Modifikasi Pemberian *Digital Signature* pada Arsip Citra Menggunakan *Gray Code*

Unggul Satrio Respationo – NIM : 13506062

Teknik Informatika, Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung
Jl. Ganesha 10, Bandung, Jawa Barat
e-mail: if16062@students.if.itb.ac.id

Abstrak

Digital Signature atau tanda tangan digital adalah teknik untuk memberikan otentikasi pada suatu arsip. Secara umum pemberian tanda tangan digital meliputi 2 proses, yaitu pembuatan message digest, dan enkripsi *message digest* menggunakan *Asymmetric Cryptosystem* menjadi tanda tangan yang akan ditambahkan ke dalam arsip.

Pemberian tanda tangan digital pada arsip citra merupakan masalah tersendiri karena file merupakan *binary file* yang jika di tambahkan tanda tangan hanya dengan menambahkan begitu saja akan menyebabkan arsip menjadi rusak. Untuk itu perlu teknik untuk menambahkan tanda tangan sehingga arsip masih dapat dibuka dan tidak rusak. Salah satu teknik yang dapat digunakan adalah watermarking.

Pada teknik pemberian tanda tangan digital banyak sekali digunakan representasi bit dari suatu data seperti saat pembuatan *message digest* dan penyisipan tanda tangan pada arsip. Representasi bit yang biasa saja akan menyebabkan kriptanalis dapat melakukan serangan yang sudah ada. Jika digunakan representasi bit yang berbeda akan menyebabkan kriptanalis tidak menduga bahwa ternyata trepresentasu bitnya berbeda. Untuk itu penulis akan mencoba mengimplementasikan teknik representasi bit menggunakan *Gray code* untuk beberapa teknik dalam pemberian tanda tangan digital.

***Kata kunci**: Digital Signature,* Tandatangan Digital, *Gray Code,*

1. PENDAHULUAN

Dalam bahasan kriptografi, menjaga kepemilikan suatu data digital dan integritasnya merupakan hal yang penting. Hasil karya seseorang dalam bentuk digital dapat dengan mudah disalin, dimanipulasi oleh orang lain. Bahkan orang lain dapat melakukan klaim hak milik terhadap suatu digital, karena mudahnya untuk melakukan penyalinan terhadap data digital. Untuk itu diperlukan metoda *digital signature* (tanda tangan digital) dan *watermarking* yang dapat menjaga kepemilikan seseorang terhadap suatu data digital dan keasliannya. Dengan mengkombinasikan kedua metode ini kita dapat membubuhkan tanda tangan pada suatu arsip citra.

Pemberian tanda tangan digital pada arsip citra sangatlah penting, karena arsip citra merupakan salah satu media dalam dunia digital yang paling sering digunakan seperti logo suatu perusahaan yang pastinya merupakan arsip citra. Dengan memberikan tanda tangan digital, kita dapat menjamin bahwa suatu arsip itu asli dan tidak diubah-ubah sebelumnya oleh pihak yang tidak berwenang.

Algoritma pembuatan digital signature dari suatu arsip merupakan kombinasi dari algoritma public key cryptography dan algortima fungsi hash. Pada algoritma dengan menggunakan algoritma RSA dan MD5 akan menghasilkan tanda tangan dengan ukuran 128 bit, yang relatif kecil jika dibandingkan dengan ukuran arsip citra yang relatif besar. Sehingga penyisipan tanda tangan pada arsip citra tentunya hanya akan mengubah sedikit data pada arsip citra yang tidak akan dapat ditangkap dengan mata kepala manusia perbedaannya dengan aslinya.

2. *DIGITAL SIGNATURE* PADA ARSIP CITRA

2.1. Algoritma Pembuatan Tanda Tangan Digital

Pembuatan tanda tangan digital pada suatu arsip media melalui 2 tahap yaitu *hashing* dan enkripsi hasil hash yang menjadi tanda tangan digital. Pada kasus ini dibahas algoritma pemberian tanda tangan dengan RSA dan MD5.

2.1.1. Fungsi Hash MD5

MD5 (Message Digest 5) merupakan algoritma hash satu arah yang dikembangkan oleh Ronald Rivest pada tahun 1991. Algoritma ini menerima masukan data dengan panjang berapapun dan mengembalikan nilai message digest dengan panjang 128 bits (16 byte). Secara garis besar langkah-langkah dalam fungsi MD5 adalah :

- Penambahan bit-bit pengganjal (*padding bits*) pada akhir data yang akan dikenakan fungi hash sehingga panjang data menjadi ekivalen 448 (mod 512). Bit pengganjal diawali dengan 1 dan diikuti dengan 0.
- Penambahan nilai panjang pesan semula pada akhir data dengan panjang informasi sebanyak 64 bit.
- Inisialisasi *MD (Message Digest) buffer*. MD5 menggunakan buffer sejumlah 4 buah dengan panjang masing-masing 32 bit.
- Pengolahan pesan dalam blok berukuran 512 bit. Data akan dibagi-bagi menjadi N blok dengan panjang tiap blok adalah 512 bit. Tiap blok akan dikenakan proses terhadap *buffer* yang akan menghasilkan data sepanjang 128 bit.
- Masing-masing hasil blok diproses dengan XOR sehingga didapatkan hasil berupa message digest.

Contoh hasil dari fungsi MD5 terhadap *string* "Arsip Citra" adalah `8216 3813 CB0D 4987 9834 0A71 0C7E 0889.`

2.1.2. Enkripsi RSA

Algoritma RSA diambil dari nama ketiga orang pengembangnya yaitu Ron (R)ivest, Adi (S)hamir, dan Leonard (A)dleman. Algoritma ini memiliki kekuatan berupa sulitnya mencari faktor prima dari bilangan yang besar. Secara garis besar algoritma ini cukup sederhana, hanya terdiri dari fungsi pangkat dan modulo saja. Berikut algoritma secara singkat :

- Pilih 2 bilangan prima p dan q yang cukup besar.
- Hitung $n = p * q$. Dan hitung $\phi(n) = (p-1) * (q-1)$.
- Pilih kunci publik e yang relatif prima terhadap $\phi(n)$.
- Bangkitkan kunci private dengan ketentuan

$$e * d \equiv 1 (mod\ \phi(n))$$

- Enkripsi dengan cara membagi plainteks menjadi blok-blok sehingga setiap blok merepresentasikan nilai dalam selang $[0..n-1]$.
- Tiap blok akan dikenakan fungsi

$$c_i \equiv {p_i}^e (mod\ n)$$

- Dekripsi dilakukan dengan fungsi yang sama terhadap setiap blok hanya dengan kunci yang berbeda.

$$p_i \equiv {c_i}^d (mod\ n)$$

Pada pemberian tanda tangan digital, enkripsi dilakukan dengan kunci privat. Dan saat verifikasi tanda tangan dilakukan dekripsi dengan kunci publik.

2.2. Format Arsip Citra

Sebuah arsip citra, baik itu yang bertipe *raster* maupun *vector* pasti akan memiliki header yang merupakan *metadata* dari data yang ada di dalamnya. Dan setiap arsip pasti akan ditampilkan dalam pixel dengan skala tingkatan warna yang berbeda tiap arsipnya.

Karena pada dasarnya setiap bit image jika memiliki warna 24-bit maka setiap pixelnya akan bernilai 3 bytes. Sehingga untuk arsip yang katakanlah berukuran 8 megapixel akan memiliki ukuran arsip sebesar 24.000.000 bytes. Ini merupakan nilai yang sangat besar. Untuk itu beberapa format arsip citra akan memiliki tingkatan kompresi.

Beberapa arsip yang umum digunakan terutama untuk penggunaan dalam internet adalah JPEG, GIF, dan PNG. Akan tetapi untuk mengubah nilai byte pada arsip ini akan memerlukan teknik dan metoda khusus karena akan ada konvensi yang harus dipatuhi didalamnya. Maka untuk penyederhanaan arsip yang digunakan berbasis BMP, yaitu format arsip media yang terdiri dari matriks nilai warna tiap pixel. Format ini belum memiliki kompresi sehingga memiliki ukuran besar, tetapi memiliki keuntungan yaitu simplicity, dapat diterima di mana saja.

2.3. Steganografi pada Arsip Citra

Steganografi adalah teknik penyisipan suatu pesan ke dalam data lainnya dimana data yang digunakan untuk menyisipkan pesan (*covertext / coverimage*) tidak akan mengalami perbedaaan yang mencolok sehingga tidak akan mudah dikenali. Untuk melakukan penyisipan pesan dalam arsip citra dapat digunakan teknik spatial domain, dimana kita mengubah langsung nilai byte dari *cover-object* yang merepresentasikan nilai intensitas warna per pixel.

Tekniknya adalah mengubah nilai LSB dari byte yang akan diubah dengan nilai bit dari pesan yang akan dimasukkan. Sehingga untuk menyisipkan tanda tangan digital dengan panjang 128 bit kita hanya membutuhkan 128 byte data yang akan diubah.

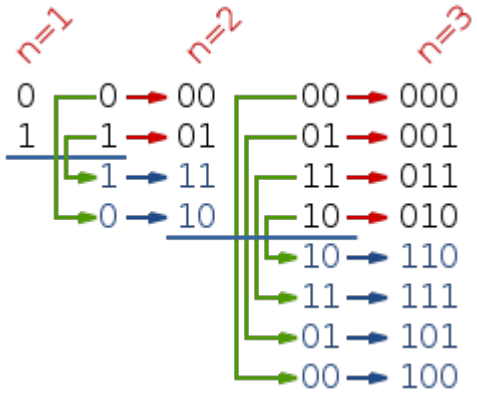
Karena tanda tangan digital tidak masalah jika diketahui orang lain maka posisi pemberian tanda tangan tidak perlu dipermasalahkan, dapat di awal file, tengah, akhir, bahkan di acak dengan kunci. Tentunya ketika penyisipan kita harus memastikan bahwa tanda tangan dan *cover-object* yang notabene adalah arsip yang ditandatangani harus dapat diambil kembali sesuai dengan aslinya. Bahkan kelemahan metode LSB yaitu lemah terhadap modifikasi *cover-object* tidak menjadi masalah, karena justru perubahan *cover-object* sama saja dengan mengubah arsip yang di tanda

tangani yang akan membuat arsip menjadi tidak valid.

3. *GRAY CODE*

Gray code, juga dikenal sebagai *reflected binary code* dinamakan atas penemunya yaitu *Frank Gray.* Gray code adalah suatu sistem penomoran biner dengan 2 bilangan yang berdekatan hanya memiliki tepat 1 bit beda. Gray code ditujukan untuk mencegah terjadinya keluaran yang rusak pada suatu switch elektromagnetik. Dewasa ini gray code banyak digunakan untuk berbagai pemeriksa galat pada aplikasi komunikasi digital seperti sistem TV kabel.

Untuk mendapatkan suatu deret gray code dengan ukuran n-bit, kita dapat memulainya dari 1 bit terlebih dahulu yaitu 0 dan 1. Kemudian kita dapat menuliskannya berulang dengan urutan berbeda dan diawali dengan bit 1 sehingga didapatkan 2-bit gray code dan seterusnya.



Gambar 1 Diagram pembangunan Gray code

Berikut merupakan contoh representasi bilangan decimal dalam gray code dan perbandingannya dengan representasi biner biasa :

Dec	Gray	Binary
0	000	000
1	001	001
2	011	010
3	010	011
4	110	100
5	111	101
6	101	110
7	100	111

Secara algoritma bilangan gray code dapat dihasilkan dengan algoritma sebagai berikut :

- Bilangan decimal yang akan diubah dikonversikan terlebih dahulu ke dalam representasi binernya. Contoh : 14 →1110.
- Dimulai dari LSB, dalam contoh adalah 0, kita cek nilai bit disebelah kirinya, jika bernilai 1 maka LSB = 1 – LSB, jika tidak biarkan saja. Pada contoh maka nilai LSB akan menjadi 1.
- Untuk bilangan berikutnya proses yang dilakukan sama, dengan mengasumsikan nilai dari MSB tidak akan diubah. Sehingga akan didapatkan hasil gray code dari bilangan decimal 14 adalah 1001.

Untuk mendapatkan representasi biner suatu bilangan decimal dari graycode dapat dilakukan algoritma berikut :

- Dimulai dari LSB atau bit ke-n kita kenakan perhitungan dengan g_n adalah nilai bit dari gray code pada posisi ke – n.

$$\Sigma_n \equiv \sum_{i=1}^{n-1} g_n (mod\ 2)$$

- Jika nilai Σ_n adalah 1, tukar nilai g_n dengan $1 - g_n$, jika tidak biarkan saja.
- Dan selanjutnya dilakukan perhitungan yang sama, sehingga akan didapatkan bilangan biner yang bersesuaian dengan gray code yang ada.

4. IMPLEMENTASI

Pada kriptografi, seringkali kita berkerja dalam mengubah dan memanipulasi bit / byte data. Jika kita mengubah representasi bit ini dengan yang lainnya tentu akan mengakibatkan kriptanalis semakin sulit menganalisa data yang akan di serangnya. Dalam kasus ini kita terapkan gray code dalam pemberian tanda tangan digital.

Pergantian representasi bit dapat kita terapkan pada data hasil dari message digest yaitu data sepanjang 128 bit. Sebagai contoh hasil MD5 dari string "Arsip Citra" adalah `8216 3813 CB0D 4987 9834 0A71 0C7E 0889.` Hasil ini jika direpresentasikan dalam bit kemudian nilai bit tadi kita representasikan ke dalam hexa menjadi `C315 2C12 AE0B 6DC4 DC26 0F41 0A49 0CCD.` Nilai yang didapat akan jauh berbeda. Sehingga jika sebelumnya tanda tangan yang dihasilkan dari adalah `EDC4 478F 8CA0 8998 21AF 31CA 4F9A 8F43,` setelah dilakukan perubahan message digest dengan gray code tanda tangan akan menjadi `6A14 7D1B 9F2B A2D7 0601 311A CC9A A075.` Nilai yang berbeda ini tentunya akan membuat kriptanalis sulit untuk melakukan kriptanalisis.

Bahkan, kita bias melakukan representasi gray code lagi terhadap tanda tangan yang telah kita dapatkan sehingga tanda tangan akan menjadi `5F16 4B1E D83E F3B4 0501 211F AADF F047.`

Tentunya beberapa serangan yang bisa dilakukan pada tangan digital seperti pencarian kolisi pasti akan gagal selama sang kriptanalis tidak mengetahui bahwa representasi bit pada hasil hash

menggunakan gray code. Begitu juga untuk pencarian kunci privat jika diketahui kunci publik ketika melakukan dekripsi tanda tangan.

Akan tetapi, pada sisi penerima juga harus mengetahui bahwa representasi bit dalam gray code. Hal ini tentunya harus dirahasiakan dari orang lain.

5. KESIMPULAN

Kesimpulan dari studi implementasi *Gray code* pada pemberian tanda tangan digital arsip citra adalah :

1. Pemberian tanda tangan digital pada arsip citra merupakan masalah tersendiri, karena file citra merupakan *binary file* sehingga penambahan data yang tidak sesuai akan menyebabkan perubahan yang drastic. Untuk itu dapat diimplementasikan teknik steganografi untuk pemberian tanda tangan digital.
2. Serangan pada tanda tangan digital dapat berupa serangan pada algoritma enkripsi maupun pada algoritma fungsi hash. Karena sebagian besar algoritma tersebut bekerja pada level byte bahkan bit, perubahan representasi dapat memperkuat algoritma.
3. *Gray code* mengubah representasi biner standard sehingga akan menyulitkan serangan pada algoritma.
4. Implementasi *Gray code* ini juga bias dikembangkan untuk berbagai metode kriptografi lainnya, terutama yang bekerja dalam level byte / bit.

DAFTAR REFERENSI

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

[2] Sloane, N. J. A. Sequence A014550 in "The On-Line Encyclopedia of Integer Sequences."

[3] Stinson, D. R. 1995. *Cryptography: Theory and Practice*. CRC Press.

[4] Talbot, John and Dominic Welsh. 2006. *Complexity and Cryptography*. Cambridge University Press.

[5] Weisstein, Eric W. "Gray Code." From MathWorld - A Wolfram Web Resource. http://mathworld.wolfram.com/GrayCode.html. Waktu akses: 26 Desember 2007, pukul: 20.00.