Seminar Nasional Teknologi Komputer & Sains (SAINTEKS)
SAINTEKS 2020
ISBN: 978-602-52720-7-3
Februari 2020
Hal 426-428

Pengamanan Transfer File Menggunakan Secret Sharing Asmuth-Bloom

Imam Saputra[1], Nyata Manalu[1], Azlan[2]

[1] Program Studi Teknik Informatika, STMIK Budi Darma, Medan Indonesia
[2] STMIK Triguna Dharma, Medan Indonesia
Email: saputraimam69@gmail.com[1],nyatamanalu11@gmail.com[1], azlansaja@gmail.com[2]

Abstrak−File merupakan kumplan data-data yang telah dioalah untuk menjadi sebuah informasi. File dapat bersifat pribadi maupun bersifat umum tergantung pada pemiliknya. File yang bersifat pribadi merupakan file yang bersifat rahasia yang mana hanya dapat diketahui oleh pemiliknya. Namun, kadang kala file pribadi ini tidak akan terjamin keamanannya akibat lemahnya keamanan data (*security*) data. Oleh sebab itu, perlu adanya dibuat sebuah system yang dapat mengamankan file tersebut apalagi file tersebut tidak boleh diketahui oleh umum. Untuk itu kriptografi sangat membantu dalam hal ini. Dengan menerapkan metode yang ada dalam kriptografi memungkinkan keamaman dan kerahasiaan file akan terjamin. Salah satu metode yang cocok untuk digunakan dalam mengamankan file yaitu metode *secret sharing asmuth-bloom*. Metode ini merupakan metode yang sangat cukup kuat dalam mengamankan file, dimana metode ini sulit untuk dibobol dan dihancurkan oleh pihak-pihak yang tidak bertanggung jawab, yang mencuri file dan menggunakannya untuk sesuatu hal yang buruk.

Kata Kunci: Pengamanan, *File, Secret Sharing Asmuth-Bloom*, Kriptografi

1. PENDAHULUAN

Informasi murupakan data-data yang telah diproses sehingga dapat diolah dan dibaca. Informasi dapat tergolong dalam dua bentuk, yaitu informasi untuk konsumsi umum dan pribadi. Data pribadi kadang kala menjadi data rahasia pribadi. Kadang kala, data pribadi yang bersifat rahasia sering disalah gunakan oleh orang lain, bahkan dibuat menjadi konsumsi umum. Untuk itu, data tersebut memerlukan pengaman yaitu dengan menerapkan kriptografi dalam mengamankan data

Kriptografi di dalanya terdiri dari beberapa metode dan agoritma, yang memang dikhususkan dalam melindungi data, baik itu gambar, text dan dokumen lainnya yang memang dianggap rahasia. contohnya dengan menggunakan algoritma *secret sharing.* Agorita ini, memungkinkan data rahasia dapat tersimpan dengan aman sebab menggunakan penyandian tertentu untuk meindungi data.

Dalam peneitian ini, akan dibahas bagaimana menerapkan agoritma *secret sharing* dalam mengamankan data. Dilihat dari penelitian sebelumnya bahwa algoritma ini cukup mampu dalam mengamankan data.

2. METODOLOGI PENELITIAN

2.1 Kriptografi

Ilmu yang mempelajari bagaimana cara menjaga agar data atau pesan tetap aman saat dikirimkan dari pengirim ke penerima tanpa mengalami gangguan dari pihak ketiga disebut kriptografi. Algoritma yang sering digunakan dalam kriptografi yaitu algoritma simetris dan algoritma asimetris. Algoritma simetris ini merupakan algoritma yang mana kunci enkripsi yang digunakan sama dengan kunci dekripsi. Sedangkan algoritma asimetris merupakan suatu algoritma yang mana kunci enkripsi yang digunakan tidak sama dengan kunci dekripsi.

2.2 Secret Sharing Asmuth-Bloom

Secret sharing merupakan salah satu metode untuk mengamankan kerahasiaan suatu data atau file dengan membagi atau mengirim rahasia tersebut menjadi beberapa bagian yang dinamakan *share*, setiap bagian dari rahasia tersebut tidak memberikan informasi apa-apa mengenai rahasia yang dimaksud bila tidak digabungkan dengan bagian yang lainnya. Salah satu alasan adanya secret sharing adalah perlindungan terhadap ancaman kehilangan kunci kriptografi. Data rahasia dibagikan sedemikian rupa sehingga sembarangan m dari n orang ($m \leq n$) memiliki pecahan data rahasia yang cukup untuk mengetahui atau menggabungkan kembali pecahan data rahasia tersebut.

Dalam allgoritma *Secret Sharing,* terdapat beberapa proses kerja yang hampir sama dengan cara kerja dengan algoritma kriptografi konvensional, yang dimana bisa dibagi menjadi 3 bagian seperti berikut ini:

a. Proses pembentukan kunci berguna dalam hal menghasilkan kunci yang akan digunakan saat proses pembentukan dan kombinasi *shadow.*
b. Proses pembentukan bayangan (*shadow*) pecahan pesan memiliki fungsi untuk menghasilkan beberapa bayangan (*shadow*) dari pesan. Proses ini hampir dengan cara kerja enkripsi yang ada pada algoritma kriptografi tentang konvensional.
c. Proses penggabungan bayangan (*shadow*) memiliki fungsi untuk membentuk pesan awal dengan menggunakan sejumlah bayangan (sesuai dengan prosedur yang ditentukan pada proses pembentukan kunci). Proses ini hampir sama dengan proses dekripsi pada algoritma kriptografi tentang konvensional. Tetapi bedaannya adalah dengan algoritma kriptografi konvensional adalah: Algoritma ini bisa menghasilkan lebih dari 1 (satu) buah bayangan (pecahan pesan) yang hampir sama dengan *ciphertext* pada algoritma konvensional.
d. Jumlah *user* yang melakukan proses pembentukan dan pengkombinasian *shadow* hampir sama dengan proses enkripsi dan dekripsi juga bisa berbeda banyaknya.

Di dalam algoritma *Secret Sharing Asmuth-Bloom* ini, ada beberapa fungsi pendukung yang digunakan diantaranya adalah:

ISBN: 978-602-52720-7-3
Februari 2020
Hal 426-428

a. Metode Rabin-Miller biasanya digunakan dalam melakukan proses pengujian dan pembangkitan bilangan prima.
b. *Fast Exponentiation*, biasanya digunakan dalam menghitung nilai dari operasi pangkat modulo bilangan terbesar.
c. Algoritma *Euclidean* biasanya digunakan dalam menghitung nilai GCD antara dua buah bilangan.
d. Algoritma *Extended Euclidean* biasanya digunakan dalam menghitung nilai invers modulo.
e. Teorema *Chinese Remainder*, biasanya digunakan dalam melakukan proses pengkombinasian *shadow*

3. ANALISA DAN PEMBAHASAN

Berikut adalah contoh penerapan algoritma *secret sharing asmuth-bloom*. Akan dikirim sebuah file rahasia kepada seseorang Adapun proses yang dilakukan algoritma *Secret Sharing Asmuth-Bloom* antara lain:

a. Input: misalkan file M = 4, yaitu "Data", jadi isi filenya hanya ada 4, yaitu:
   'D', *ASCII Code* $M_1 = 68$
   'A', *ASCII Code* $M_2 = 65$
   'T', *ASCII Code* $M_3 = 84$
   'A' *ASCII Code* $M_4 = 65$
b. Nilai *threshold scheme* yang akan digunakan: (4, 5) dimana m = 4 dan n = 5.
c. Nilai bilangan prima yang disimbolkan dengan p digunakan = 601.
d. Pilih 4 buah bilangan d_i:
   1. $d_1 = 127$
   2. $d_2 = 509$
   3. $d_3 = 521$
   4. $d_4 = 599$
e. Pilihlah sebuah bilangan random r, contohnya diambil r = 17.
f. Hitung nilai M' untuk setiap karakter pesan :
   1. $M_1' = M_1 + r * p = 69 + 17 * 601 = 10286$
   2. $M_2' = M_2 + r * p = 86 + 17 * 601 = 10303$
   3. $M_3' = M_3 + r * p = 73 + 17 * 601 = 10290$
g. Menghitung pecahan file yang akan didistibusikan untuk satu orang
   $k_{1,1} = M_1' \bmod d_1 = 10286 \bmod 127 = 81$ $k_{1,2} = M_2' \bmod d_1 = 10303 \bmod 127 = 81$ $k_{1,3} = M_3' \bmod d_1 = 10290 \bmod 127 = 81$
   Output: Pecahan pesan untuk orang pertama = 81, 81, 81
   $k_{2,1} = M_1' \bmod d_2 = 10286 \bmod 509 = 20$
   $k_{2,2} = M_2' \bmod d_2 = 10303 \bmod 509 = 20$
   $k_{2,3} = M_3' \bmod d_2 = 10290 \bmod 509 = 20$
h. Cntohnya, saat menggabungkan pesan dipilih, maka proses penggabungan pecahan file yaitu sebagai berikut:
   1. *Input*: Untuk mencari karakter-1:
      $k_{1,1} = M_1 \bmod d_1 = 14 = M_1' \bmod 127$ $k_{2,1} = M_1' \bmod d_2 = 121 = M_1' \bmod 509$
      $k_{3,1} = M_1 \bmod d_3 = 402 = M_1' \bmod 521$
      Nilai M_1 bisa didapat dengan menggunakan bantuan teorema *Chinese Remainder* seperti berikut:
      Soal:
      $X = 14 \pmod{127}$
      $X = 121 \pmod{509}$
      $X = 402 \pmod{521}$
      Cara kerja:
      Persamaan pertama: $X = 14 \pmod{127}$ Maka $x = 14 + 127k_1$ untuk nilai k_1 Substitusi ke dalam kongruen 2 menjadi :
      $X = 121 \pmod{509}$ $14 + 127k_1 = 121 \pmod{509}$ $127k_1 = 107 \pmod{509}$ $k_1 = 107 \pmod{509}$ $127^{-1} \pmod{509}$ $k_1 = 107 \pmod{509}$ $505\ k_1 = 54035 \pmod{509}$ {bentuk tidak valid} $k_1 = 81 \pmod{509}$ {bentuk valid} atau : $k_1 = 81 + 509k_2$ untuk beberapa nilai k_2
      Maka, didapatkan:
      $X = 14 + 127k_1 = 14 + 127(81 + 509k_2) = 14 + 10287 + 64643k_2$
      $X = 10301 + 64643k_2$; yang bisa memenuhi 2 persamaan pertama Substitusi ke dalam kongruen 3 menjadi:
      $X = 402 \pmod{521}$ $10301 + 64643k_2 = 402 \pmod{521}$ $64643k_2 = -9899 \pmod{521}$ {bentuk tidak valid} $64643k_2 = 0 \pmod{521}$ {bentuk valid} $k_2 = 0 \pmod{521}$. $64643^{-1} \pmod{521}$ $k_2 = 0 \pmod{521}$. $334\ k_2 = 0 \pmod{521}$ atau: $k_2 = 0 + 521k_3$ untuk beberapa nilai k_3 Jadi, didapatkan :
      $X = 10301 + 64643k_2 = 10301 + 64643(0 + 521k_3) = 10301 + 0 + 33679003k_3$
      $X = 10301 + 33679003k_3$; yang memenuhi 3 kongruen pertama.
      Berdasarkan hasil perhitungan yang telah dilakukan di atas, solusinya yaitu : $X = 10301$
      Berdasarkan hasil perhitungan di atas, maka diperoleh nilai $M_1' = 10301$.
      Selanjutnya menghitung nilai M_1 dengan cara sebagai berikut:

      M_1' $= M_1 + r *$
      p

ISBN: 978-602-52720-7-3
Februari 2020
Hal 426-428

10286 = M_1 + 10217
M_1 = 10286 – 10217
Output: M_1 = 68 → karakter 'D'

i. *Output*: Dari hasil perhitungan M_1, M_2 dan M_3 di atas ,maka diperoleh file awal = 'DATA'.

Hal yang serupa juga bisa dilakukan untuk menggabungkan file dengan menggunakan penggabungan pecahan file lainnya,

4. KESIMPULAN

Berdasarkan pembahasan di atas, maka dapat diambil kesimpulan bahwa, untuk mengamankan file rahasia yang akan didistribusikan dapat menggunakan algoritma *secret sharing asmuth-boom.* Mengapa? Sebab algoritma ini cukup kuat untuk mengamankan file sehingga tidak bisa diketahui oleh orang lain selain penerima serta sulit untuk dipecahkan.

REFERENCES

[1] Evi Amelia Tarigan, "Perancangan Aplikasi Terenskripsi dengan Mengugunakan Metode Secret Sharing Asmuth-Bloom", JURIKOM (Jurnal Riset Komputer), Vol. 5, No. 6, pp. 648-652, 2018.
[2] Marto Sihombing and Erik Gunawan, "Perangkat Lunak Pembelajaran Protokol dengan Algoritma Secret Sharing Asmuth-Bloom", STMIK IBBI.
[3] Dafit, "Kriptografi Kunci Simetris Menggunakan Crypton", Jurnal Ilmiah, Vol. 2, No. 3, 2006.
[4] Reinaldi Munir, "Kriptografi", Bandung, Informatika, 2006
[5] C.K. Chu, W. G. Zheng, "Optimal Resilent Threshold Signature, Informatika. Vol. 8, No. 177, pp. 1834-1851.
[6] A. Shamir, "How To Share Secret", Commun, ACM 22, No. 11, pp. 612-613, 1976.