Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

Penerapan Teknik Digital Signature Dalam Pengamanan Piagam Penghargaan Menggunakan Algoritma SHA-1 dan RSA

Nur Hanyfa Sari
Prodi Teknik Informatika Universitas Budi Darma, Medan, Indonesia
Jl. Sisingamangaraja No. 338, Medan, Indonesia
Email: hani.filianie@gmail.com

Abstrak - Persaingan kerap terjadi antar individu untuk mencapai suatu posisi atau jabatan tertentu, seperti memperoleh beasiswa pasca sarjan, melamar pekerjaan, dan mengejar jabatan penting pemerintahan. Selain penelaian dari aspek kemampuan dan latar belakang Pendidikan, pengalaman juga menjadi salah satu factor pendukung seseorang untuk berhasil mencapai posisi yang diinginkannya. Untuk mengantisipasi terjadinya manipulasi atau perubahan, *file digital* bisa dijadikan solusi. Tentunya *file digital* pun masih membutuhkan pengamanan guna untuk menjamin keamanan dan keabasahan (tidak adanya perubahan pada isi *file* tersebut). Oleh sebab itu diperlukan solusi untuk masalah tersebut, salah satu teknik kriptografi yang dapat dimanfaatkan adalah tanda tangan digital *(digital signature)*. Memilih *digital signature* sebagai solusi dirasa sangatlah tepat, karena *digital signature* sebagai tanda pada data, yang dapat memastikan bahwa data tersebut nyata (tidak berubah). Sehingga dengan diterapkannya teknik *digital signature* pada piagam penghargaan, otentikasi dan identifikasi kepemilikan pada piagam penghargaan secara konseptual dapat terjamin. Untuk melindungi validitas suatu *file digital* secara maksimal dengan teknik *digital signature*, maka algoritma yang akan digunakan adalah algoritma SHA-1 dan RSA yang dianggap cukup untuk memastikan keamanan suatu *file digital*.

Kata Kunci: Kriptografi, Tanda Tangan Digital *(Digital Signature),* Algoritma SHA-1, Algoritma RSA

Abstract - Competition often occurs between individuals to achieve a certain position or position, such as obtaining postgraduate scholarships, applying for jobs, and pursuing important government positions. In addition to research from the aspects of ability and educational background, experience is also one of the factors supporting a person to succeed in achieving the position he wants. To anticipate manipulation or change, digital files can be a solution. Of course, even digital files still need security in order to ensure safety and authenticity (there are no changes to the contents of the file). Therefore we need a solution to this problem, one of the cryptographic techniques that can be used is a digital signature. Choosing a digital signature as a solution is considered very appropriate, because the digital signature is a sign on the data, which can ensure that the data is real (unchanged). So that by applying the digital signature technique to the award certificate, authentication and identification of ownership of the award certificate can be conceptually guaranteed. To protect the maximum validity of a digital file with a digital signature technique, the algorithms that will be used are the SHA-1 and RSA algorithms which are considered sufficient to ensure the security of a digital file.

Keywords: Cryptography, Digital Signature, SHA-1 Algorithm, RSA Algorithm

1. PENDAHULUAN

Pada masa yang serba kompetitif seperti sekarang ini, persaingan biasanya terjadi antar individu untuk mencapai suatu posisi atau jabatan tertentu, seperti memperoleh beasiswa pasca sarjana, melamar pekerjaan, dan mengejar jabatan penting pemerintahan. Selain penilaian dari aspek kemampuan dan latar belakang pendidikan, pengalaman juga menjadi salah satu faktor pendukung seseorang untuk berhasil mencapai posisi yang diinginkannya. Meski sering dianggap kurang penting, namun pada dasarnya piagam penghargaan memiliki banyak kegunaan. Memiliki banyak piagam penghargaan merupakan salah satu bukti bahwa orang tersebut memiliki banyak pengalaman, hal tersebut dapat menjadi suatu pertimbangan bagi perusahaan untuk menerima karyawan. Piagam penghargaan juga salah satu syarat jika ingin kuliah lewat jalur beasiswa berprestasi.

Dengan telah berkembangnya bidang teknologi informasi proses manipulasi, modifikasi, maupun duplikasi terhadap gambar, teks, atau dokumen termasuk dokumen piagam penghargaan dapat dilakukan dengan mudah dan cepat. Hal tersebut dapat menimbulkan celah bagi pihak tertentu untuk mengambil keuntungan dari jeri payah orang lain. Untuk mengantisipasi terjadinya manipulasi atau perubahan, *file digital* bisa dijadikan solusi. Tentunya *file digital* pun masih membutuhkan pengamanan guna untuk menjamin keamanan dan keabasahan (tidak adanya perubahan pada isi *file* tersebut). Penerapan salah satu teknik kriptografi yaitu tandatangan digital *(digital signature)* dirasa sangatlah tepat untuk masalah diatas. Terdapat sebuah penelitian dengan topik "Peran Kriptografi Sebagai Keamanan Sistem Pada Usaha Kecil dan Menengah" menyimpulkan bahwa kriptografi masih merupakan sistem yang efektif dalam hal keamanan dan proteksi serta dapat digunakan secara luas di berbagai bidang usaha dan teknologi [1].

Digital signature berfungsi sebagai tanda pada data, yang dapat memastikan bahwa data tersebut adalah data nyata (tidak berubah). Sehingga dengan diterapkannya teknik *digital signature* pada piagam penghargaan, otentikasi dan

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS
identifikasi kepemilikan pada *file digital* piagam penghargaan secara konseptual dapat terjamin. *Digital Signature* adalah suatu untaian string yang diperoleh dengan cara memetakan suatu masukan string dengan suatu fungsi hash, kemudian nilai *hash* tersebut dienkripsi dengan menggunakan algoritma kriptografi kunci-asimetris. Hasil enkripsi nilai *hash* tersebutlah yang menjadi *digital signature* masukan string awal. [1]-[2]-[3]
Secure Hashing Algorithm (SHA) adalah salah satu jenis dari algoritma fungsi *hash*. SHA terdiri dari 4 macam yaitu SHA-1, SHA-256, SHA-384, SHA-521. SHA-1[4] dikatakan aman karena secara matematis tidak mungkin menemukan dua pesan yang berbeda yang menghasilkan nilai *hash* yang sama [2]. *Rivest Shamir Adleman* (RSA) merupakan algoritma enkripsi-dekripsi non simetri dimana kunci pribadi dan kunci publik dihasilkan dari olahan dua buah bilangan prima. Tingkat keamanan dari algoritma RSA adalah sulitnya menguraikan angka menjadi faktor-faktor prima. Pemfaktoran dilakukan untuk memperoleh kunci pribadi. Selama pemfaktoran bilangan besar menjadi faktor-faktor prima belum ditemukan, maka keamanan algoritma RSA tetap terjamin [1].

2. METODE PENELITIAN

2.1 Kriptografi

Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi, seperti kerahasiaan data, keabsahan data, integritas data, serta autentikasi data. Menurut ahli kriptografi yaitu sebuah studi teknik informatika yang berkaitan aspek keamanan informasi seperti kerahasiaan data dan integritas data, kriptografi tidak hanya penyediaan keamanan informasi saja tetapi juga sebuah himpunan tekik-teknik [3]. Sehingga dapat disimpulkan bahwa dapat dsimpulkan bahwa kriptografi adalah suatu bidang ilmu, teknik dan seni yang bertujuan untuk menjaga kerahasiaan suatu pesan yang berupa data-data dari pihak lain yang tidak berhak.

2.2 Digital Signature

Digital signature berfungsi sebagai tanda pada data, yang dapat memastikan bahwa data tersebut adalah data nyata (tidak berubah). Tanda tangan digital adalah nilai terenkripsi, yang bergantung pada konten datanya sendiri dan kunci yang digunakan untuk menghasilkan nilai terenkripsi, sehingga nilai setiap tanda tangan digital selalu berbeda sesuai dengan data yang ditandatangani. Tanda tangan digital yang dilampirkan pada data dapat memverifikasi sumber data. Tanda tangan digital memberikan rasa aman bagi penerima data karena dapat mengetahui pengirim datanya. Tanda tangan digital yang dihasilkan dari pesan dan dienkripsi dengan kunci tertentu harus digunakan untuk membuktikan keaslian pesan. Ini berarti bahwa fungsi hash dan proses enkripsi yang diharapkan harus membuat tanda tangan menjadi unik dan hanya mengandalkan data dan kunci input. Selain itu, kunci yang digunakan untuk proses enkripsi dan dekripsi haruslah kunci yang kuat, yang tidak dapat diperoleh dengan mudah oleh pihak ketiga tanpa kunci pribadi pengirim [4].

2.3 Algoritma SHA-1

SHA-1 adalah fungsi hash satu arah. SHA-1 menerima pesan dengan ukuran maksimum 2 ^ 64bit (2.147.483.648 gigabyte) dan menghasilkan nilai hash dengan panjang 160 bit. Kemudian, nilai hash digunakan di DSA untuk menghitung tanda tangan digital dari pesan tersebut. Penerima pesan dapat memperoleh nilai hash pesan saat menerima pesan dari pengirim dengan memasukkan nilai di fungsi SHA-1. Dikatakan bahwa SHA-1 aman, karena secara matematis tidak mungkin menemukan dua pesan berbeda yang menghasilkan nilai hash yang sama, atau tidak mungkin menemukan pesan asli jika nilai hash diberikan [2].
Cara kerja kriptografi algoritma SHA-1 adalah menerima input berupa pesan dengan ukuran sembarang dan menghasilkan *message digest* yang memiliki panjang 160 bit. Langkah-langkah pembuatan *message digest* dengan algoritma SHA-1 adalah sebagai berikut [5]:

a. *Input* Pesan yang akan di *hash* SHA-1.
b. Ubah pesan menjadi deretan biner.
c. Penambahan *Padding* Bit.
   Misalnya Panjang M = 200 bit. Proses berikutnya adalah dengan menambahkan padding bit 1 dan sisanya 0 sejumlah k, dengan persamaan berikut:

$$200 + 1 + K = 448 \bmod 512 \quad (2.1)$$

$$K = 448 - 201 \bmod 512$$

$$K = 247$$

   Lakukan penambahan bit sebanyak 247 bit.

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

d. Penambahan Panjang, penambahan panjang append dilakukan dengan penambahan Panjang pesan sebanyak 64 bit di akhir.

e. *Parsing* pesan (Mengelompokan Pesan)

f. Penjadwalan pesan, langkah ini diawali dengan mengubah setiap blok pesan menjadi heksadesimal dengan persamaan sebagai berikut:

$$Wt = \begin{cases} M_t^{(t)} & 0 \le t \le 15 \\ ROTL\ (Wt-3) \oplus (Wt-8) \oplus (Wt-14) \oplus (Wt-16) & 16 \le t \le 79 \end{cases} \quad (2.2)$$

g. Inisialisasi Variabel Kerja, Inisialisasi penyangga MD SHA-1 membutuhkan 5 buah penyangga, A, B, C, D, E, yang masing panjang penyangga adalah 32 bit. Dengan persamaan sebagai berikut:

A=67452301
B=EFCDAB89
C=98BADCFE
D=10325476
E=C3D2E1F0

$$T = ROTL5\ (a) + ft\ (b.c,d) + e + Kt + Wt \quad (2.3)$$

$$A = T^1 \quad (2.4)$$

$$B = A \quad (2.5)$$

$$C = ROTL^{30}\ (b) \quad (2.6)$$

$$D = C \quad (2.7)$$

$$E = D \quad (2.8)$$

$$f_t(b, c, d) \begin{cases} \text{Ch} & (b, c, d) = (b \wedge c) \oplus (b \wedge d) & 0 \le t \le 19 \\ \text{Parity} & (b, c, d) = b \oplus c \oplus d & 20 \le t \le 39 \\ \text{Maj} & (b, c, d) = (b \wedge c) \oplus (b \wedge d) \oplus (c \wedge d) & 40 \le t \le 59 \\ \text{Parity} & (b, c, d) = b \oplus c \oplus d & 60 \le t \le 79 \end{cases} \quad (2.9)$$

$$Kt = \begin{cases} 5A027999 & 0 \le t \le 19 \\ 6ED9EBA1 & 20 \le t \le 39 \\ 8F1BBCDC & 40 \le t \le 59 \\ CA62C1D6 & 60 \le t \le 79 \end{cases} \quad (2.10)$$

Dan selanjutnya mencari nilai dari Ch, Parity, Maj dan Parity

Ch (b, c, d) = ((b $\wedge$ c) $\oplus$ (b $\wedge$ d) (2.11)
= EFCDAD89 $\wedge$ 98BADCFE $\oplus$ EFCDAD89 $\wedge$ 10325476
= 88888C88 $\oplus$ 00000000
= 88888C88

Parity (b, c, d) = (b $\oplus$ c $\oplus$ d) (2.12)
= EFCDAB89 $\oplus$ 98BADCFE $\oplus$ 10325476
= 77777777 $\oplus$ 10325476
= 67452301

Maj (b, c, d) = (b $\wedge$ c)$\oplus$ (b $\wedge$ d) $\oplus$ (c $\wedge$ d) (2.13)
= EFCDAB89 $\oplus$ 98BADCFE
= EFCDAB89 $\oplus$ 10325476
= 98BADCFE $\oplus$ 10325476
= 88888888 $\oplus$ 00000000 $\oplus$ 10325476 = 98BADCFE

h. Tambahkan A, B, C, D, E dengan penyangga.

A = [A] 79 + A = (2.14)
B = [B] 79 + B = (2.15)
C = [C] 79 + C = (2.16)
D = [D] 79 + D = (2.17)
E = [E] 79 + E = (2.18)

i. Lalu gabungkan hasil penjumlahan tersebut.

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

2.3 Algoritma RSA

Rivest Shamir Adleman (RSA) merupakan algoritma enkripsi-dekripsi non simetri dimana kunci pribadi dan kunci publik dihasilkan dari olahan dua buah bilangan prima. Algoritma ini ditemukan pada tahun 1976 oleh tiga peneliti dari MIT *(Massachussets Institute of Technology)*, yaitu Ron Rivest, Adi Shamir, dan Len Adleman. Tingkat keamanan dari algoritma RSA adalah sulitnya menguraikan angka menjadi faktor-faktor prima. Ada 3 tahapan dalam penggunaan RSA, yaitu: pemasangan kunci, enkripsi, dan deskripsi [6]:

a. Pembangkit kunci
   Algoritma pembangkitan kunci mengambil sebuah masukan sebagai parameter *n* dan sebuah parameter tambahan *b*. Algoritma ini membangkitkan pasangan *public key* dan *private key* RSA dengan ketentuan sebagai berikut:
   1. Bangkitkan 2 buah bilangan prima yaitu *p* dan *q*
   2. Hitung n = p x q. (2.19)
   3. Hitung $\phi(n) = (p-1) \times (q-1)$. (2.20)
   4. Pilih e yang nilai nya relatif prima terhadap $\phi(n)$.
   5. Tentukan kunci privat d dengan persamaan,
      $d = (1 + a*\phi(n))/e$, (2.21)
      dengan a adalah bilangan bulat yang dapat memenuhi.
   6. Sehingga didapat pasangan kunci publik (e, n) dan kunci privat (d, n).

b. Enkripsi
   Pada proses ini kunci privat (e, n) akan digunakan. Untuk mengenkripsi *plaintext* menjadi *ciphertext* dapat dilakukan dengan cara berikut:
   $$C = M^e \text{ Mod } n \quad (2.22)$$

c. Untuk mendekripsi ciphertext C digunakan private key (d, n), dengan cara berikut:
   $$M = C^d \text{ Mod } n \quad (2.23)$$

3. HASIL DAN PEMBAHASAN

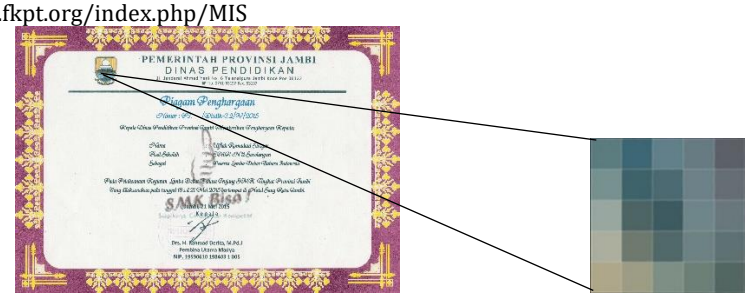
3.1 Analisa Proses Penandatanganan

Manipulasi terhadap teks, gambar, atau dokumen termasuk piagam penghargaan, dengan mudah dapat dilakukan. Hal tersebut dapat menimbulkan celah bagi pihak tertentu untuk mengambil keuntungan dari jeri payah orang lain. Dokumen digital bisa dijadikan solusi, tentunya dokumen digital pun masih membutuhkan pengamanan guna untuk menjamin keamanan dan keabasahan (tidak adanya perubahan pada isi dokumen tersebut). Masalah yang akan di analisa adalah bagaimana proses pengamanan piagam penghargaan, serta bagaimana proses verifikasi tanda tangan digital tersebut guna untuk memastikan bahwa dokumen tersebut adalah asli dan belum pernah dimodifikasi.
Untuk memberikan metode proteksi terbaik untuk validitas tanda tangan digital, penulis menggunakan algoritma SHA-1 dan RSA. Karena algoritma ini memiliki kelebihan, maka dianggap cukup. Oleh karena itu, algoritma SHA-1 dirancang agar secara komputasi tidak dapat menemukan pesan yang sesuai dengan intisari pesan yang diberikan.Keuntungan dari Algortima RSA adalah sulit untuk menguraikan sejumlah bilangan besar menjadi faktor-faktor primanya.
Proses penandatanganan menggunakan *digital signature* dimulai dengan mengubah *file digital* dokumen berekstensi .jpeg menjadi *messeage digest* mengunakan Algoritma SHA-1, lalu *message digest* tersebut akan di enkripsi menggunakan Algoritma RSA. Hasil enkripsi kemudian disematkan pada dokumen .jpeg.

3.2 Analisa SHA-1

Yang menjadi objek pada penelitian ini adalah dokumen citra hasil pemindaian dari piagam penghargaan. Dokumen hasil pemindaian berformat .jpg dengan resolusi 3393 x 2348 *pixel*. Untuk menyederhanakan proses analisis maka diambil sampel dari hasil pemindaian piagam penghargaan tersebut berukuran 5 x 5 *pixel*. Adapun nilai pixel 25*pixel* diambil menggunakan MatlabR2013a.

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS



Gambar 1. Pixel Citra Sampel

Sebelum menerapkan algoritma SHA-1, dilakukan penyesuaian masukan dalam bentuk bilangan biner. Untuk melakukan ini, ubah nilai desimal piksel dari gambar input menjadi bilangan biner.

Tabel 1. Nilai *Input* Desimal

114	93	115	119	99
139	88	96	129	109
161	93	82	130	110
172	120	97	126	105
164	157	134	118	98

Tabel 2. Nilai *Input* Desimal ke Biner

01110010	01011101	01110011	01110111	01100011
10001011	01011000	01100000	10000001	01101101
10100001	01011101	01010010	10000010	01101110
10101100	01111000	01100001	01111110	01101001
10100100	10011101	10000110	01110110	01100010

a. Penambahan *Padding* Bit
   1. Dari table diatas diketahui bahwa Panjang M = 200 bit. Proses berikutnya adalah dengan menambahkan padding bit 1 dan sisanya 0 sejumlah k, dengan persamaan berikut:
   2. 200 + 1 + K = 448 mod 512
      K = 448 – 201 mod 512
      K = 247
      Lakukan penambahan bit sebanyak 247 bit.

Tabel 3. Penambahan *Padding* Bit

M0	01110010	01011101	01110011	01110111
M1	01100011	10001011	01011000	01100000
M2	10000001	01101101	10100001	01011101
M3	01010010	10000010	01101110	10101100
M4	01111000	01100001	01111110	01101001
M5	10100100	10011101	10000110	01110110
M6	01100010	**10000000**	**00000000**	**00000000**
M7	**00000000**	**00000000**	**00000000**	**00000000**
M8	**00000000**	**00000000**	**00000000**	**00000000**
M9	**00000000**	**00000000**	**00000000**	**00000000**
M10	**00000000**	**00000000**	**00000000**	**00000000**
M11	**00000000**	**00000000**	**00000000**	**00000000**
M12	**00000000**	**00000000**	**00000000**	**00000000**
M13	**00000000**	**00000000**	**00000000**	**00000000**

Biner yang dihitamkan dalam tabel diatas adalah nilai dari tambahan bit sebanyak 247 bit.

b. Penambahan panjang pesan
   Penambahan Panjang append dilakukan dengan penambahan Panjang pesan sebanyak 64 bit di akhir.

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

Tabel 4. Penambahan Panjang Pesan

M0	01110010	01011101	01110011	01110111
M1	01100011	10001011	01011000	01100000
M2	10000001	01101101	10100001	01011101
M3	01010010	10000010	01101110	10101100
M4	01111000	01100001	01111110	01101001
M5	10100100	10011101	10000110	01110110
M6	01100010	10000000	00000000	00000000
M7	00000000	00000000	00000000	00000000
M8	00000000	00000000	00000000	00000000
M9	00000000	00000000	00000000	00000000
M10	00000000	00000000	00000000	00000000
M11	00000000	00000000	00000000	00000000
M12	00000000	00000000	00000000	00000000
M13	00000000	00000000	00000000	00000000
M14	**00000000**	**00000000**	**00000000**	**00000000**
M15	**00000000**	**00000000**	**00000000**	**10001000**

c. *Parsing* pesan (Mengelompokan Pesan)
Melakukan pengelompokan pesan dari penambahan bit sampai ke penambahan panjang pesan.

Tabel 5. Pengelompokan Pesan

M0	01110010	01011101	01110011	01110111
M1	01100011	10001011	01011000	01100000
M2	10000001	01101101	10100001	01011101
M3	01010010	10000010	01101110	10101100
M4	01111000	01100001	01111110	01101001
M5	10100100	10011101	10000110	01110110
M6	01100010	10000000	00000000	00000000
M7	00000000	00000000	00000000	00000000
M8	00000000	00000000	00000000	00000000
M9	00000000	00000000	00000000	00000000
M10	00000000	00000000	00000000	00000000
M11	00000000	00000000	00000000	00000000
M12	00000000	00000000	00000000	00000000
M13	00000000	00000000	00000000	00000000
M14	00000000	00000000	00000000	00000000
M15	00000000	00000000	00000000	10001000

d. Penjadwalan pesan
Langkah ini diawali dengan mengubah setiap blok pesan menjadi heksadesimal dengan ketentuan sebagai berikut:

$$Wt = \begin{cases} M_t^{(t)} & 0 \le t \le 15 \\ ROTL\,(Wt-3) \oplus (Wt-8) \oplus (Wt-14) \oplus (Wt-16) & 16 \le t \le 79 \end{cases}$$

Untuk penjadwalan pesan ke 16 sampai 79 dilakukan perhitungan sebagai berikut:

$Wt = ROTL\ 1(W16-3 \oplus W16-8\ \oplus W16-14\ \oplus W16-16)$

$Wt = ROTL\ 1(W13\ \oplus W8\ \oplus W2\ \oplus W0)$

Wt = ROTL 1 (00000000 ⊕ 00000000 ⊕ 816DA15D ⊕ 725D7377)

W13	= 00000000 00000000 00000000 00000000	
W8	= 00000000 00000000 00000000 00000000	
W2	= 10000001 01101101 10100001 01011101	
W0	= 01110010 01011101 01110011 01110111	⊕
W16	= 11110011 00110000 11010010 00101010	

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

W16 = 11100110 01100001 10100100 01010101
E6 61 A4 55

Wt = $ROTL\ 1(W17-3 \oplus W17-8 \oplus W17-14 \oplus W17-16)$

Wt = $ROTL\ 1(W14 \oplus W9 \oplus W3 \oplus W1)$

Wt = ROTL 1 (00000000 ⊕ 00000000 ⊕ 52826EAC ⊕ 638B5860)

W14 = 00000000 00000000 00000000 00000000
W9 = 00000000 00000000 00000000 00000000
W3 = 01010010 10000010 01101110 10101100
W1 = 01100011 10001011 01011000 01100000 ⊕
W17 = 00110001 00001001 00110110 11001100

W17 = 01100010 00010010 01101101 10011000
62 12 6D 98

Dan langkah selanjutnya sampai ke putaran tujuh puluh sembilan. Dari hasil keseluruhan dari proses SHA-1 diatas, maka diperoleh hasil yang menggunakan nilai heksadesimal dari peroses hasil tersebut. Sebagaimana terlihat pada tabel dibawah ini sebagai berikut:

Tabel 6. Penjadwalan Pesan

W0 = 725D7377	W1 = 638B5860	W2 = 816DA15D	W3 = 52826EAC	W4 = 78617E69
W5 = A49D8676	W6 = 62800000	W7 = 00000000	W8 = 00000000	W9 = 00000000
W10 = 00000000	W11 = 00000000	W12 = 00000000	W13 = 00000000	W14 = 00000000
W15 = 00000088	W16 = E661A455	W17 = 62126D98	W18 = F219BE69	W19 = 20FC991E
W20 = F1E627E2	W21 = AD08703E	W22 = 84F9323C	W23 = F1E6276A	W24 =96D3A8D6
W25 = CDD6BF49	W26 = 07FF3206	W27 = 6C5E6391	W28 = 78613156	W29 = 55EE8561
W30 = 1D8DEBF0	W31 = D72AF659	W32 =AE8A6F17	W33 = 256B407F	W34 =A654BBA9
W35 = 9E41CB4C	W36 = 502AC9EE	W37 = 5EA8D339	W38 =23CD74AD	W39 = 76614F29
W40 = C21C4DFC	W41 = 4E5DD015	W42 = 5F57EFA1	W43 =CBDAC080	W44 = F73786BA
W45 = 06769F41	W46 = B6206194	W47 = E62EFF6A	W48 = 99680C07	W49 = 86AE7564
W50 = 9E0EC504	W51 = 24B7A9E5	W52 =04FC9D3A	W53 = 61638CAB	W54 = E68DE240
W55 = B5DDFAD9	W56 = CA8045E2	W57 = CB490F63	W58 = 0776AD8D	W59 = 473770EC
W60 = 1D44EAEF	W61 = 0C9A821B	W62 = 1DE5FE7F	W63 = 90333471	W64 = 82F81DE5
W65 = E96A5B3A	W66 = 1B6F8384	W67 = 0036908F	W68 =2CBF9D5E	W69 = 8696EFDB
W70 = 63BD92A5	W71 = 8430B92B	W72 = 933034A3	W73 = 0D536C20	W74 = 0ADAFB9B
W75 = B156ADB7	W76 = 429BCBDD	W77 = 21CB4455	W78 = 9BEDB910	W79 = 7FE43B7B

e. Inisialisasi Variabel Kerja

Inisialisasi penyangga MD SHA-1 membutuhkan 5 buah penyangga yang masing panjang penyangga adalah 32bit. Penyangga SHA-1 adalah sebagai berikut:

A=67452301
B=EFCDAB89
C=98BADCFE
D=10325476
E=C3D2E1F0

T = $ROTL5\ (a) + ft\ (b.c,d) + e + Kt + Wt$

A = T^1 D = C
B = A E = D
C = $ROTL^{30}$ (b)

Nilai (a) akan ROTL sebanyak 5x kekiri dan jumlahkan dengan nilai Ch, Parity, Maj, Parity, c, Kt dan Wt untuk mencari nilai A dan (b) akan di ROTL kan sebanyak 30x kekiri untuk mencari nilai C. Langkah pertama menggunakan nilai Ch = 88888888 dan Kt = 5A827999 dari 0 ≤ 19.

T = $ROTL^5$(a) + ft (b. c, d) + e + Kt + Wt
(a) = 01100111 01000101 00100011 00000001
$ROTL^5$(a) = 11101000 10100100 01100000 00101100
= E8A4602C + 88888888 + C3D2E1F0 + 5A827999 + 725D7377
A = 8B2103BA
B = A
= 67452301

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

C = ROTL^{30}(b) EFCDAB89
= 01111011 11110011 01101010 11100010
= 7BF36AE2
D = C
= 98BADCFE
E = D
= 10325476
T_0 = 8B2103BA, 67452301, 7BF36AE2, 98BADCFE, 10325476

Demikian seterusnya hingga T_{19}. Dan langkah selanjutnya menggunakan nilai Parity = 67452301 dan Kt = 6ED9EBA1 dari 20 ≤ 39.

T = ROTL^5(a) + ft (b. c, d) + e + Kt + Wt
(a) = 01100011 10011101 11010111 00110111
ROTL^5 (a) = 01110011 10111010 11100110 11101100
= 73BAE6EC + 67452301 + 947FA913 + 6ED9EBA1 +F1E627E2
A = 1FBFA0BD
B = A
= 639DD737
C = ROTL^{30}(b) 5828A0B1
= 01010110 00001010 00101000 00101100
= 56A0282C
D = C
= 66F563B6
E = D
= 693DC147
T_{20} = 1FBFA0BD, 639DD737, 56A0282C, 66F563B6, 693DC147

Demikian seterusnya hingga T_{39}. Dan langkah selanjutnya menggunakan nilai Maj = 98BADCFE dan Kt = 8F1BBCDC dari 40 ≤ 59.

T = ROTL^5(a) + ft (b. c, d) + e + Kt + Wt
(a) = 01011101 00111001 01001111 00100111
ROTL^5 (a) = 10100111 00101001 11100100 11101011
= A729E4EB+ 98BADCFE + 26709779 + 8F1BBCDC + C21C4DFC
A = 54E45E4C
B = A
= 5D394F27
C = ROTL^{30}(b) B956E976
= 10101110 01010101 10111010 01011101
= AE55BA5D
D = C
= AA54C5EF
E = D
= 510C6A03
T_{40} = 54E45E4C, 5D394F27, AE55BA5D, AA54C5EF, 510C6A03

Demikian seterusnya hingga T_{59}. Dan langkah selanjutnya menggunakan nilai Parity = 67452301 dan Kt = CA62C1D6 dari 60 ≤ 79

T = ROTL^5(a) + ft (b. c, d) + e + Kt + Wt
(a) = 10010101 10000010 01110111 11101110
ROTL^5 (a) = 10110000 01001110 11111101 11010010
= B04EFDD2+ 67452301 + 3F3882F0 + CA62C1D6 + 1D44EAEF
A = 3F15771A
B = A
= 958277EE
C = ROTL^{30}(b) CF3A0EA9
= 01110011 11001110 10000011 10101010
= 73CE83AA

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

D = C
= 7621A958
E = D
= 19E1AD57
T_{60} = 3F15771A, 958277EE, 73CE83AA, 7621A958, 19E1AD57

Tabel 7. Hasil Inisialisasi Variabel Kerja

Round	A	B	C	D	E
0	8B2103BA	67452301	7BF36AE2	98BADCFE	10325476
1	98BADCFE	7BF36AE2	59D148C0	7BF36AE2	98BADCFE
2	79ACC66A	C5938A56	A2C840EE	59D148C0	7BF36AE2
3	CEE33810	79ACC66A	B164E295	A2C840EE	59D148C0
4	2FDDC5A1	CEE33810	9E6B319A	B164E295	A2C840EE
5	2FE783AC	2FDDC5A1	33B8CE04	9E6B319A	B164E295
6	FD1E6601	2FE783AC	4BF77168	33B8CE04	9E6B319A
7	EFAD00B4	FD1E6601	0BF9E0EB	4BF77168	33B8CE04
8	14122988	EFAD00B4	7F479980	0BF9E0EB	4BF77168
9	1BB8B17B	14122988	3BEB420D	7F479980	0BF9E0EB
10	AEE53E99	1BB8B17B	4EFAD083	3BEB420D	7F479980
11	71EABBA4	AEE53E99	C6EE2C5E	4EFAD083	3BEB420D
12	D4B6C792	71EABBA4	6BB94FA6	C6EE2C5E	4EFAD083
13	0A28D3C8	D4B6C792	1C7AAEE9	6BB94FA6	C6EE2C5E
14	51FEA44E	0A28D3C8	B52DB1E4	1C7AAEE9	6BB94FA6
15	866737F5	51FEA44E	028A34FA	B52DB1E4	1C7AAEE9
16	A4F7051D	866737F5	947FA913	028A34FA	B52DB1E4
17	9BD58ED9	A4F7051D	6199CDFD	947FA913	028A34FA
18	5828A0B1	9BD58ED9	693DC147	6199CDFD	947FA913
19	639DD737	5828A0B1	66F563B6	693DC147	6199CDFD
20	1FBFA0BD	639DD737	56A0282C	66F563B6	693DC147
21	3A5D6E7A	1FBFA0BD	D8E775CD	56A0282C	66F563B6
22	A03D566D	3A5D6E7A	47EFE82F	D8E775CD	56A0282C
23	A9700A52	A03D566D	8E975B9E	47EFE82F	D8E775CD
24	69A95FEE	A9700A52	680F559B	8E975B9E	47EFE82F
25	B68E620B	69A95FEE	AA5C0294	680F559B	8E975B9E
26	5138E04E	B68E620B	9A6A57FB	AA5C0294	680F559B
27	2AD1F760	5138E04E	EDA39882	9A6A57FB	AA5C0294
28	819F1767	2AD1F760	944E3813	EDA39882	9A6A57FB
29	F5FAF6CA	819F1767	0AB47DD8	944E3813	EDA39882
30	46EC628C	F5FAF6CA	E067C5D9	0AB47DD8	944E3813
31	97305762	46EC628C	BD7EBDB2	E067C5D9	0AB47DD8
32	E1A8363D	97305762	46EC628C	BD7EBDB2	E067C5D9
33	F9968ABA	E1A8363D	A5CC15D8	46EC628C	BD7EBDB2
34	206799E4	F9968ABA	786A0D8F	A5CC15D8	46EC628C
35	DDC25DE4	206799E4	BE65A2AE	786A0D8F	A5CC15D8
36	4431A80D	DDC25DE4	0819E679	BE65A2AE	786A0D8F
37	A96B17BE	4431A80D	26709779	0819E679	BE65A2AE
38	B956E976	A96B17BE	510C6A03	26709779	0819E679
39	5D394F27	B956E976	AA54C5EF	510C6A03	26709779
40	54E45E4C	5D394F27	AE55BA5D	AA54C5EF	510C6A03
41	947B13BE	54E45E4C	D74E53C9	AE55BA5D	AA54C5EF
42	6DC03DBE	947B13BE	15391793	D74E53C9	AE55BA5D
43	CA29AD32	6DC03DBE	A51EC4EF	15391793	D74E53C9
44	27ED1308	CA29AD32	9B700F6F	A51EC4EF	15391793
45	F94C89F4	27ED1308	B28A6B4C	9B700F6F	A51EC4EF

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

46	2D0EFBC6	F94C89F4	09FB44C2	B28A6B4C	9B700F6F
47	CB20E8E2	2D0EFBC6	3E53227D	09FB44C2	B28A6B4C
48	585E1A30	CB20E8E2	8B43BEF1	3E53227D	09FB44C2
49	9337178F	585E1A30	B2C83A38	8B43BEF1	3E53227D
50	D11E76A9	9337178F	1617868C	B2C83A38	8B43BEF1
51	9B9BA20C	D11E76A9	E4CDC5E3	1617868C	B2C83A38
52	D2E186B3	9B9BA20C	74479DAA	E4CDC5E3	1617868C
53	3CE5BC7F	D2E186B3	26E6E883	74479DAA	E4CDC5E3
54	8956C864	3CE5BC7F	F4B861AC	26E6E883	74479DAA
55	FCE20BC0	8956C864	CF396F3D	F4B861AC	26E6E883
56	6786B55C	FCE20BC0	2255B219	CF396F3D	F4B861AC
57	D886A561	6786B55C	3F3882F0	2255B219	CF396F3D
58	CF3A0EA9	D886A561	19E1AD57	3F3882F0	2255B219
59	958277EE	CF3A0EA9	7621A958	19E1AD57	3F3882F0
60	3F15771A	958277EE	73CE83AA	7621A958	19E1AD57
61	5AF239BC	3F15771A	A5609DFB	73CE83AA	7621A958
62	98A4827B	5AF239BC	8FC55DC6	A5609DFB	73CE83AA
63	5A4A1A7F	98A4827B	16BC8E6F	8FC55DC6	A5609DFB
64	C3FC2D22	5A4A1A7F	E629209E	16BC8E6F	8FC55DC6
65	B40D4073	C3FC2D22	D692869F	E629209E	16BC8E6F
66	215CE14A	B40D4073	B0FF0B48	D692869F	E629209E
67	60A47B82	215CE14A	ED0350C1	B0FF0B48	D692869F
68	4385895A	60A47B82	88573852	ED0350C1	B0FF0B48
69	EBFF2D0C	4385895A	98291EE0	88573852	ED0350C1
70	5C7C812E	EBFF2D0C	90E1625A	98291EE0	88573852
71	2ED04665	5C7C812E	3AFFCB43	90E1625A	98291EE0
72	7C360431	2ED04665	917F204B	3AFFCB43	90E1625A
73	B6556A82	7C360431	4BB41199	917F204B	3AFFCB43
74	57AF825A	B6556A82	5F0D810C	4BB41199	917F204B
75	78FE2461	57AF825A	AD955AA0	5F0D810C	4BB41199
76	BBCCB4BC	78FE2461	95EBE096	AD955AA0	5F0D810C
77	AA77B019	BBCCB4BC	5E3F8918	95EBE096	AD955AA0
78	D5A90252	AA77B019	2EF32D2F	5E3F8918	95EBE096
79	F2087360	D5A90252	6A9DEC06	2EF32D2F	5E3F8918

f. Tambahkan A, B, C, D, E dengan penyangga.
A = [A] 79 + A = 594D9661
B = [B] 79 + B = C576ADDB
C = [C] 79 + C = 0358C904
D = [D] 79 + D = 3F2581A5
E = [E] 79 + E = 22126B08

g. Lalu gabungkan hasil penjumlahan tersebut. Sehingga didapatlah *hash e*-dokumen (*message digest)* sebagai berikut: “594D9661 C576ADDB 0358C904 3F2581A5 22126B08”.

3.3 Analisa RSA

3.3.1 Analisa Pembangkit Kunci

Berikut langkah-langkah pembangkitan kunci pada Algoritma RSA.

a. Bangkitkan 2 bilangan prima p dan q, misalnya p = 7 dan q =13
Cari n = p*q
n = 7*13
n = 91

b. Cari $\phi(n)= (p-1) * (q-1)$
$\phi(n) = (7-1) * (13-1)$
$\phi(n) = 72$

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

c. Pilih bilangan bulat sebagai kunci publik yang relatif prima terhadap ϕ(n). Penulis memilih 5 karena 5 relatif prima terhadap 72.

d. Tentukan kunci privat d dengan persamaan d = (1 + k*ϕ(n))/e. Dimana k = 1, 2, 3,… (cari d dengan hasil bilangan bulat dengan mencoba nilai-nilai k)

k = 1 → d = 1 + 1*72 = 73/5
= 14.6 (bukan bilangan bulat)
k = 2 → d = 1 + 2*72 = 145/5
= 29 (bilangan bulat)

Sehingga didapatkan kunci public (n = 91, e = 5) dan kunci privat (n = 91, d = 29)

3.3.2 Proses Enkripsi

Setelah dokumen telah diubah menjadi *message digest* maka proses selanjutnya adalah meenkripsi *message digest* menggunakan algoritma RSA dan kunci privat yang telah dibangkitan. Pesan yang akan dienkripsi adalah hasil *hash* pada *hash* diatas yaitu "594D9661 C576ADDB 0358C904 3F2581A5 22126B08".

a. Pertama, ubah setiap "karakter" menjadi "desimal".
"53 57 52 68 57 54 54 49 67 53 55 54 65 68 68 66 48 51 53 56 67 57 48 52 51 70 50 53 56 49 65 53 50 50 49 50 54 66 48 56"

b. Selanjutnya ubah ke *chipertext* dengan rumus C= mod^e n

53^5 mod 91 = 79	67^5 mod 91 = 58
57^5 mod 91 = 57	57^5 mod 91 = 57
52^5 mod 91 = 26	48^5 mod 91 = 55
68^5 mod 91 = 87	52^5 mod 91 = 26
57^5 mod 91 = 57	51^5 mod 91 = 25
54^5 mod 91 = 45	70^5 mod 91 = 70
54^5 mod 91 = 45	50^5 mod 91 = 85
49^5 mod 91 = 56	53^5 mod 91 = 79
67^5 mod 91 = 58	56^5 mod 91 = 49
53^5 mod 91 = 79	49^5 mod 91 = 56
55^5 mod 91 = 48	65^5 mod 91 = 39
54^5 mod 91 = 45	53^5 mod 91 = 79
65^5 mod 91 = 39	50^5 mod 91 = 85
68^5 mod 91 = 87	50^5 mod 91 = 85
68^5 mod 91 = 87	49^5 mod 91 = 56
66^5 mod 91 = 40	50^5 mod 91 = 85
48^5 mod 91 = 55	54^5 mod 91 = 45
51^5 mod 91 = 25	66^5 mod 91 = 40
53^5 mod 91 = 79	48^5 mod 91 = 55
56^5 mod 91 = 49	56^5 mod 91 = 49

Sehingga diperoleh nilai chipertext nya yaitu: "79 57 26 87 57 45 45 56 58 79 48 45 39 87 87 40 55 25 79 49 58 57 55 26 25 70 85 79 49 56 39 79 85 85 56 85 45 40 55 49".

3.3.3 Proses Verifikasi

Proses verifikasi dokumen untuk membuktikan apakah dokumen sah atau tidak dimulai dengan merubah digital signature menjadi hash yang semula dengan melakukan deskripsi menggunakan kunci public. Lalu, hasil dekripsi dibandingkan dengan message digest dokumen. Jika sama maka dokumen tersebut sah dan juka tidak sama maka dokumen tersebut tidak sah.

3.3.4 Proses Dekripsi

a. Ubah *chipertext* e-dokumen menjadi pesan semula dengan persamaan M = C^d mod n
C = 79 57 26 87 57 45 45 56 58 79 48 45 39 87 87 40 55 25 79 49 58 57 55 26 25 70 85 79 49 56 39 79 85 85 56 85 45 40 55 49
D = 29

79^{29} mod 91 = 53	58^5 mod 91 = 57

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

$57^{29} \bmod 91 = 57$	$57^{5} \bmod 91 = 57$
$26^{29} \bmod 91 = 52$	$55^{5} \bmod 91 = 48$
$87^{29} \bmod 91 = 68$	$26^{5} \bmod 91 = 52$
$57^{29} \bmod 91 = 57$	$25^{5} \bmod 91 = 51$
$45^{29} \bmod 91 = 54$	$70^{5} \bmod 91 = 70$
$45^{29} \bmod 91 = 54$	$85^{5} \bmod 91 = 50$
$56^{29} \bmod 91 = 49$	$79^{5} \bmod 91 = 53$
$58^{29} \bmod 91 = 67$	$49^{5} \bmod 91 = 56$
$79^{29} \bmod 91 = 53$	$56^{5} \bmod 91 = 49$
$48^{29} \bmod 91 = 55$	$39^{5} \bmod 91 = 65$
$45^{29} \bmod 91 = 54$	$79^{5} \bmod 91 = 53$
$39^{29} \bmod 91 = 65$	$85^{5} \bmod 91 = 50$
$87^{29} \bmod 91 = 68$	$85^{5} \bmod 91 = 50$
$87^{29} \bmod 91 = 68$	$56^{5} \bmod 91 = 49$
$40^{29} \bmod 91 = 66$	$85^{5} \bmod 91 = 50$
$55^{29} \bmod 91 = 48$	$45^{5} \bmod 91 = 54$
$25^{29} \bmod 91 = 51$	$40^{5} \bmod 91 = 66$
$79^{29} \bmod 91 = 53$	$55^{5} \bmod 91 = 48$
$49^{29} \bmod 91 = 56$	$49^{5} \bmod 91 = 56$

Sehingga diperoleh nilai “53 57 52 68 57 54 54 49 67 53 55 54 65 68 68 66 48 51 53 56 67 57 48 52 51 70 50 53 56 49 65 53 50 50 49 50 54 66 48 56”

b. Ubah hasil yang telah diperoleh menjadi karakter. Sehingga menjadi seperti ini “594D9661 C576ADDB 0358C904 3F2581A5 22126B08”.
c. Bandingakan hasil deskripsi dengan *hash* e-dokumen. Jika hasil deskripsi sama dengan *hash* e-dokumen maka dokumen yang diverifikasi adalah dokumen yang sah.

4. KESIMPULAN

Untuk melindungi validitas suatu *file digital* secara maksimal dengan teknik *digital signature*, algoritma yang digunakan adalah algoritma SHA-1 dan RSA sebagai algoritma pembangkit kunci. SHA-1 dikatakan aman karena secara matematis tidak mungkin menemukan dua pesan yang berbeda yang menghasilkan nilai *hash* yang sama. Tingkat keamanan dari algoritma RSA adalah sulitnya menguraikan angka menjadi faktor-faktor prima, pemfaktoran dilakukan untuk memperoleh kunci pribadi, selama pemfaktoran bilangan besar menjadi faktor-faktor prima belum ditemukan, maka keamanan algoritma RSA tetap terjamin. Sehingga dengan penggabungan dua algoritma tersebut dianggap cukup untuk memastikan keamanan suatu *file digital*.

Berdasarkan pengujian yang telah dilakukan dapat disimpulkan bahwa proses pengamanan dan verifikasi pada *file digital* piagam penghargaan menggunakan algoritma SHA-1 dan RSA berhasil dilakukan, penerapkan teknik *digital signature file digital* pada piagam penghargaan menggunakan algoritma SHA-1 dan RSA berhasil mendeteksi adanya perubahan yang terjadi pada *file digital* piagam penghargaan meski perubahan dilakukan hanya pada satu nilai *pixel* saja.

REFERENCES

[1] M. Bobbi, K. Nasution, A. Karim, and S. Esabella, “Sistem Pendukung Keputusan Penilaian Kinerja Ketua Program Studi Menerapkan Metode WASPAS dengan Pembobotan ROC,” vol. 4, no. 1, pp. 130–136, 2022, doi: 10.47065/bits.v4i1.1619.

[2] M. Mesran, S. D. Nasution, S. Syahputra, A. Karim, and E. Purba, “Implementation of the Extended Promethee II in Upgrade Level of Mechanic,” Int. J. Sci. Res. Sci. Technol., vol. 4, no. 2, pp. 125–130, 2018.

[3] S. Dharma Hardi et al., “Implementation of Computer Based Systems for Effective Deci-sions in Acceptance of Vikar,” Int. J. Eng. Technol., vol. 7, no. 3, pp. 101–104, 2018, [Online]. Available: www.sciencepubco.com/index.php/IJET.

[4] Abdul Karim, “Implementasi Metode Multi-Objective Optimization On The Basis Of Ratio Analysis dalam Seleksi Mahasiswa Program Indonesia Pintar,” Bull. Comput. Sci. Res., vol. 3, no. 5, pp. 351–356, 2023, doi: 10.47065/bulletincsr.v3i5.283.

Management of Information System Journal
Vol 2, No 3, Juli 2024, Hal. 55 - 67
ISSN: 2964-9455 (media online)
DOI: 10.47065/mis.v2i3.1465
https://journal.fkpt.org/index.php/MIS

[5] B. S. Hasugian, "PERANAN KRIPTOGRAFI SEBAGAI KEAMANAN SISTEM INFORMASI PADA USAHA KECIL DAN MENENGAH," Jurnal Warta, vol. 53, Juli 2017.
[6] S. and P. D. Atika, "DIGITAL SIGNATURE DENGAN ALGORITMA SHA-1 DAN RSA SEBAGAI AUTENTIKASI," Jurnal Cendikia, vol. XVI, Oktober 2018.
[7] M. A. J, P. O. C van and S. A. Vanstone, Handbook of Applied Cryptography, USA: CRC Press, 1996.
[8] H. Jogiyanto, Analisa dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis, Yogyakarta: ANDI, 2005.
[9] R. Munir, Kriptografi, Yogyakarta: Informatika Bandung, 2006.
[10] M. Hanafi, "Aplikasi Pengajuan Surat Digital Menggunakan Algoritma RSA Pada LPPM UIN SUSKA," Pekan Baru, 2019.