BULLETIN OF COMPUTER SCIENCE RESEARCH
ISSN 2774-3659 (Media Online)
Vol 1, No 2, April 2021
Hal 48-56
https://hostjournals.com/bulletincsr

Implementasi Metode *Secure Hash Algorithm*-1 Untuk Mendeteksi Keaslian *File* Dokumen

Dermawan Lumban Toruan, Rivalri Kristianto Hondro

Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia
Email: [1]dermawan123@gmail.com, [2]rivalryhondro@gmail.com

Abstrak−File dokumen merupakan suatu sarana transformasi informasi dari satu orang ke orang lain atau dari suatu kelompok kelompok lain. Perkembangan teknologi komputerisasi ini sudah sangat meninggkat. File Dokumen sangat rentan terhadap penipuan, penyandapan maupun pencurian data oleh pihak- pihak yang tidak bertanggung jawab. Demi menjaga keamanan File dokumen dapat dilakukan dengan pemanfaatan teknik kriptografi. Kriptografi yaitu ilmu untuk menjaga keamanan data. Kriptografi merupakan salah satu metode pengamanan data yang dapat digunakan menjaga keaslian data, kerahasiaan data, serta keaslian pengiriman data. SHA merupakan singkatan *Secure Hash Algorithm* merupakan fungsi *hash standard* yang dipublikasi oleh NIST *(National Institut e of Standard and Technology)*, (NIST, 1995a). Penelitian ini akan menggunakan Metode SHA-1 untuk mengamankan suatu keaslian File dokumen, kerahasiaan dokumen, integritas dokumen, dan auntentikasi dokumen. Penelitian ini menguraikan proses pengamanan untuk mendeteksi keaslian *file* dokumen dengan menggunakan Metode SHA-1 dalam bentuk pendeteksian agar dokumen yang bersifat rahasia yang dikirim melalui telekomunikasi umum tidak dapat dirubah atau dimodifikasi oleh orang yang tidak berhak atau orang yang tidak berkepentingan. Hal ini dilakukan sebagai upaya untuk meminimalisir tindakan-tindakan penipuan, hoax, ataupun penyalahgunaan *file* dokumen.

Kata Kunci: Kriptografi; *File* Dokumen; SHA-1

Abstract−Document file is a means of transforming information from one person to another or from a group of other groups. The development of computerized technology has increased greatly. Document files are very vulnerable to fraud, interception and data theft by irresponsible parties. For the sake of maintaining the security of document files can be done by using cryptographic techniques. Cryptography is the science of maintaining data security. Cryptography is a data security method that can be used to maintain data authenticity, data confidentiality, and data transmission authenticity. SHA stands for Secure Hash Algorithm, a standard hash function published by NIST (National Institute of Standards and Technology), (NIST, 1995a). This research will use the SHA-1 method to secure the authenticity of document files, document confidentiality, document integrity, and document authentication. This study describes the security process to detect the authenticity of document files using the SHA-1 method in the form of detection so that confidential documents sent via public telecommunications cannot be changed or modified by unauthorized persons or unauthorized persons. This is done as an effort to minimize acts of fraud, hoaxes, or misuse of document files.

Keywords: Cryptography; Document Files; SHA-1

1. PENDAHULUAN

Perkembangan tegnologi umumnya pada bidang komputer sudah menjadi suatu kebutuhan bagi ribuan orang atau diseluruh dunia. Karena dengan tegnologi ini atau memakai komputer ini banyak pekerjaan yang dapat diselesaikan dengan cepat, akurat, dan efesien bahkan ratusan organisasi contohnya perusahaan, lembaga keuangan, lembaga negara dan lain sebagainya telah menggunakan komputer sebagai alat terpenting di instansi tersebut. Namun kemajuan tegnologi ini selalu memiliki sisi buruk dan sisi baik dari tegnologi itu sendiri. Keamanan suatu *file* dokumen adalah hal yang sangat penting bagi setiap orang baik itu pribadi ataupun umum, karena *file* dokumen ada yang bersifat penting dan rahasia sehingga pemilik data tidak ingin dokumen yang mereka miliki diketahui atau bahkan di ubah oleh orang yang tidak berhak. Sehingga menimbulkan tuntutan akan tersedianya suatu sistem pengamanan data dan akhirnya banyak orang mengembangkan atau melakukan berbagai cara untuk mengatasi persoalan keamanan data. Intinya adalah bagaimana agar orang lain tidak dapat menipulasi keaslian *file* yang kita miliki,atau bagaimana orang yang tidak berhak atau orang yang tidak berkepentingan, tidak dapat merubah, merusak atau memodifikasi keaslian *file* dokumen tersebut.

Kriptografi yaitu ilmu untuk menjaga keamanan data. Kriptografi merupakan salah satu metode pengamanan data yang dapat digunakan menjaga keaslian data, kerahasiaan data, serta keaslian pengiriman data. Metode ini bertujuan agar dokumen yang bersifat rahasia yang dikirim melalui telekomunikasi umum tidak dapat dirubah atau dimodifikasi oleh orang yang tidak berhak atau orang yang tidak berkepentingan. Teknik kriptografi terdiri dari simetri dan asimetri. Teknik ini digunakan untuk mengamankan suatu keaslian dokumen, kerahasiaan dokumen, integritas dokumen, dan auntentikasi dokumen. Salah satu pengamanan yang dapat dilakukan adalah dengan cara membuktikan atau mendeteksi keaslian *file* dokumen apakah *file* tersebut masih asli atau sudah dimodifikasi orang lain atau orang yang tidak berkepentingan sehingga pemilik *file* dokumen tersebut dapat mengetahui perubahan isi dari *file* dokumennya. *Algoritma Secure Hash* dirancang oleh NSA-Badan Keamanan Nasional yang merupakan Standar Pemrosesan Informasi Federal (FIPS) A.S. yang diterbitkan oleh Institut Standar dan Teknologi NIST-Nasional Amerika Serikat. SHA-1 merupakan keluarga fungsi *hash* satu-arah, fungsi *hash* SHA yang paling umum digunakan adalah SHA-1 yang telah diimplementasikan didalam bebagai aplikasi dan protokol keamanan seperti TLS, SSL, PGP, SSH, S/MIME, dan *Ipsec*. Penelitian sebelumnya telah menggunakan

Algorithm ini untuk pengamanan informasi dengan judul Implementasi *Secure Hash Algorithm*-1 Untuk Pengamanan Data Dalam Library Pada Pemrograman Java[1].

2. METODOLOGI PENELITIAN

2.1 File Dokumen

Dokumen merupakan suatu sarana transformasi informasi dari satu orang ke orang lain atau dari suatu kelompok ke kelompok lain. Dokumen meliputi berbagai kegiatan yang diawali dengan bagaimana suatu dokumen dibuat[9].

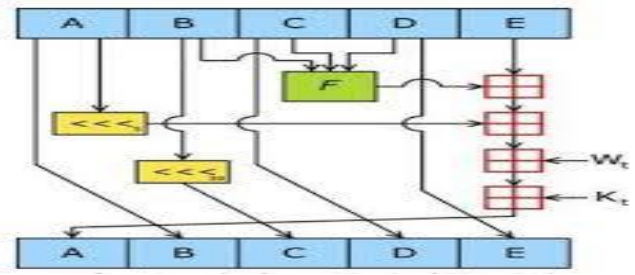
File adalah entitas dari data yang disimpan didalam sistem file yang dapat diakses dan diatur oleh pengguna. Sebuah *file* memiliki nama yang unik dalam direktori di mana ia berada. Alamat direktori dimana suatu berkas ditempatkandiistilahkan dengan path. Sebuah *file* berisi aliran data (atau data stream) yang berisi sekumpulan data yang saling berkaitan serta atribut berkas yang disebut dengan properties yang berisi informasi mengenai file yang berisi informasi mengenai *file* yang bersangkutan seperti informasi mengenai kapan sebuah berkas dibuat*udio* (Suara) adalah fenomena fisik yang dihasilkan oleh getaran suatu

2.2 *Secure Hash Algorithm* (SHA-1)

SHA merupakan singkatan *Secure Hash Algorithm* merupakan fungsi *hash standard* yang dipublikasi oleh NIST *(National Institute of Standard and Technology)*, (NIST, 1995a). Langkah-langkah pembuatan *message digest* dengan algoritma SHA-1 adalah sebagai berikut[7]:

1. *Input* Pesan yang akan di *hash* SHA-1.
2. Ubah pesan menjadi deretan biner.
3. Penambahan bit-bit pengganjal, yaitu dengan menambahkan pesan dengan sejumlah bit pengganjal sedemikian sehingga panjang pesan (dalam satuan bit) kongruen dengan 448 mod 512.
4. Penambahan nilai panjang pesan semula, yaitu pesan ditambah lagi dengan 64 bit yang representasi *biner* dari panjang pesan asli.
5. Inisialisasi Nilai *Hash*, pada algoritma SHA-1 nilai *hash*, H(0) terdiri dari 5 *words* dengan besar 32 bit dalam notasi *hexadecimal*.
6. *Output* nilai *hash* adalah nilai terakhir dari *buffer*.

Berdasarkan tahapan yang ada pada Fungsi *Hash* SHA-1, maka skema Fungsi *Hash* SHA-1 dapat dilihat pada gambar berikut ini :



Gambar 1 Skema Fungsi *Hash* SHA-1

3. HASIL DAN PEMBAHASAN

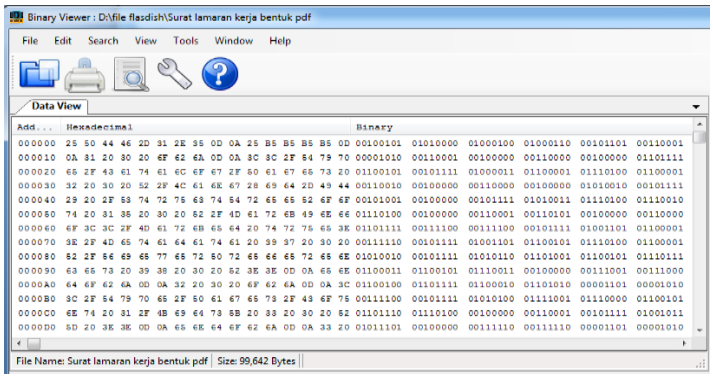
Adapun pendeteksian keaslian file dokumen adalah dengan cara membandingkan hasil kunci yang didapatkan oleh metode *Secure Hash Algorithm-1 (SHA-1*).

Langkah langkah Metode *Secure Hash Algorithm-1 (SHA-1):*

1. Penambahan bit-bit pengganjal dan nilai panjang pesan semula
2 .Inisialisasi penyangga nilai Hash Message Digest (MD)
3. Pengolahan pesan dalam blok berukuran 512 bit (Parsing)

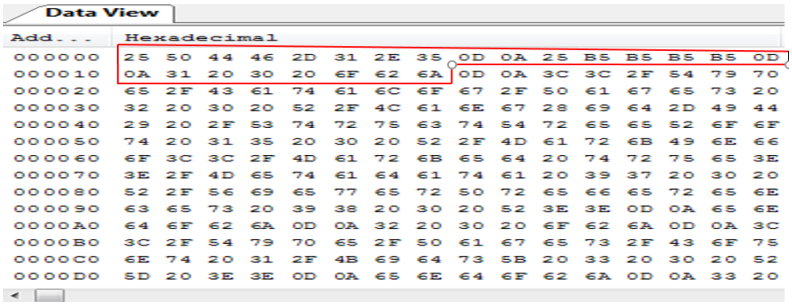
Pada kasus asli dalam proses ini seperti dijelaskan dalam analisa yaitu *file* dokumen Lamaran kerja (Surat Lamaran Kerja.pdf). data yang diambil hanya sebanyak 24 byte untuk plainteks, cara pengambilan nilai hex data *file* dokumen menggunakan aplikasi Binary Viewer, seperti dibawah ini:

Gambar di bawah ini adalah data heksadesimal dari *file dokumen SHA-1* Surat lamaran kerja bentuk *pdf* menggunakan aplikasi *Binary Viewer*.



Gambar 2. Data *Dokumen PDF*

Dari data tersebut diambil sebanyak 24 byte atau 48 karakter heksa desimal dan dikonversi ke biner, yang berguna untuk mengetahui nilai biner dari bilangan tersebut.



Gambar 3. Data *byte dokumen* "Surat lamaran kerja bentuk *pdf*

CLS_S yaitu *Circular Left Shift* dengan maksud pergeseran atau rotasi bit ke kiri sebanyak s kali, untuk a CLS_5 berarti sebanyak 5 kali dan untuk b CLS_{30} berarti 30 kali. Berikut bentuk CLS :
(a) = 0110 0111 0100 0101 0010 0011 0000 0001
CLS_5 (a) = 1110 1000 1010 0100 0110 0000 0010 1100

Tabel 1. Lima penyangga a, b, c, d, dan e

A	B	C	D	E
67452301	EFCDAB89	98BADCFE	10325476	C3D2E1F0

Tabel 2. Hasil t_0 berisi a_0, b_0, c_0, d_0, dan e_0

a_0	b_0	c_0	d_0	e_0
DC2C348A	67452301	7BF36A62	98BADCFE	10325476

t_0 = DC2C348A 67452301 7BF36A62 98BADCFE 10325476
Dan selanjutnya untuk mengerjakan t_1 yaitu :
A1= DC2C348A
B1= 67452301
C1=7BF36A62
D1=98BADCFE
E1= 10325476

CLS_S yaitu *Circular Left Shift* dengan maksud pergeseran atau rotasi bit ke kiri sebanyak s kali, untuk a CLS_5 berarti sebanyak 5 kali dan untuk b CLS_{30} berarti 30 kali. Berikut bentuk CLS :
a1 =1101 1100 0010 1100 0011 0100 1000 1010
CLS_5 (a1) =1000 0101 1000 0110 1001 0001 0101 1011

Tabel 3. Lima penyangga a, b, c, d, dan e

A	B	C	D	E
DC2C348A	67452301	7BF36A62	98BADCFE	10325476

ISSN 2774-3659 (Media Online)
Vol 1, No 2, April 2021
Hal 48-56
https://hostjournals.com/bulletincsr

Tabel 4. Hasil t_1 berisi a_1, b_1, c_1, d_1, dan e_1

a_1	b_1	c_1	d_1	e_1
0C4E020E	DC2C348A	59D148C0	7BF36A62	98BADCFE

t_1 = 0C4E020E DC2C348A 59D148C0 7BF36A62 98BADCFE
Proses untuk t_1 hingga t_{79} sangat panjang, disini langsung dibuat hasil t keseluruhan.

Tabel 5. Hasil a, b, c, d, dan e untuk t_0 hingga t_{79}

Round	A	B	C	D	E
t_0	DC2C348A	67452301	7BF36A62	98BADCFE	10325476
t_1	0C4E020E	DC2C348A	59D148C0	7BF36A62	98BADCFE
t_2	69DA4214	0C4E020E	B70B0D22	59D148C0	7BF36A62
t_3	5186A4A5	69DA4214	83138083	B70B0D22	59D148C0
t_4	09224BBC	5186A4A5	1A769085	83138083	B70B0D22
t_5	C70B2438	09224BBC	1A769085	1A769085	83138083
t_6	1318FBE	C70B2438	EEE670E1	1A769085	1A769085
t_7	CC0A000B	1318FBE	BB3845B6	EEE670E1	1A769085
t_8	C9E5C96E	CC0A000B	ACC621A1	BB3845B6	EEE670E1
t_9	9FE6FAF6	C9E5C96E	F3022002	ACC621A1	BB3845B6
t_{10}	50B9D56F	9FE6FAF6	B279725B	F3022002	ACC621A1
t_{11}	F79D7A07	50B9D56F	A7F9BEBD	B279725B	F3022002
t_{12}	42A97D03	F79D7A07	D46E755B	A7F9BEBD	B279725B
t_{13}	92FADB53	42A97D03	FDE75E81	D46E755B	A7F9BEBD
t_{14}	B739E8CE	92FADB53	D0AA5F40	FDE75E81	D46E755B
t_{15}	44CC9469	B739E8CE	D0AA5F40	D0AA5F40	FDE75E81
t_{16}	74536887	44CC9469	ADCE7A33	D0AA5F40	D0AA5F40
t_{17}	56527838	74536887	5133251A	ADCE7A33	D0AA5F40
t_{18}	64810A2D	56527838	DD14DA21	5133251A	ADCE7A33
t_{19}	96473A34	64810A2D	15949E0E	DD14DA21	5133251A
t_{20}	9DB4518A	96473A34	5920428B	15949E0E	DD14DA21
t_{21}	FAB7587C	9DB4518A	2591CE8D	5920428B	15949E0E
t_{22}	C4151F2E	FAB7587C	A76D1462	2591CE8D	15949E0E
t_{23}	92521856	C4151F2E	3EADD61F	A76D1462	2591CE8D
t_{24}	9695CA2F	92521856	B10547CB	3EADD61F	A76D1462
t_{25}	CDC36219	9695CA2F	A4948615	B10547CB	3EADD61F
t_{26}	86C41CB2	CDC36219	E5A5728B	A4948615	B10547CB
t_{27}	26DFC15C	86C41CB2	7370D886	E5A5728B	A4948615
t_{28}	F613ACF3	26DFC15C	A1B1072C	A1B1072C	E5A5728B
t_{29}	1022C2C7	F613ACF3	09B7F057	A1B1072C	7370D886
t_{30}	77C2C7CA	1022C2C7	FD84EB3C	09B7F057	A1B1072C
t_{31}	25A3A732	77C2C7CA	C408B0B1	FD84EB3C	A1B1072C
t_{32}	504FAC54	25A3A732	9DF0B1F2	C408B0B1	FD84EB3C
t_{33}	F60D2724	504FAC54	8968E9CC	9DF0B1F2	C408B0B1
t_{34}	D0AB57EA	F60D2724	1413EB15	8968E9CC	9DF0B1F2
t_{35}	8A7CE8B9	D0AB57EA	3D8349C9	1413EB15	8968E9CC
t_{36}	FD2427F6	8A7CE8B9	B42AD5FA	3D8349C9	1413EB15
t_{37}	D6B0F3C4	FD2427F6	629F3A2E	B42AD5FA	3D8349C9
t_{38}	C268985E	D6B0F3C4	BF4909FD	629F3A2E	B42AD5FA
t_{39}	B150E233	C268985E	35AC3CF1	BF4909FD	629F3A2E
t_{40}	B5646936	B150E233	B09A2617	35AC3CF1	BF4909FD
t_{41}	C4EBE642	B5646936	EC54388C	B09A2617	35AC3CF1
t_{42}	A4A5768E	C4EBE642	AD591A4D	EC54388C	B09A2617
t_{43}	C9AD01CB	A4A5768E	B13AF990	AD591A4D	EC54388C
t_{44}	86695B97	C9AD01CB	A9295DA3	B13AF990	AD591A4D
t_{45}	A72AE1AA	86695B97	F26B4072	A9295DA3	B13AF990
t_{46}	B50780E0	A72AE1AA	E19156E5	F26B4072	A9295DA3
t_{47}	98269011	B50780E0	A9CAB86A	E19156E5	F26B4072

Round	A	B	C	D	E
t_{48}	C39DC3A6	98269011	2D41E038	A9CAB86A	E19156E5
t_{49}	F5D2A00	C39DC3A6	6608A404	2D41E038	A9CAB86A
t_{50}	6690E670	F5D2A00	B0E770E9	6608A404	2D41E038
t_{51}	07443189	6690E670	0BD74A80	B0E770E9	6608A404
t_{52}	E8DD562B	07443189	19A4399C	0BD74A80	B0E770E9
t_{53}	69992CA1	E8DD562B	41D10C62	19A4399C	0BD74A80
t_{54}	191C9321	69992CA1	FA37558A	41D10C62	19A4399C
t_{55}	672C9DED	191C9321	5A664B28	FA37558A	41D10C62
t_{56}	5A51FEA1	672C9DED	565725C8	5A664B28	FA37558A
t_{57}	763AB7CA	5A51FEA1	59CB277B	565725C8	5A664B28
t_{58}	B9D96D18	763AB7CA	56947FA8	59CB277B	565725C8
t_{59}	C8DCDEF8	B9D96D18	9D8EADF2	56947FA8	59CB277B
t_{60}	0118CCC2	C8DCDEF8	2D765B46	9D8EADF2	56947FA8
t_{61}	64674FB3	0118CCC2	323737BE	2D765B46	9D8EADF2
t_{62}	27124362	64674FB3	80463330	323737BE	2D765B46
t_{63}	B51C924A	27124362	D919D3EC	80463330	323737BE
t_{64}	259D1A22	B51C924A	89C490D8	D919D3EC	80463330
t_{65}	55C8A800	259D1A22	AD472492	89C490D8	D919D3EC
t_{66}	8CC19B45	55C8A800	89675E88	AD472492	89C490D8
t_{67}	F9CE89E2	8CC19B45	15722A00	89675E88	AD472492
t_{68}	A1C43505	F9CE89E2	633066D1	15722A00	89675E88
t_{69}	1825D61B	A1C43505	BE73A278	633066D1	15722A00
t_{70}	16B0A0DB	1825D61B	68710D41	BE73A278	633066D1
t_{71}	D07B8F33	16B0A0DB	C6097586	68710D41	BE73A278
t_{72}	B1CE0FDA	D07B8F33	C5AC2836	C6097586	68710D41
t_{73}	A5B3147A	B1CE0FDA	F406E3CC	C5AC2836	C6097586
t_{74}	B14EF2AD	A5B3147A	AC7303F6	F406E3CC	C5AC2836
t_{75}	7E6D1A02	B14EF2AD	A96CC51E	AC7303F6	F406E3CC
t_{76}	BC649E10	7E6D1A02	6C53BCAB	A96CC51E	AC7303F6
t_{77}	89340CAA	BC649E10	9F9B4680	6C53BCAB	A96CC51E
t_{78}	71BC30F0	89340CAA	2F192784	9F9B4680	6C53BCAB
t_{79}	32DE40D2	71BC30F0	A24D032A	2F192784	9F9B4680

Selanjutnya setelah didapatkan a, b, c, d, dan e untuk t_{79}, maka nilai t_{79} digunakan untuk mendapatkan *digest* dalam SHA-1 dengan cara di XOR dengan nilai a, b, c, d, dan e awal (penyangga).

$H_i = a_0 \oplus a_n$

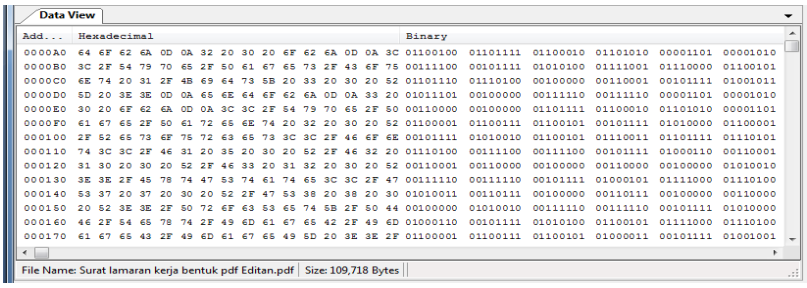
H_0 = 67452301 $\oplus$ 32DE40D2

01100111 01000101 00100011 00000001
= 00110010 11011110 01000000 11010010
01010101 10011011 01100011 11010011

= 559B63D3

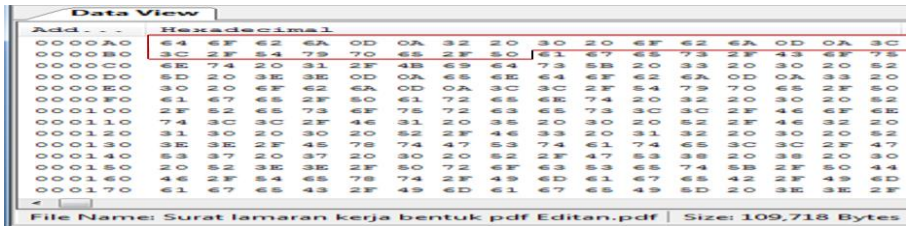
Concate h0, h1, h2, h3, dan h4. Hasil *concate* sebanyak 20 Byte 40 karakter tersebut disebut *message digest* :
559B63D3 9E719B79 B129148D 3F2B73F1 5C49A770

Contoh kasus Editan dalam proses ini yaitu *file dokumen* (Surat Lamaran Kerja Bentuk Pdf). Data yang di ambil sebanyak 24 *byte* untuk *plainteks*, cara pengambilan nilai *hexadecimal file dokumen* menggunakan aplikasi *Binary Viewer*, seperti dibawah ini.



Gambar 4. Data *file* dokumen *Editan*

Dari data tersebut diambil sebanyak 24 *byte* atau 48 karakter heksadesimal dan dikonversi ke biner sebagai sampel metode, yang berguna untuk mengetahui nilai biner dari bilangan tersebut.



Gambar 5. Data Byte Dokumen pdf

Tabel 6. Lima penyangga a, b, c, d, dan

A	B	c	D	E
67452301	EFCDAB89	98BADCFE	10325476	C3D2E1F0

Tabel 7. Hasil t_0 berisi a_0, b_0, c_0, d_0, dan e_0

A_0	B_0	C_0	D_0	E_0
9D3312A7	67452301	7BF36AE2	98BADCFE	10325476

Tabel 8. Lima penyangga a, b, c, d, dan e

A	B	C	D	D
9D3312A7	67452301	7BF36AE2	98BADCFE	10325476

Dikerjakan terlebih dahulu $f_t(b,c,d)$ dengan fungsi logika AND ($\wedge$), NAND (~), OR ($\vee$), $f_t(b \wedge c) \vee (\sim b \wedge d)$

Tabel 9 Hasil t_1 berisi a_1, b_1, c_1, d_1, dan e_1

A_1	B_1	C_1	D_1	E_1
0291ED20	9D3312A7	59D148C0	7BF36AE2	98BADCFE

t_1 = 0291ED20 9D3312A7 59D148C0 7BF36AE2 98BADCFE
Proses untuk t_1 hingga t_{79} sangat panjang, disini langsung dibuat hasil t keseluruhan

Tabel 10. Hasil a, b, c, d, dan e untuk t_0 hingga t_{79}

Round	A	B	C	D	E
t_0	9D3312A7	67452301	7BF36AE2	98BADCFE	10325476
t_1	0291ED20	9D3312A7	59D148C0	7BF36AE2	98BADCFE
t_2	820BC93E	0291ED20	E74CC4A9	59D148C0	7BF36AE2
t_3	10B6A0E1	820BC93E	00A47B48	E74CC4A9	59D148C0
t_4	EEE670E1	10B6A0E1	B279725B	00A47B48	E74CC4A9
t_5	D0AA5F40	EEE670E1	9FE6FAF6	B279725B	00A47B48
t_6	DD14DA21	D0AA5F40	A7F9BEBD	92521856	B279725B
t_7	A76D1462	DD14DA21	96473A34	A7F9BEBD	92521856
t_8	7370D886	A76D1462	74536887	96473A34	A7F9BEBD
t_9	C408B0B1	7370D886	9DB4518A	74536887	96473A34
t_{10}	92FADB53	C408B0B1	ADCE7A33	9DB4518A	74536887
t_{11}	D0AA5F40	92FADB53	64810A2D	ADCE7A33	9DB4518A
t_{12}	DD14DA21	D0AA5F40	92FADB53	64810A2D	ADCE7A33
t_{13}	96473A34	DD14DA21	64810A2D	92FADB53	64810A2D
t_{14}	FDE75E81	96473A34	56527838	64810A2D	D46E755B
t_{15}	E5A5728B	FDE75E81	1022C2C7	56527838	64810A2D
t_{16}	A4948615	E5A5728B	ADCE7A33	1022C2C7	565278380
t_{17}	3EADD61F	A4948615	370D886	ADCE7A33	1022C2C7
t_{18}	5920428B	3EADD61F	5133251A	370D886	ADCE7A33
t_{19}	CDC36219	5920428B	15949E0E	5133251A	370D886
t_{20}	2591CE8D	CDC36219	9DB4518A	15949E0E	5133251A
t_{21}	FAB7587C	2591CE8D	A1B1072C	9DB4518A	15949E0E

t_{22}	25A3A732	FAB7587C	F613ACF3	A1B1072C	9DB4518A
t_{23}	1022C2C7	25A3A732	A76D1462	F613ACF3	A1B1072C
t_{24}	9695CA2F	1022C2C7	26DFC15C	A76D1462	F613ACF3
t_{25}	A1B1072C	9695CA2F	A4948615	26DFC15C	A76D1462
t_{26}	86C41CB2	A1B1072C	E5A5728B	A4948615	26DFC15C
t_{27}	26DFC15C	86C41CB2	77C2C7CA	E5A5728B	A4948615
t_{28}	35AC3CF1	26DFC15C	B0E770E9	77C2C7CA	E5A5728B
t_{29}	B0E770E9	35AC3CF1	323737BE	B0E770E9	77C2C7CA
t_{30}	629F3A2E	B0E770E9	C268985E	323737BE	B0E770E9
t_{31}	8968E9CC	629F3A2E	C9AD01CB	C268985E	323737BE
t_{32}	8968E9CC	8968E9CC	B150E233	C9AD01CB	C268985E
t_{33}	3D8349C9	8968E9CC	FD2427F6	B150E233	C9AD01CB
t_{34}	D0AB57EA	3D8349C9	D0AB57EA	FD2427F6	B150E233
t_{35}	EC54388C	D0AB57EA	3D8349C9	D0AB57EA	FD2427F6
t_{36}	B09A2617	EC54388C	1413EB15	3D8349C9	D0AB57EA
t_{37}	35AC3CF1	B09A2617	C268985E	1413EB15	3D8349C9
t_{38}	F26B4072	35AC3CF1	A72AE1AA	C268985E	1413EB15
t_{39}	B5646936	F26B4072	07443189	A72AE1AA	C268985E
t_{40}	5A664B28	B5646936	191C9321	07443189	A72AE1AA
t_{41}	AFF8A297	5A664B28	5A664B28	191C9321	07443189
t_{42}	8370B828	AFF8A297	B50780E0	5A664B28	191C9321
t_{43}	2AFB17F2	8370B828	672C9DED	B50780E0	5A664B28
t_{44}	B1289201	2AFB17F2	7AF072AB	672C9DED	B50780E0
t_{45}	1210AAF9	B1289201	A9CAB86A	7AF072AB	672C9DED
t_{46}	1227BA07	1210AAF9	191C9321	A9CAB86A	7AF072AB
t_{47}	3B8A8233	1227BA07	C39DC3A6	191C9321	A9CAB86A
t_{48}	3B8A8233	3B8A8233	0118CCC2	C39DC3A6	191C9321
t_{49}	0A28FFF0	3B8A8233	B9D96D18	0118CCC2	C39DC3A6
t_{50}	2208B211	0A28FFF0	27124362	B9D96D18	0118CCC2
t_{51}	21792F2A	2208B211	323737BE	27124362	B9D96D18
t_{52}	22BBB02B	21792F2A	64674FB3	323737BE	27124362
t_{53}	0AF19928	22BBB02B	C8DCDEF8	64674FB3	323737BE
t_{54}	22239738	0AF19928	55C8A800	C8DCDEF8	64674FB3
t_{55}	322AF2BF	22239738	85F81547	55C8A800	C8DCDEF8
t_{56}	82872828	322AF2BF	D3A3B1A8	85F81547	55C8A800
t_{57}	97F00220	82872828	1825D61B	D3A3B1A8	85F81547
t_{58}	7AF93228	97F00220	323737BE	1825D61B	D3A3B1A8
t_{59}	AF1B1FF8	7AF93228	D5217E05	323737BE	1825D61B
t_{60}	DDD8B26E	AF1B1FF8	7F4AE100	D5217E05	323737BE
t_{61}	33919E45	DDD8B26E	DEF7C093	7F4AE100	D5217E05
t_{62}	A082AF09	33919E45	129F9D95	DEF7C093	7F4AE100
t_{63}	12899992	A082AF09	D07B8F33	129F9D95	DEF7C093
t_{64}	B8F2022F	12899992	F01B8A22	D07B8F33	129F9D95
t_{65}	BF839337	B8F2022F	32DE40D2	F01B8A22	D07B8F33
t_{66}	12073128	BF839337	15722A00	32DE40D2	F01B8A22
t_{67}	2A12B228	12073128	F406E3CC	15722A00	32DE40D2
t_{68}	38A7F7E2	2A12B228	4EAF2271	F406E3CC	15722A00
t_{69}	6C53BCAB	38A7F7E2	9F9B4680	4EAF2271	F406E3CC
t_{70}	F8FBA0B2	6C53BCAB	2F192784	9F9B4680	4EAF2271
t_{71}	AFB28422	F8FBA0B2	C5AC2836	2F192784	9F9B4680
t_{72}	792A102B	AFB28422	8990A37E	C5AC2836	2F192784
t_{73}	09278992	792A102B	098F7272	8990A37E	C5AC2836
t_{74}	2B2129AF	09278992	3AF22AFF	098F7272	8990A37E
t_{75}	023282BA	2B2129AF	1928F7F2	3AF22AFF	098F7272
t_{76}	2132AF27	023282BA	787EA22B	1928F7F2	3AF22AFF
t_{77}	668CD7ED	2132AF27	E885562B	787EA22B	1928F7F2
t_{78}	79860772	668CD7ED	A3E7767D	E885562B	787EA22B

t_{79}	17421783	79860772	DFB865FA	A3E7767D	E885562B

Selanjutnya setelah didapatkan a, b, c, d, dan e untuk t_{79}, maka nilai t_{79} digunakan untuk mendapatkan *digest* dalam SHA-1 dengan cara di XOR dengan nilai a, b, c, d, dan e awal (penyangga).

3.2 Hasil Pengujian

Dengan menggunakan aplikasi *Hasher Pro* pada pengujian implementasi metode SHA-1 untuk mendeteksi file dokumen maka didapat sebuah hasil berikut ini yang merupakan data hasil perbandingan mendeteksi *file* dokumen yang diperoleh dari data *file* dokumen Asli dan *file* dokumen editan pada table 4.1 di bawah ini:

Tabel 11. Hasil Perbaikan Citra

No	*File* Dokumen Asli	SHA-1	*File* Dokumen Editan	SHA-1	Kesimpulan
1.	Lamaran Kerja.Pdf Size 98 KB	559B63D3 9E719B79 B129148D 3F2B73F1 5C49A770	Lamaran Kerja .Pdf Size 108 KB	71873482 964BACFB 4702C904 B3D52208 B3D5220	Dari Hasil Perbandingan metadata *file* dokumen Asli dan editan dinyatakan berbeda berdasarkan kode dari metode yang di dapatkan
2.	Kartu Keluarga.Pdf Size 182 KB	9aad71f3531 6b2fe142aed 1 1df46c46ec9 dc4274	Kartu Keluarga. Pdf Size 141 KB	8c8f7b3c8a68e 423eb09bb2ab5 73e1e533	Dari Hasil Perbandingan metadata *file* dokumen Asli dan editan dinyatakan sama atau asli berdasarkan kode dari metode yang di dapatkan
3.	Izazah.Pdf Size 179 KB	cf932582768 eb7b61bcca5 7241b7c71ae cbd789c	Izazah.Pdf Size 135 KB	284a34601fa8b b65aa6d80ef57 22af2c1f	Dari hasil perbandingan meta data *file* dokumen asli dan editan dinyatakan berbeda berdasarkan kode dari metode yang didapatkan
4	Foto.Pdf Size 56 KB	6397153d08 2ec1015bda5 b0433a29818 3c	Foto.Pdf Size 147 KB	12e8c91e6d4f0 457efcc4a6f5f0 a577727fb08d8	Dari hasil perbandingan meta data *file* dokumen asli dan editan dinyatakan berbeda berdasarkan kode dari metode yang didapatkan
5.	Daftar Riwayat hidup .Pdf Size 60 KB	c45091afa9d 517d643bf39 0fa34ae5636 cbe189f	Daftar Riwayat Hidup.Pdf Size 36 KB	2edf18a68c58d a9a2934f9cc5ef 301eff6	Dari hasil perbandingan meta data *file* dokumen asli dan editan dinyatakan berbeda berdasarkan kode dari metode yang didapatkan

Berdasarkan data hasil pengujian mendeteksi keaslian *file* dokumen menggunakan metode SHA-1 menunjukkan bahwa perubahan sekecil apapun sangat mempengaruhi hasil dari pendeteksi atau keaslian dari *file* tersebut sehingga tingkat akurat dari perbedaan *file* dokumen asli dan yang telah di rubah/diedit sangat besar perbedaanya

4. KESIMPULAN

Kesimpulan yang didapat dari penelitian bahwa algoritma SHA-1 dapat mengamankan atau menjaga keaslian *file* dokumen dengan aman dari pihak lain yang tidak berkepentingan dapat diketahui perubahan yang terjadi melalui kode *hash* yang dihasilkan. Pengujian aplikasi dengan menggunakan *Hasher Pro* yang telah selesai diuji dengan desain minimalis diharapkan dapat berguna dalam mendeteksi keaslian *file* dokumen. Mengetahui hasil pengujian dari metode SHA-1 dan aplikasi *Hasher Pro* dalam mendeteksi keaslian *file* dokumen.

REFERENCES

[1] K. Aryasa and Y. T. Paulus, "Implementasi Secure Hash Algorithm-1 Untuk Pengamanan Data Dalam Library Pada Pemrograman Java," *Creat. Inf. Technol. J.*, vol. 1, no. 1, p. 57, 2018.
[2] Doroty L, Satoto K I, Nurhayati O D, PERANCANGAN IMPLEMENTASI SISTEM INFORMASI PERPUSTAKAAN DIPROGRAM STUDI TEKNIK LINGKUNGAN FAKULTAS TEKNIK UNDIP Vol.2, No.4, Oktober 2014 (e-ISSN:2338-0403)
[3] M. K. Emy Setyaningsih, S.Si., *Kriptografi & Implementasinya menggunakan MATLAB*. Yogyakarta: ANDI, 2015.
[4] Rifki Sadikin, *Kriptografi untuk keamanan jaringan dan implementasi dalam Bahasa Java*. Yogyakarta: ANDI, 2012.
[5] Dony Ariyus, *PENGANTAR ILMU KRIPTOGRAFI Teori Analisis & Implementasi*. Yogyakarta, 2008.
[6] Dimaz A Wijaya, *MENGENAL BITCOIN & CRYPTOCURRENCY*. Medan, 2016.
[7] 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, p. 61, 2018.
[8] Sugiartowo, Ambo S N, "Implementasi Simulasi Media Pembelajaran Rangkaian Kombinasional Berbasis Kolaburasi Multimedia Simulator Dan Pemograman Delphi,"Junal Informatika Upgris, Vol. 4, No. 2, (2018) P/E-ISSN: 2460-4801/2447-6645 170
[9] Pabokory F.N Astuti I.F Kridalaksana A.H IMPLEMENTASI KRIPTOGRAFI PENGAMANAN DATA PADA PESAN TEKS, ISI FILE DOKUMEN, DAN FILE DOKUMEN MENGGUNAKAN ALGORITMA ADVANCED ENCRYPTION STANDARD. Vol. 10 No. 1 Februari 2015
[10] Crystanti C.Y, Wardati I.U, Sistem Pengolahan Data Simpan Khusus Perempuan (SPP) Pada Unit Pengelola Kegiatan (UPK) Mintra Usaha Mandiri Program Nasional Pemberdayan Masyarakat Mandiri Perdesaan PNPM-MPD Kecamatan Pringkuku Kabupaten Pacitan. *Journal Speed – Sentra Penelitian Engineering dan Edukasi – Volume 3 No 1 - 2011 - ijns.org*
[11] http://www.den4b.com/products/hasher/