JOISIE Journal Of Information System And Informatics Engineering
Vol. 4, No.1 Juni 2020, Hlm 1-6

p- ISSN: 2503-5304
e- ISSN: 2527-3116

PEMERIKSAAN INTEGRITAS DOKUMEN DENGAN DIGITAL SIGNATURE ALGORITHM

Diki Arisandi[1)], Sukri[2)], Moh. Baharudin Yusuf[3)].
[1,2,3] Fakultas Teknik, Universitas Abdurrab, Jalan Riau Ujung No.73 Kota Pekanbaru, Riau
email: [1]diki@univrab.ac.id, [2]Sukri@univrab.ac.id, [3]moh.baharudin.y@student.univrab.ac.id

Abstract

The vital aspect that must be contained in the document is the signature, which is the spesific identifier for document accountability. However, sometimes a person who sign document or signers unable to sign by reasons. This can be resolved by digitized signature but through the authentication process and the integrity checking of a document using DSA (Digital Signature Algorithm). DSA uses private and public key, where the private key is to generate a digital signature and a public key to verify a digital signature. This reserch explains the DSA is succeed to identify the set of its key, document, and signatures to verify that the received document consistently maintains its integrity from the sender to the recipient.

Keywords: *DSA, Public Key, Private Key, Integrity Check, Digital Signature.*

Abstrak

Hal penting yang harus ada dalam sebuah dokumen adalah tandatangan, yang merupakan suatu ciri bahwa dokumen tersebut dapat dipertanggungjawabkan. Akan tetapi, terkadang seseorang yang akan menandatangani dokumen tidak dapat menandatanganinya karena alasan tertentu. Permasalahan ini dapat diatasi dengan penggunaan tandatangan yang terdigitasi namun melalui otentikasi dan pemeriksaan integritas sebuah dokumen dengan menggunakan DSA (Digital Signature Algorithm). DSA menggunakan kunci publik dan kunci privat, dimana kunci privat untuk membuat tandatangan digital dan kunci publik untuk memverifikasi tandatangan digital. Algoritma ini diuji pada dokumen yang dibuat di aplikasi perkantoran. Dalam penelitian ini dijelaskan bagaimana DSA dapat mengidentifikasi kecocokan pasangan kunci, dokumen, dan tanda tangan untuk memastikan bahwa dokumen yang diterima secara konsisten terjaga integritasnya dari pengirim ke penerima.

Kata Kunci*: DSA, Kunci Publik, Kunci Privat, Pemeriksaan integritas, Tandatangan Digital.*

1. PENDAHULUAN

Dokumen sering berisi informasi penting seperti kontrak resmi, transaksi keuangan, dan lain-lain (Pattah, 2013). Hal terpenting yang disertakan pada dokumen yaitu tanda tangan dan pada saat ini sudah banyak dokumen elektronik (e-dokumen) atau dokumen digital untuk keperluan formal (Wahyudi, 2012). Dengan adanya dokumen elektronik, mutlak diperlukan verifikasi dokumen untuk memastikan keabsahannya, baik isi dokumen maupun yang menandatanganinya (Widodo and Harjoko, 2015).

Contoh kasus seperti seorang bawahan yang berada di kantor cabang membutuhkan sebuah dokumen yang harus di tanda tangani oleh atasannya yang berada di kantor pusat. Akan memakan waktu yang lama apabila harus pergi ke kantor pusat hanya untuk sebuah tanda tangan, hal ini dapat di selesaikan dengan cepat melalui sebuah tanda tangan digital dan mekanisme pemeriksaan integritas dokumen yang telah ditandatangani melalui sebuah metode *digital signature* (Sulaiman, Ihwani and Rizki, 2016). Dengan demikian, dokumen yang telah ditandatangani dan dikirim akan sama dengan dokumen yang diterima, sehingga keabsahan dan integritas dokumen dapat terjaga.

DSA merupakan salah satu algoritma kriptografi seperti AES, Blowfish, DES,

IDEA, RC4, dan lain-lain (Jasman, Arisandi and Sukri, 2017). DSA merupakan salah satu kriptografi kunci public yang digunakan untuk otentikasi, pengamanan data, dan perangkat anti sangkal (Nurhasanah, 2013). Pada DSA dibutuhkan program khusus untuk membangkitkan kunci dan masalah yang timbul adalah kepercayaan pengguna pada program tersebut. Parameter yang digunakan yaitu menggunakan parameter kunci *public* dan kunci *private* yang dinamis yaitu bernilai berbeda untuk tiap proses pembuatan tanda tangan digital (Yassein *et al.*, 2017). Algoritma DSA yang digunakan untuk penandatanganan pesan, fungsi SHA juga dilibatkan sebagai pembangkit *message digest* dari pesan (Pajčin and Ivaniš, 2011). Algoritma yang digunakan yaitu:

1. Membuat pasangan kunci:
   P: bilangan prima antara panjang 412 dan 1024 bit
   q: faktor bilangan prima p – 1, panjang 160 bit
   g: h-(p-1)/q (mod p) > 1, dan h_ < p - 1
   x <q :kunci rahasia, mempunyai panjang 160 bit
2. Proses pengiriman:
   k < q: bilangan acak
   r = (gk mod p) (mod q)
   s = k-1 (h + xr) (mod q), h = H(m) adalah fungsi Hash satu arah untuk pesan m.
   (r, s): tanda tangan digital
3. Proses pembuktian tanda tangan digital:
   w = s-1 (mod q)
   ul = h x w (mod q)
   u2 = r x w (mod q)
   v = (gu1yu2 (mod p)) (mod q)
   jika v = r, maka tanda tangan digital telah terbukti keasliannya.

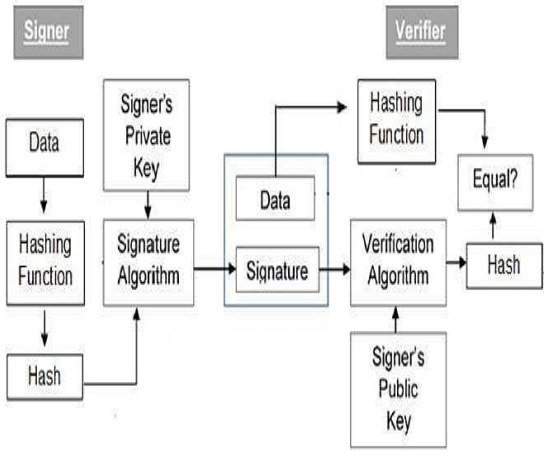
Secara umum, algoritma DSA adalah algoritma yang lazim digunakan, institusi perbankan merupakan salah satu pengguna algoritma ini (Wanda, 2016). Ditinjau dari segi ukuran dan waktu, DSA memiliki kunci yang relatif lebih pendek dibandingkan algortima lainnya (Saputro, Hidayati and Ujianto, 2020) serta memerlukan proses enkripsi/dekripsi dengan waktu yang lebih singkat (Basri, 2015). Selain itu, dikarenakan DSA memiliki bentuk yang ringkas maka proses autentikasi pesan yang dikirim dapat langsung diketahui dari ciphertext yang diterima, karena kunci hanya diketahui oleh pengirim dan penerima pesan saja (Ihwani, 2016). Maka pengirim dan penerima harus dapat memastikan bahwa saluran komunikasi pada saat mengirimkan pesan dan kunci benar-benar aman.

2. METODE PENELITIAN

Penelitian ini dimulai dari tahap indentifikasi masalah. Adapun hal yang diidentifikasi yaitu keraguan akan keaslian dokumen digital yang sudah ditandatangani. Penerima dokumen digital perlu sebuah *digital identifier* yang menyatakan bahwa dokumen tersebut memang berasal dari pengirim, bukan dari pihak lain yang tidak bertanggungjawab.

Setelah permasalahan dapat diidentifikasi, maka tahap berikutnya yaitu analisa kebutuhan dan desain sistem. Hal yang dibutuhkan adalah sebuah antar muka yang dapat dipergunakan dari sisi pengirim maupun penerima dokumen. Pengirim dapat mengirimkan dokumen dan disertai dengan *key* yang telah di *generate* oleh aplikasi. Dari sisi penerima, penerima dapat memverifikasi dokumen dan *key* yang diberikan pengirim. Jika key milik penerima dan pengirim cocok, maka dokumen yang dikirimkan dapat diakui keabsahannya.

Berdasarkan uraian pada pendahuluan dan paragraf sebelumnya maka peneliti menetapkan bahwa algortima yang cocok untuk kebutuhan ini adalah DSA. Berdasarkan beberapa penelitian terdahulu, algoritma ini mampu melakukan *generating* and *identifying* pada key dan dokumen yang dikirim ke penerima sehingga keabsahan dan integritas dokumen dapat terjaga (Nuraeni, Agustin and Muharam, 2018; Al Shaikhli *et al.*, 2012; Lakshmanan and Madheswaran, 2012). Algoritma DSA mempunyai alur kerja seperti pada gambar 1 berikut:



Gambar 1. Alur Kerja Algoritma DSA

Signer sebagai pengirim dokumen melakukan *generating private key*, *key* yang sudah di *generate* akan digunakan sebagai *key generator* untuk menghasilkan *signature* dan *public key* yang akan disertakan bersama dokumen yang dikirim. Dokumen, *key* dan *signature* kemudian dikirim ke penerima atau *verifier*. *Verifier* menerima *key* dan dicocokkan melalui proses verfikikasi bersama dengan *signature* dan dokumen dari *signer*. Bila *signature* milik *key* dan *signer* dari *verifier* cocok atau bernilai sama setelah di verifikasi, maka dokumen yang dikirim dapat dipastikan keabsahan dan integritasnya.

3. HASIL DAN PEMBAHASAN

3.1.Perancangan Sistem

Hal yang terpenting dari algoritma DSA adalah proses *key generation* dan *signature generation* seperti terlihat pada *code* berikut:

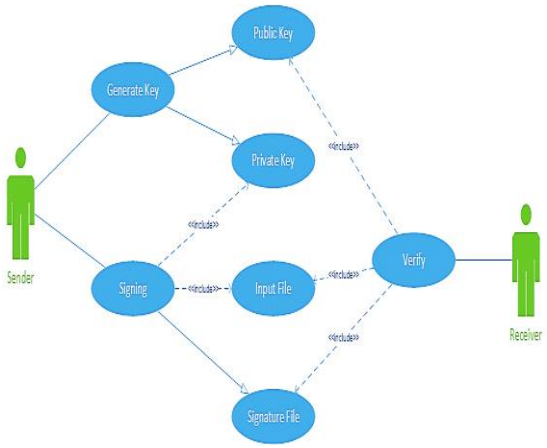
```
P = Prime[7085473]; FactorInteger[p-1]
{{2,1},{3581,1},{17389,1}}

PrimeQ[17389]
True

p = 124540019;q=17389;
h = Random[Integer,{1,p}];h=110217528;
g = Mod[h^((p-1)/q),p];
x = Random[Integer,{1,q-1}];x=12496;
y = Mod[g^x,p];
Print[“Public key is:
p=”,p,”,q=”,q,”,g=”,g,”,y=”,y];
k = Random[Integer,{1,p}];k=9557;
r = Mod[Mod[g^k,p],q];
xx = ExtendedGCD[k,q][2,1];Hm=5246;
s = mod[xx*(Hm+x*r),q];
Print[Digital signature is: r=”,r,”,s=”,s];
```

```
Public key is: p=124540019, q=17389,
g=10083255, y=11994625
Digital signature is: r=34, s=13049
```

Dan proses untuk memverifikasi tandantangan digital pada DSA terlihat pada *code* berikut:

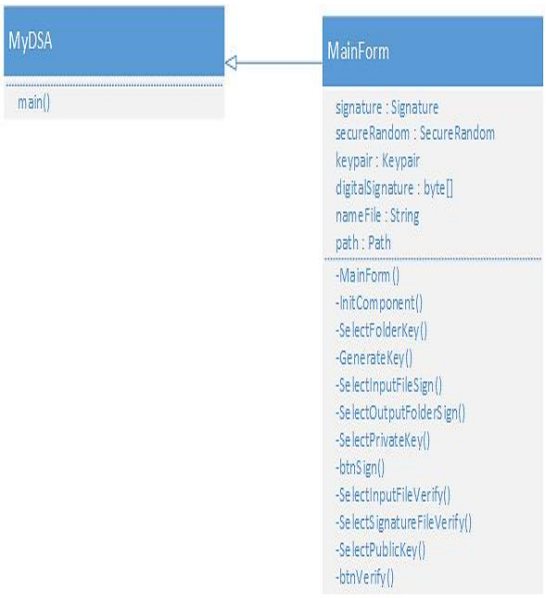
```
If[Or[r>q,r<0,s>q,s<0],Print[“Digital
signature is invalid.”]]
  Ww = ExtendedGCD[s,q][[2,1]];
  u1 = Mod[ww*Hm,q];u2 = Mod[r*ww,q];
  v = Mod[Mod[g^u1*y^u2,p],q];Print[“v=”,v];
  If[v==r,Print[“Digital signature is
valid,”],
    Print[“Digital signature is invalid.”]]

v = 34
Digital signature is Valid.
```

Untuk melihat Hubungan antara *signer* sebagai pengirim, dan *verifier* sebagai penerima digambarkan dalam diagram use case berikut:



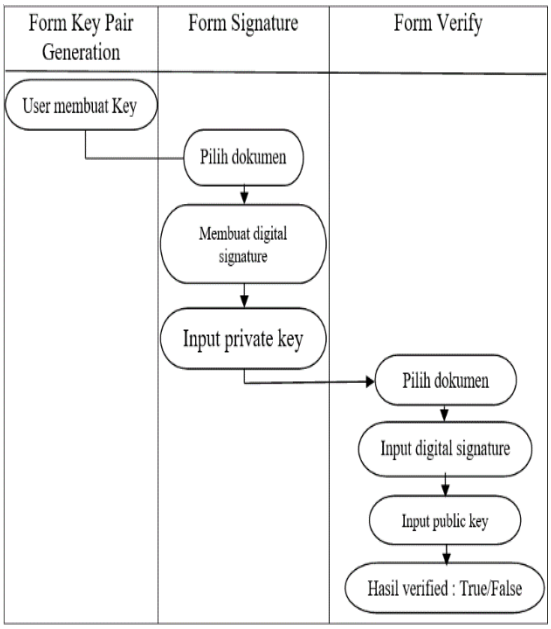
Gambar 2. Diagram Use Case

Dalam menggambarkan struktur dan deskripsi *Class* yang menggambarkan keadaan sistem berbasis DSA yang dirancang, maka dibuat class diagram sebagai berikut:



Gambar 3. Class Diagram Untuk Sistem Berbasis DSA

Untuk menggambarkan logika prosedural, proses bisnis, jalur kerja, dan urutan aktivitas dalam suatu proses dalam sistem berbasis DSA yang dirancang, maka dibuat activity diagram seperti berikut:



Gambar 4. Activity Diagram Sistem Berbasis DSA yang Dirancang

3.2.Implementasi dan Pengujian

Pada tahap awal implementasi, dilakukan key generation untuk membuat private key dan public key seperti terlihat pada gambar 5 berikut.

Gambar 5. Key Generation

Tahap selanjutnya yaitu membuat *signature*, Pada tahapan ini langkah yang harus dilakukan adalah memilih dokumen dan membuat *Signature* yang berfungsi sebagai data sign untuk keabsahan dari dokumen yang akan dikirim dengan cara menggunakan *private key* yang telah di *generate*. Tahap signature ini dapat dilihat pada gambar 6.

Gambar 6. Pembuatan Signature

Untuk melakukan pemeriksaan keabsahan dokumen, maka dilakukan proses *verify* oleh pihak penerima. Proses verify ini dilakukan untuk mengetahui bahwa dokumen masih terjaga integritas nya atau tidak. Langkah ini dilakukan dengan cara memilih dokumen yang ingin di validasi

kemudin menginputkan data signature dan public key yang telah diberikan oleh pihak pengirim dokumen atau signer. Bila data *signature* dan *public* key yang di inputkan cocok dengan dokumen, maka dokumen bernilai *true* yang artinya dokumen masih tetap terjaga integritasnya dan keabsahannya, serta benar bahwa orang yang bersangkutanlah yang menandatangani dokumen tersebut seperti pada gambar 7.

Gambar 7. Hasil Verifikasi Dokumen, Key, dan Signature

Peneliti juga melakukan skenario pengujian dari algoritma DSA ini, yang gunanya untuk mengetahui sejauh mana keberhasilan dan respon yang dihasilkan sistem apabila diuji dengan menggunakan dokumen, *key*, dan *signature* yang sama maupun yang berbeda. Pengujian yang dilakukan melibatkan tiga belas jenis file yang berasal dari aplikasi perkantoran / office. Skenario Pengujian dilakukan dengan mencocokkan antara dokumen, *key* dan *signature*. Bila respon yang dihaslkan adalah *True*, maka bisa dipastikan tidak ada perbedaan antara dokumen, signature dan key yang dikirim hingga sampai ke penerima. Dengan kata lain bahwa dokumen yang dikirim teruji integritasnya. Namun jika ada perbedaan diantara salah satunya (*key*, dokumen atau *signature*), maka respon yang dihasilkan adalah *false*, yang menandakan bahwa dokumen yang dikirim diragukan integritasnya. Hasil pengujian dapat dilihat pada tabel 1.

Tabel 1. Hasil Pengujian Sistem Dengan Algoritma DSA

No	Jenis File	Hal yang di uji	Respon		Positive rate
			True	False	
1	Dokumen dari aplikasi office (.doc, .docx, .rtf, .odt, .xls, .xlsx, .csv, .ppt, .pptx, .pps, ppsx, .pdf, .txt)	Key, dokumen, dan Signature sama	√		100%
2		Key dan dokumen sama, signature berbeda		√	100%
3		Key sama, dokumen dan signature berbeda		√	100%
4		Key, dokumen, dan signature berbeda		√	100%
5		Key dan dokumen, berbeda, signature sama		√	100%
6		Key beda, dokumen dan signature sama		√	100%

4. SIMPULAN

Berdasarkan hasil implementasi dan pengujian yang dilakukan, algoritma DSA berhasil mengidentifikasi kecocokan *key*, dokumen, dan *signature* pada dokumen office yang berasal dari aplikasi perkantoran. Identifikasi kecocokan yang dilakukan dapat menjadi acuan bahwa dokumen yang dikirim dan diterima masih tetap terjaga integritasnya.

5. UCAPAN TERIMAKASIH

Penulis mengucapkan terimakasih kepada program studi S1 Teknik Informatika Universitas Abdurrab atas bantuan materi dan saran dalam proses penelitian ini.

6. DAFTAR PUSTAKA

Basri, B. (2015) 'Pendekatan Kriptografi Hybrid pada Keamanan Dokumen Elektronik dan HypertextTransfer Protocol Secure (HTTPS) (Analisis Potensi Implementasi Pada Sistem Keamanan)', *Jurnal Ilmiah Ilmu Komputer*, 1(2), pp. 31–36. doi: 10.35329/jiik.v1i2.70.

Ihwani, M. (2016) 'Model Keamanan Informasi Berbasis Digital Signature Dengan Algoritma Rsa', *CESSJournal Of Computer Engineering System And Science*, 1(1), pp. 15–20.

Jasman, J., Arisandi, D. and Sukri, S. (2017) 'Rancang Bangun Aplikasi Enkripsi Coding Berbasis Php Program Menggunakan Algoritma Aes', in *2th Celscitech-UMRI 2017 Vol*. Pekanbaru: LP2M-UMRI, pp. 49–61.

Lakshmanan, T. and Madheswaran, M. (2012) 'A novel secure hash algorithm for public key digital signature schemes.', *Int. Arab J. Inf. Technol.*, 9(3), pp. 262–267.

Nuraeni, F., Agustin, Y. H. and Muharam, I. M. (2018) 'Implementasi Tanda Tangan Digital Menggunakan RSA dan SHA-512 Pada Proses Legalisasi Ijazah', *Konferensi Nasional Sistem Informasi (KNSI) 2018*.

Nurhasanah, F. (2013) 'Pembuatan Tanda Tangan Digital Menggunakan Digital Signature Algorithm', *MATHunesa*, 2(2).

Pajčin, B. R. and Ivaniš, P. N. (2011) 'Analysis of Software Realized DSA Algorithm for Digital Signature', *Guest Editorial W*. Citeseer, p. 73.

Pattah, S. H. (2013) 'Pemanfaatan kajian bibliometrika sebagai metode evaluasi dan kajian dalam ilmu perpustakaan dan informasi', *Khizanah al-Hikmah: Jurnal Ilmu Perpustakaan, Informasi, dan Kearsipan*, 1(1), pp. 47–57.

Saputro, T. H., Hidayati, N. H. and Ujianto, E. I. H. (2020) 'Survei Tentang Algoritma Kriptografi Asimetris', *Jurnal Informatika Polinema*, 6(2), pp. 67–72. doi: 10.33795/jip.v6i2.345.

Al Shaikhli, I. F. *et al.* (2012) 'Protection of integrity and ownership of PDF documents using invisible signature', in *2012 UKSim 14th International Conference on Computer Modelling and Simulation*. IEEE, pp. 533–537.

Sulaiman, O. K., Ihwani, M. and Rizki, S. F. (2016) 'Model Keamanan Informasi Berbasis Tanda Tangan Digital Dengan Data Encryption Standard (DES) Algorithm', *InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan*, 1(1), pp. 14–19.

Wahyudi, J. (2012) 'Dokumen elektronik sebagai alat bukti pada pembuktian di pengadilan', *Perspektif*, 17(2), pp. 118–126.

Wanda, P. (2016) 'Model Pengamanan End-to-End pada M-Banking Berbasis Algoritma Kurva Hyper Elliptic', *Jurnal Buana Informatika*, 7(4), pp. 245–254. doi: 10.24002/jbi.v7i4.765.

Widodo, A. W. and Harjoko, A. (2015) 'Sistem Verifikasi Tanda Tangan Off-Line Berdasar Ciri Histogram Of Oriented Gradient (HOG) Dan Histogram Of Curvature (HoC)', *Jurnal Teknologi Informasi dan Ilmu Komputer*, 2(1), pp. 1–10.

Yassein, M. B. *et al.* (2017) 'Comprehensive study of symmetric key and asymmetric key encryption algorithms', in *2017 international conference on engineering and technology (ICET)*. IEEE, pp. 1–7.