Keamanan pada Digital Signature dan Potensinya di Masa Depan

Nama: Dannis Muhammad Mangan
NIM: 13507112
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
e-mail: dannis_m@students.itb.ac.id

Abstrak - Tanda tangan fisik telah menjadi fondasi dalam otentikasi bisnis dan transaksi kepemerintahan selama ratusan bahkan ribuan tahun. Namun sekarang ini dengan tools dan kebutuhan yang sudah bergeser ke media digital, kebutuhan untuk otentikasi atas dokumen-dokumen digital sangatlah mendesak. Kendati sudah mulai diimplementasikan, seringkali dokumen yang diberi tanda tangan fisik dianggap lebih "resmi" dibandingkan dokumen dengan tanda tangan digital. Kendala terbesar yang dihadapi adalah keamanan akan tanda tangan digital tersebut, misalnya apakah bisa dipalsukan atau tidak.
Makalah ini lebih ditujukan untuk mempelajari digital signature yang sudah ada dan mengetahui kekurangan-kekurangannya melalui serangan sehingga nantinya dapat dievaluasi lebih lanjutdan potensinya untuk menggantikan tanda tangan fisik.

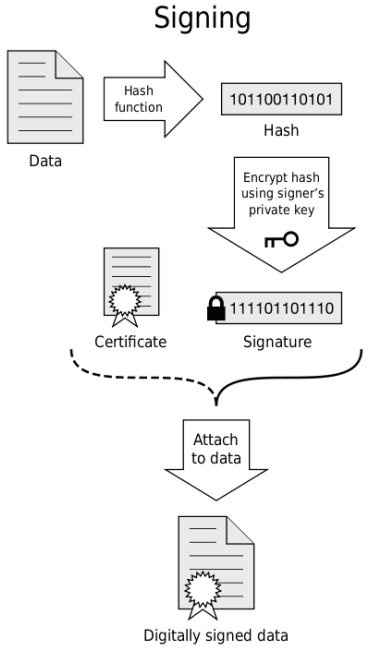
Kata kunci:Tanda tangan digital, RSA, Keamanan, Serangan, Otentikasi

I. PENDAHULUAN

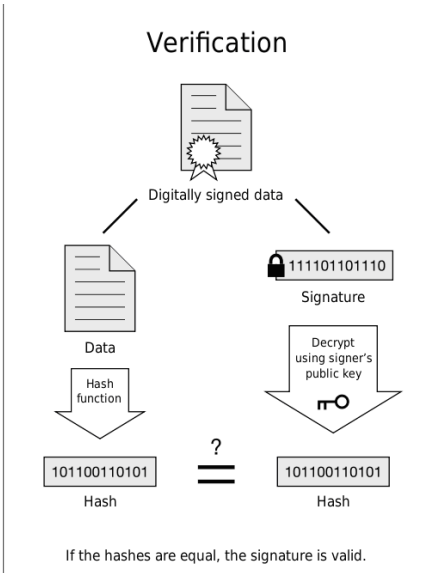
Tanda tangan digital (*Digital signature*) merupakan salah satu topik yang hangat pada pengaplikasian teknologi pada bisnis. Sebelumnya, tanda tangan fisik telah menjadi fondasi dalam otentikasi bisnis dan transaksi kepemerintahan selama ratusan bahkan ribuan tahun. Namun sekarang ini dengan tools dan kebutuhan yang sudah bergeser ke media digital, kebutuhan untuk otentikasi atas dokumen-dokumen digital sangatlah mendesak.
Walaupun sudah mulai diimplementasikan terutama di negara maju, seringkali dokumen yang diberi tanda tangan fisik dianggap lebih "resmi" atau "legal" dibandingkan dokumen dengan tanda tangan digital. Kendala terbesar yang dihadapi adalah keamanan akan tanda tangan digital tersebut, misalnya apakah bisa dipalsukan atau tidak. Salah satu algoritma yang umum dipakai dalam skema tanda tangan digital adalah algoritma RSA.

II. TANDA TANGAN DIGITAL

Tanda tangan digital (*digital signature*) berbeda dengan *electronic signature*. Tanda tangan digital adalah suatu skema untuk pembuktian keabsahan suatu dokumen atau pesan digital. Yang dimaksud dokumen digital disini adalah dokumen-dokumen yang berwujud elektronik dan bukan pada kertas. Dokumen digital yang telah di tandatangani secara digital (*digitally signed documents*) adalah dokumen digital. Dokumen tersebut tidak dapat dipisahkan dari tanda tangan digital-nya (merupakan satu kesatuan).



Gambar 1. Skema pembubuhan tanda tangan digital



Gambar 2. Skema pengecekan keabsahan dokumen bertanda tangan digital

III. ALGORITMA RSA

Algoritma RSA (dinamakan RSA dari inisial ketiga pembuatnya Rivest-Shamir-Adleman) merupakan algoritma *public-key cryptography* yang banyak diimplementasikan pada protocol *electronic commerce* dan dipercaya mempunyai tingkat keamanan yang tinggi

Langkah-langkah pembuatan algoritma RSA sebagai berikut:
- Generate kunci
1. Pilih 2 bilangan prima berbeda p dan q
2. Hitung n = p x q
3. Hitung totient = (p-1) x (q-1)
4. Pilih e sedemikian sehingga e < totient dan e saling prima dengan totient
5. Cari d yang memenuhi de kongruen 1 (mod totient)

-Proses pembubuhan tanda tangan digital
Untuk setiap karakter pada hasil Hash dokumen, $c = m^d \bmod n$
Hasilnya adalah “tanda tangan digital”.
-Proses verifikasi
Untuk setiap karakter pada “tanda tangan digital”, $m = c^e \bmod n$

Contoh kode program RSA (disadur dari http://www.cryptopp.com/wiki/RSA):

```
// Generate or Load keys
RSA::PrivateKey privateKey = ...;
RSA::PublicKey publicKey = ...;

// Message
string message = "RSA Signature",
signature;

////////////////////////////////////////////////
// Sign and Encode
RSASSA_PKCS1v15_SHA_Signer
signer( privateKey );

StringSource( message, true,
   new SignerFilter( rng, signer,
      new StringSink( signature )
   ) // SignerFilter
); // StringSource

////////////////////////////////////////////////
// Verify and Recover
RSASSA_PKCS1v15_SHA_Verifier
verifier( publicKey );

StringSource( message+signature, true,
   new SignatureVerificationFilter(
      verifier, NULL,
      SignatureVerificationFilter::THROW_
EXCEPTION
   ) // SignatureVerificationFilter
); // StringSource

cout << "Verified signature on message" <<
endl;
```

IV. POTENSI PENGGUNAAN

Penggunaan tanda tangan digital banyak dianggap sebagai salah satu prioritas yang harus dikembangkan karena mempunyai potensi bisnis yang sangat besar. Penggunaan tanda tangan digital dianggap dapat merevolusi industri bisnis.
Keunggulan terbesar dari tanda tangan digital (karena bersifat dokumen elektronik) adalah penghematan waktu dan biaya, kemudian di masa depan kecenderungannya adalah dokumen semakin berwujud elektronik.
Secara bisnis dan tekonologi akan terus maju ke depan, penggunaan kontrak elektronik dan tanda tangan digital dalam masa depan mungkin akan se-umum penanda-tanganan dokumen jarak jauh via faximile seperti pada saat ini.
Tanda tangan digital adalah tulang punggung dari sistem transfer data elektronik, dan pada akhirnya keberhasilannya tergantung pada sejauh mana perusahaan dan instansi pemerintah menggunakan teknologi untuk memfasilitasi hal-hal birokrasi.

V. ANALISIS CELAH KEAMANAN

Pada bagian ini akan dijelaskan beberapa celah keamanan yang dapat dilewati pada teknologi tanda tangan digital saat ini, dan ditujukan untuk usulan perbaikan keamanan dari sistem saat ini.

V.I Pencarian Private Key

Pada bagian pencarian private Key ini, algoritma yang akan diserang adalah RSA karena merupakan algoritma yang sangat umum digunakan

V.I.I Menghitung-ulang kunci

Kondisi:
1.Mengetahui skema hash korban
2.Korban memakai Algoritma RSA
3.Mengetahui public key korban
4.Mengetahui nilai n korban

Langkah:
- Dari informasi nilai n, cari 2 bilangan prima p dan q | pq = n
- Hitung (P-1)*(q-1) = T
- Hitung bilangan d sedemikian sehingga $(kunci\_publik)^d \bmod T = 1$
- Simpan nilai d
- Buat nilai hash dari dokumen asli. Kemudian hasilnya di dekripsi dengan d.

Jika hasilnya sama dengan “tanda tangan digital”maka **kunci privat ditemukan.**

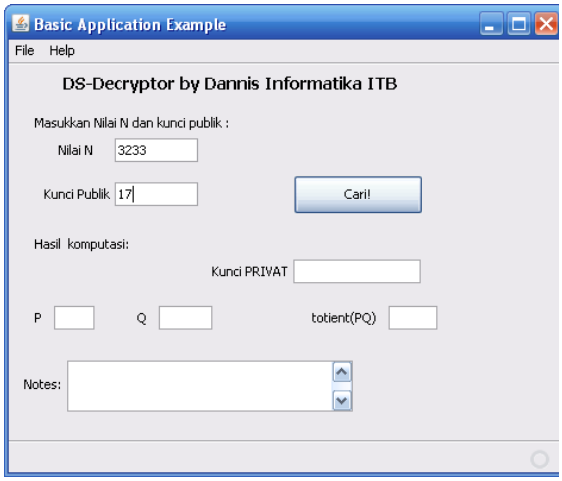
Kode program di Java untuk implementasinya:

```
public class degenerator {
double NilaiN;
double NilaiP;
double NilaiQ;
double NilaiTotient;
double NilaiPublic;
double NilaiPrivate;

degenerator (double N,double PublicKey){
NilaiN = N;
decrypt(N, (double)2);
NilaiP = NilaiN/NilaiQ;
NilaiTotient = (NilaiP-1) *(NilaiQ-1);
NilaiPublic = PublicKey;
CariPrivateKey(PublicKey, NilaiTotient);
}
public boolean IsPrime(double n) {
                    boolean prime = true;
                    for (double i = 3; i <=
Math.sqrt(n); i += 2)
                                if (n % i == 0) {
                                        prime =
false;
                                        break;
                                }
                    if (( n%2 !=0 && prime &&
n > 2) || n == 2) {
                                return true;
                    } else {
                                return false;
                    }
          }
public boolean IsDivisor(double N,double P){
if (Math.IEEEremainder(N, P)==0){
return true;
}else{
return false;
}
}

void decrypt(double N, double P){
//System.out.println("P: "+P);
if (IsPrime(P)){
if(IsDivisor(N, P)){
double Qtemp = N/P;
if (IsPrime(Qtemp)){
NilaiQ = Qtemp;
}else{
   decrypt(N,P+1);
}
}else{
   decrypt(N,P+1);
}
}else {
   decrypt(N,P+1);
}
}

void CariPrivateKey(double PublicKey,
double Totient){
double coba = 1;
          while ((PublicKey * coba) % Totient !
= (double)1)
          {
             coba++;
          }
NilaiPrivate = coba;
}
```

Pemakaian (masukkan: Nilai N dan kunci publik)

```
degenerator test = new degenerator
(Nilai_N,kunci_publik);
//elemen-elemen hasil komputasinya
tersimpan //secara otomatis pada dataslot
dari variable
// test
```

ScreenShot GUI Program (dibuat dengan Java dan IDE NetBeans):



Gambar 3. Contoh tampilan program dengan GUI Java Swing untuk pengaplikasian algoritma diatas

Kelemahan:
- Jika ada variable yang sangat besar, karena penulis merepresentasikan bilangan sebagai "double".
Solusi: mengganti representasi dengan BigInteger atau semacamnya
- Proses pemfaktoran N menjadi 2 buah bilangan prima membutuhkan waktu cukup lama jika N semakin besar.
Solusi: Penggunaan prosesor yang lebih cepat, tercatat bahwa bilangan N dengan 313 digit sudah berhasil dipecahkan pada tahun 2007.

V.I.II Menggunakan Brute force

Kondisi:
1.Mengetahui skema hash korban
2.Korban memakai Algoritma RSA
3.Mengetahui public key korban
4.Mengetahui nilai n korban

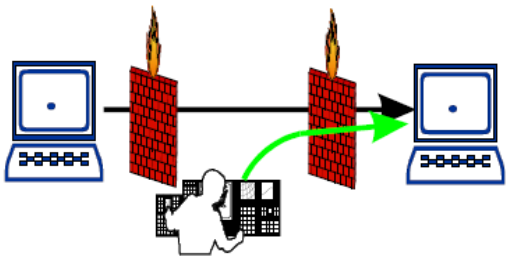
Seperti yang ditunjukkan oleh namanya, teknik ini memakai cara "coba-coba" untuk "mensimulasikan" private key korban. Skema teknik ini sebagai berikut:

- Kita mempunyai nilai n dan public key korban
- Generate Hash dari dokumen, sebut sebagai H(x)
- Generate sebarang bilangan / random. Jika bilangan tersebut sudah pernah di-generate, buang dan generate ulang. Bilangan itu disebut C
- Aplikasikan algoritma RSA dengan public key C dan nilai N
- Hasilnya dibandingkan dengan "signature" korban, jika masih berbeda, kembali ke langkah generate bilangan random. Jika hasilnya sudah sama maka **kunci privat** ditemukan.

Kelemahan: Untuk membuat "signature" dari H(x) dibutuhkan waktu yang cukup lama. Dan algoritma ini diraca kurang "cerdas" karena terlalu mengandalkan pencarian secara naïf/coba-coba.

V.II Celah Firewall

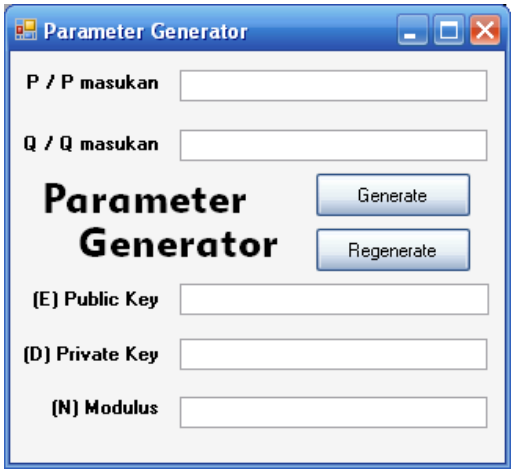
Pada dewasa ini, hampir seluruh komputer memakai pengaman firewall untuk mencegah data-data yang tidak dikenal masuk ke komputer tersebut. Namun ada kesenjangan keamanan pada firewall yang bisa dijadikan celah. Misalkan ada dua buah konfigurasi firewall:
1. Firewall dikonfigurasi sedemikian rupa sehingga tidak memungkinkan dokumen melalui isi yang tidak dapat diperiksa olehnya.
Akibat: Dalam cara ini berarti dokumen yang di-*cipher* dan dokumen yang ditanda tangani digital tidak akan pernah sampai karena tidak dikenali.
partai.
2. Firewall diatur membiarkan melalui pesan ciphered tanpa pengawasan. Maka kedua pihak dapat melakukan pertukaran dokumen yang di-*cipher* dan dokumen yang ditanda tangani digital.
Akibat: Seorang penyerang dapat masuk menembuh firewall untuk menyerang, dengan menggunakan program berbahaya (virus, Trojan, dsb.) yang di-enkripsi dengan kunci public dari calon korban, sehingga file tersebut dapat melewati firewall



Gambar 4. Skema penyerangan dengan menggunakan celah firewall

VI. HASIL PERCOBAAN

Kebenaran dari hasil percobaan ini adalah program dengan menggunakan souce code seperti yang telah disebut sebelumnya dan diuji dengan variabel-variabel yang dihasilkan dari program "Parameter Generator" yang dibuat oleh 3 orang mahasiswa ITB (Dannis Muhammad, Aqsath Rasyid, Raditya Arief)



Gambar 5. Parameter Generator untuk Digital Signature yang dibuat oleh Dannis-Aqsath-Radit

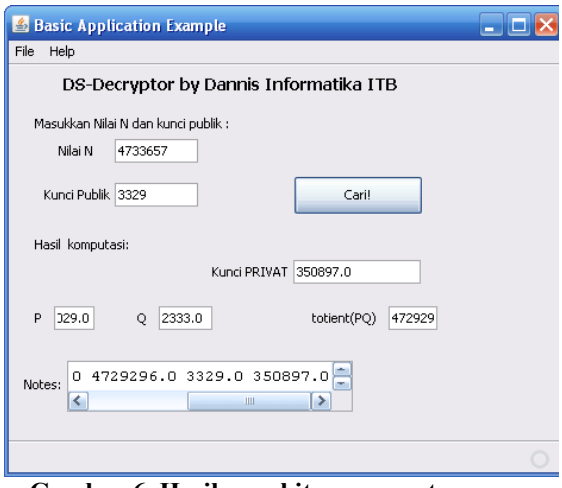
Dari program didapat variabel-variabel sebagai berikut:

P =2029
Q = 2333
Kunci public = 3329
Kunci privat = 350897
N = 4733657

Dan

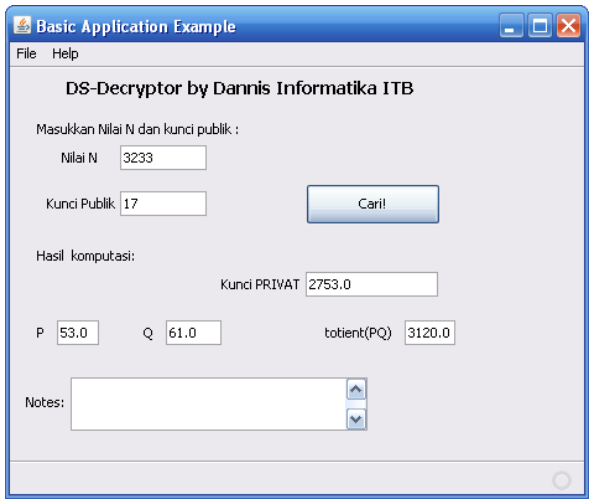
P =53
Q = 61
Kunci public = 17
Kunci privat = 2753
N = 3233

Hasil eksekusi program:



Gambar 6. Hasil penghitungan pertama

Kunci privat = 350897 => sama
P =2029 => sama
Q = 2333 =>sama



Gambar 7. Hasil penghitungan kedua

Kunci privat = 2753 => sama
P =53 => sama
Q = 61 => sama

Dari hasil percobaan diatas dapat dilihat bahwa program dapat menemukan dengan tepat kunci privat dengan menggunakan informasi public yaitu kunci public dan nilai N dari korban. Percobaan diatas .dimana menggunakan panjang kunci sedang dan pendek , mendapatkan waktu komputasi yang sangat cepat / dapat diabaikan. Sedangkan untuk kunci panjang (belum diuji oleh penulis) diperkirakan akan memakan waktu cukup banyak karena kompleksitas dari pencarian tersebut tergolong eksponensial, namun poin pentingnya adalah selama komputer yang dipakai belum kehabisan memori, nantinya kunci privatnya akan **ditemukan**.

VII. KESIMPULAN

Potensi penggunaan tanda tangan digital pada aktivitas bisnis adalah kemungkinan akan dipakai secara penuh dan aman pada beberapa tahun mendatang. Dokumen yang disahkan dengan tanda-tangan digital (diantaranya "kontrak elektronik" dalam bisnis) tersebut mungkin tidak akan menggantikan secara penuh dokumen dengan tanda tangan konvensional namun jelas akan mempercepat waktu pengesahan kontrak.
Keamanan dari tanda tangan digital sangatlah bergantung pada keamanan komputer dan juga panjang kunci serta jenis algoritma yang digunakan.
Tanda tangan digital bisa jadi lebih sulit di sangkal (non-repuditation) karena itu bahaya

jika dipalsukan (dengan menggunakan private key korban) sangatlah tinggi. Dari hasil percobaan diatas, pencarian private key korban masih cukup mudah dilakukan, karena itu sebelum dicari solusi keamanannya tanda tangan digital belumlah tepat untuk dijadikan standar keabsahan kepemilikan suatu dokumen.

VIII. REFERENSI

[1] Tanenbaum, Andrew. *Computer Networks Fourth Edition*. 2003. Prentice Hall PTR
[2].http://www.govetech.com/gt/94791?id=94791&full=1&story_pg=1
[3]http://www.youdzone.com/signature.html
[4]http://en.wikipedia.org/wiki/Digital_signature
[5]http://www.fleitold.com/publications/eicar2006.pdf
[6]http://www.corbinball.com/articles/art-digitalcontracts.htm
[7]http://www.fdic.gov/regulations/information/fils/banktechbulletin.html
[8]http://en.wikipedia.org/wiki/RSA
[9] http://www.cryptopp.com/wiki/RSA

Dengan ini, saya menyatakan bahwa makalah ini dibuat oleh saya sendiri dan bukan merupakan hasil plagiasi.

Dannis Muhammad Mangan,