Jurnal Riset Mahasiswa Matematika
Volume 1(2) (2021), Pages 51-63
p-ISSN: 2808-1552; e-ISSN: 2808-4926

Implementasi Fungsi Hash MD5 dan Kriptografi Algoritma RSA Pada Pembuatan Tanda Tangan Digital

Annisa Hardiningsih HR[1], Muhammad Khudzaifah[2*], M. Nafie Jauhari[3]

[1,2,3]Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang, Indonesia

Email: xxnisaaaa@gmail.com, khudzaifah@uin-malang.ac.id*, nafie.jauhari@uin-malang.ac.id

ABSTRAK

Penggunaan dokumen elektronik mengalami peningkatan karena kebijakan *Work from Home* (WFH) yang dikeluarkan oleh pemerintah Indonesia. Namun penggunaan dokumen elektronik masih memiliki kelemahan dalam hal keautentikannya. Salah satu cara untuk mengatasinya adalah dengan menerapkan tanda tangan digital. Penelitian ini membahas penerapan tanda tangan digital dengan menggabungkan dua algoritma, yaitu fungsi *hash* MD5 dan kriptografi algoritma RSA pada tiga puluh dokumen elektronik berformat *Portable Document Format* (PDF). Tanda tangan digital dibuat dengan memberikan fungsi *hash* MD5 pada isi dokumen, sehingga menghasilkan *message digest*. Selanjutnya *message digest* dienkripsi menggunakan algoritma RSA. Tanda tangan digital diverifikasi dengan memberikan fungsi *hash* MD5 pada isi dokumen elektronik dan mendekripsi tanda tangan digital dokumen elektronik. Hasil pengujian menunjukkan tanda tangan digital yang dihasilkan adalah berbeda-beda dari setiap dokumen elektronik. Dokumen elektronik yang menghasilkan nilai dekripsi tanda tangan digital dan *message digest* modulo *n* yang sama pada saat verifikasi menunjukkan bahwa dokumen elektronik tidak mengalami perubahan pada isinya. Sebaliknya, dokumen elektronik yang tidak menghasilkan nilai dekripsi dan tanda tangan digital dan *message digest* modulo *n* yang sama pada saat verifikasi menunjukkan bahwa dokumen elektronik telah mengalami perubahan pada isinya. Penelitian ini diharapkan dapat menjadi salah satu upaya dalam meminimalkan risiko pemalsuan dokumen elektronik.

Kata kunci: Dokumen elektronik; MD5; RSA; tanda tangan digital

ABSTRACT

The use of electronic document has increased due to the Work from Home (WFH) policy issued by Indonesian government. However, the use of electronic document still has weaknesses in terms of its authenticity. One way to overcome this is using digital signature. This study discusses the application of digital signature using two algorithms, namely MD5 hash function and RSA algorithm cryptography on thirty electronic documents in Portable Document Format (PDF). The digital signature is created by assigning an MD5 hash function to the document's content, so it obtains the message digest. Then, the message digest is encrypted using the RSA algorithm. The digital signature is verified by assigning an MD5 hash function to electronic document's content and decrypting electronic document's digital signature. The results show that the digital signatures produced are different from each electronic document. Electronic document that

Submitted: 16 November 2021 Reviewed: 18 November 2021 Accepted: 22 November 2021
DOI: https://doi.org/10.18860/jrmm.v1i1.13992

produces the same decryption and message digest modulo n values indicate that the electronic document has not changed its content. On the other hand, electronic documents that do not produce the same decryption and message digest modulo n values indicate that electronic document has changed in its content. This study is expected to be one of the efforts to minimize the risk of falsification of electronic document.

Keywords: digital signature; electronic document; MD5; RSA

PENDAHULUAN

Pandemi COVID-19 di Indonesia membuat pemerintah mengeluarkan kebijakan-kebijakan baru. Salah satunya yaitu kebijakan *Work from Home* (WFH). WFH merupakan suatu konsep di mana para pekerja dapat bekerja secara fleksibel dari rumah karyawan tersebut secara *online* [1]. Kebijakan WFH ini mengakibatkan beralihnya sistem manual ke sistem digital [2], salah satunya yaitu penggunaan dokumen. Dokumen yang sebelumnya berbentuk *print-out*, sekarang berbentuk dokumen elektronik.

Dokumen elektronik merupakan dokumen berbentuk analog, digital, atau sejenisnya yang dibuat, diteruskan, dikirimkan, diterima atau disimpan. Dokumen elektronik dalam penggunaannya memiliki kekurangan dalam hal keotentikannya. Hal ini karena dokumen elektronik sangat mudah untuk diedit [3]. Oleh karena itu dibutuhkan suatu mekanisme guna menjaga keotentikan dokumen elektronik, yaitu tanda tangan digital.

Penelitian sebelumnya mengenai tanda tangan digital pada "*Authentication System for E-Certificate by Using RSA's Digital Signature*" [4] menunjukkan bahwa proses penandatanganan dan pengecekan diselesaikan dengan cepat ketika aplikasi penandatanganan diterapkan dengan Chinese Remainder Theorem (CRT). Selanjutnya pada "*Data Integrity and Security using Keccak and Digital Signature Algorithm* (DSA)" [5] menunjukkan bahwa algoritma Keccak dapat diterapkan pada sistem DSA, serta diperoleh perbandingan waktu eksekusi proses *signing* dan *verifying* antara DSA dan RSA ketika menggunakan algoritma Keccak.

Penelitian ini bertujuan untuk mengetahui hasil tanda tangan digital dan verifikasinya pada dokumen elektronik berformat PDF menggunakan fungsi *hash* MD5 dan kriptografi algoritma RSA. Penelitian ini diharapkan dapat menjadi salah satu upaya dalam meminimalkan risiko pemalsuan dokumen elektronik, khususnya yang berformat PDF.

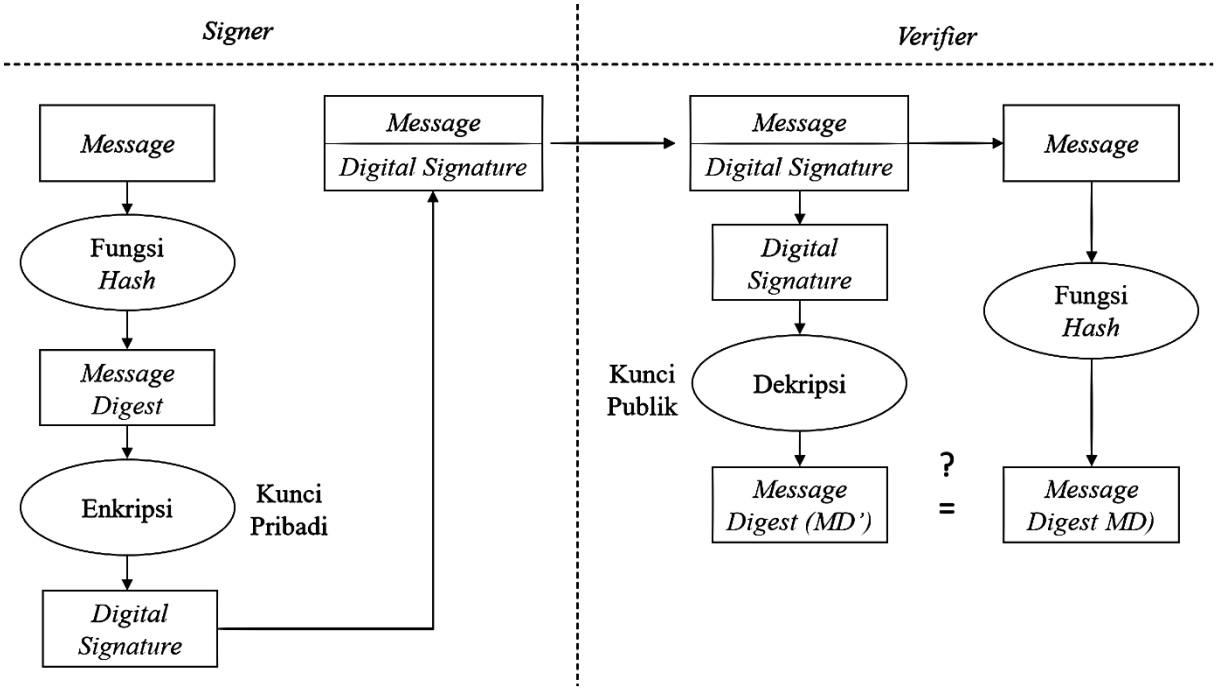
Tanda Tangan Digital

Tanda tangan digital memiliki kemiripan dalam aspek kegunaannya, yaitu menjamin keaslian dan persetujuan dokumen oleh penandatanganan. Namun tanda tangan digital bukanlah tanda tangan yang di-digitalisasi oleh alat *scanner* atau tanda tangan yang dibuat dengan pena elektronik [6]. Tanda tangan digital merupakan suatu nilai kriptografi yang memiliki ketergantungan pada isi pesan dan pengirim pesan [7]. Oleh karena itu, tanda tangan digital dari pesan yang berbeda, walaupun dengan pengirim yang sama, akan memiliki tanda tangan digital yang berbeda [8].

Dalam menjaga validitas data, pengirim diharuskan menandatangani dahulu dokumen yang hendak dikirim. Kemudian penerima dapat memeriksa tanda tangan pada dokumen untuk memastikan bahwa dokumen yang diterima masih asli. Segala aktivitas yang dilakukan oleh pengirim disebut *signing* dan aktivitas yang dilakukan oleh penerima disebut *verifying* [9]. Tanda tangan digital menggunakan algoritma *hashing* akan

membentuk sebuah kombinasi karakter yang khas disebut *message digest* [10] dan *message digest* ini akan dienkripsi menggunakan algoritma kunci pribadi. Hasil enkripsi dari *message digest* inilah yang disebut sebagai tanda tangan digital dari suatu dokumen. Dengan cara ini pengirim bertanggungjawab terhadap isi dokumen dan penerima dapat mengecek keaslian dokumen [11].

Tanda tangan digital dapat dilakukan dengan salah satu dari dua cara, salah satunya yaitu tanda tangan digital dengan penggabungan algoritma *hashing* dan algoritma kunci publik [12]. Tanda tangan digital pada penelitian ini dibuat dengan menggunakan kombinasi algoritma *hashing*, yaitu MD5 dan kriptografi algoritma kunci publik, yaitu RSA. Berikut merupakan skema tanda tangan digital menggunakan algoritma *hashing* dan kriptografi algoritma kunci publik



Gambar 1. Skema Tanda Tangan Digital

Berdasarkan Gambar di atas, proses penandatanganan yang dilakukan oleh pengirim (*signer*) adalah sebagai berikut:

1. Pengirim pesan menghitung nilai *message digest* pesan awal menggunakan fungsi *hash*.
2. *Message digest* dienkripsi menggunakan kunci pribadi pengirim pesan. Hasil enkripsi ini merupakan tanda tangan digital *S*.
3. Hasil tanda tangan digital dilekatkan ke pesan awal, lalu keduanya dikirim kepada penerima.

Proses verifikasi tanda tangan yang dilakukan oleh penerima (*verifier*) adalah sebagai berikut:

1. Penerima menghitung *message digest* pesan awal menggunakan fungsi *hash*.
2. Tanda tangan digital *S* didekripsi menggunakan kunci publik pengirim pesan sehingga menghasilkan *message digest*.
3. Lihat apakah hasil dekripsi *message digest* dan *message digest* pesan awal bernilai sama atau tidak. Jika bernilai sama, maka tanda tangan yang diterima adalah otentik dan isi pesan tidak mengalami perubahan walau satu karakter.

MD5

MD5 merupakan fungsi *hash* yang dibuat oleh Ronal Rivest pada tahun 1991. Pada MD5, pesan masukan berukuran sembarang dan *message digest* yang dihasilkan memiliki panjang 128-bit atau 32 karakter heksadesimal [13]. *Message digest* diperoleh dengan membahkan *padding bits*, menambahkan nilai panjang pesan semula, menginisialisasi *buffer* MD, dan mengolah pesan dalam blok berukuran 512 bit, di mana pengolahan ini terdiri dari empat buah putaran yang setiap putarannya terdapat 16 kali operasi dasar MD5 [12].

RSA

Pada tahun 1977, Ron Rivest, Adi Shamir, dan Leonard Adleman membuat algoritma RSA. Algoritma RSA merupakan algoritma kunci publik yang dalam proses pengerjaannya membutuhkan konsep matematika, yaitu Faktor Persekutuan Terbesar (FPB), algoritma Euclid, relatif prima, bilangan prima, aritmetika modular, dan kekongruenan. Keamanan algoritma RSA dilihat dari susahnya memfaktorkan bilangan-bilangan prima besar dari proses pembangkitan sepasang kunci [14]. Hasil dari algoritma ini adalah kunci publik (e, n) yang digunakan untuk enkripsi dan kunci pribadi (d, n) yang digunakan untuk dekripsi dengan e, d, dan n merupakan bilangan bulat positif [15]. Kunci yang digunakan pada enkripsi dan dekripsi pada pesan biasa berbeda dengan kunci yang digunakan pada tanda tangan digital. Pada tanda tangan digital, kunci pribadi (d, n) digunakan untuk mengenkripsikan pesan dan kunci publik (e, n) digunakan untuk mendekripsi pesan.

METODE

Tahapan-tahapan penelitian dilakukan sebagai berikut:

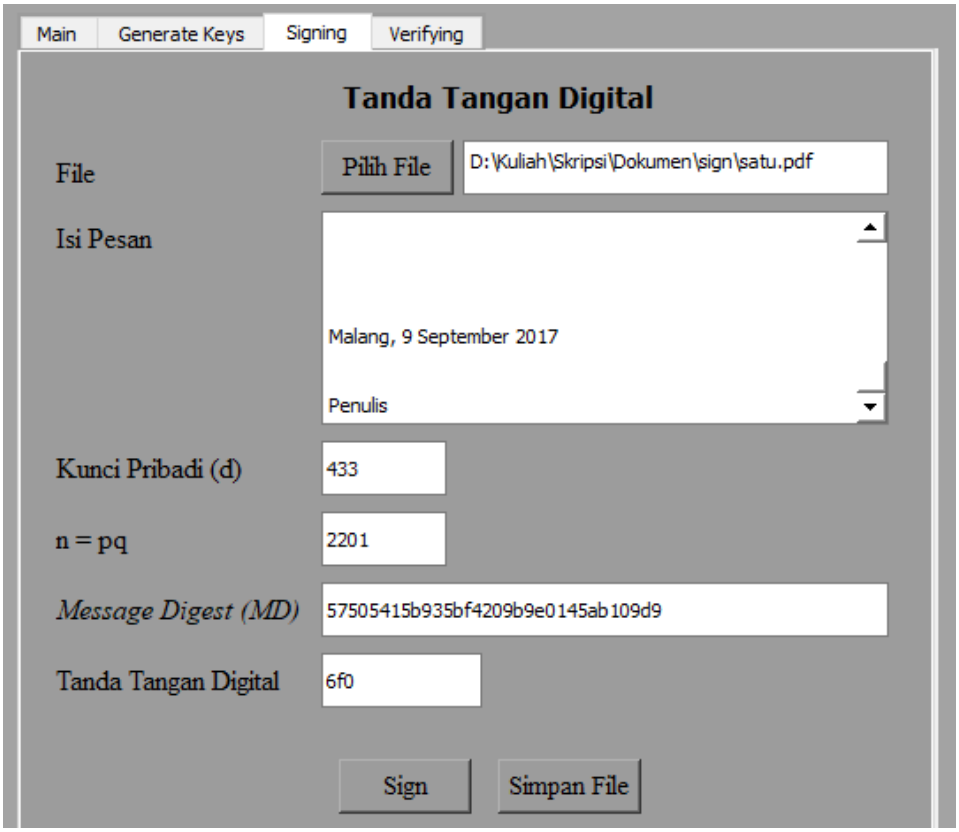
1. Membuat tanda tangan digital menggunakan fungsi *hash* MD5 dan algoritma enkripsi kriptografi RSA dengan prosedur sebagai berikut:
   a) Meng*input*kan pesan (teks) berformat PDF.
   b) Mengubah pesan menjadi *message digest* dengan cara sebagai berikut:
      i. Menambahkan *padding bits* pada pesan dengan menambahkan 1 dan sejumlah 0 sampai panjang pesan kongruen dengan 448 modulo 512.
      ii. Menambahkan 64-bit pada pesan yang telah diberi *padding bits*.
      iii. Inisialisasi *buffer* MD, yaitu A = 67452301, B = EFCADB89, C = 98BADCFE, dan D = 10325476.
      iv. Pemecahan pesan menjadi $Y_0, Y_1, Y_2, \dots, Y_{L-1}$ berukuran 512 bit.
      v. Melakukan proses MD5 sebanyak empat buah putaran.
   c) Mengenkripsi *message digest* dengan kriptografi algoritma RSA sebagai berikut:
      i. Memilih bilangan prima p dan q.
      ii. Menghitung nilai $n = pq$.
      iii. Menghitung nilai $\phi(n) = (p-1)(q-1)$
      iv. Memilih kunci publik e, dimana $\gcd(e, \phi(n)) = 1$
      v. Membangkitkan kunci pribadi dengan persamaan $ed \equiv 1 \ (mod \ (\phi(n))$
      vi. Mengubah *message digest* yang diperoleh ke dalam bentuk desimal.
      vii. Mengenkripsi *message digest* dalam bentuk desimal menggunakan kunci pribadi d dengan rumus $c = m^d \bmod n$.

viii. Mengubah hasil enkripsi *message digest* yang berbentuk desimal ke dalam bentuk heksadesimal.

2. Memverifikasi tanda tangan digital dengan menggunakan fungsi *hash* MD5 dan algoritma dekripsi kriptografi RSA sebagai berikut:
   a) Menghitung *message digest* dari pesan yang diperoleh dengan fungsi *hash* MD5 seperti pada langkah 1 b).
   b) Mengubah hasil *message digest* ke dalam bentuk desimal.
   c) Mengubah hasil tanda tangan digital yang diperoleh ke dalam bentuk desimal.
   d) Meng*input*kan kunci publik e.
   e) Mendekripsi tanda tangan digital yang diperoleh dengan algoritma dekripsi dengan rumus $m = c^e \bmod n$.
   f) Membandingkan hasil *message digest* dan dekripsi tanda tangan digital.
3. Membuat program menggunakan bahasa pemrograman Python.
4. Mengimplementasikan program pada tiga puluh dokumen elektronik yang digunakan.

HASIL DAN PEMBAHASAN

Proses penandatanganan dokumen elektronik menggunakan fungsi *hash* MD5 dan kriptografi algoritma RSA pada dokumen pertama menggunakan program yang telah dibuat dapat dilihat pada Gambar 2 berikut:



Gambar 2. Hasil Tanda Tangan Digital Dokumen Pertama

Berdasarkan Gambar 2, diperoleh *message digest* adalah

$$57505415b935bf4209b920145ab109d9$$

Selanjutnya *message digest* ini diubah ke dalam bentuk desimal, sehingga diperoleh

116059924825488289768352942606727055833

Message digest dalam bentuk desimal ini akan dienkripsi menggunakan kunci pribadi. Kunci pribadi diperoleh dengan cara sebagai berikut:

1. Memilih nilai $p = 31$ dan $q = 71$.
2. Menghitung nilai n

$$\begin{aligned} n &= pq \\ &= 31 \times 71 \\ &= 2201 \end{aligned}$$

3. Menghitung nilai $\phi(n)$ dengan persamaan

$$\begin{aligned} \phi(n) &= (p-1)(q-1) \\ &= 30 \times 70 \\ &= 2100 \end{aligned}$$

4. Dipilih $e = 97$, dimana $\gcd(e, \phi(n)) = 1$. Lalu menghitung d dari persamaan

$$ed \equiv 1 \bmod (\phi(n))$$

$$d \equiv \frac{1 \bmod (\phi(n))}{e}$$

$$= \frac{1 + (k \times 2100)}{97}$$

dengan mencoba nilai-nilai $k = 1, 2, 3, \ldots$, diperoleh nilai $d = 433$. Maka diperoleh kunci pribadi untuk enkripsi *message digest*, yaitu $d = 433$.

Message digest dienkripsi dengan

$c = m^d \bmod n$

$= 116059924825488289768352942606727055833^{433} \bmod 2201$

$= 1776$

Hasil enkripsi yang diperoleh adalah 1776 yang dalam bentuk heksadesimal adalah $6f0$. Maka tanda tangan digital dokumen pertama adalah $6f0$.

Hasil tanda tangan digital dari tiga puluh dokumen elektronik dapat dilihat pada Tabel berikut ini:

Tabel 1. Hasil Tanda Tangan Digital dari Tiga Puluh Dokumen Eletronik

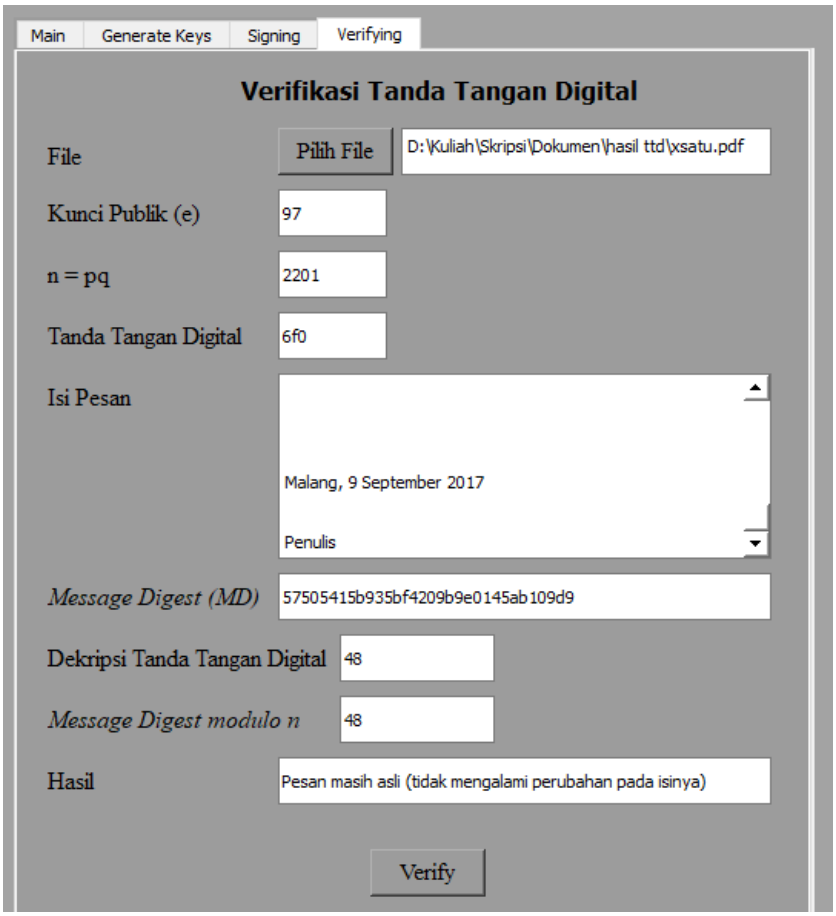
Nama Dokumen	*Message Digest*	Kunci Pribadi (d, n)	Tanda Tangan Digital
Satu	57505415b935bf4209b9e0145ab109d9	$433,\ 2201$	$6f0$
Dua	3a02a5a773c9596470a91bd2277dc0cd	$2275,\ 2701$	$a77$
Tiga	2004c1f59373182df37b15132a3ed255	$5777, 9797$	$132c$
Empat	8148710b4171f2206b579c0b9c9ff0a2	$123, 1837$	122
Lima	9217013676aeacd967bfa2f3a6917569	$361, 3233$	$1ec$
Enam	aec44c5ba60336a61787da7e882d3461	$3037, 6283$	$af7$
Tujuh	d2ca5b4f7b382d1a3ee0da973bc01c7b	$797,\ 3901$	$67e$
Delapan	eecd6169923cfa4da41ad1c8172c68c4	$117,\ 391$	20

Nama Dokumen	*Message Digest*	Kunci Pribadi (d, n)	Tanda Tangan Digital
Sembilan	faade28365e90f633b43e5ebb51aabe6	10909, 13561	266e
Sepuluh	0b0b2f73869527a81903b8021e30d9d9	8251, 11581	aae
Sebelas	c45b78ecc3e0cb8efd72e2b2af5f3ca6	3799, 7807	31e
Duabelas	4270c6ae8ee8bc184e0895586ef5edef	4451, 8371	1a89
Tigabelas	2e451bedc659a4970afb0e3c9f410fad	1859, 4009	14b
Empatbelas	ad1ca2962c7be7738601463ac180331d	13279, 16867	2ba2
Limabelas	7b0486e17b6256452e99e724a144448b	5713, 17233	2169
Enambelas	b63aeab57d006a0a916c12361c92b6c4	367, 2149	825
Tujuhbelas	4beb5c0f9797193897982916570fa445	1889, 2603	92
Delapanbelas	2571d496903bd67cef1ca695174a3022	827, 6739	4a0
Sembilan belas	e0e110ed1ffe640d4dbc9c5cb6ec4cbf	743, 5959	a82
Duapuluh	45d254fd2e83db58c3b8902315e65157	5393, 9599	965
Duapuluh satu	869f352382abfa98ab2c07f540a65e2e	8513, 20987	2c2b
Duapuluhdua	108f049431bc40c3c1eee2f530726ac4	8273, 9841	1d18
Duapuluhtiga	4f08e5b44d49c26c087dd4782654f6f2	4117, 8549	e6f
Duapuluh empat	c8fb70c17723bc26ebaae5e0b3dc52eb	47, 721	10f
Duapuluh lima	35f7c674b56b1e7b387e40cd05bb162b	1891, 4399	e71
Duapuluh enam	8170167029d09a0cfeb875a3610889bc	637, 1739	616
Duapuluh tujuh	7f6fc3fc79ba1a9a11c99af52bf2a022	8303, 13957	238c
Duapuluh delapan	88e2ea7a51199b91b88dc8f081194247	415, 3683	58b
Duapuluh	c475733c4391924e5bc2b4b456c7b68	1149, 6931	1abc

Nama Dokumen	*Message Digest*	Kunci Pribadi (d, n)	Tanda Tangan Digital
sembilan	1		
Tigapuluh	910edae55304e1ebc8da24702499676d	771, 11233	$20a7$

Berdasarkan hasil pengujian pada Tabel 1, dapat dilihat bahwa tanda tangan digital yang dihasilkan berbeda-beda dari setiap dokumen. Hal ini karena tanda tangan digital yang dihasilkan bergantung pada nilai *message digest* dan kunci pribadi yang digunakan. Selanjutnya dokumen yang telah diberi tanda tangan digital akan dikirimkan kepada penerima untuk diverifikasi menggunakan kunci publik.

Proses verifikasi dokumen elektronik menggunakan fungsi *hash* MD5 dan kriptografi algoritma RSA pada dokumen pertama menggunakan program yang telah dibuat dapat dilihat pada Gambar 3 berikut:



Gambar 3. Hasil Verifikasi Dokumen Pertama

Berdasarkan Gambar 3, tanda tangan digital, yaitu $6f0$ yang dalam desimal adalah 1776 didekripsi menggunakan kunci publik. Kunci publik yang diperoleh dari proses penandatangan dokumen pertama yang bernama satu.pdf, yaitu $e = 97$. Tanda tangan digital didekripsikan dengan

$m = c^e \bmod n$

$= 1776^{97} \bmod 2201$

$= 72$

Hasil dekripsi yang diperoleh adalah 72 yang dalam heksadesimal adalah 48. Selanjutnya isi dokumen akan dikenakan fungsi *hash*, sehingga diperoleh *message digest*

$$57505415b935bf4209b9e0145ab109d9$$

Selanjutnya perhitungan *message digest* modulo n, dengan n bernilai 2201 menghasilkan nilai 48. Hasil dekripsi tanda tangan digital dan *message digest* menghasilkan nilai yang sama, yaitu 48.

Hasil pengujian verifikasi tanda tangan digital dari tiga puluh dokumen elektronik dapat dilihat pada Tabel 2 berikut:

Tabel 2 Hasil Verifikasi dari Tiga Puluh Dokumen Elektronik

Nama Dokumen	Message Digest	Kunci Publik (e, n)	Tanda Tangan Digital	Dekripsi Tanda Tangan Digital	*Message Digest* modulo n
xsatu	57505415b935bf4209b9e0145ab109d9	97, 2201	$6f0$	48	48
xdua	3a02a5a773c9596470a91bd2277dc0cd	139, 2701	$a77$	898	898
xtiga	2004c1f59373182df37b15132a3ed255	113, 9797	132c	2344	2344
xempat	8148710b4171f2206b579c0b9c9ff0a2	27, 1837	122	299	299
xlima	9217013676aeacd967bfa2f3a6917569	121, 3233	1ec	35a	35a
xenam	aec44c5ba60336a61787da7e882d3461	133, 6283	$af7$	$d1c$	$d1c$
xtujuh	00f2f16d5e31452efa09bc55e6e5ef9f	549, 3901	$67e$	459	44
xdelapan	22ecc56816ef1677a2c8273efd077725	349, 391	20	24	131
xsembilan	9e3c499e9f0f3b8bf6040ee6689fb987	89, 13561	$266e$	3093	$2d86$
xsepuluh	dff3fe1235562e1649c6f068d6ccd2ef	211, 11581	aae	$c46$	$e40$
xsebelas	6ded9e7b97014ab8af1f0dba4a90522b	199, 7807	$31e$	$2d4$	$6e7$
xduabelas	90f62379b698ccb14596ea6082f98f8f	251, 8371	$1a89$	ada	83
xlima	45d05d021e65ed70f6fd773c069aac45	3719, 4009	$14b$	$7b1$	$8ad$
xenam	097a161d04ff3ef878231473c862471f	3319, 16867	$2ba2$	$1c88$	$3a7a$
xlimabelas	ba9bfc4fd0ebad48ddcbbd51151f1b36	157, 17233	3508	2445	$b6a$
xenambelas	be40cb309db2450426963405ceb51222	1831, 2149	$4b0$	$2e2$	700
xtujuhbelas	9aae020fbf85dad179cf75415b0ee0c7	2321, 2603	92	$6d7$	319
xdelapanbelas	9db4d32322c5cddecdaab9d4c0ec2236	435, 6739	$4a0$	$11aa$	$104e$

Nama Dokumen	Message Digest	Kunci Publik (e, n)	Tanda Tangan Digital	Dekripsi Tanda Tangan Digital	*Message Digest* modulo n
xsembilanbelas	9f48e5eb0e726dc4920c6188833842b7	$1007,\ 5959$	$a82$	224	47
xduapuluh	99f6500771b08bdd2c004230cce5cb98	$257,\ 9599$	965	$1b10$	$dd1$
xduapuluhsatu	6b231e65a3a1157ad55e75b74fe806d6	$617, 20987$	$2c2b$	913	4933
xduapuluhdua	391ee7b8c53da3f476c7985857568ae7	$545, 9841$	1710	269	$8b9$
xduapuluhtiga	6b70afa9c5e09ebc145a5518b8d08f55	$193, 8549$	1807	1543	1505
xduapuluhempat	17c9d5d9199a66bedb1c13cbc650a94a	$599, 721$	217	$24a$	255
xduapuluhlima	2ec83a6d69ce5de13af3dc6da3bf6305	$115,\ 4399$	$e71$	$26c$	$96c$
xduapuluhenam	b02c772c99017d26260764b2af63d523	$13,\ 1739$	616	$3c5$	652
xduapuluhtujuh	2daca0119af6f4c557091bbda0228eaf	$207, 13957$	$238c$	2768	ef
xduapuluhdelapan	231b433ee003b7a4b716f56222b2b6e7	$3511, 3683$	$58b$	dfa	$8e7$
xduapuluhsembilan	78f46ee71357f021deb7df0d7360f302	$29, 6931$	$1abc$	$9be$	$119c$
xtigapuluh	820c6681994fd814d790d7e9cad00059	$71, 11233$	$20a7$	$1b2a$	$b10$

Berdasarkan hasil pengujian pada Tabel 2, dapat dilihat bahwa dokumen pertama sampai dokumen keenam menghasilkan nilai dekripsi tanda tangan digital dan *message digest* modulo *n* yang sama, sedangkan pada dokumen ketujuh sampai dokumen ketiga puluh menghasilkan nilai dekripsi tanda tangan digital dan *message digest* modulo *n* yang tidak sama. Hal ini karena dokumen ketujuh sampai ketiga puluh telah mengalami perubahan pada isinya, yaitu berupa perubahan, penambahan, pengurangan, dan pembalikan kata ataupun angka.

KESIMPULAN

Setiap dokumen elektronik menghasilkan tanda tangan digital yang berbeda-beda satu sama lain. Hal ini karena tanda tangan digital yang dihasilkan sangat bergantung pada nilai *message digest* yang diperoleh dari isi pesan dan kunci pribadi yang digunakan pengirim untuk mengenkripsi *message digest*. Dokumen elektronik yang tidak mengalami perubahan pada isinya menghasilkan nilai dekripsi tanda tangan digital dan *message digest* modulo *n* bernilai sama. Sedangkan dokumen elektronik yang telah mengalami perubahan pada isinya menghasilkan nilai dekripsi dan *message digest* modulo *n* bernilai tidak sama. Hal ini menunjukkan bahwa segala bentuk perubahan pada isi dokumen akan merubah nilai *message digest*.

DAFTAR PUSTAKA

[1] Suspahariati dan R. Susilawati, "Penerapan Sistem WFH (Work From Home) dan Dampaknya terhadap Kinerja Staf dan Dosen Unipdu Jombang selama Pandemi Covid-19," *Jurnal Manajemen dan Pendidikan Islam,* vol. VI, no. 2, pp. 229-240, 2020.

[2] T. Yuniati dan M. F. Sidiq, "Literature Review: Legalisasi Dokumen Elektronik Menggunakan Tanda Tangan Digital sebagai Alternatif Pengesahan Dokumen di Masa Pandemi," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi),* vol. IV, no. 6, pp. 1058-1059, 2020.

[3] M. Benedict, M. A. Budiman dan D. Rachmawati, "Perbandingan Algoritma Message Digest 5 (MD5) Dan GOST Pada Hashing File Dokumen," *Jurnal Teknik Informatika Kaputama (JTIK),* vol. I, no. 1, pp. 50-61, 2017.

[4] K. Somsuk dan M. Thakong, "Authentication System for E-certificate by Using RSA's Digital Signature," *TELKOMNIKA (Telecommunication Computing Electronics and Control),* vol. XVIII, no. 6, pp. 2948-2955, 2020.

[5] M. A. Nazal, R. Pulungan dan M. Riasetiawan, "Data Integrity and Security using Keccak and Digital Signature Algorithm (DSA)," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems),* vol. XIII, no. 3, pp. 273-282, 2019.

[6] E. Army, Bukti Elektronik dalam Praktik Peradilan, Jakarta: Sinar Grafika, 2020.

[7] I. P. A. Swastika dan I. G. L. A. R. Putra, Audit Sistem Informasi dan Tata Kelola Teknologi Informasi, Yogyakarta: CV ANDI OFFSET, 2016.

[8] D. Ariyus, Pengantar Ilmu Kriptografi: Teori Analisis dan Implementasi, Yogyakarta: CV ANDI OFFSET, 2008.

[9] R. A. Azdy, "Tanda tangan Digital Menggunakan Algoritme Keccak dan RSA," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI),* vol. V, no. 3, pp. 184-191, 2016.

[10] Y. Anshori, A. Y. E. Dodu dan D. M. P. Wedanata, "Implementasi Algoritma Kriptografi Rivest Shamir Adleman (RSA) pada Tanda Tangan Digital," *Techno.Com,* vol. XVIII, no. 2, pp. 110-121, 2019.

[11] E. C. Prabowo dan I. Afrianto, "Penerapan Digital Signature Dan Kriptografi Pada Otentikasi Sertifikat Tanah Digital," *Komputa : Jurnal Ilmiah Komputer dan Informatika,* vol. II, no. 6, pp. 83-90, 2017.

[12] R. Munir, Kriptografi, Bandung: Informatika, 2019.

[13] Sumarno, I. Gunawan, H. S. Tambunan dan E. Irawan, "Analisis Kinerja Kombinasi Algoritma Message-Digest Algortihm 5 (MD5), Rivest Shamir Adleman (RSA) dan Rivest Cipher 4 (RC4) Pada Keamanan

E-Dokumen," *Jurnal Sistem Informasi dan Ilmu Komputer Prima (JUSIKOM PRIMA),* vol. II, no. 1, pp. 41-48, 2018.

[14] R. H. Sianipar, Java untuk Kriptografi, Yogyakarta: Penerbit Andi, 2017.

[15] Pahrizal dan D. Pratama, "Implementasi Algoritma Rsa Untuk Pengamanan Data Berbentuk Teks," *Pseudocode,* vol. III, no. 1, pp. 44-49, 2016.

[16] R. M. McLeod, K. Ranson dan L. Biehl, The generalized Riemann integral, JSTOR, 1980.

[17] C. Godsil dan G. F. Royle, Algebraic graph theory, vol. 207, Springer Science \& Business Media, 2013.

[18] A. Gara, M. A. Blumrich, D. Chen, G.-T. Chiu, P. Coteus, M. E. Giampapa, R. A. Haring, P. Heidelberger, D. Hoenicke, G. V. Kopcsay dan others, "Overview of the Blue Gene/L system architecture," *IBM Journal of Research and Development,* vol. 49, no. 2, pp. 195-212, 2005.

[19] J. France, J. H. Thornley dan others, Mathematical models in agriculture., Butterworths, 1984.

[20] F. E. Browder, "Nonexpansive nonlinear operators in a Banach space," *Proceedings of the National Academy of Sciences,* vol. 54, no. 4, pp. 1041-1044, 1965.

[21] M. J. Berger dan J. Oliger, "Adaptive mesh refinement for hyperbolic partial differential equations," *Journal of computational Physics,* vol. 53, no. 3, pp. 484-512, 1984.

[22] R. H. Sianipar, Java untuk Kriptografi, Yogyakarta: Penerbit ANDI, 2017.