Departemen Teknik Informatika
Institut Teknologi Bandung

Tugas III IF5054 Kriptografi

Tanda-tangan Digital pada Dokumen dengan Menggunakan Algoritma *ElGamal*

Batas pengumpulan : 9 Desemebr 2005
Tempat pengumpulan : Lab IRK (disedikan tempat khusus di alemari)
Arsip pengumpulan :
- disket/cd berisi program, arsip *readme.txt*, laporan, arsip contoh, arsip parameter dan kunci.
- kertas A4 untuk laporan (*print 2up*)

Deskripsi tugas :

Tanda-tangan digital dapat digunakan untuk otentikasi data digital, seperti pesan yang dikirim melalui saluran komunikasi dan dokumen elektronis yang disimpan dalam komputer.

Pada tugas ke-3 ini, anda diminta mengimplementasikan algoritma *ElGamal* untuk memberi tanda-tangan digital pada dokumen (*file*) elektronis, dan algoritma *MD5* untuk membangkitkan nilai *hash* dari dokumen. Dalam hal ini, anda sebagai pemilik dokumen mempunyai sepasang kunci, yaitu kunci publik dan kunci privat. (catatan: algoritma ElGamal yang digunakan adalah algoritma *ElGamal* untuk enkripsi/dekripsi seperti yang sudah dijelaskan di dalam dikat kuliah, bukan algoritma *ElGamal* untuk *signature*. Untuk diketahui, ada algoritma *ElGamal* yang khusus untuk *digital signature*).

Dokumen yang akan diberi tanda-tangan digital dibatasi hanya arsip teks saja (`*.txt`, `*.pas`, `*.c`, `*.cpp`, `*.bas`, dan lain-lain) meskipun sebenarnya arsip dengan format selain teks dapat juga diberi tanda-tangan digital (namun memerlukan pengetahuan format arsip sebelum kita dapat mengimplementasikan tanda-tangan digital, tetapi hal ini diluar cakupan tugas). Pemilihan arsip teks pada tugas ini dimaksudkan agar arsip yang sudah diberi tanda-tangan digital dapat disunting oleh editor teks, sehingga anda dapat melihat apa yang terjadi pada hasil verifikasi jika nilai tanda-tangan digital atau isi dokumen diubah.

Tanda-tangan digital dapat dilekatkan (*embedded*) di awal atau di akhir dokumen, tetapi, pada tugas ini tanda-tangan digital dilekatkan di akhir dokumen. Tanda-tangan digital selanjutnya digunakan untuk membuktikan keaslian isi dokumen dan keaslian pemilik dokumen. Dokumen harus dapat diekstraksi kembali dari arsip yang sudah diberi tanda-tangan digital sehingga dokumen dapat dibuka dan diproses oleh program aplikasi yang bersesuaian. Begitu juga tanda-tangan digital harus dapat diekstraksi dari dokumen.

Tanda tangan digital bergantung pada isi dokumen dan kunci. Tanda-tangan digital direpresentasikan sebagai karakter-karakter heksadesimal dan ditaruh pada awal dokumen. Untuk membedakan tanda-tangan digital dengan isi dokumen, maka tanda-tangan digital diawali dan diakhiri dengan *tag* `<ds>` dan `</ds>`, atau penandaan dengan cara lain (diserahkan kepada anda)
Contoh: `<ds>4EFA7B223CF901BAA58B991DEE5B7A</ds>`.

Oleh karena algoritma *ElGamal* menghasilkan pasangan cipherteks *a* dan *b*, maka program harus dapat membedakan nilai *a* dan *b* di dalam tanda-tangan digital tersebut (mungkin dengan suatu separator atau penanda lain).

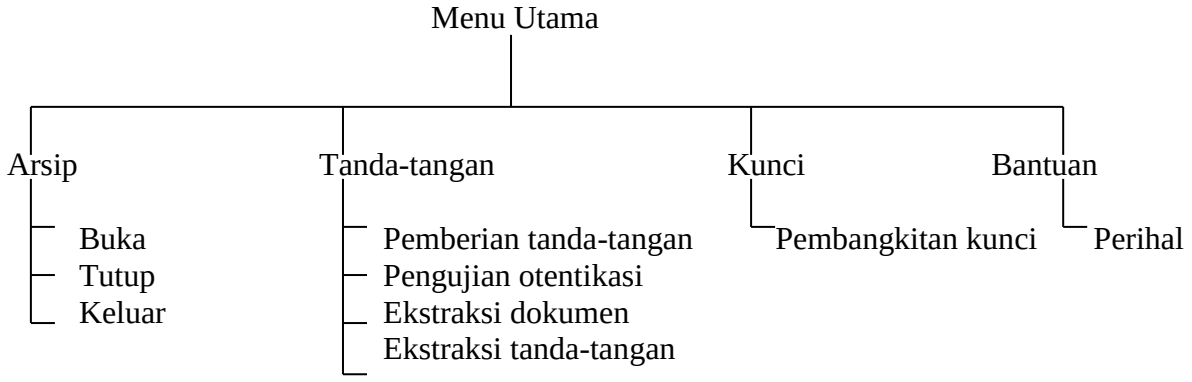
Berhubung algoritma *ElGamal* menggunakan parameter bilangan bulat yang panjang (besar), maka program anda harus mampu menggunakan bilangan yang besar dengan membuat tipe data khusus untuk bilangan bulat besar dan primitif-primitif operasi aritmetikanya (lihat penjelasannya pada Bab 17 di dalam diktat kuliah). Penggunaan tipe *long integer* sebaiknya digunakan dalam keadaan terpaksa apabila anda tidak berhasil membuat tipe data khusus tersebut beserta primitif operasinya.

Spesifikasi program:

1. Program mampu memberikan tanda-tangan digital pada arsip dengan format teks. Keluarannya adalah arsip bertipe teks yang sudah berisi dokumen semula + tanda-tangan digital (disimpan sebagai arsip baru, tidak menimpa arsip lama).
2. Program mampu memverifikasi otentikasi dokumen. Keluarannya adalah informasi di layar bahwa tanda-tangan digital sah atau tidak.
3. Program mampu membangkitkan bilangan acak untuk parameter algoritma *ElGamal*.
4. Program mampu membangkitkan pasangan kunci (kunci publik dan kunci privat). Kunci privat/publik sebaiknya dapat disimpan.
5. Sedapat mungkin program menerima sedikit masukan (hanya nama arsip, kunci publik/privat).
6. Program mampu menangani operasi aritmetika dengan bilangan bulat yang panjang (melebihi *long integer*).
7. Program mampu mengekstraksi dokumen atau tanda-tangan dari arsip yang sudah diberi tanda-tangan digital.
8. Khusus dokumen yang bertipe teks, program dapat menampilkan baik arsip yang teks yang belum diberi tanda-tangan digital maupun yang sudah diberi tanda-tangan digital (manfaatkan editor sederhana untuk menampilkan arsip teks dari program tugas 2).

Lain-lain

1. Struktur menu minimal di dalam program adalah kira-kira sebagai berikut (anda dapat menambahkan menu lain jika perlu)



Ket: Pada menu "Buka", hanya dapat membuka dan menampilkan arsip.
Pada menu "Ekstraksi dokumen", masukannya adalah nama arsip yang sudah diberi tanda-tangan digital, dan keluarannya adalah dokumen semula yang belum diberi tanda-tangan digital (nama arsip hasil ekstraksi boleh sama dengan nama arsip lama atau nama

baru (*save as*). Pengaktifan menu "Ekstraksi dokumen" boleh membutuhkan masukan kunci privat pemilik arsip agar tidak sembarang orang dapat melakukannya. Penjelasan serupa juga berlaku untuk menu "Ekstraksi tanda-tangan"

2. Bilangan prima dapat diketikkan sendiri oleh pengguna program atau dibangkitkan secara acak oleh program pembangkit. Bilangan prima yang dibangkitkan panjangnya sembarang.

3. Nama arsip yang sudah diberi tanda-tangan digital sebaiknya memakai ekstensi khusus, misalnya `.dsf`, namun arsip tersebut masih tetap dalam format teks.
5. Tugas dikerjakan berkelompok yang terdiri maksimal 3 orang.
6. Program dibuat sebaiknya mempunyai antarmuka yang bagus.
7. Program diberi nama yang singkat, menarik, dan memiliki makna.
8. Program harus mengandung komentar yang jelas.
9. Lampirkan di dalam disket program anda arsip contoh dan arsip parameter & kunci.
10. Program *MD5* sangat dianjurkan dibuat sendiri (lebih memberi kepuasan tersendiri). Jika program *MD5* diambil dari internet, anda harus menyebutkan *URL* yang mengandung program *MD5* tersebut. Ada perbedaan bobot nilai antara membuat sendiri progarm *MD5* atau mengambil/memodifikasi program *MD5* dari sumber-sumber di internet. Moral dari *point* 10 ini, kejujuran lebih diutamakan.

Isi laporan :

1. Deskripsi masalah.
2. Dasar teori.
3. Strategi penyelesaian masalah (lingkungan implementasi dan trik khusus).
4. Struktur data dan spesifikasi subrutin.
5. Pengujian dan analisis hasil. Pengujian menggunakaan arsip contoh yang disertakan di dalam teks.
   Pengujian meliputi otentikasi dengan kasus-kasus berikut:
   - karakter di dalam teks diubah (dihapus, ditambah)
   - karakter di dalam tanda-tangan digital diubah
   - kunci privat yang digunakan tidak berpadanan dengan pasangan kunci publiknya.
   - tanda-tangan digital dihapus dari dokumen
6. Lampiran yang berisi:
   - antarmuka program
   - contoh arsip masukan
   - contoh arsip keluaran yang sudah diberi tanda-tangan digital.
   - contoh nilai-nilai paramater *ElGamal* yang digunakan
7. Kesimpulan dan saran.