplikasi berbasis web dengan menggunakan Javascript dan XML.

Seperti aplikasi messenger pada umumnya, didalam sebuah jaringan , data percakapan yang akan dikirimkan lewat internet akan melalui router yang menjadi penghubung antara dunia luar dengan jaringan itu sendiri. Karenanya, tindak kejahatan seperti serangan "Man In The Middle" sulit untuk dihindari. Oleh karena itu, diperlukan suatu mekanisme untuk mengamankan isi pesan tersebut dalam hal ini adalah penggunaan kriptografi untuk mengenkripsi pesan sebelum keluar dari komputer user.

Inti dari aplikasi ini adalahmengenkripsi setiap pesan yang akan dikirimkan dengan memanfaatkan kunci publik penerima sebelum pesan keluar dari Komputer pengirim dengan menggunakan fungsi javascript. Artinya, pesan yang dikirimkan lewat jaringan sudah berupa cipherteks. Kemudian saat pesan telah sampai di pihak penerima, pesan tersebut akan didekripsi kembali menjadi pesan yang dapat dimengerti dengan kunci privat si penerima sesuai algoritma RSA. Dengan demikian diharapkan pihak yang tidak berhak tidak dapat mengerti isi pesan dari percakapan tersebut.

***Kata kunci**: RSA*, Javascript, XML, AJAX, cipherteks, enkripsi, dekripsi, *Man In The Middle*.

1. Pendahuluan

Di zaman sekarang, aplikasi komunikasi *real time* untuk melakukan *chatting* tidak lagi hanya sebatas pada aplikasi berbasis aplikasi desktop, tetapi merambah ke aplikasi di berbagai tempat seperti aplikasi mobile karena telah berkembangnya platform untuk membangun aplikasi disini dan berkembangnya teknologi web sehingga memungkinkan untuk melakukan komunikasi *real time* dengan menggunakan aplikasi berbasis web.

Salah satu jenis teknologi yang dipakai sekarang dalam membangun komunikasi *real time* di dalam aplikasi web, adalah teknologi ajax atau *asynchronous JavaScript and XML*. Di masa kini, telah banyak aplikasi messenger berbasis web dengan menggunakan teknologi ajax, namun masih banyak kemungkinan untuk terjadinya kejahatan seperti *Man In The Middle Attack*.

2. Dasar Teori

2.1 Kriptografi Kunci-Publik

2.1.1. Deskripsi

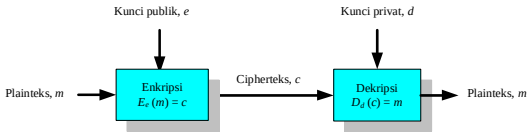
Sampai akhir tahun 1970, hanya ada sistem kriptografi kunci-simetri. Satu masalah besar dalam sistem kriptografi: bagaimana mengirimkan kunci rahasia kepada penerima? Mengirim kunci rahasia pada saluran publik (telepon, internet, pos) sangat tidak aman. Oleh karena itu, kunci harus dikirim melalui saluran kedua yang benar-benar aman. Saluran kedua tersebut umumnya lambat dan mahal.

Ide kriptografi kunci-nirsimetri (*asymmetric-key cryptography*) muncul pada tahun 1976 Makalah pertama perihal kriptografi kunci-publik ditulis

oleh Diffie-Hellman (ilmuwan dari Stanford University) di IEEE Judul makalahnya "*New Directions in Cryptography*".Namun pada saat itu belum ditemukan algoritma kriptografi kunci-nirsimetri yang sesungguhnya.

Kriptografi kunci-nirsimtri disebut juga kriptografi kunci-publik Pada kriptografi kunci-publik, masing-masing pengirim dan penerima mempunyai sepasang kunci:

Kunci publik: untuk mengenkripsi pesan $E_e(m) = c$
Kunci privat: untuk mendekripsi pesan. dan $D_d(c) = m$



2.2 Algoritma Kriptografi Kunci-Publik RSA

2.2.1 Deskripsi

RSA merupakan salah satu algoritma kunci-publik yang paling populer diimplementasikan pada berbagai aplikasi. Algoritma RSA diciptakan pada tahun 1976 oleh tiga orang peneliti dari *Massachussets Insititute of Technology* (MIT), yaitu *Ron Rivest, Adi Shamir* dan *Leonard Adleman*. Nama RSA diambil dari gabungan inisial nama belakang ketiga penemunya tersebut. Sulitnya untuk memfaktorkan bilangan besar menjadi faktor-faktor prima dimanfaatkan oleh algoritma RSA untuk menjamin keamanan algoritma kriptografi ini.

2.2.2 Pembangkitan Pasangan Kunci

RSA melibatkan kunci-publik dan kunci-privat dalam proses enkripsi dan dekripsinya. Kunci-publik sifatnya tidak rahasia dan boleh diketahui oleh orang lain dan dipergunakan untuk mengenkripsi pesan. Pesan yang dienkripsi dengan kunci-publik penerima dapat didekripsi dengan mempergunakan kunci-privat penerima . Pembangkitan kunci untuk algoritma RSA dilakukan dengan cara berikut:
1. Pilih dua buah bilangan prima sembarang, *p* dan *q*.
2. Hitung $n = p.q$ (sebaiknya $p \neq q$, sebab jika $p = q$ maka $n = p^2$ sehingga *p* dapat diperoleh dengan menarik akar pangkat dua dari *n*).
3. Hitung $\phi(n) = (p - 1)(q - 1)$.
4. Pilih kunci public, *e*, yang relatif prima terhadap $\phi(n)$.
5. Bangkitkan kunci privat dengan menggunakan persamaan $e.\ d \equiv 1 \pmod{\phi(n)}$.

Algoritma di atas akan menghasilkan pasangan kunci yaitu:
1. Kunci-publik adalah pasangan (*e*, *n*).
2. Kunci-privat adalah pasangan (*d*, *n*).

2.2.3. Algoritma Enkripsi/Dekripsi Pesan

Berikut adalah algoritma enkripsi pesan yang dilakukan dalam algoritma kriptografi RSA:

1. Ambil kunci-publik penerima pesan, yaitu e dan modulus n.
2. Nyatakan plainteks *m* menjadi blok-blok $m_1, m_2, \ldots, m_n$ sedemikian sehingga setiap blok merepresentasikan nilai di dalam selang $[0, n-1]$.
3. Setiap blok m, dienkripsi menjadi blok c_i dengan rumus $c_i = m_i^e \bmod n$.

2.3 Ajax

2.3.1 Deskripsi

Ajax atau asynchronous JavaScript + XML adalah grup dari teknik web development yang digunakan untuk membuat aplikasi web yang interaktif.
Dengan Ajax, aplikasi web dapat menerima data dari server secara asinkronus dalam background tanpa menginterferansi dengan display dan page yang tersedia.Data diterima dengan menggunakan XMLHttpRequest object.
Jesse James Garrett memikirkan konsep Ajax ketika sedang berada di shower, ketika ia menyadari keperluan sebuah cara singkat untuk merepresentasikan teknologi yang diajukannya kepada klien. Seperti pada java language pada tahun 1995 yang dapat mempersilahkan compiled client-side code untuk me-load data secara asinkron dari web server sete;aj sebuah page di load.
Tahun 1996 Internet Explorer mengenalkan IFrame element ke HTML, tahun 1999 Microsoft menciptakan XMLHTTPActiveX control di IE 5 yang sekarang juga di support oleh browser lainnya sebagai XMLHttpRequest Object.5 April 2006, World Wide Web Consortium (W3C) merilis draft pertama spesifikasi object untuk dijadikan web standart.

3. Implementasi

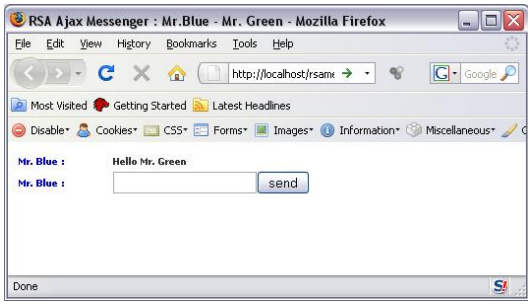
3.1 Skenario

Skenario yang terjadi pada aplikasi ini adalah kedua klien melakukan login pada aplikasi, kemudian klien pertama memilih lawan bicaranya, server menyampaikan permintaan klien pertama dan jika klien kedua mengabulkan permintaan tersebut, maka tiap klien menciptakan kunci publik dan privatnya dalam hal ini kunci di-generate oleh sistem ,kemudian kunci publik dari kedua klien dipertukarkan. Kemudian komunikasi dapat dimulai, setiap klien akan mengirim pesannya, fungsi javascript pada klien tersebut akan mengenkripsi pesan tersebut dengan kunci publik lawan bicaranya, sehingga pesan yang sampai di kirim via server adalah pesan terenkripsi. Kemudian pada sisi lawan bicara, pesan tersebut akan didekripsi dengan kunci privat penerima.

3.2 Antarmuka dan skenario pengujian

Pada percakapan ini ada 2 klien yaitu Mr. Blue sebagai pihak pertama dan Mr. Green sebagai pihak kedua. Berikut ini adalah penjelasan skenario dengan antar muka :

a. Mr. Blue memulai percakapan dengan Mr. Green maka sistem akan men-generate kunci publik dan privat untuk keduanya di computer masing-masing dengan fungsi javascript, pada hal ini sebagai contoh hanya ditampilkan kunci milik Mr. Green karena Mr. Blue yang akan mengirim pesan ke Mr. Green, kunci yang digunakan berukuran 1024 Bit heksadesimal.



Kunci publik Mr. Green :

- Parameter n :
  130ebebd67b16a9ab2c53a437badb
  f8f01a80c750095a7fcfe95742c3d5
  ed1abb318babc5cb5d9350fee4da65
  ee074f65e1758117e6945f0fcfc813
  7528053ce9d1da8618890dee24e5e
  0bf8c87795bb1d09eddd544640824
  ee0dd0ea9fd908d27b0f8a1ae5c37f
  3647fbf2f5795500ad76c195b3387d
  0458a8f51b701472301

- Parameter e : 10001

Kunci privat Mr. Green :

- Parameter n :
  130ebebd67b16a9ab2c53a437badb
  f8f01a80c750095a7fcfe95742c3d5
  ed1abb318babc5cb5d9350fee4da65
  ee074f65e1758117e6945f0fcfc813
  7528053ce9d1da8618890dee24e5e
  0bf8c87795bb1d09eddd544640824
  ee0dd0ea9fd908d27b0f8a1ae5c37f
  3647fbf2f5795500ad76c195b3387d
  0458a8f51b701472301

- Parameter d :
  12e8da920d4599458e84ec5ef1656
  161807f427d05eb79182b7418259d
  6f6c14364d1f5caf9130c8d9d9d6ea
  71d1bdbc87781a46a16bcb9e67281
  4fed3b9c96ddffe0a1b0955ae68055
  c8f92fef518a04fc32a2ea8390e617c
  c5556a251f9ae9eee70a32e579cb3e
  9f298848a9b3aaf634f5930ffbf7447
  3f7cb6c0cefee1751

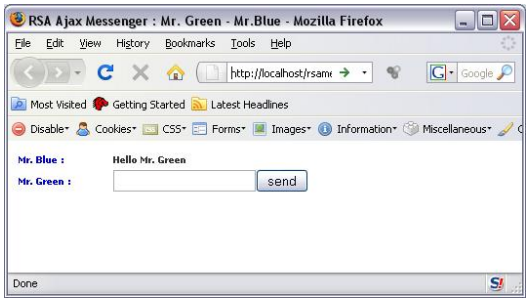
b. Sistem di komputer masing-masing akan mengirim kunci publiknya untuk dipertukarkan

c. Ketika Mr. Blue menuliskan pesan sebagai berikut :

Teks yang ditulis oleh Mr. Blue adalah “Hello Mr. Green”, tetapi ketika dikirimkan, teks yang terkirim setelah dienkripsi menggunakan kunci publik Mr. Green adalah :

“093cba4c88197a8742403bf463feb5061a92b1e6
a797baba121b76d27dca0c0000c0b2877a0dd403
c8d11f1f20181c032a6893d18fdecd6c1271c234c
f22034fdf0ec05de50f02e1fc2865119eb3f4df506
ac75247a3842a21107eeea89b0d4887fa83e1bd36
5e6f13017d3150c422926380e02af635da001634
e0faeb4b31fd”

d. Kemudian pesan yang dikirikan didekripsi dengan kunci privat milik Mr. Green dan kembali menjadi tulisan “Hello Mr. Green”.



4. Kesimpulan

Keberadaan algoritma kunci publik dan kunci privat membuat keamanan pesan menjadi lebih aman sekaligus menjadi sangat murah karena tidak perlu membuat saluran dedicated untuk 2 klien dalam melakukan komunikasi.
Dengan aplikasi web messenger yang dikembangkan dengan menggunakan Ajax, kini para konsumen dari layanan ini yang pada umumnya adalah masyarakat biasa, tidak perlu lagi khawatir untuk adanya penyadapan.
Selain itu dengan menggunakan bilangan prima yang besar, proses dekripsi menggunakan brute force akan memakan waktu lama.

5. DAFTAR PUSTAKA

[1] Munir, Rinaldi. (2004). Bahan Kuliah IF5054 Kriptografi. Departemen Teknik Informatika, Institut Teknologi Bandung.

[2] RSA Algorithm *http://www.en.wikipedia.org* Tanggal akses: *Wikipedia.org.* http://en.wikipedia.org. *Tanggal akses 5 Mei 2009.*

[3] Ajax tutorial http://www.w3school.com Tanggal akses: *Wikipedia.org.* http://en.wikipedia.org. *Tanggal akses 5 Mei 2009.*