Jurnal TICOM: *Technology of Information and Communication*
Volume 10, Nomor 3, Mei 2022 Hal: 181-186 E-ISSN: 2962-7982

Implementasi Tanda Tangan Digital Menggunakan Algoritme RSA dan SHA-512 dengan *Salt* Berbasis Web

Agnes Aryasanti[1], Mardi Hardjianto[2*], Goenawan Brotosaputro[3], Ririt Roeswidiah[4]

[1,3]Fakultas Teknologi Informasi, Sistem Informasi, Universitas Budi Luhur Jakarta, Indonesia
[2]Fakultas Teknologi Informasi, Teknik Informatika, Universitas Budi Luhur Jakarta, Indonesia
Jl. Raya Ciledug, Petukangan Utara, Kebayoran Lama, Jakarta Selatan 12260
E-mail: [1]agnes.aryasanti@budiluhur.ac.id, [2*]mardi.hardjianto@budiluhur.ac.id, [3]goenawan.brotosaputro@budiluhur.ac.id, [4]ririt.roeswidiah@budiluhur.ac.id
(*: corresponding author)

Abstrak—**Dokumen adalah surat tercetak atau tertulis yang digunakan sebagai alat bukti. Seiring dengan perkembangan teknologi, saat ini dokumen juga diterbitkan dalam bentuk digital tidak hanya dalam bentuk fisik atau cetak. Dokumen digital memiliki fungsi yang sama dengan dokumen fisik, namun memiliki kelemahan yaitu sangat rentan terhadap kemungkinan dimodifikasi, dipindahtangankan oleh siapapun dan sulitnya pembuktikan keaslian dokumen. Salah satu cara untuk mencegahnya adalah dengan membuat tanda khusus untuk memastikan bahwa dokumen tersebut asli. Tanda khusus ini adalah tanda tangan digital. Penelitian ini menggunakan fungsi hash SHA-512 yang diulang sebanyak sepuluh kali dan disertai dengan *salt* agar tidak mudah untuk dimodifikasi oleh orang lain. Enkripsi kunci publik yang digunakan adalah algoritme RSA. Hasil pengujian menunjukkan bahwa sistem tanda tangan digital menggunakan algoritme RSA dan SHA-512 dengan *salt* lebih dapat mengamankan dokumen penting dari perubahan oleh orang yang tidak bertanggung jawab. Jangka waktu pembuatan tanda tangan digital tidak terlalu lama. Ukuran file sebesar 5 megabyte hanya membutuhkan waktu sekitar 0,07 detik.**

Kata kunci — **RSA, Tanda Tangan Digital, SHA-512**

Abstract— ***A document is a written or printed letter used as evidence. Along with the development of technology, currently documents are not only published in physical or printed form, but also in digital form. Digital documents have the same function as physical documents, but have a weakness that is very vulnerable to the possibility of modification, being transferred by anyone and challenging to prove the document authenticity. One way to prevent this is to make a special mark to ensure that the document is genuine. This particular signature is a digital signature. This research uses the hash function SHA-512 which is repeated ten times and is accompanied by salt so others do not easily modify it. The public key encryption used is the RSA algorithm. The test results show that the digital signature system using the RSA and SHA-512 algorithms with salt is better able to secure important documents from changes by irresponsible people. The time period for creating a digital signature is not too long. A file size of 5 megabytes only takes about 0.07 seconds.***

Keywords— ***RSA, Digital Signature, SHA-512***

I. PENDAHULUAN

Dokumen adalah surat yang tercetak atau tertulis yang dapat digunakan sebagai bukti keterangan, seperti akte lahir, surat nikah, ijazah, sertifikat, dll. Kelemahan dokumen tercetak antara lain dalam hal penyimpanannya, rawan kerusakan fisik, ketidakefisienan dalam transmisi, dan masalah keamanan. Seiring kemajuan teknologi dan kelemahan kertas, dokumen saat ini tidak hanya diterbitkan dalam bentuk fisik saja, namun juga dalam bentuk digital. Dokumen digital merupakan dokumen yang dapat dibaca melalui perangkat eletronik tanpa adanya kertas fisik. Dokumen digital memiliki fungsi yang sama dengan dokumen fisik.

Salah satu kelemahan dari dokumen digital adalah mudahnya diubah, diperbaiki dan dipindahkan oleh siapapun. Dokumen digital bila dikirimkan via internet, sangat beresiko terhadap kemungkinan modifikasi, serta sulit untuk membuktikan keaslian dokumen tersebut. Pengirim dapat dengan mudah membantah bahwa dialah yang telah menulis atau mengirim dokumen tersebut. Penggunaan file digital juga berisiko terhadap perubahan data oleh pihak yang tidak bertanggung jawab, sehingga timbul kekhawatiran adanya tindak pemalsuan dokumen. Salah satu cara untuk mengatasinya adalah dengan dibuatkan suatu tanda khusus untuk membuktikan bahwa dokumen tersebut valid.

Sistem kriptografi dapat digunakan untuk memeriksa masalah keaslian suatu dokumen. Salah satu teknik kriptografi yang dapat digunakan adalah tanda tangan digital. Istilah tanda tangan digital mulai dikenal pada tahun 1991 saat dimana penggunaan komputer makin massif [1]. Tanda tangan digital merupakan mekanisme otentikasi yang memungkinkan pembuat pesan menambahkan sebuah kode sebagai tanda tangannya [2]. Tanda tangan digital bukanlah suatu tanda tangan yang didapat melalui proses *scanner*, namun diperoleh dari suatu nilai kriptografi yang ditentukan dari sebuah pesan dan pemilik pesan [3]. Bila sebuah dokumen tersebut disimpan ulang, tanda tangan digital yang telah dibuat akan hilang. Hal ini mengakibatkan dokumen

menjadi tidak asli lagi. Keuntungan menggunakan tanda tangan digital karena menggunakan teknik matematika dan kriptografi sehingga dapat menjaga kerahasiaan, integritas dan anti penyangkalan [4].

Tanda tangan digital memiliki kekuatan hukum setara dengan tanda tangan basah. Kekuatan dari tanda tangan basah terletak pada proses hashing. Hashing berperan memastikan integritas file, data ataupun informasi berbasis digital. Sebagai contoh, bila sebuah file digital dihitung nilai hash-nya akan menghasilkan suatu nilai yang merupakan ciri dari file digital tersebut. Apabila terjadi perubahan pada file digital walau satu bit saja, maka bila dihitung kembali nilai hash-nya akan terjadi perubahan. Nilai hash ini dapat digunakan sebagai bukti integritas file[5].

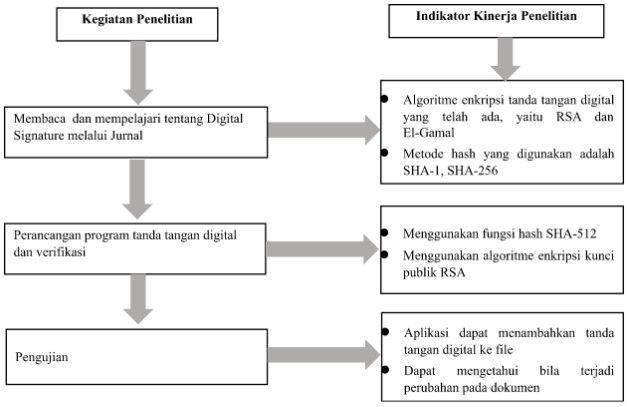
Pada penelitian sebelumnya ada yang menggunakan tanda tangan digital untuk memverifikasi data *cloud*. Algoritme yang digunakan adalah Keccak dan Digital Signature Algorithm (DSA). Pada algoritme DSA fungsi hash masih menggunakan *Secure Hash Algorithm* (SHA-1). Saat ini algoritme SHA-1 dianggap tidak aman, sehingga diganti dengan algoritme Keccak[6]. Algoritme El-Gamal dan SHA-1 juga dapat digunakan untuk membuat tanda tangan digital. File yang akan diberi tanda tangan digital dihitung nilai hash-nya dengan SHA-1 lalu nilai hash ini dienkripsi dengan El-Gamal[3]. Pemanfaatan tanda tangan digital juga dapat digunakan untuk menguji keutuhan dan otentikasi dokumen sertifikat tanah digital, serta dapat mendeteksi perubahan dokumen sertifikat tanah digital dari hasil manipulasi oleh orang yang tidak berhak. Peneliti menggunakan SHA-256 untuk algoritme hash dan RSA untuk algoritme kunci publik[7]. Peneliti lain ada juga yang menggunakan DSA, dimana fungsi hash menggunakan SHA-1 dan enkripsi menggunakan El-Gamal. Enkripsi yang dilakukan mengambil bilangan p = 59419 dan q = 3301[8]. Selain enkripsi RSA dan El-Gamal, algoritme Schnorr dan fungsi hash dapat digunakan untuk membuat tanda tangan digital[9]. Algoritme kriptografi Data Encryption Standard dapat juga digunakan dalam pembuatan tanda tangan digital. Proses enkripsi dilakukan dua kali, yaitu untuk membuat tanda tangan digital dan untuk mengenkripsi dokumen yang sudah terdapat tanda tangan digital [10]. Selain menggunakan fungsi hash SHA, tanda tangan digital dapat pula menggunakan fungsi hash MD5 [11]–[14]

Dalam penelitian ini, kami mencoba untuk membuat sebuah model tanda tangan digital mengunakan algoritme kriptografi RSA dan fungsi hash SHA-512. Perbedaan dengan penelitian sebelumnya, kami melakukan pengulangan fungsi hash SHA-512 sebanyak sepuluh kali. Dalam setiap pengulangan, kami menukar posisi nilai hash antara 128 bit kiri dengan 128 bit kanan. Nilai hash yang telah ditukar posisinya ini ditambahkan *salt*. Hal ini dimaksudkan untuk mempersulit orang lain yang ingin mencoba untuk mengubah keaslian pesan.

II. METODE PENELITIAN

A. Langkah-Langkah Penelitian

Tahapan kegiatan penelitian ditunjukkan pada Gambar 1. Pada langkah awal, kami melakukan pengkajian terhadap penelitian terdahulu mengenai tanda tangan digital. Pembuatan tanda tangan digital melalui dua proses, yaitu mencari dan melakukan enkripsi nilai hash. Banyak penelitian sebelumnya menggunakan fungsi hash *Secure Hash Algorithm*, baik itu SHA-1 maupun SHA-256. Proses enkripsi menggunakan algoritme kunci publik, yaitu RSA atau El-Gamal.



Gambar 1. Langkah-langkah Penelitian

Selanjutnya kami merancang program tanda tangan digital dan verifikasi menggunakan fungsi hash SHA-512 dan enkripsi kunci publik RSA. Sebagai algoritme kunci publik, RSA memiliki dua kunci, yaitu kunci publik dan privat. Kunci publik boleh disebar ke semua orang, sedangkan kunci privat hanya boleh diketahui oleh pemilik kunci. Algoritme RSA terdiri dari tiga proses, yaitu pembentukan kunci, enkripsi dan dekripsi.

B. Proses Pembentukan Kunci

Berikut adalah skema pembentukan kunci publik dan privat algoritme RSA[15]:

Pertama, kami memilih dua buah bilangan prima yang sangat besar, yaitu p dan q. Kedua bilangan prima ini tidak boleh sama. Kemudian, kami menghitung n yang merupakan perkalian dari p dan q.

Langkah berikutnya memilih bilangan bulat d yang berupakan bilangan acak besar yang relatif prima dengan (p-1) * (q-1). Bilangan bulat d perlu diperiksa apakah memenuhi

PBB (d, (p-1) * (q-1)) = 1
Di mana
PBB = Perbagi bersama terbesar

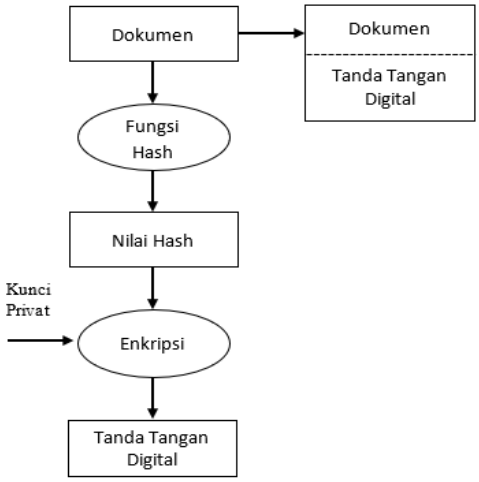
Langkah terakhir, kami menentukan bilangan bulat e yang didapat dari

(e * d) *mod* (p-1) * (q-1) = 1
dari proses di atas diperoleh kunci publik (e, n) dan kunci privat (d, n)

C. Proses Pembuatan Tanda Tangan Digital

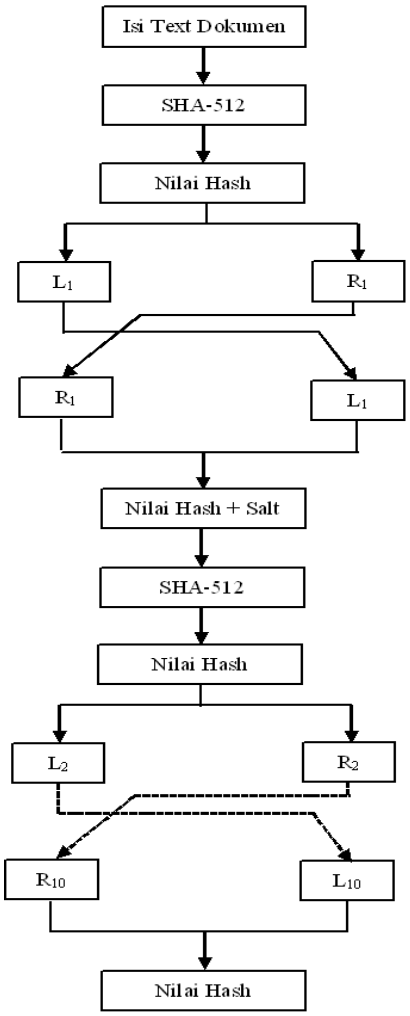
Proses pembuatan tanda tangan digital ditunjukkan pada Gambar 2. Tahap awal pembuatan tanda tangan digital, sebuah dokumen dihitung nilai hash-nya menggunakan metode SHA-512. Nilai hash yang dihasilkan ditukar posisi bit-nya antara 128 bit kiri dengan 128 bit kanan lalu ditambahkan salt pada bagian awal dan akhir. Proses ini

diulang selama sepuluh kali. Pengulangan dan penambahan *salt* ini bertujuan untuk mempersulit bila ada orang lain yang berusaha untuk mengubah pesan. Nilai hash terakhir yang dihasilkan memiliki panjang 512-bit atau 64 Bytes.



Gambar 2. Proses Pembuatan Tanda Tangan Digital

Proses pembuatan nilai hash ditunjukkan Gambar 3:



Gambar 3. Proses Pembuatan Nilai Hash

Berikut algoritme pembuatan nilai hash:

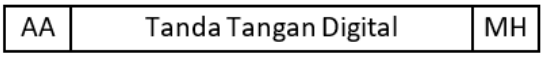
```
TextDokumen ← isi file dokumen
NilaiHash ← SHA512(TextDokumen)
for I ← 1 to 9
    cTemp ← right(NilaiHash, 32) + left(NilaiHash, 32)
    cTemp ← "DiGiTaL SiGnAtUrE" + cTemp + "TaNdA
            TaNgAn DiGiTaL"
    NilaiHash ← SHA512(cTemp)
Next
```

Salt bagian awal digunakan kalimat "DiGiTaL SiGnAtUrE", sedangkan bagian akhir digunakan kalimat "TaNdA TaNgAn DiGiTaL". Nilai hash ini akan dienkripsi menggunakan algoritme RSA dengan kunci privat dari pemilik dokumen sehingga menghasilkan tanda tangan digital. Berikut proses enkripsi nilai hash:

```
ct ← ""
for i ← 0 to 63
        temp ← substr(NilaiHash, i*2, 2)
        nTemp ← ubah_Hex_ke_Des(temp)
        for j ← 1 to kunci_privat
            nTemp ← (nTemp * nTemp) mod n
        next
        ct ← ct + char(nTemp >> 8) + char(nTemp
        and 255);
next
TandaTanganDigital ← "AA" + ct + "MH"
```

Di mana
kunci_private = 1373
n = 42781

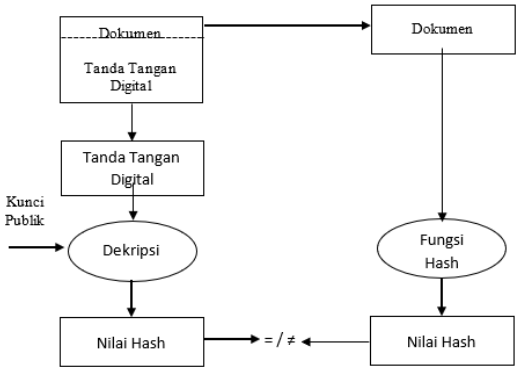
Tanda tangan digital ini diberikan ciri, yaitu ditambahkan kode awal AA dan kode akhir MH agar dapat dikenali pada saat dilakukan verifikasi. Tanda tangan digital yang sudah dibuat ditambahkan pada akhir dokumen. Format tanda tangan digital ditunjukkan pada Gambar 4.



Gambar 4. Format Tanda Tangan Digital

D. Proses Verifikasi Tanda Tangan Digital

Proses verifikasi tanda tangan digital ditunjukkan pada Gambar 5.



Gambar 5. Proses Verifikasi Tanda Tangan Digital

Proses verifikasi dimulai dengan mengambil tanda tanda digital dari file dokumen. Tanda tangan digital ini didekripsi menggunakan algoritme RSA dengan kunci publik dari pemilik dokumen sehingga menghasilkan nilai hash. Berikut proses mencari nilai hash dari tanda tangan digital.

Hash1 ← “”
TextDokumen ← isi file dokumen yang ada tanda tangan digital
cTemp ← *right*(TextDokumen, 132)
If (*left* (cTemp, 2) <> “AA” *or right* (cTemp, 2) <> “MH”) *Then*
    Print “File dokumen tidak ada tanda tangan digital”
    Proses berhenti
endif
TandaTanganDigital ← *substr*(cTemp, 2, 128)
for i ← 1 *to* 64
    temp ← *substr*(TandaTanganDigital, i*2, 2)
    nTemp ← *ord*(*left*(temp, 1)) * 256 +
    ord(*right*(temp, 1))

    for j ← 1 *to* kunci_publik
        nTemp ← (nTemp * nTemp) *mod* n
    next
    cTemp ← ubah_dec_ke_hex(nTemp)
    cTemp ← *right*(“00” + cTemp, 2)
    Hash1 ← Hash1 + cTemp
Next

Di mana
kunci_publik = 2129
n = 42781

File dokumen yang tanpa tanda tangan digital dihitung nilai hash-nya sama seperti pada saat proses pemberian tanda tangan digital. Berikut proses perhitungan nilai hash dari file dokumen yang tanpa tanda tangan digital:

TextDokumen ← isi file dokumen yang ada tanda tangan digital
Dokumen←*left*(TextDokumen, *LengthOf*(TextDokumen) – 132)
Hash2 ← SHA512(Dokumen)
for i ← 1 *to* 9
    cTemp ← *right*(NilaiHash, 32) + *left*(NilaiHash, 32)
    cTemp ← “DiGiTaL SiGnAtUrE” + cTemp + “TaNdA TaNgAn DiGiTaL”
    NilaiHash ← SHA512(cTemp)
Next
Hash2 ← NilaiHash

Kedua nilai hash, Hash1 dan Hash2 dibandingkan. Bila kedua nilai hash sama berarti dokumen dinyatakan asli dan tidak ada perubahan, namun bila berbeda berarti sudah mengalami perubahan atau tidak asli.

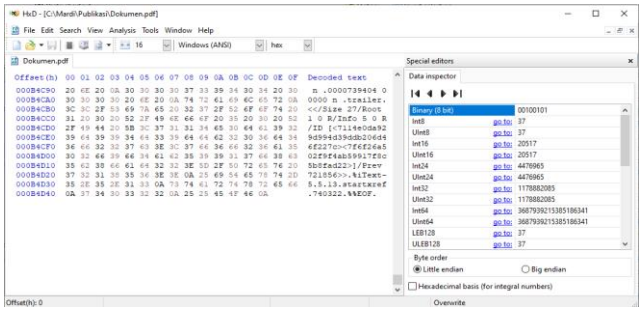
III. HASIL DAN PEMBAHASAN

Setelah program selesai dibuat, maka perlu dilakukan pengujian. Berikut ini diberikan implementasi dan hasil pengujian aplikasi tanda tangan digital.

A. Penyisipan Tanda Tangan Digital *Pada File PDF*

Dokumen yang digunakan untuk penambahan tanda tangan digital adalah berjenis pdf dan ditunjukkan pada Gambar 6. Bila file dokumen dibuka menggunakan aplikasi HxD maka terlihat isi pada akhir file seperti ditunjukkan pada Gambar 7.

Gambar 6. Contoh File Dokumen Yang Diberikan Tanda Tangan Digital



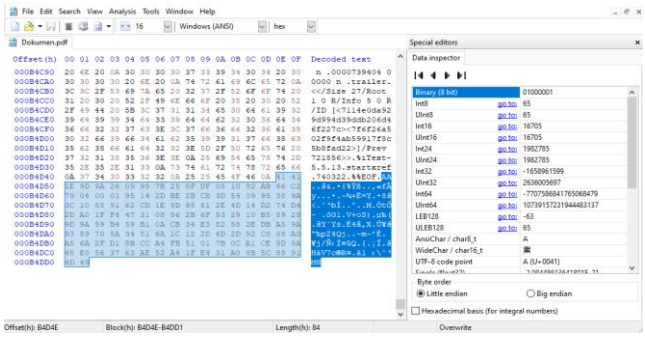
Gambar 7. Isi File Dokumen

Dokumen ini nanti akan diunggah di aplikasi untuk diberikan tanda tangan digital. Aplikasi tanda tangan digital ini dibuat berbasis web seperti ditunjukkan pada Gambar 8. Untuk memberikan tanda tangan digital ke sebuah dokumen dengan mengklik tautan **Buat**. Setelah diklik tautan **Buat** akan menampilkan layar seperti pada Gambar 9.

Gambar 8. Aplikasi Tanda Tangan Digital

Gambar 9. Halaman Membuat Tanda Tangan Digital

Setelah file dokumen diberi tanda tangan digital dan diunduh, maka terdapat perbedaan dengan file aslinya di bagian akhir file. Isi bagian akhir file dokumen yang sudah diberi tanda tangan digital dapat dilihat pada Gambar 10.



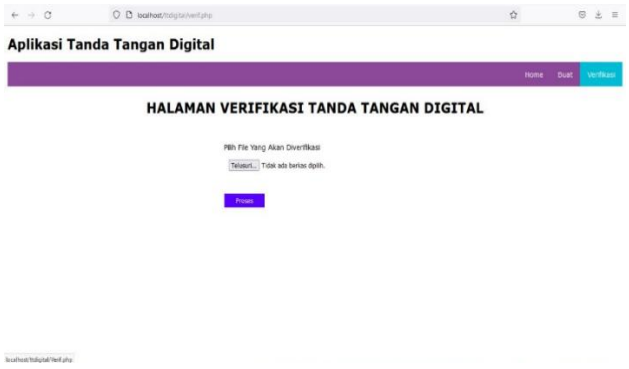
Gambar 10. Penambahan Tanda Tangan Digital di Akhir File

Walaupun sudah ditambahkan tanda tangan digital, file dokumen masih dapat dibuka dengan aplikasi pembuka file pdf seperti ditunjukkan pada Gambar 11.

Gambar 11. File Dokumen PDF Yang Sudah Ditambahkan Tanda Tangan Digital

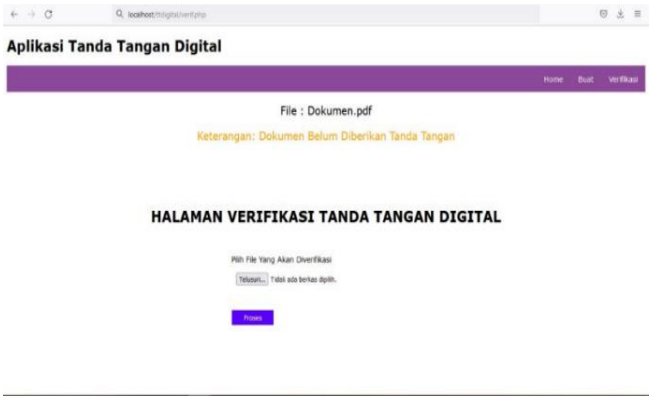
B. Verifikasi File Dokumen

Setelah file dokumen ditambahkan tanda tangan digital, maka perlu dilakukan ujicoba verifikasi untuk menentukan apakah file dokumen masih asli atau sudah berubah. Proses verifikasi dilakukan dengan mengklik tautan **Verifikasi**. Halaman **Verifikasi** dokumen ditunjukkan pada Gambar 12.



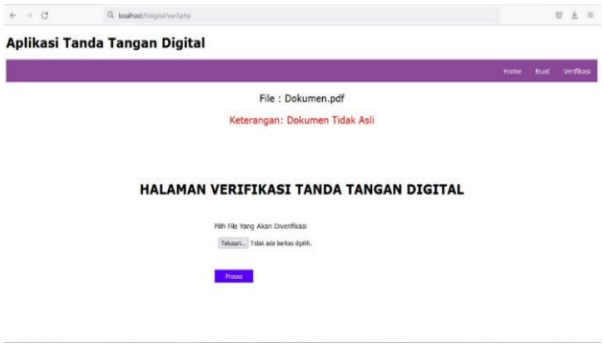
Gambar 12. Halaman Verifikasi Tanda Tangan Digital

Setelah dipilih dokumen yang akan diverifikasi, maka aplikasi akan memeriksa apakah file dokumen pernah diberikan tanda tangan digital atau belum. Pemeriksaan dilakukan dengan membaca 132 byte terakhir dari file dokumen. Apabila 132 byte tersebut diawali dengan karakter **AA** dan diakhiri dengan **MH**, berarti file dokumen tersebut sudah diberi tanda tangan digital. Tanda tangan digital ini bisa hilang bila file dokumen dibuka dengan sebuah aplikasi lalu disimpan**.** Bila belum pernah diberikan tanda tangan digital, akan muncul keterangan seperti ditunjukkan pada Gambar 13.



Gambar 13. Dokumen Belum Diberi Tanda Tangan Digital

Dokumen yang sudah diberi tanda tangan digital akan diperiksa apakah isinya telah berubah atau tidak. Pemeriksaan dilakukan dengan memisahkan isi dokumen dengan tanda tangan digital. Dokumen ini akan dihitung nilai hash-nya dan tanda tangan digital akan didekripsi dengan algoritme RSA menggunakan kunci publik pemilik dokumen untuk menghasilkan nilai hash. Bila kedua nilai hash ini sama, berarti dokumen masih asli, sedangkan kalau nilainya berbeda berarti dokumen sudah mengalami perubahan. Bila dokumen sudah berubah akan muncul pesan seperti ditunjukkan pada Gambar 14. Sebaliknya, bila tidak berubah atau masih asli akan muncul pesan seperti ditunjukkan pada Gambar 15.



Gambar 14. Dokumen Tidak Asli

Gambar 15. Dokumen Asli

C. *Hasil Pengujian Lainnya*

Aplikasi tanda tangan digital ini perlu diujikan pada beberapa file untuk memastikan berjalan dengan baik. Jenis file yang digunakan untuk pengujian adalah .pdf, .docx, .xlsx dan .jpg. Proses pemberian tanda tangan digital sangat cepat sekali, masih di bawah satu detik. Hasil pengujian ditunjukkan pada Tabel 1.

TABEL 1. HASIL PENGUJIAN PENYISIPAN TANDA TANGAN DIGITAL

No.	Nama File	Ukuran File	Waktu Proses
1.	Sertifikat_CEH.pdf	1,331,553 bytes	0,022 detik
2.	Ijazah.pdf	104,693 bytes	0,010 detik
3.	Dokumen.docx	3,210,902 bytes	0,060 detik
4.	Katsinovmeter.xlsx	5,262,336 bytes	0.070 detik
5.	Gambar.jpg	155,355 bytes	0,011 detik

IV. KESIMPULAN

Setelah dilakukan pengujian terhadap aplikasi tanda tangan digital ini, maka dapat diambil kesimpulan bahwa sistem tanda tangan digital menggunakan algoritme RSA dan SHA-512 dapat mengamankan dokumen-dokumen penting dari perubahan oleh orang-orang yang tidak bertanggung jawab. Proses pembuatan tanda tangan digital tidak membutuhkan waktu yang lama.

REFERENSI

[1] S. Bhatia and A. D. W. de Hernandez, "Blockchain Is Already Here. What Does That Mean for Records Management and Archives?," *J. Arch. Organ.*, vol. 16, no. 1, pp. 75–84, 2019.

[2] Y. Anshori, A. Y. Erwin Dodu, and D. M. P. Wedananta, "Implementasi Algoritma Kriptografi Rivest Shamir Adleman (RSA) pada Tanda Tangan Digital," *Techno.Com*, vol. 18, no. 2, pp. 110–121, 2019.

[3] Y. A. Putri and E. V. Manullang, "Implementasi Kriptografi Digital Signature Menggunakan Secure Hash Algorithm (SHA-1)," *J. Teknol. Inf.*, vol. 6, no. 1, pp. 1–8, 2018.

[4] U. P. Patel, A. K. Patel, and F. A. Suthar, "Science and Applications the Study of Digital Signature," *Journal of Information, Knowledge and Research in Computer Science and Applications*, vol. 1, no. 2, pp. 38–43, 2019.

[5] D. Drescher, *Blockchain basics: A Non-Technical Introduction in 25 Steps*. California: Apress, 2017.

[6] M. A. Nazal, R. Pulungan, and M. Riasetiawan, "Data Integrity and Security using Keccak and Digital Signature Algorithm (DSA)," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 13, no. 3, p. 273, 2019.

[7] E. C. Prabowo and I. Afrianto, "Penerapan Digital Signature dan Kriptografi pada Otentikasi Sertifikasi Tanah Digital," *J. Ilmu Komputer dan Informatika*, vol. 6, no. 2, pp. 83–90, 2017.

[8] A. Saepulrohman. and A. Ismangil, "Data Integrity and Security of Digital Signatures on Electronic Systems Using the Digital Signature Algorithm (DSA)," *International Journal of Electronics and. Communications Systems*, vol. 1, no. 1, pp. 11-15, 2021.

[9] H. Manurung, "Perancangan Perangkat Lunak Pembelajaran Authentication dan Digital Signature Scheme dengan Metode Simulasi Schnorr," *J. Inform. Kaputama*, vol. 3, no. 2, pp. 51–59, 2019.

[10] O. K. Sulaiman, M. Ihwani, and S. F. Rizki, "Model Keamanan Informasi Berbasis Tanda Tangan Digital Dengan Data Encryption Standard (Des) Algorithm," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 1, no. 1, pp. 14–19, 2016.

[11] M. Ihwani, "Model Keamanan Informasi Berbasis Digital Signature Dengan Algoritma Rsa," *CESS Journal Comput. Eng. Syst. Sci.*, vol. 1, no. 1, pp. 15–20, 2016.

[12] B. K. Hutasuhut, S. Efendi, and Z. Situmorang, "Digital Signature untuk Menjaga Keaslian Data dengan Algoritma MD5 dan Algoritma RSA," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 3, no. 2, pp. 164–169, 2019.

[13] N. Zaatsiyah and D. Djuniadi, "Implementing Digital Signature With Rsa and Md5 in Securing E-Invoice Document," *Cybersp. J. Pendidik. Teknol. Inf.*, vol. 5, no. 2, p. 129, 2021.

[14] Arpan, N. Mayasari, "Pembangunan, P. Budi, and M.-S. Utara, "Membangkitkan Digital Signature Dengan Algoritma MD5 dan Algoritma RSA untuk Memastikan Keaslian File," vol. 6, no. 2, pp. 8–13, 2019.

[15] R. L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 1-15, 1978.