JURNAL CYBER TECH STMIK TRIGUNA DHARMA

P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

Implementasi Digital Signature Pada Faktur Dengan Menggunakan Kombinasi Algoritma SHA-256 Dan RSA-CRT

Syahfira Chairunnisa Lubis[1], Khairi Ibnutama[2], Syarifah Fadillah Rezky[3]

[1,2,3] Sistem Informasi, STMIK Triguna Dharma
Email: [1]raralubis53@gmail.com, [2]mr.ibnutama@gmail.com, [3]ikic5500@gmail.com
Email Penulis Korespondensi: **raralubis53@gmail.com**

Abstrak− Pemodifikasian dokumen kerap terjadi di dunia bisnis, salah satunya adalah faktur. Permasalahan faktur yang sering terjadi adalah perbedaan isi antara faktur yang diterbitkan oleh pengirim dengan faktur yang di terima oleh penerima. Sehingga solusi yang tepat untuk mengatasi permasalahan tersebut dengan menggunakan mekanisme *Digital Signature*. *Digital Signature* akan dibuat dengan menggunakan kombinasi algoritma SHA-256 dan RSA-CRT. SHA-256 merupakan jenis kriptografi fungsi *hash* yang dapat digunakan untuk proses pengecekan integritas data, dan mempunyai waktu paling cepat dalam proses otentikasi. Menghasilkan 256 bit *message digest* lalu di enkripsi menggunakan kunci publik RSA-CRT. Kombinasi algoritma tersebut merupakan kombinasi yang tepat dalam penggunaan *Digital Signature* untuk memperketat keamanan data. *Digital Signature* akan dibubuhkan di setiap faktur yang telah diterbitkan dalam bentuk *QR Code*. Proses verifikasi keabsahan faktur dapat dilakukan dengan pemindaian *QR Code* yang ada di dalam faktur.

Kata Kunci: *Digital Signature*; Faktur; Kriptografi; *QR Code*; RSA-CRT (*Rivest Shamir Adleman - Chinesse Reminder Theorem*); SHA-256 (*Secure Hash Algoritm*-256).

Abstrac− *Document modifications often occur in the business world, one of which is invoices. An invoice problem that often occurs is the difference in content between the invoice issued by the sender and the invoice received by the recipient. So the right solution to overcome this problem is to use the Digital Signature mechanism. Digital Signature will be created using a combination of the SHA-256 and RSA-CRT algorithms. SHA-256 is a type of cryptographic hash function that can be used to check data integrity, and has the fastest time in the authentication process. Generates a 256 bit message digest and then encrypts it using the RSA-CRT public key. This combination of algorithms is the right combination for using Digital Signature to tighten data security. Digital Signature will be affixed to every invoice that has been issued in the form of a QR Code. The process of verifying the validity of the invoice can be done by scanning the QR Code on the invoice.*

Keywords: *Cryptography; Digital Signature; Invoice; QR Code; RSA-CRT (Rivest Shamir Adleman - Chinese Reminder Theorem); SHA-256 (Secure Hash Algorithm-256).*

1. PENDAHULUAN

Dunia bisnis sangat melekat pada teknologi dan sangat berpengaruh bagi perusahaan besar maupun kecil dikarenakan tuntutan pasar yang selalu mengikuti perkembangan zaman [1]. Perusahaan harus mengamankan catatan - catatan dan hartanya salah satunya faktur [2],[3]. Faktur merupakan dokumen yang harus disimpan oleh pihak yang berwenang [4]. Faktur adalah bukti transaksi berisi tentang rincian pengiriman barang terkait dengan penagihan untuk pembayaran. Pada umumnya dokumen ini berisi tentang jumlah barang, perhitungan pembayaran, harga satuan, dan harga total [3]. Faktur dibagi menjadi dua ada faktur penjualan dan faktur pembelian. Faktur penjualan merupakan bukti transaksi yang diberikan oleh penjual ketika barang telah sampai kepada pembeli dan pembayaran bisa dilakukan dengan kredit [5], sedangkan faktur pembelian merupakan bukti yang dibuat setelah transaksi ketika barang telah diterima dan diberikan langsung oleh penjual kepada pembeli dibayar dengan lunas [6]. Banyak permasalahan yang terjadi mengenai faktur salah satunya pemalsuan yang menyebabkan kerugian terhadap perusahaan dan merupakan tindak pidana [7].

Kasus pemalsuan faktur kerap terjadi. Faktur yang diterbitkan oleh pengirim berbeda dengan faktur yang diterima oleh penerima. Adanya perubahan isi faktur yang tidak sesuai dengan kesepakatan, yaitu penambahan dalam jumlah transaksi. Hal tersebut merugikan perusahaan karena mendapatkan keluhan serta merusak citra perusahaan. Dalam rangka meningkatkan kemanan dan menjaga keaslian faktur, maka dibuat mekanisme untuk menghasilkan faktur disertai dengan *Digital Signature*.

Digital Signature adalah sebuah teknik dalam kriptografi yang dapat digunakan untuk menanda tangani dokumen digital [8], berfungsi sebagai alat otentikasi dan verifikasi keaslian data dengan menggunakan kode acak sebagai tanda tanganya [9]. Kriptografi adalah ilmu yang mempelajari bagaimana agar pesan atau dokumen aman, tidak bisa dibaca oleh pihak yang tidak berhak [10]. Algoritma yang akan digunakan untuk menghasilkan *Digital Signature* pada faktur pembelian distributor ini adalah kombinasi SHA-256 dan algoritma RSA-CRT. Kombinasi algoritma

P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index
digunakan agar memperkuat kemanan data dari pemodifikasian. *Digital Signature* akan dibuat berbasis web agar dapat memudahkan dalam prorses memverifikasi faktur dan mencegah terjadinya pemalsuan faktur.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Metode penelitian digunakan untuk mengumpulkan informasi atau data dalam menyelesaikan masalah. Berikut metode penelitian yang digunakan dalam penelitian ini:

1. Observasi
   Observasi adalah salah satu teknik pengumpulan data yang dimana data diambil dengan melakukan tinjauan dan pengamatan secara langsung ke objek penelitian.
2. Wawancara
   Wawancara adalah salah satu teknik pengumpulan data yang dimana data diambil dengan melakukan tanya jawab kepada pihak yang bersangkutan mengenai hal yang akan dijadikan informasi.
3. Studi Pustaka
   Referensi merupakan hal yang penting di dalam sebuah penelitian, sebab referensi berguna sebagai penguat sebuah karya tulis agar menjadikan karya tulis penelitian menjadi relevan.

2.2 Kriptografi

Berasal dari bahasa Yunani kriptografi (*cryptography*) yaitu *cryptos* dan *graphia* yang berarti penulisan rahasia [11]. Rinaldi, Munir berpendapat bahwa kriptografi merupakan ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek kemanan informasi misalnya kerahasiaan, integritas data, otentikasi pengirim/penerima data, dan otentikasi data [12]. Seni dan ilmu yang mampu mengubah data atau pesan menjadi kode tertentu dan kode tersebut dapat menjadi data atau pesan kembali jikadengan mengetahui kunci rahasia, sehingga berfungsi sebagai dapat terjaga kerahsiaannya disebut kriptografi. [13].

2.3 Fungsi Hash

Fungsi *hash* merupakan suatu fungsi matematika yang mengambil masukan panjang variabel dan mengubahnya ke dalam urutan biner dengan panjang yang tetap [14]. Fungsi *hash* mengubah pesan asli menjadi serangkaian kode acak yang disebut dengan *message digest* [15]. Pesan yang telah diubah menjadi *message digest* tidak dapat dikembalikan lagi menjadi pesan semula karena sifatnya satu arah dan pesan yang berbeda akan menghasilkan nilai *hash* yang berbeda [16]. Adanya perubahan 1 (satu) bit saja akan mengubah keluaran hash secara drastis. Fungsi *hash* biasanya digunakan untuk menjamin integritas dan *digital signature* [17]. Fungsi *hash* digunakan untuk menjamin bahwa data atau pesan yang dikirim tidak mengalami modifikasi atau pemalsuan selama proses transmisi. Suatu fungsi *hash* akan memetakan bit-bit *string* dengan panjang sembarang ke sebuah *string* dengan panjang tertentu misalnya *n*. Proses pemetaan suatu input *string* output tersebut disebut dengan proses *hashing*. Nilai *hash*, kode *hash* atau hasil *hash* merupakan sebutan dari output fungsi *hash* [18].

2.4 Digital Signature

Digital Signature pertama kali ditemukan oleh Diffie dan Hellman pada tahun 1976 [19]. *Digital Signature* atau tanda tangan digital adalah mekanisme otentikasi yang memungkinkan pembuat pesan dapat melampirkan sebuah kode yang bertindak sebagai tanda tangan berfungsi untuk menguji keutuhan dan otentikasi suatu dokumen digital, serta dapat mendeteksi perubahan dokumen dari hasil manipulasi [10], [20]. Tanda tangan digital yang dibubuhkan dalam suatu dokumen digital dapat memastikan bahwa pesan yang ditandatangani dan diterima adalah asli [21]. Tanda tangan digital dapat menjamin kemanan dokumen yang di tanda tanganinya dalam aspek *non-repudation* yang artinya anti penyangkalan apabila dokumen valid maka pengirim tidak bisa menyangkal keberadaan dokumen bahwa memang benar dikirim oleh pengirim yang bersangkutan [8].

2.5 SHA-256

Secure Hash Algorithm 256 (SHA-256) adalah sebuah kriptografi fungsi hash satu arah yang dirancang oleh *National Security Agency (NSA)* dan dipublikasikan oleh *National Institute of Standart and Technology (NIST)* sebagai sebuah *Federal Information Processing Standart (FIPS)* oleh *U.S* pada tahun 2002 [22]. Ada empat algoritma untuk keamanan fungsi hash yaitu SHA-0, SHA-1, SHA-2, dan SHA-3 [23]. SHA 256 merupakan pengembangan dari SHA 0 dan SHA 1 [24]. SHA-256 menghasilkan *message digest* sepanjang 256 bit [25]. Algoritma SHA-256 dapat digunakan untuk melakukan pengecekan integritas data, pembuatan *Digital Signature*, dan lain-lain [26]. Dalam *sign* dan *verify* SHA 2 yang mempunyai waktu paling cepat dan baik dalam melakukan proses otentikasi [27]. SHA –256 tergolong aman karena didesain sedemikian rupa sehingga tidak memungkinkan mendapatkan pesan yang berhubungan dengan *message digest* yang sama [28].

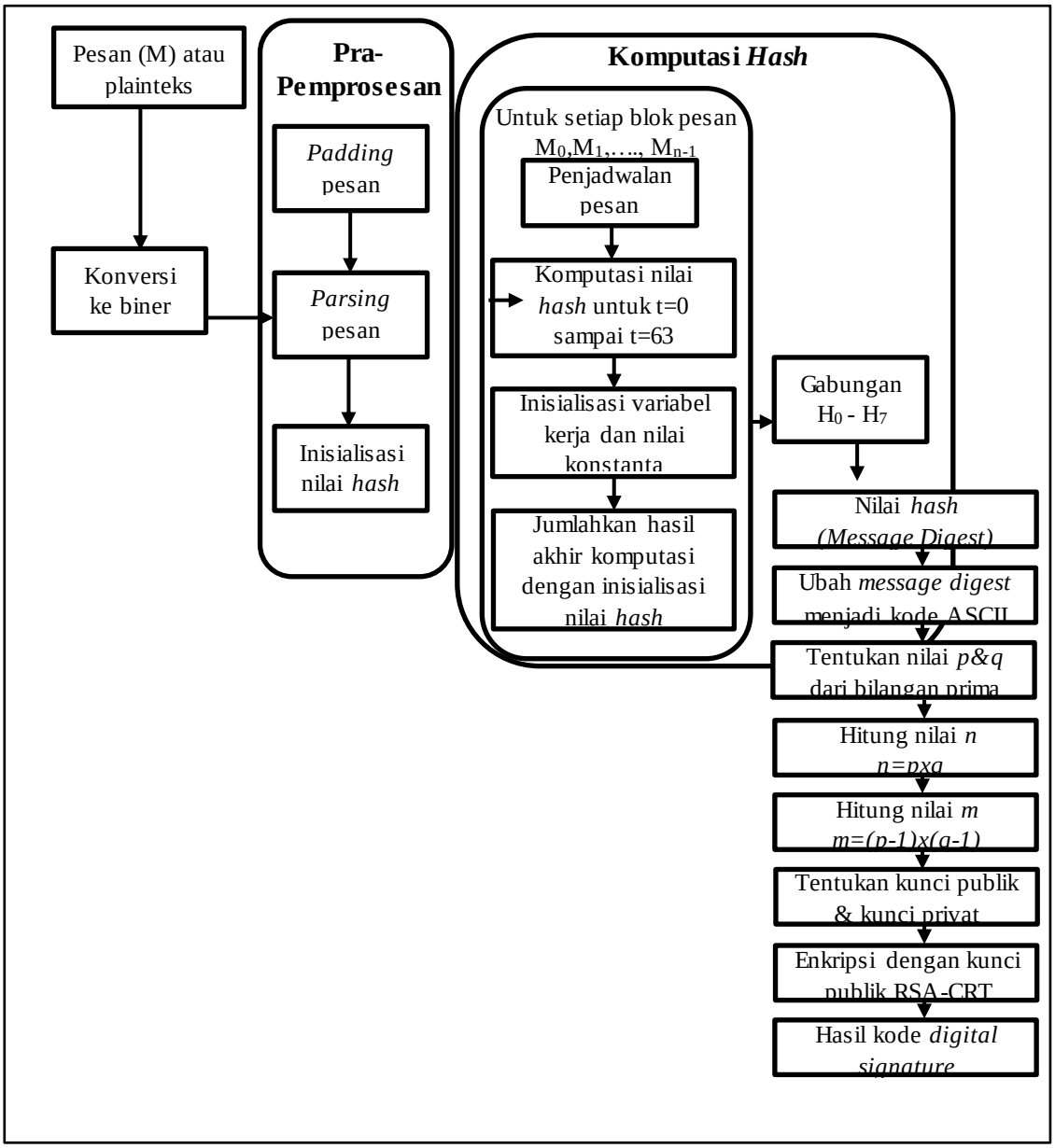
JURNAL CYBER TECH STMIK TRIGUNA DHARMA
P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

2.6 RSA-CRT

RSA-CRT (*Rhivest-Shamir-Adleman* dengan *Chinese Remainder Theorem*) merupakan suatu metode kriptografi yang sama dengan RSA biasa, namun memanfaatkan teorema CRT untuk memperpendek ukuran bit eksponen dekripsi [29]. CRT (*Chinese Remainder Theorem*) merupakan suatu algoritma untuk mengurangi perhitungan aritmatika modular dengan modulus besar untuk perhitungan yang sama untuk masing-masing faktor dari modulus [30]. Pada dasarnya, proses enkripsi pesan dengan RSA tidak membutuhkan proses perhitungan dengan teori seperti CRT. Hal ini dikarenakan nilai pemangkatan eksponensial modulus yang diproses tidaklah besar. CRT hanya diperlukan untuk menurunkan nilai pemangkatan yang besar pada saat dekripsi pesan untuk menghemat waktu pengerjaan [31].

2.7 Kerangka Kerja Algoritma SHA-256 dan RSA-CRT

SHA-256 dan RSA-CRT adalah algoritma yang digunakan dalam proses pembentukan *Digital Signature* pada penelitian ini. Plainteks akan di enkripsi dengan fungsi *hash* SHA-256 dan menghasilkan *message digest* kemudian *message digest* akan dienkripsi dengan kunci publik dari algoritma RSA-CRT.



Gambar 1. Diagram Algoritma SHA-256 dan RSA-CRT

P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

3. HASIL DAN PEMBAHASAN

Pada bagian ini berisi hasil dan pembahasan dari topik penelitian, yang bisa di buat terlebih dahulu metodologi penelitian. Bagian ini juga merepresentasikan penjelasan yang berupa penjelasan, gambar, tabel dan lainnya. Banyaknya kata pada bagian ini berkisar.

3.1 Penerapan Algoritma SHA-256 dan RSA-CRT

a. Tentukan Plainteks

Data yang akan diolah adalah nomer permintaan faktur yang tertera di dalam faktur. Data nomer faktur yang digunakan adalah “PO21100461”.

b. Konversi Plainteks ke Biner

Plainteks yang digunakan berjumlah 10 digit dan akan diubah ke dalam bentuk biner.

(P = 01010000, O = 01001111, 2 = 00000010, 1 = 00000001, 1 = 00000001, 0 = 00000000, 0 = 00000000, 4 = 00000100, 6 = 00000110, 1 = 00000001)

Panjang pesan (l) = 10 x 8 = 80 bit.Maka panjang pesannya adalah 80 bit.

c. Tambahkan Bit Padding

Tahap berikutnya adalah menambahkan bit-bit pengganjal sehingga total panjangnya 512 bit. *Padding* dilakukan dengan cara menambahkan bit 1 dan sisanya adalah bit 0 sejumlah k dengan persamaan sebagai berikut:

$l + 1 + k \equiv 448 \bmod 51$ (1)

$80 + 1 + k \equiv 448 \bmod 512$

$81 + k \equiv 448 \bmod 512$

$k = 448 - 81 \bmod 512$

$k = 367 \bmod 512$

Karena k = 367, maka banyak bit 0 yang akan ditambahkan adalah sebanyak 367 bit. Lalu pada akhir pesan yang di*padding*, ditambahkan 8 bit nilai l yaitu 80 = 10100000. Hasil pesan yang di*padding* adalah sebagai berikut:

Tabel 1. Hasil *Padding* Pesan

01010000	01001111	00000010	00000001	00000001	00000000	00000000	00000100
00000110	00000001	10000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000	00000000	00000000	00000000	10100000

d. Lakukan Parsing Pesan

Selanjutnya adalah membagi setiap blok 512 bit menjadi 16 blok dan masing-masing blok berukuran 32 bit. Berikut adalah hasil *parsing* pesan:

Tabel 2. Parsing Pesan

Blok	Biner
$M_1^{(0)}$	01010000 01001111 00000010 00000001
$M_2^{(0)}$	00000001 00000000 00000000 00000100
$M_3^{(0)}$	00000110 00000001 10000000 00000000
$M_4^{(0)}$	00000000 00000000 00000000 00000000
$M_5^{(0)}$	00000000 00000000 00000000 00000000
$M_6^{(0)}$	00000000 00000000 00000000 00000000

JURNAL CYBER TECH STMIK TRIGUNA DHARMA
P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

$M_7^{(0)}$	00000000	00000000	00000000 00000000
$M_8^{(0)}$	00000000	00000000	00000000 00000000
$M_9^{(0)}$	00000000	00000000	00000000 00000000
$M_{10}^{(0)}$	00000000	00000000	00000000 00000000
$M_{11}^{(0)}$	00000000	00000000	00000000 00000000
$M_{12}^{(0)}$	00000000	00000000	00000000 00000000
$M_{13}^{(0)}$	00000000	00000000	00000000 00000000
$M_{14}^{(0)}$	00000000	00000000	00000000 00000000

Blok	Biner
$M_{15}^{(0)}$	00000000 00000000 00000000 00000000
$M_{16}^{(0)}$	00000000 00000000 00000000 10100000

e. Inisiallisasi Nilai Hash

Nilai *hash* terdiri dari 8 kata 32 bit dalam bentuk heksadesimal, sebagai berikut:

a = H0 (0) = 6A09E667
b = H1 (0) = BB67AE85
c = H2 (0) = 3C6EF372
d = H3 (0) = A54FF53A
e = H4 (0) = 510E527F
f = H5 (0) = 9B05688C
g = H6 (0) = IF83D9AB
h = H7 (0) = 5BE0CD19

f. Lakukan Penjadwalan Pesan

Mengubah setiap blok pesan menjadi bilangan heksadesimal yang diberi label W_0, W_1, W_2, sampai W_{63} dengan ketentuan sebagai berikut :

$$Wt = \begin{cases} M_t^{(t)} & 0 \le t \le 15 \\ \sigma_1^{(256)}(Wi-2) + Wi - 7 + \sigma_0^{(256)}(Wi-15) + Wi - 16 & 16 \le t \le 63 \end{cases} \quad (2)$$

Dimana :

$$\sigma_1^{(256)}(W_{i-2}) = ((W_{i-2})ROTR\ 17) \oplus ((W_{i-2})ROTR\ 19 \oplus ((W_{i-2})SHR10) \quad (3)$$

$$\sigma_0^{(256)}(W_{i-15}) = ((W_{i-15})ROTR\ 7) \oplus ((W_{i-15})ROTR\ 18 \oplus ((W_{i-15})SHR3) \quad (4)$$

Keterangan :

Wt = Blok pesan yang baru
Mt = Blok pesan yang lama
W_{i-2} = Blok pesan dari W ke i-2
W_{i-15} = Blok pesan dari W ke i-15
ROTR = Rotate Right
SHR = Shift Right
$\oplus$ = Operator XOR

Tabel 3. Penjadwalan Pesan

W0	504F0201	W16	924F020A	W32	3D869ACC	W48	19581D66
W1	01000004	W17	8E423148	W33	C8FE85A2	W49	A5248819
W2	06018000	W18	214FC144	W34	E2F7CFE2	W50	90E57231
W3	00000000	W19	9669BE2D	W35	B834068B	W51	DE872D61
W4	00000000	W20	B44ED8CA	W36	3F0DFB3A	W52	945BF77B
W5	00000000	W21	99998833	W37	5E39265B	W53	1F099B95
W6	00000000	W22	CCEE77DC	W38	FD234098	W54	79522E12
W7	00000000	W23	924E5809	W39	1E9031CD	W55	216EA6F2
W8	00000000	W24	8DEC873D	W40	CB132BF2	W56	00F35B5B
W9	00000000	W25	3D3DE8E9	W41	8B47F883	W57	DDD6FB8F

W10	00000000	W26	0E54F166	W42	209704E9	W58	D563B330
W11	00000000	W27	ED884433	W43	B662E66C	W59	B41F2CF8
W12	00000000	W28	601A4F32	W44	08E5BA2A	W60	56D231E1
W13	00000000	W43	1843E197	W45	D9F3C438	W61	D854B08D
W14	00000000	W30	CE9CA6E5	W46	7112A99A	W62	89B1D4EC
W15	000000A0	W31	30FAD8AD	W47	4651BEB7	W63	3692E6E2

Untuk perhitungan penjadwalan pesan ke 16 sampai ke 63 dilakukan dengan penyelesaian dengan persamaan rumus sebagai berikut:

$\sigma_1^{(256)}(W_{i-2}) = ((W_{i-2})ROTR\ 17) \oplus ((W_{i-2})ROTR\ 19 \oplus ((W_{i-2})SHR10)$

= (00000000 00000000 00000000 00000000) ⊕(00000000 00000000 00000000 00000000) ⊕ (0000000000000000 00000000 00000000)

= 00000000 00000000 00000000 00000000 = 00000000

$((W_{16-7})$ $= W_9$

$= 00000000$

$\sigma_0^{(256)}(W_{i-15}) = ((W_{i-15})ROTR\ 7) \oplus ((W_{i-15})ROTR\ 18 \oplus ((W_{i-15})SHR3$

= (00000010 0000000 00000000 00001000) ⊕(0100000 00000000 00000000 00000001) ⊕ (00000000 00000000 00000000 00000000)

= 11001111 10000100 01010001 00000000 = 42000009

(W_{16-16}) $= W_0$

$= 504F0201$

W_t $= \sigma_1^{(256)}(W_{16-2}) + W_{16-7} + \sigma_0^{(256)}(W_{16-15}) + W_{16-16}$

$= 00000000 + 00000000 + 42000009 + 504F0201$

$= 924F020A$

Perhitungan dilakukan sampai dengan W_{63}.

g. Inisialisasi Variabel Kerja dan Konstanta

Setiap variabel kerja a, b, c, d, e, f, g, dan h daimbil dari inisial nilai *hash*.

Tabel 4. Nilai Konstanta

428A2F98	71374491	B5C0FBCF	E9B5DBA5	3956C25B	59F111F1	923F82A4	AB1C5ED5
D807AA98	12835B01	243185BE	550C7DC3	72BE5D74	80DEB1FE	9BDC06A7	C19BF174
E49B69C1	EFBE4786	0FC19DC6	240CA1CC	2DE92C6F	4A7484AA	5CB0A9DC	76F988DA
983E5152	A831C66D	B00327C8	BF597FC7	C6E00BF3	D5A79147	06CA6351	14434367
27B70A85	2E1B2138	4D2C6DFC	53380D13	650A7354	766A0ABB	81C2C92E	92722C85
A2BFE8A1	A81A664B	C24B8B70	C76C51A3	D192E819	D6990624	F40E3585	106AA070
19A4C116	1E376C08	2748774C	34B0BCB5	391C0CB3	4ED8AA4A	5B9CCA4F	682E6FF3
48F82EE	78A5636F	84C87814	8CC70208	90BEFFFA	A4506CEB	BEF9A3F7	C67178F2

h. Hitung komputasi Fungsi Hash

Komputasi *hash* dilakukan untuk mencari nilai t. Proses penyelesaian komputasi fungsi *hash* dilakukan dari t0 hingga t63.

Tabel 5. Nilai Komputasi

JURNAL CYBER TECH STMIK TRIGUNA DHARMA
P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
Init	6A09E667	BB67AE85	3C6EF372	A54FF53A	510E527F	9B05688C	1F83D9AB	5BE0CD19
t=0	4C578A4E	6A09E667	BB67AE85	3C6EF372	E916E4A3	510E527F	9B05688C	1F83D9AB
t=1	7F03F68E	4C578A4E	6A09E667	BB67AE85	B1DC843B	E916E4A3	510E527F	9B05688C
t=2	9E3BDC12	7F03F68E	4C578A4E	6A09E667	C7366809	B1DC843B	E916E4A3	510E527F
t=3	A767190D	9E3BDC12	7F03F68E	4C578A4E	0AF2D4C4	C7366809	B1DC843B	E916E4A3
t=4	C4618819	A767190D	9E3BDC12	7F03F68E	13C46A73	0AF2D4C4	C7366809	B1DC843B
t=5	7131E3B5	C4618819	A767190D	9E3BDC12	B1DC1C0F	13C46A73	0AF2D4C4	C7366809
t=6	3266AD5E	7131E3B5	C4618819	A767190D	64D7DBAD	B1DC1C0F	13C46A73	0AF2D4C4
t=7	D2D2174B	3266AD5E	7131E3B5	C4618819	3D1C982C	64D7DBAD	B1DC1C0F	13C46A73
t=8	450A5C8F	D2D2174B	3266AD5E	7131E3B5	90420140	3D1C982C	64D7DBAD	B1DC1C0F
t=9	B2D5917E	450A5C8F	D2D2174B	3266AD5E	B57AD57F	90420140	3D1C982C	64D7DBAD
t=10	51708407	B2D5917E	450A5C8F	D2D2174B	400013DE	B57AD57F	90420140	3D1C982C
t=11	ACF46289	51708407	B2D5917E	450A5C8F	F7FF2E05	400013DE	B57AD57F	90420140
t=12	CBE60D99	ACF46289	51708407	B2D5917E	B1019067	F7FF2E05	400013DE	B57AD57F
t=13	AA333E78	CBE60D99	ACF46289	51708407	0D130FF4	B1019067	F7FF2E05	400013DE
t=14	37A1FA8A	AA333E78	CBE60D99	ACF46289	C86BD349	0D130FF4	B1019067	F7FF2E05
t=15	F886545D	37A1FA8A	AA333E78	CBE60D99	1B838CDB	C86BD349	0D130FF4	B1019067
t=16	EB0D388D	F886545D	37A1FA8A	AA333E78	36B0A107	1B838CDB	C86BD349	0D130FF4
t=17	501F8C8B	EB0D388D	F886545D	37A1FA8A	747F3090	36B0A107	1B838CDB	C86BD349
t=18	FB31C256	501F8C8B	EB0D388D	F886545D	6C5551B5	747F3090	36B0A107	1B838CDB
t=19	7D341E46	FB31C256	501F8C8B	EB0D388D	0F43F71D	6C5551B5	747F3090	36B0A107
t=20	CEE44A45	7D341E46	FB31C256	501F8C8B	83B3FCC7	0F43F71D	6C5551B5	747F3090
t=21	D6E6C9BD	CEE44A45	7D341E46	FB31C256	7713997A	83B3FCC7	0F43F71D	6C5551B5
t=22	72E4BF28	D6E6C9BD	CEE44A45	7D341E46	EBC04537	7713997A	83B3FCC7	0F43F71D
t=23	6301592C	72E4BF28	D6E6C9BD	CEE44A45	924A9321	EBC04537	7713997A	83B3FCC7
t=24	A00FABA1	6301592C	72E4BF28	D6E6C9BD	D596F0EE	924A9321	EBC04537	7713997A
t=25	F260A3B5	A00FABA1	6301592C	72E4BF28	5BA14635	D596F0EE	924A9321	EBC04537
t=26	050532EC	F260A3B5	A00FABA1	6301592C	B22EDA4E	5BA14635	D596F0EE	924A9321
t=27	1E410A35	050532EC	F260A3B5	A00FABA1	5E51EB36	B22EDA4E	5BA14635	D596F0EE
t=28	CFCAB889	1E410A35	050532EC	F260A3B5	4788DC79	5E51EB36	B22EDA4E	5BA14635
t=29	9AACA407	CFCAB889	1E410A35	050532EC	E06D15C7	4788DC79	5E51EB36	B22EDA4E
t=30	8A76C70D	9AACA407	CFCAB889	1E410A35	7CAB6FE6	E06D15C7	4788DC79	5E51EB36
t=31	4406C77E	8A76C70D	9AACA407	CFCAB889	556B594A	7CAB6FE6	E06D15C7	4788DC79
t=32	92BEA4BD	4406C77E	8A76C70D	9AACA407	26742D3E	556B594A	7CAB6FE6	E06D15C7
t=33	FCA16147	92BEA4BD	4406C77E	8A76C70D	59457381	26742D3E	556B594A	7CAB6FE6
t=34	CE6F5D01	FCA16147	92BEA4BD	4406 C77E	CE6F5D01	59457381	26742D3E	556B594A
t=35	D43DE0FE	CE6F5D01	FCA16147	92BE A4BD	1113A074	CE6F5D01	59457381	26742D3E
t=36	831F219F	D43DE0FE	CE6F5D01	FCA16147	0E766B6B	1113A074	CE6F5D01	59457381
t=37	FF7B8017	831F219F	D43DE0FE	CE6F5D01	369259A9	0E766B6B	1113A074	CE6F5D01
t=38	0D2A7971	FF7B8017	831F219F	D43DE0FE	FBA8FCFB	369259A9	0E766B6B	1113A074
t=39	DE3E71D2	0D2A7971	FF7B8017	831F219F	7079E67E	FBA8FCFB	369259A9	0E766B6B
t=40	E6A9D97A	DE3E71D2	0D2A7971	FF7B8017	65C880B7	7079E67E	FBA8FCFB	369259A9
t=41	36DC9791	E6A9D97A	DE3E71D2	0D2A7971	9314F5AC	65C880B7	7079E67E	FBA8FCFB
t=42	47B7A0F8	36DC9791	E6A9D97A	DE3E71D2	DAC470BC	9314F5AC	65C880B7	7079E67E
t=43	2BA6F28B	47B7A0F8	36DC9791	E6A9D97A	7020102F	DAC470BC	9314F5AC	65C880B7
t=44	603B46F0	2BA6F28B	47B7A0F8	36DC9791	59B3A096	7020102F	DAC470BC	9314F5AC
t=45	86C2674E	603B46F0	2BA6F28B	47B7A0F8	97510A01	59B3A096	7020102F	DAC470BC
t=46	FCE98B70	86C2674E	603B46F0	2BA6F28B	8FA52FC3	97510A01	59B3A096	7020102F
t=47	0480911B	FCE98B70	86C2674E	603B46F0	88ABF5D3	8FA52FC3	97510A01	59B3A096
t=48	9EC7F1EC	0480911B	FCE98B70	86C2674E	550D6990	88ABF5D3	8FA52FC3	97510A01
t=49	2F150BCD	9EC7F1EC	0480911B	FCE98B70	62034364	550D6990	88ABF5D3	8FA52FC3
t=50	1D16B678	2F150BCD	9EC7F1EC	0480911B	C9F7A134	62034364	550D6990	88ABF5D3
t=51	A7FEA8F6	1D16B678	2F150BCD	9EC7F1EC	9F0AE0C6	C9F7A134	62034364	550D6990
t=52	D768BAC5	A7FEA8F6	1D16B678	2F150BCD	3230B25E	9F0AE0C6	C9F7A134	62034364
t=53	6FA2FA0C	D768BAC5	A7FEA8F6	1D16B678	0619C83C	3230B25E	9F0AE0C6	C9F7A134
t=54	5DC4A17B	6FA2FA0C	D768BAC5	A7FEA8F6	528CAB05	0619C83C	3230B25E	9F0AE0C6
t=55	ACA556F2	5DC4A17B	6FA2FA0C	D768BAC5	26951A8F	528CAB05	0619C83C	3230B25E
t=56	C9C1 08D4	ACA556F2	5DC4A17B	6FA2FA0C	A99CD6FB	26951A8F	528CAB05	0619C83C
t=57	0D9CFFEA	C9C108D4	ACA556F2	5DC4A17B	3E8599EA	A99CD6FB	26951A8F	528CAB05
t=58	4A0CF720	0D9CFFEA	C9C108D4	ACA556F2	0A833E7E	3E8599EA	A99CD6FB	26951A8F
t=59	6B78727C	4A0CF720	0D9CFFEA	C9C108D4	3632DD07	0A833E7E	3E8599EA	A99CD6FB
t=60	C7B4864F	6B78727C	4A0CF720	0D9CFFEA	5893E266	3632DD07	0A833E7E	3E8599EA
t=61	3EC9B568	C7B4864F	6B78727C	4A0CF720	E0EB7080	5893E266	3632DD07	0A833E7E
t=62	45484CB6	3EC9B568	C7B4864F	6B78727C	550D4EF7	E0EB7080	5893E266	3632DD07
t=63	432AD4AF	45484CB6	3EC9B568	C7B4864F	DD386DC6	550D4EF7	E0EB7080	5893E266

Perhitungan t=0 sampai t=63 dilakukan dengan menggunakan rumus:

$T_1 = h + \sum_1^{(256)}(e) + Ch(e,f,g) + K_t^{(256)} + W_t$ (5)

$= 5BE0CD19 + 3587272B + 1F85C985 + 428A2F98 + 504F0201 = 43C6EF69$

$T_2 = \sum_0^{(256)}(a) + Maj\ (a,b,c)$ (6)

$= CE20B47E + 3A6FE667 = 08909AE5$

$e = d + T_1$ (7)

$= A54FF53A + 43C6EF69 = E916E4A3$

$a = T_1 + T_2$ (8)

$= 43C6EF69 + 08909AE5 = 4C578A4E$

P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

i. Jumlahkan Hasil Akhir dengan Nilai Hash Value

Tabel 6. Penjumlahan Dengan Nillai *Hash Value*

Variabel	Inisial Nilai *Hash* Value	+	Variabel Kerja	Hasil
$H_0^{(0)}$	6A09E667	+	432AD4AF	AD34BB16
$H_1^{(0)}$	BB67AE85	+	45484CB6	00AFFB3B
$H_2^{(0)}$	3C6EF372	+	3EC9B568	7B38A8DA
$H_3^{(0)}$	A54FF53A	+	C7B4864F	6D047B89
$H_4^{(0)}$	510E527F	+	DD386DC6	2E46C045
$H_5^{(0)}$	9B05688C	+	550D4EF7	F012B783
$H_6^{(0)}$	IF83D9AB	+	E0EB7080	006F4A2B
$H_7^{(0)}$	5BE0CD19	+	5893E266	B474AF7F

j. Menghasilkan Message Digest

Setelah melakukan perhitungan, maka nilai *message digest* yang dihasilkan dari pesan "PO21100461" yaitu:
AD34BB1600AFFB3B7B38A8DA6D047B892E46C045F012B783006F4A2BB474AF7F

k. Ubah Message Digest ke Bentuk ASCII

Message digest yang diperoleh dari hasil hash SHA-256 akan diubah dalam bentuk kode ASCII, karena proses enkripsi RSA-CRT menggunakan bilangan desimal. Maka hasil konversi *message digest* dari nilai hash SHA-256 kedalam bentuk desimal adalah: 10 13 3 4 11 11 1 6 0 0 10 15 15 11 3 11 7 11 3 8 10 8 13 10 6 13 0 4 7 11 8 9 2 14 4 6 12 0 4 5 15 0 1 2 11 7 8 3 0 0 6 15 4 10 2 11 11 4 7 4 10 15 7 15

l. Tentukan Nilai p dan q

$p = 151$ dan $q = 137$ (9)

m. Hitung Nilai n

$n = p.q = 151 \times 137 = 20687$ (10)

n. Tentukan Kunci Publik dan Kunci Privat

1. Hitung $\phi(m)$ Dengan Menggunakan Persamaan Pada Rumus

$\phi(m) = (p - 1) . (q - 1)$ (11)

$\phi(m) = (151 - 1) . (137 - 1)$

$= (150) . (136) = 20400$

2. Tentukan Nilai *e* Acak Sebagai Kunci Publik

Dengan syarat memenuhi *Greater Common Divisor* (GCD) dimana $(e, \phi(m)) = 1$, $1 < e < \phi(m)$. dengan menggunakan persamaan pada rumus

$r_0 = q_1 r_1 + r_2, 0 < r_2 < r_1$ (12)

$r_1 = q_2 r_2 + r_3, 0 < r_3 < r_2$

$r_n = \ldots\ldots\ldots$

Perhitungan akan berhenti jika nilai r berahir pada angka 0. Jika nilai sebelum *r* terakhir adalah 1 maka angka yang menjadi percobaan benar sebagai *e*.

Maka gcd (e,20400) = 1

Percobaan pertama nilai e = 6

$r_0 = 6$

$r_1 = 20400$

$r_0 = q_1 r_1 + r_2$

$6 = 0.20400 + 6$

$r_1 = q_2 r_2 + r_3, 0 < r_3 < r_2$

$20400 = 3400. 6 + 0$

Karena angka pada *r* terakhir sebelum 0 tidak 1, tetapi 6 maka 6 bukanlah nilai *e*, dilakukan percobaan kedua.

Percobaan kedua dengan nilai $e = 7$

$r_0 = 7$

$r_1 = 204000$

$r_0 = q_1 r_1 + r_2$

$7 = 0.20400 + 7$

$r_1 = q_2 r_2 + r_3, 0 < r_3 < r_2$

$20400 = 4314.7 + 2$

$r_2 = q_3 r_3 + r_4, 0 < r_4 < r_5$

$7 = 3.2 + 1$

$r_3 = q_4 r_4 + r_5, 0 < r_5 < r_6$

$2 = 2.1 + 0$

P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

Dari perhitungan diatas, angka pada *r* terakhir sebelum 0 adalah 1, maka 7 adalah angka yang tepat untuk nilai *e*, maka e = 7.

3. Membangkitkan Kunci Privat *d* Dengan Menggunakan Persamaan Rumus

e x d mod m = 1 (13)

7 x 8743 mod 20400 = 1

61201 x mod 20400 = 1

4. Tentukan Nilai *dP* Dengan Menggunakan Persamaan Rumus

$dP = e^{-1} \bmod (p-1) = d \bmod (p-1)$ (14)

dP = 8743 mod (151-1)

dP = 8743 mod 150

dP = 43

5. Tentukan Nilai d*Q* Dengan Menggunakan Persamaan Rumus

$dQ = e^{-1} \bmod (q-1) = d \bmod (q-1)$ (15)

dQ = 8743 mod (137-1)

dQ = 8743 mod 136

dQ = 39

6. Menentukan Nilai *qInv* Dengan Menggunakan Persamaan Rumus

$qInv = q^{-1} \bmod p$ (16)

$qInv = 137^{-1} \bmod p$

$qInv$ = 137 x 97 mod 151

artinya 137 x 97 ≡ 1 mod 151 , sama dengan 137 x 97 mod 151 = 1

Dari hasil perhitungan diatas, maka dapat disimpulkan bahwa:

Kunci Publik adalah pasangan dari (e = 7 dan n = 20687)

Kunci Privat adalah pasangan dari (dP= 43, dQ= 39, $QInv$= 97, p= 151, q= 137)

o. Enkripsi dengan Kunci Publik RSA-CRT

Didalam proses enkripsi *message digest* yang sudah dikonversi menjadi kode ASCII akan dirubah menjadi chiperteks. Proses ini akan menggunakan kunci publik (*e*,*n*). Untuk mempermudah perhitungan maka, *message digest* dipecah menjadi blok yang lebih kecil. Dimulai dari M_1 sampai M_{64}. Dimulai dari M_1= 10, M_2= 13,M_3= 3 sampai dengan M_{64}. Rumus enkripsi adalah sebagai berikut:

$C = M^e \pmod n$ (17)

$C_1 = M_1^e \pmod n$

$= 10^7 \bmod 20687$

= 8179

$C_2 = M_2^e \pmod n$

$= 13^7 \bmod 20687$

= 4846

Diteruskan sampai dengan C_{64}.

Hasil enkripsi dari RSA-CRT yaitu:

8179 4846 2187 16384 17 17 1 1005 0 0 8179 5442 5542 17 2187 17 16750 17 2187 7765 8179 7765 4846 8179 11005 4846 0 16384 16750 17 7765 4247 128 13239 16384 11005 1924 0 16384 16064 5442 0 1 128 17 16750 7765 2187 0 0 1105 5442 16384 8179 128 17 17 16384 16750 16384 8179 5442 16750 5442

p. Hasil Kode Digital Signature

Maka hasil enkripsi RSA-CRT dikonversikan kembali ke dalam heksadesimal yaitu:

IFF312EE88B4000111112AFD001FF3154215421188B11416E1188B1E551FF31E5512EE1FF32AFD12EE 04000416E111E5510B08033B740002AFD784040003EC01542018011416E1E5588B002AFD154240001FF3 8011114000416E40001FF31542416E1542

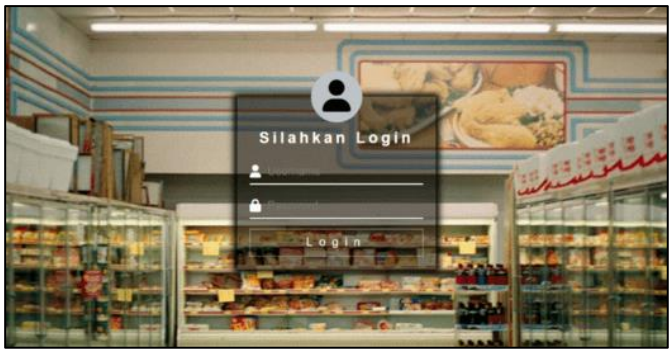
3.2 Implementasi Sistem

Implementasi sistem merupakan penerapan dari proses perancangan sistem yang dioperasikan secara menyeluruh.

a. Rancangan Form Login

Form login ini merupakan *form* yang harus diisi oleh admin. Hanya admin yang mempunyai hak akses untuk masuk ke dalam sistem. Admin harus mengisi *username* dan *password* yang sesuai dengan sistem agar sistem terbuka ke halaman berikutnya dan siap mengolah data faktur yang berada di dalam sistem.

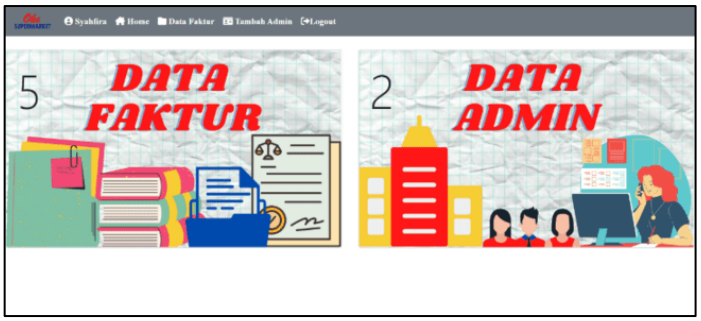
JURNAL CYBER TECH STMIK TRIGUNA DHARMA
P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index



Gambar 2. Tampilan *Form Login*

b. Rancangan Halaman *Home*

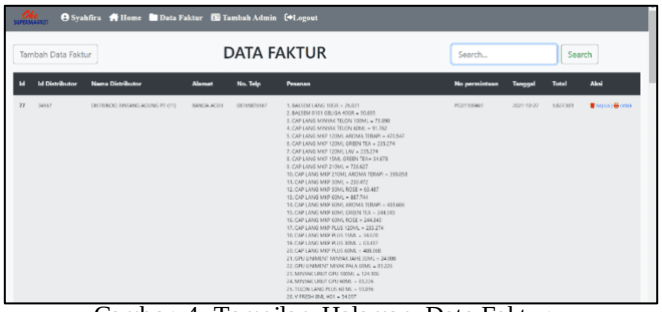
Halaman *Home* akan terbuka ketika proses *login* telah berhasil. Halaman *Home* berisi informasi jumlah data faktur dan data admin yang ada pada sistem.



Gambar 3. Tampilan Halaman *Home*

c. Rancangan Halaman Data Faktur

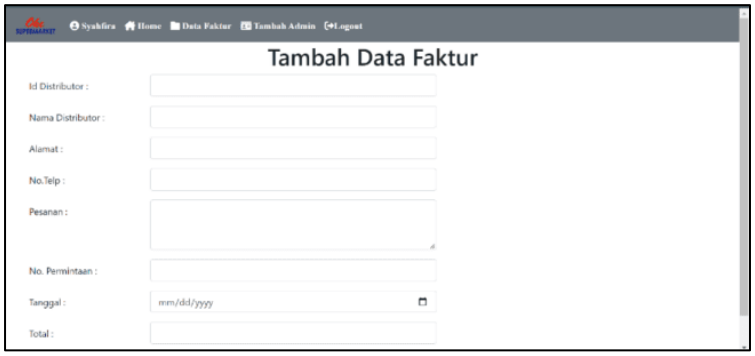
Pada halaman data faktur terdapat daftar data faktur yang telah di input oleh admin. Data faktur tersebut berisi informasi faktur yang akan diterbitkan.



Gambar 4. Tampilan Halaman Data Faktur

d. Tampilan *Form* Tambah Data Faktur

Halaman tambah data faktur berisi *form* yang harus diisi oleh admin berkaitan dengan isi faktur yang akan diterbitkan. Ketika *form* sudah diisi serta disimpan maka data faktur akan tersimpan ke *database* sistem dan akan tampil di halaman data faktur.



Gambar 5. Tampilan Form Tambah Data Faktur

JURNAL CYBER TECH STMIK TRIGUNA DHARMA
P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

e. Tampilan Halaman Cetak Faktur

Halaman cetak faktur merupakan halaman print preview pada faktur pembelian distributor yang akan dicetak melalui aksi cetak pada halaman data faktur. Faktur pembelian distributor yang akan dicetak sudah disertai *QR Code* yang memudahkan para distributor untuk melakukan verifikasi pada faktur.

Gambar 6 Tampilan Cetak Faktur

d. Tampilan Halaman Verifikasi Faktur

Halaman verifikasi faktur akan terbuka ketika para distributor sudah melakukan *scan QR Code* pada faktur. *QR Code* yang berada di faktur akan mengarahkan ke halaman verifikasi faktur. Apabila faktur tersebut asli maka halaman verifikasi faktur akan menampilkan data data faktur yang sesuai dengan faktur yang telah diterima oleh distributor dan menyatakan bahwa faktur asli, *signature* ditemukan. Namun, jika faktur tersebut palsu dan mengalami modifikasi maka halaman verifikasi faktur akan menampilkan data data faktur kosong dan menyatakan bahwa faktur palsu, *signature* tidak ditemukan.

Gambar 7. Tampilan Halaman Verifikasi

4. KESIMPULAN

Permasalahan dalam pemodifikasian faktur dapat diatasi dengan mengimplementasikan sistem *Digital Signature* berbasis web. *Digital Signature* dapat menjamin dan memenuhi tujuan kriptografi yaitu menjaga integritas data, kerahasiaan data, otentikasi serta penyangkalan. Kombinasi algoritma SHA-256 dan RSA-CRT yang digunakan dalam pembuatan *Digital Signature* membantu menciptakan kode kode acak yang rumit sehingga sulit untuk dipecahkan. Untuk membuktikan bahwa faktur asli dikirim oleh pengirim yaitu dengan cara memvalidasi *QR Code* yang berada di

P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

faktur, jika *QR Code* yang dikirim oleh pengirim mengeluarkan informasi yang sama dengan faktur yang diterima menandakan bahwa faktur tersebut asli, namun jika *QR Code* tidak memunculkan informasi apapun yang tidak sama dengan faktur yang diterima maka sudah terjadi pemodifikasian faktur. Dengan diterapkannya digital signature pada faktur berbasis web dapat memudahkan para penggguna dalam proses verifikasi.

UCAPAN TERIMAKASIH

Terima kasih disampaikan kepada Bapak Khairi Ibnutama dan Ibu Syarifah Fadillah Rezky, serta pihak-pihak yang telah mendukung dalam proses penyelesaian penelitian ini.

REFERENCES

[1] L. Y. Siregar and M. I. P. Nasution, "Perkembangan Teknologi Informasi Terhadap Peningkatan Bisnis Online," *HIRARKI J. Ilm. Manaj. dan Bisnis*, vol. 02, no. 01, pp. 71–75, 2020, [Online]. Available: http://journal.upp.ac.id/index.php/Hirarki%0APERKEMBANGAN.
[2] C. A. Hassan Basrie, Fahri Fuady, "Analisis Sistem Pengendalian Intern Atas Penjualan dan Penerimaan Kas," *J. Akunt. Keuang.*, vol. 2, no. 2, pp. 203–216, 2011.
[3] A. Yulianto and A. Ariani, "Perancangan Sistem Informasi Pembuatan E-Invoice Pada PT. Hasta Perkasa Graha Berbasis Web," *REMIK (Riset dan E-Jurnal Manaj. Inform. Komputer)*, vol. 4, no. 2, p. 39, 2020, doi: 10.33395/remik.v4i2.10555.
[4] F. Revando Rawung and K. Kunci, "Analisis Efektivitas Sistem Akuntansi Penjualan Dan Penerimaan Kas Pada Pt. Surya Wenang Indah Manado Analysis of Effectiveness of Sales Accounting System and Cash Receipts in Pt. Surya Wenang Indah Manado," *J. Berk. Ilm. Efisiensi*, vol. 16, no. 01, pp. 795–805, 2016.
[5] S. Rosyafah, P. S. Akuntansi, F. Ekonomi, U. Bhayangkarasurabaya, and P. Kredit, "ANALISIS PERANCANGAN SISTEM DAN PROSEDUR," pp. 97–105.
[6] H. Supriono, "Analisis Pelaksanaan Sistem Akuntansi Pembelian Untuk Meningkatkan Efektivitas Sistem Pengendalian Manajemen," *Ris. Mhs. Akunt.*, vol. xx, pp. 1–14, 2015.
[7] B. Erlina, M. Safitri, R. Setya, and C. Phourtuna, "PERUSAHAAN DISTRIBUTOR LAMPU BOHLAM BERBAGAI MEREK," vol. 4, no. 1, pp. 231–242, 2021.
[8] R. A. Azdy, "Tanda tangan Digital Menggunakan Algoritme Keccak dan RSA," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 5, no. 3, pp. 184–191, 2016, doi: 10.22146/jnteti.v5i3.255.
[9] 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, doi: 10.33633/tc.v18i2.2166.
[10] E. Cahyo Prabowo and I. Afrianto, "Ilmiah Komputer dan PENERAPAN DIGITAL SIGNATURE DAN KRIPTOGRAFI PADA Teknik Informatika – Universitas Komputer Indonesia Ilmiah Komputer dan," *J. Ilm. Komput. dan Inform.*, vol. 6, no. 2, 2017.
[11] Romindo, "Analisa Perbandingan Algoritma Monoalphabetic Cipher Dengan Algoritma One Time Pad Sebagai Pengamanan Pesan Teks," *Sink. (Jurnal dan PenelitianTeknik Inf.*, vol. 2, no. 2, pp. 62–66, 2018.
[12] A. Ginting, R. R. Isnanto, and I. P. Windasari, "Implementasi Algoritma Kriptografi RSA untuk Enkripsi dan Dekripsi Email," *J. Teknol. dan Sist. Komput.*, vol. 3, no. 2, p. 253, 2015, doi: 10.14710/jtsiskom.3.2.2015.253-258.
[13] F. N. Pabokory, I. F. Astuti, and A. H. Kridalaksana, "Implementasi Kriptografi Pengamanan Data Pada Pesan Teks, Isi File Dokumen, Dan File Dokumen Menggunakan Algoritma Advanced Encryption Standard," *Inform. Mulawarman J. Ilm. Ilmu Komput.*, vol. 10, no. 1, p. 20, 2016, doi: 10.30872/jim.v10i1.23.
[14] J. Rekayasa, S. Komputer, F. Mipa, U. Tanjungpura, J. Prof, and H. H. Nawawi, "PENERAPAN ALGORITMA KNAPSACK DAN FUNGSI HASH PADA SISTEM E-VOTING (Studi Kasus: Pemilihan Raya Mahasiswa Universitas Tanjungpura Pontianak)," vol. 07, no. 01, 2019.
[15] F. Nuraeni, Y. H. Agustin, and I. M. Muharam, "Implementasi Tanda Tangan Digital Menggunakan RSA dan SHA-512 Pada Proses Legalisasi Ijazah," *Konf. Nas. Sist. Inf.*, pp. 864–869, 2018.
[16] I. Saputra and S. D. Nasution, "Analisa Algoritma SHA-256 Untuk Mendeteksi Orisinalitas Citra Digital," *Pros. Semin. Nas. Ris. Inf. Sci.*, vol. 1, no. September, p. 164, 2019, doi: 10.30645/senaris.v1i0.20.
[17] T. Abdurrachman and B. R. Suteja, "Pengembangan Sistem Informasi Asosiasi Jasa Konstruksi dengan Menerapkan Tanda Tangan Digital," *J. Tek. Inform. dan Sist. Inf.*, vol. 7, no. 1, pp. 261–273, 2021, doi: 10.28932/jutisi.v7i1.3431.
[18] R. Prasetyo and A. Suryana, "Aplikasi Pengamanan Data dengan Teknik Algoritma Kriptografi AES dan Fungsi Hash SHA-1 Berbasis Desktop," *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 5, no. 2, pp. 61–65, 2016, doi: 10.32736/sisfokom.v5i2.40.
[19] E. V. Waruwu, A. A, F. Sonata, and I. Zulkarnain, "Penerapan Digital Signature Menggunakan Metode Rsa Untuk Menvalidasi Keaslian Ijazah Sma Swasta Bina Artha," *J-SISKO TECH (Jurnal Teknol. Sist. Inf. dan Sist. Komput. TGD)*, vol. 3, no. 2, p. 45, 2020, doi: 10.53513/jsk.v3i2.2033.
[20] H. Agung and Ferry, "Kriptografi Menggunakan Hybrid Cryptosystem dan Digital Signature," *J. Tek. Inform. dan Sist. Inf.*, vol. 3, no. 1, pp. 34–45, 2016.
[21] H. Tuban, "i RANCANG BANGUN SISTEM E- VOTING DENGAN MENERAPKAN HASH DAN DIGITAL SIGNATURE UNTUK VERIFIKASI DATA HASIL VOTING ...," 2014.
[22] M. B. Kurniawan *et al.*, "Aplikasi Nilai Online Menggunakan One Time Password Dengan Algoritma Sha 512 Berbasis Web Pada Smp Pgri 336," *Skanika Vol. 1 No. 1 Maret 2018 Apl.*, vol. 1, no. 1, pp. 411–416, 2018.

P-ISSN : 2988-2508 ; E-ISSN : 2987-9604
Volume 5. Nomor 1, Edisi Januari 2022
https://ojs.trigunadharma.ac.id/index.php/jct/index

[23] Z. Panjaitan, E. F. Ginting, and Y. Yusnidah, "Modifikasi SHA-256 dengan Algoritma Hill Cipher untuk Pengamanan Fungsi Hash dari Upaya Decode Hash," *J. SAINTIKOM (Jurnal Sains Manaj. Inform. dan Komputer)*, vol. 19, no. 1, p. 53, 2020, doi: 10.53513/jis.v19i1.225.

[24] R. Ardilla, "Rancang Bangun Sistem E – Voting Dengan Metode," pp. 1–10, 2018, [Online]. Available: http://repository.unim.ac.id/id/eprint/276.

[25] A. Fauzi, "Ekstraksi Citra Pada Proses Keamanan Kriptografi Memanfaatkan Algoritma Secure Hash (Sha)," *J. Inform. Kaputama*, vol. 4, no. 1, 2020, [Online]. Available: https://jurnal.kaputama.ac.id/index.php/JIK/article/view/222.

[26] Y. Anugrah, M. Hannats, H. Ichsan, and A. Kusyanti, "Implementasi Algoritme SHA-256 Menggunakan Protokol MQTT pada Budidaya Ikan Hias," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 4, pp. 4066–4074, 2019.

[27] L. Refialy, E. Sediyono, and A. Setiawan, "Pengamanan Sertifikat Tanah Digital menggunakan Digital Signature SHA-512 dan RSA," *J. Tek. Inform. dan Sist. Inf.*, vol. 1, no. 3, pp. 229–234, 2015, doi: 10.28932/jutisi.v1i3.400.

[28] A. Y. Mulyadi, E. P. Nugroho, and R. R. J. P, "Implementasi Algoritma AES 128 dan SHA – 256 Dalam Pengkodean pada Sebagian Frame Video CCTV MPEG-2," *JATIKOM J. Teor. dan Apl. Ilmu Komput.*, vol. 1, no. 1, pp. 33–39, 2018.

[29] E. B. Setiawan and Y. S. Nugraha, "Kriptografi Citra Menggunakan Metode Rivest-Shamir-Adleman Chinese Remainder Theorem Di Konsultan XYZ," *J. Ultim.*, vol. 7, no. 2, pp. 82–90, 2016, doi: 10.31937/ti.v7i2.357.

[30] A. Arief and R. Saputra, "Implementasi Kriptografi Kunci Publik dengan Algoritma RSA-CRT pada Aplikasi Instant Messaging," *Sci. J. Informatics*, vol. 3, no. 1, pp. 46–54, 2016, doi: 10.15294/sji.v3i1.6115.

[31] Z. Panjaitan, K. Ibnutama, and M. G. Suryanata, "Penggunaan Chinese Reminder Theorem (CRT) pada Algoritma RSA," *J. SAINTIKOM (Jurnal Sains Manaj. Inform. dan Komputer)*, vol. 18, no. 1, p. 41, 2019, doi: 10.53513/jis.v18i1.102.