Paper

Implementasi Algoritma SHA-3 Dan ElGamal Untuk Otentikasi Piagam Penghargaan Berbasis Digital Signature

Author: Annisa Apriliani, Nelly Astuti Hasibuan, Dito Putro Utomo

SNASTIKOM Ke 9 Oktober Tahun 2022
ISNN: 2964-1950

Implementasi Algoritma SHA-3 Dan ElGamal Untuk Otentikasi Piagam Penghargaan Berbasis Digital Signature

Annisa Apriliani[1], Nelly Astuti Hasibuan [2], Dito Putro Utomo [3*]
[1,2,3*]Universitas Budi Darma, Medan, Indonesia
[1]aapriliani819@email.com, [2]nelly.ahsb@email.com, [3*]ditoputro12@gmail.com

Abstrak

Perkembangan teknologi yang pesat saat ini sangat memungkinkan terjadinya berbagai pemalsuan dan penipuan yang dilakukan untuk mengambil keuntungan secara sepihak. Salah satunya adalah pemalsuan sebuah piagam penghargaan yang dapat dengan mudah dilakukan oleh pihak-pihak yang tidak bertanggung jawab. Piagam penghargaan adalah sebuah *file* penting yang membuktikan prestasi dari seseorang dalam sebuah kegiatan atau perlombaan tertentu. Hal inilah yang menimbulkan banyak sekali orang-orang yang berusaha untuk memalsukan piagam penghargaan untuk kepentingan pribadi baik pekerjaan maupun pendidikan. Oleh karena itu diperlukan sebuah sistem untuk menguji keaslian dari piagam penghargaan tersebut. Dengan serangkaian langkah pengujian yang dilakukan maka dapat disimpulkan apakah piagam penghargaan tersebut otentik atau tidak. Untuk menghindari adanya pemalsuan *file* dokumen maka perlu diterapkan sebuah *digital signature* dengan metode kriptografi. *Digital signature* atau lebih dikenal sebagai tanda tangan digital dapat membantu menguji otentikasi dari sebuah dokumen khusunya piagam penghargaan. Ketika *digital signature* telah diterapkan pada piagam penghargaan, maka penerima piagam dapat mengecek apakah *file* tersebut otentik atau telah mengalami modifikasi. Penerapan *digital signature* melalui 2 tahapan yaitu proses *signing* dan *verification*. *Signing* adalah tahapan pembentukan *digital signature* dengan cara mengombinasikan algoritma SHA-3 dan ElGamal. Keotentikan dari *file* dapat dilihat dengan melakukan proses *verification* yaitu proses membandingkan nilai *hash* dari *file* yang diterima dengan nilai *hash file* yang sebenarnya. Penelitian ini menghasilkan sebuah analisa terkait proses otentikasi dengan menggunakan digital signature.

Kata Kunci: Kriptografi, Digital Signature, Otentikasi, SHA-3, ElGamal

Abstract

The rapid development of technology today is very possible for the occurrence of various forgeries and frauds that are carried out to take profits unilaterally. One of them is the falsification of a certificate of appreciation that can be easily carried out by irresponsible parties. An award certificate is an important file that proves the achievements of a person in a particular activity or competition. This is what gives rise to a lot of people who try to forge certificates of appreciation for their personal interests, both work and education. Therefore, a system is needed to test the authenticity of the award charter. With a series of testing steps carried out, it can be concluded whether the award certificate is authentic or not. To avoid falsification of document files, it is necessary to apply a digital signature with a cryptographic method. Digital signatures or better known as digital signatures can help test the authentication of a document, especially a certificate of appreciation. When the digital signature has been applied to the award certificate, the award recipient can check whether the file is authentic or has been modified. The application of digital signatures goes through 2 stages, namely the signing and verification process. Signing is the stage of forming a digital signature by combining the SHA-3 and ElGamal algorithms. The authenticity of the file can be seen by carrying out the verification process, which is the process of comparing the hash value of the received file with the actual file hash value. This research produces an analyzation about process of authentication using digital signature.

Keywords: *Cryptography, Digital Signature, Authentication, SHA-3, ElGamal*

1. PENDAHULUAN

Prestasi akademik maupun non-akademik saat ini dianggap sebagai tolak ukur kemampuan dan gambaran potensi seseorang. Prestasi tersebut dapat disimbolkan melalui sebuah piagam penghargaan, sehingga piagam penghargaan dianggap sebagai bukti penting yang menunjukkan kemampuan seseorang maupun keaktifannya dalam sebuah ajang perlombaan. Tidak jarang piagam penghargaan digunakan sebagai berkas pendukung yang membantu seseorang untuk menempuh pendidikan yang lebih tinggi atau bahkan dalam proses rekrutmen pegawai atau melamar pekerjaan. Oleh karena itu banyak sekali orang-orang yang menjadikan piagam penghargaan sebagai alasan utama untuk ikut serta dalam sebuah perlombaan.

Untuk memperoleh piagam penghargaan tentu saja tidak mudah dan hanya bisa diperoleh jika seseorang memenangkan sebuah perlombaan atau meraih posisi tertentu. Hal tersebut memerlukan usaha dan latihan yang tidak mudah. Oleh karena itu, banyak sekali terjadi kasus pemalsuan dokumen khususnya piagam penghargaan. Ini merupakan salah satu dampak negatif dari perkembangan teknologi dimana seseorang dapat dengan mudah memanipulasi suatu dokumen demi kepentingannya sendiri. Memanipulasi dokumen merupakan tindakan illegal yang sangat merugikan sehingga hal ini sangat dilarang dan memiliki landasan hukum yang kuat. Meskipun begitu, untuk mengetahui apakah suatu dokumen khususnya piagam penghargaan tersebut asli atau tidak diperlukan sebuah sistem yang mumpuni. Hal ini dikarenakan piagam penghargaan asli dan palsu tidak dapat dibedakan dengan kasat mata. Terutama jika yang melakukan manipulasi adalah orang yang handal.

Keaslian dari suatu *file* dapat diketahui dengan menerapkan *digital signature* atau yang dikenal sebagai tanda tangan digital. *Digital signature* merupakan salah satu konsep yang dikembangkan dalam ilmu kriptografi modern yang berfungsi untuk memeriksa keotentikan dari suatu *file* dan bersifat anti penyangkalan [1]. Kriptografi merupakan teknik pengamanan pesan dengan mengubah isi dari *file* kedalam bentuk simbol-simbol yang tidak mudah dipahami oleh orang lain [2]. Salah satu fungsi kriptografi adalah menjaga isi pesan serta menjamin keotentikannya sehingga penerima yakin bahwa pesan tersebut tidak mengalami perubahan dalam proses pengiriman. Dalam proses pembuatan *digital signature* diterapkan algoritma kriptografi yang terdiri dari metode SHA-3 dan algoritma ElGamal yang termasuk kedalam algoritma kriptografi asimetris untuk meningkatkan keakuratan dalam proses identifikasi piagam. Algoritma SHA-3 adalah salah satu jenis algoritma SHA, dimana ia memiliki kinerja yang lebih baik daripada pendahulunya karena memiliki ketahanan yang lebih baik dalam menghadapi serangan-serangan luar [3]. Algoritma ElGamal adalah algoritma kunci asimetris yang cara kerjanya didasarkan pada perhitungan logaritma diskrit serta menerapkan 2 kunci berbeda pada proses enkripsi dan deskripsinya [4]. Penerima dapat melakukan proses verifikasi untuk menguji keotentikan *file* dengan membandingkan *message digest* dari *file* yang diterima dengan *message digest* sebenarnya dari *file* piagam penghargaan, jika *message digest* (nilai *hash*) dari *file* yang diterima sama dengan *message digest* (nilai *hash*) sesungguhnya, maka dapat disimpulkan bahwa *file* tersebut bersifat otentik.

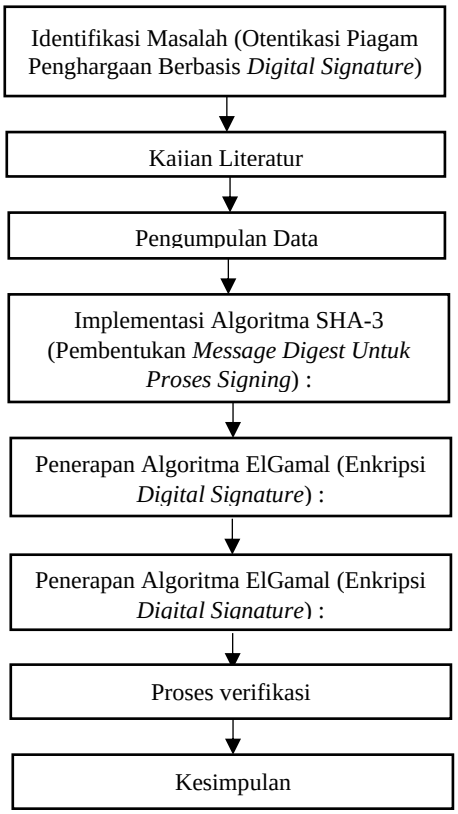
Penelitian terdahulu terkait *digital signature* adalah penelitian oleh Egi Cahyo Prabowo, dkk pada tahun 2017 yang berjudul Penerapan Digital Signature Dan Kriptografi Pada Otentikasi Sertifikat Tanah Digital. Penelitian ini menghasilkan sebuah layanan otentikasi dokumen berupa sertifikat tanah [1]. Penelitian oleh Antika Lorien, dkk pada tahun 2019 yang berjudul Implementasi Sistem Otentikasi Dokumen Berbasis Quick Response (QR) Code dan Digital Signature. Penelitian ini menghasilkan sebuah sistem otentikasi dokumen berupa sertifikat dengan menerapkan kombinasi antara QR Code dengan digital signature [5]. Penelitian oleh Sugiyatno pada tahun 2018 yang berjudul Digital Signature Dengan Algoritma Sha-1 Dan Rsa Sebagai Autentikasi. Penelitian ini menghasilkan sebuah aplikasi otentikasi dokumen berupa SKPI (Surat Keterangan Pendamping Ijazah) dengan menerapkan digital signature [6]. Penelitian oleh Budi Kurniawan Hutasuhut pada tahun 2019 yang berjudul Analisis Rancangan Model Digital Signature Dengan Kombinasi Algoritma MD5, Algoritma Elgamal, dan Algoritma RSA Untuk Menguji Keaslian Data Dengan Akurat. Penelitian ini menghasilkan kesimpulan bahwa skema yang dirancang dengan mengombinasikan algoritma MD5, ElGamal, dan RSA lebih unggul dengan tingkat keamanan yang sangat tinggi [7]. Penelitian oleh Morita Puspita Sari pada tahun 2021 yang berjudul Analisis Algoritma SHA-3 Keamanan Pada Data Pribadi. Penelitian ini menghasilkan sebuah sistem keamanan yang dapat diterapkan untuk melindungi data pribadi serta analisis terhadap kinerja algoritma SHA-3 [3]. Penelitian oleh Nonik Indahwati, dkk pada tahun 2019 yang berjudul Penerapan Algoritma Kriptografi Asimetris Elgamal dengan Modifikasi Pembangkit Kunci terhadap Enkripsi dan Deskripsi Gambar warna. Penelitian ini menghasilkan sebuah sistem enkripsi dan deskripsi algoritma ElGamal dengan kunci yang telah dimodifikasi [8].

Berdasarkan keterangan diatas, maka penulis akan melakukan penelitian yang berjudul "IMPLEMENTASI ALGORITMA SHA-3 DAN ELGAMAL UNTUK OTENTIKASI PIAGAM PENGHARGAAN BERBASIS DIGITAL SIGNATURE". Penelitian ini akan menghasilkan sebuah analisa terkait proses otentikasi khusunya piagam penghargaan dengan menerapkan *digital signature*.

2. METODE PENELITIAN

2.1 Tahapan Penelitian

Dalam melaksanakan penelitian, penulis menyiapkan sebuah kerangka kerja penelitian yang berisi langkah-langkah pelaksanaan kegiatan guna mempermudah proses penelitian yang berlangsung agar lebih terarah dan sistematis. Adapun kerangka kerja penelitian yang dilakukan dapat dilihat pada gambar dibawah ini :



Gambar 1. Tahapan Penelitian

2.2 Kajian Pustaka

2.2.1 Kriptografi

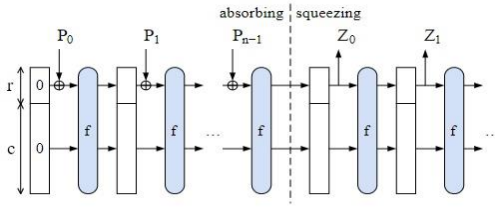
Kriptografi merupakan teknik mengubah pesan rahasia kedalam bentuk simbol acak yang disebut sebagai tahap enkripsi sesuai dengan kunci serta metode yang digunakan dan jika penerima ingin membuka pesan tersebut maka diterapkan sebuah teknik deskripsi [9].

2.2.2 *Digital Signature*

Digital signature (tanda tangan digital) adalah salah satu skema kriptografi yang tidak hanya berfokus kepada keamanan pesan tetapi juga mampu mengidentifikasi keaslian atau keotentikan dari pesan yang dikirim serta bersifat anti penyangkalan (*non-repudiation*). *Digital signature* berfungsi layaknya sebuah tanda tangan yang menunjukkan keabsahan suatu file [10].

2.2.3 Algoritma SHA-3

Algoritma SHA-3 merupakan varian terbaru dari algoritma SHA yang dirancang layaknya sebuah konstruksi spons yang diawali dengan *absorbing* dan hasilnya akan *squeezing* layaknya seperti sebuah spons. Skema konstruksi spons pada algoritma SHA-3 dapat dilihat pada gambar dibawah ini [11]:



Gambar 2. Konstruksi Spons SHA-3

Adapun tahapan penerapan algoritma SHA-3 adalah sebagai berikut [11]:
a. Menentukan panjang *digest* (*d*) bit yang ingin dibentuk.

b. Melakukan *padding* terhadap pesan *M* kedalam bentuk string *P* hingga nilai *P* habis dibagi *r* atau *n*. *Padding* pada SHA-3 mengikuti pola 10*1, pada SHA-3 dengan tipe SHAKE128 dapat ditulis :
SHAKE128(*M*,*d*) = Keccak[*c*] (*M* || 1111,*d*) (1)
c. *P* dibagi menjadi beberapa blok P_i dengan ukuran *r*-bit.
d. *b*-bit dari *state S* diinisialisasi menjadi 0.
e. Fase *absorbing* (penyerapan), XOR-kan setiap blok P_i dengan *r*-bit pertama yang berada dalam *state S*, hasil dari proses XOR dimasukkan pada fungsi permutasi (*f*) sehingga menghasilkan sebuah *state* baru *S*. Fungsi permutasi melalui 5 tahapan yaitu theta, rho, phi, chi, iota yang dilakukan hingga 23 putaran untuk menghasilkan state baru.
f. Fase *squeezing* (pemerasan).
$Z = Z \| Trunc_r(S)$ (2)
Jika *output* yang diinginkan (*d*) ≤ |*Z*|, maka $Trunc_d$

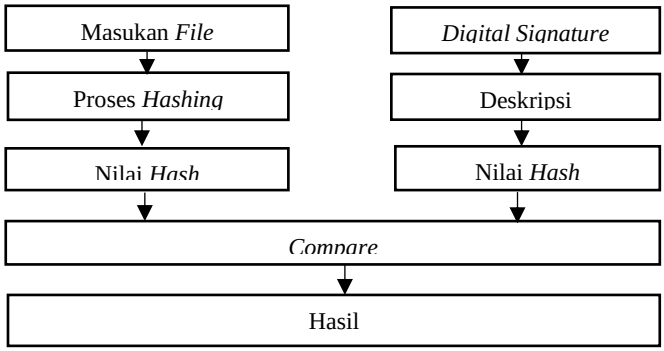
2.2.4 Algoritma ElGamal

Algoritma ElGamal adalah salah satu algoritma kunci asimetris yang cukup rumit dengan menggunakan perhitungan matematika berupa logaritma diskrit. Kunci yang digunakan pada algoritma ini terdiri atas sebuah bilangan prima dan 2 bilangan lainnya yang dipilih secara acak (*random*). Untuk kunci privat maka dibutuhkan 2 buah bilangan (*x*,*p*) sedangkan kunci publik membutuhkan 3 buah bilangan (*p*,*y*,*x*). Pemilihan bilangan untuk kunci yang digunakan harus memenuhi syarat utama yaitu *p* > *g* dan *p* > *x* serta memenuhi persamaan [8]:
$y = g^x \bmod p$ (3)
Ketentuan :
g, *x* = bilangan *random*
p = bilangan prima
Proses enkripsi algoritma elgamal menggunakan bilangan acak *k* dengan syarat 1≤ *k* ≤ *p*-2 serta harus memenuhi persamaan dibawah ini [4]:
$a = g^k \bmod p$ (4)
$b = y^k\, m \bmod p$ (5)
Setelah melalui tahap enkripsi, maka dihasilkan sebuah *ciphertext* yang terdiri atas simbol-simbol yang sulit dibaca. Untuk membaca pesan dilakukan proses deskripsi terlebih dahulu. Adapun persamaan yang digunakan pada proses deskripsi yaitu [8]:
$(ax)^{-1} = a^{p-1-x} \bmod p$ (6)
$m = b^* a^x \bmod p$ (7)

3. HASIL DAN PEMBAHASAN

3.1 Analisa Proses Otentikasi Berbasis *Digital Signature*

Proses otentikasi *file* piagam penghargaan menggunakan *digital signature* yang merupakan hasil dari proses kriptografi. Perlu diketahui bahwa *digital signature* berbeda dengan tanda tangan elektronik. Tanda tangan digital bersifat *eligible* dan lebih diakui keorisinalitasannya dibandingkan tanda tangan elektronik. Otentikasi suatu *file* umumnya terbagi 2 yaitu otentikasi pengirim dan otentikasi isi *file*. Sehingga penerapan *digital signature* harus dapat memastikan bahwa *file* yang dikirim adalah asli dari pihak pengirim dan isi *file* tidak mengalami perubahan. Adapun analisa proses otentikasi piagam penghargaan berbasis *digital signature* dapat dilihat pada gambar dibawah ini :



Gambar 3. Analisa Proses Otentikasi Berbasis *Digital Signature*

Proses otentikasi piagam penghargaan dengan algoritma SHA-3 dan ElGamal melalui tahapan *verification*. *Verification* adalah tahapan pemeriksaan keotentikan *file* dengan membandingkan nilai *hash* yang diperoleh dari tahapan *hashing file* dengan hasil deskripsi *digital signature* yang telah diterima oleh *user*. Untuk melakukan deskripsi terhadap *digital signature*, maka penerima *file* harus mendapatkan kunci privat dari penandatangan. Berdasarkan kerahasiaan kunci privat tersebut, pengirim tidak dapat menyangkal bahwa dialah yang telah mengirim *file* piagam penghargaan, sehingga *file* piagam penghargaan memiliki sifat anti penyangkalan. Apabila nilai *hash* dan deskripsi sama, maka dapat disimpulkan bahwa *file* tersebut otentik.

3.2 Implementasi Algoritma SHA-3 dan ElGamal Berbasis Digital Signature

Terdapat 2 metode yang diterapkan dalam proses otentikasi *file* piagam penghargaan yaitu metode SHA-3 yang digunakan dalam proses *hashing* dan metode ElGamal yang digunakan untuk mengamankan nilai *hash* yang telah diperoleh. Sampel piagam yang digunakan dapat dilihat pada gambar 5 berikut:

Gambar 4. Sampel Piagam Penghargaan

Piagam tersebut akan diambil nilai binernya sebanyak 128 bit yang dapat dilihat pada tabel 1 berikut:

Tabel 1. Sampel *File*

Nilai Heksadesimal	Nilai Biner
42 69 74 73 50 65 72 43 6F 6D 70 6F 6E 65 6E 74	01000010 01101001 01110100 01110011 01010000 01100101 01110010 01000011 01101111 01101101 01110000 01101111 01101110 01100101 01101110 01110100

3.2.1 Implementasi Algoritma SHA-3

Pada penelitian ini, digunakan algoritma SHA-3 dengan tipe SHAKE128. Dengan SHAKE128, maka panjang *output* yang dihasilkan dapat ditentukan oleh *user*. Adapun tahapan-tahapan implementasi algoritma SHA-3 pada *file* yang telah diterima adalah sebagai berikut :

1. Penentuan panjang *message digest* (*d*) yaitu 128 bit.
2. *Padding*
   Padding pada masukan *P* dilakukan dengan menambahkan 10*1 yang artinya bit 1 diikuti oleh satu bit 0 atau bahkan lebih dengan ketentuan jumlah bit = *r*-1 dan diakhiri dengan bit 1, hingga nilai *padding* habis dibagi *r*. Untuk SHAKE128, penambahan *padding* dapat mengikuti persamaan SHAKE128(*M*,*d*) = *Keccak*[256] 128 || 1111,128). Dengan nilai *r* = 1344 dan *c* = 256. maka penambahan *padding* adalah 1344-132 = 1212, terdiri atas 2 buah bit 1 dan 1200 bit 0.
   P' = 01000010 01101001 01110100 01110011 01010000 01100101 01110010 01000011 01101111 01101101 01110000 01101111 01101110 01100101 01101110 01110100 1111 $\underbrace{1\ldots\ldots\ldots\ldots1}_{\text{Bit 0 = 1200}}$
3. Pembagian blok (P_i)
   Blok P_i dibagi hingga P_n-1, dimana *i* adalah indeks putaran yang dimulai dari 0 dan *n* adalah jumlah putaran yang berlangsung yaitu sebanyak 24 putaran. Maka blok *P* ialah :
   $P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, P_{12}, P_{13}, P_{14}, P_{15}, P_{16}, P_{17}, P_{18}, P_{19}, P_{20}, P_{21}, P_{22}, P_{23}$
4. Inisialisasi *state* (S) ke string dengan *b*-bit *state* menjadi 0
   State pada algoritma SHA-3 terdiri atas matriks 5 x 5 x 2^6 = 1600 bit yang digambarkan dengan sumbu *x*,*y*,*z*. Setiap *state* menyimpan bilangan biner dari *block size* (*r*) dan *capacity* (*c*). Setelah melakukan inisialisasi *state*

dengan bit 0 sebanyak 1600 bit, maka ditentukan string bit yang akan di *absorb* berdasarkan masukan yang telah dikonversi kedalam biner dan telah melalui proses *padding* untuk memenuhi nilai *b* = 1600 bit. Lakukan proses xor antara *state* (*S*) bit 0 dan masukan yang akan di *absorb*.

5. *Absorb* (Penyerapan)
   Pada tahap penyerapan terjadi permutasi blok *f* terhadap *b*-bit yang terdiri atas 12 + 2*l* putaran, nilai *l* yang digunakan adalah 6, maka 12+2(6) = 24 putaran. Operasi yang dilakukan meliputi XOR, AND, ROT yang terdiri atas 5 proses yaitu theta(θ), rho(ρ), pi(π), chi(χ), iota(ι).
6. *Squeeze* (Pemerasan)
   Pada tahap ini, *state* yang diperoleh pada proses *absorb* akan di *squeeze* hingga menghasilkan *output d*, dimana *d* = 128, dan $Z = Trunc_d$, dimana *Z* jumlah keluaran total dari proses *absorb*. Bit string total pada *state* akan dipotong atau dibagi menjadi nilai *d*.
   $Z = Trunc_s$ = 11000011 01001100 00011111 11110000 10011100 00011101 00001001 01101000 11001110 01111011 11111110 00011100 00000111 11001111 00001010 11101101.
   Output yang diperoleh pada algoritma SHA-3 kemudian dikonversi kedalam heksadesimal yang dapat dilihat pada table 2 berikut :

Tabel 2. *Output* SHA-3

Output (*d* = 128 *bit*)	Heksadesimal
11000011 01001100 00011111 11110000 10011100 00011101 00001001 01101000 11001110 01111011 11111110 00011100 00000111 11001111 00001010 11101101	c3 4c 1f f0 9c 1d 09 68 ce 7b fe 1c 07 cf 0a ed

3.2.2 Implementasi Algoritma ElGamal

Algoritma ElGamal digunakan untuk melakukan proses enkripsi terhadap *message digest* yang diperoleh sebelum piagam dikirim kepada pihak penerima. Proses penerapan algoritma ElGamal menggunakan bilangan desimal, sehingga *message digest* yang diperoleh harus dikonversi terlebih dahulu. Adapun proses pembentukan kunci tersebut melalui beberapa proses sebagai berikut :

1. Konversi *Message Digest*
2. Inisialisasi nilai *p*, *g*, *x*, dengan ketentuan *p* > *g* dan *p* > *x*.
   Bilangan prima yang digunakan lebih baik memiliki nilai yang besar karena semakin besar nilai yang digunakan, maka keamanan yang terbentuk akan semakin kuat. Untuk mempermudah pembentukan kunci, maka nilai prima yang diinputkan oleh pengirim harus terdiri atas 3 digit angka.
   p = 251 ; *g* = 3 ; *x* = 5
   Nilai *p* merupakan bilangan prima sedangan *g* dan *x* merupakan bilangan acak yang harus bernilai lebih kecil dari *p*.
3. Pembangkitan bilangan acak (random) *k* dengan ketentuan $1 \leq k \leq p\text{-}2$.
   p = 131
   p-2 = 131-2 = 129
   Maka diambil nilai k yang digunakan berkisar antara 1 – 129.
4. Proses enkripsi dengan menggunakan rumus $a = g^x \bmod p$ atau $b = y^k\ m \bmod p$ yang akan dibagi kedalam blok *a* dan *b*.Sebelum melakukan enkripsi maka dicari nilai *y* dengan rumus sebagai berikut :
   $y = g^x \bmod p$
   $y = 3^5 \bmod 131$
   $y = 243 \bmod 131$
   $y = 112$
   Nilai masukkan : 195 76 31 240 156 29 9 104 206 123 254 28 7 207 10 237.
   Kunci public = {*p*,*g*,*y*} = {131,3,112}
   Kunci privat = {*p*,*x*} = {131,5}
   Kunci privat akan diberikan kepada penerima *file* sebagai inputan kunci dalam proses verifikasi. Setelah pengirim menginputkan nilai *p* dan *x*, kunci privat secara otomatis terbentuk. Berdasarkan inputan yang diberi oleh penerima yaitu *p* = 131 dan *x* = 5, maka kunci privat yang terbentuk adalah 1315. Jika nilai *m* > *p*, maka nilai *b* yang diperoleh pada proses enkripsi, diolah dihitung kembali dengan nilai *m* = *b*.

Proses perhitungan terus dilakukan hingga nilai masukan terakhir. Untuk hasil perhitungan dapat dilihat pada tabel 3 berikut :

Tabel 3. Enkripsi Algoritma ElGamal

Md	*k*	*m*	$a = g^k \bmod p$	$b = y^k m \bmod p$	Ciphertext (*a*,*b*)
c3	3	195	112	7	(112,102)
4c	4	76	81	10	(81,10)
1f	2	31	9	56	(9,56)
f0	4	240	81	4	(81,4)
9c	3	156	27	4	(27,4)
1d	4	29	81	90	(81,90)
09	5	9	112	43	(112,43)
68	5	104	112	2	(112,2)
ce	3	206	27	12	(27,12)
7b	2	123	9	125	(9,125)
fe	6	254	74	13	(74,13)
1c	4	28	81	114	(81,114)
07	2	7	9	38	(9,38)
cf	4	207	81	10	(81,10)
0a	5	10	112	106	(112,106)
ed	3	237	27	127	(27,127)

Hasil blok (*a*,*b*) digabungkan agar mendapat nilai ciphertext yang utuh. Hasil penggabungan blok (a,b) yaitu : 112, 102, 81, 10, 9, 56, 81, 4, 27, 4, 81, 90, 112, 43, 112, 2, 27, 12, 9, 125, 74, 13, 81, 114, 9, 38, 81, 10, 112, 106, 27, 127.

3.2.3 Proses Verifikasi

Verification adalah tahapan yang dilakukan oleh penerima *file* untuk menguji keabsahan atau keaslian dari *file* yang dikirim. Pada proses verification ciphertext yang diperoleh dari algoritma ElGamal akan dideskripsikan dengan kunci privat {p,x} yang diberikan oleh pengirim *file* dan kemudian dibandingkan dengan nilai hash dari algoritma SHA-3. Adapun tahapan *verification* algoritma SHA-3 dan ElGamal yaitu :

1. Konversi *sign file* kedalam bentuk heksadesimal dan biner
2. Implementasi Algoritma SHA-3
   Algoritma SHA-3 diimplementasikan pada sampel *file* dengan tujuan untuk mendapatkan nilai hash atau *message digest* dari *file* yang telah di-sign.
3. Implementasi Algoritma ElGamal
   Pada tahapan ini dilakukan proses deskripsi terhadap *ciphertext* yang telah disisipkan pada *sign file* dengan menggunakan kunci privat. Pada penelitian ini kunci privat yang diberikan oleh pengirim adalah 1315. Sistem akan secara otomatis mengambil 3 angka pertama yaitu 131 sebagai bilangan prima *p* dan angka sesudahnya yaitu 5 sebagai nilai *x*. Adapun *ciphertext* yang diperoleh pada *sign file* yaitu: 112, 102, 81, 10, 9, 56, 81, 4, 27, 4, 81, 90, 112, 43, 112, 2, 27, 12, 9, 125, 74, 13, 81, 114, 9, 38, 81, 10, 112, 106, 27, 127.
   Dalam melakukan proses deskripsi terdapat beberapa tahapan yang dilakukan antara lain :
   a. Bagi hasil *ciphertext* kedalam blok *a* dan *b*.
   b. Hitung nilai $(a^x)^{-1}$ dan *m* dari *digital signature*
   c. Setelah mendeskripsikan nilai *hash* pada *file* yang diterima, maka sistem akan membandingkan nilai *hash* dari *sign file* dengan nilai *hash* dari *file* yang dikirim oleh penandatangan, dimana nilai *hash* tersebut diperoleh melalui proses deskripsi algoritma ElGamal dengan menginputkan kunci privat.

Tabel 4. Perbandingan Nilai *Hash*

Nilai *Hash File* terkirim	NilaI *Hash File* diterima	Hasil
c3 4c 1f f0 9c 1d 09 68 ce 7b fe 1c 07 cf 0a ed	c3 4c 1f f0 9c 1d 09 68 ce 7b fe 1c 07 cf 0a ed	Otentik

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, terdapat beberapa kesimpulan yang diperoleh oleh penulis, antara lain:

1. Proses otentikasi dengan memanfaatkan *digital signature* cukup efisien karena penggunaan fungsi *hash* dapat memastikan keotentikan dari isi *file* sesuai dengan sifat dari *one-way hash* yang dimilikinya serta algoritma asimetris dapat memastikan keotentikan dari penandatangan *file* karena penggunaan kunci yang berbeda dalam proses enkripsi dan deskripsinya.
2. Implementasi Algoritma SHA-3 dan ElGamal dapat diimplementasikan untuk proses pembentukan *digital signature* dimana algoritma SHA-3 berperan sebagai algoritma pembentukan *message digest* yang kemudian dienkripsi dengan algoritma ElGamal untuk memenuhi fungsi keamanan.
3. Algoritma SHA-3 memiliki beberapa varian dengan prinsip kerja yang mirip satu sama lain, akan tetapi setiap varian memiliki perbedaan terutama untuk ukuran dari *block size* dan *capacity* yang tersedia. Varian SHAKE128, memiliki kelebihan dimana pengguna dapat menentukan panjang *output* yang diinginkan, sehingga dalam perannya untuk pembentukan *digital signature*, setiap piagam kemungkinan memiliki panjang digital signature yang berbeda. Hal ini memiliki kelemahan karena panjang *digital signature* yang berbeda akan menyulitkan proses otentikasi oleh penerima.
4. Penerapan algoritma ElGamal digunakan agar proses otentikasi piagam penghargaan hanya dapat dilakukan oleh orang yang berhak dan mengetahui kunci deskripsinya, serta dengan memanfaatkan algoritma asimetris ElGamal, pengirim tidak dapat menyangkal bahwa dialah yang telah mengirim piagam penghargaan tersebut.

DAFTAR PUSTAKA

[1] E. C. Prabowo and I. Afrianto, "Penerapan Digital Signature Dan Kriptografi Pada Otentikasi Sertifikat Tanah Digital," *Komputa J. Ilm. Komput. dan Inform.*, vol. 6, no. 2, pp. 83–90, 2017, doi: 10.34010/komputa.v6i2.2481.

[2] R. K. Hondro, "Analisis Algoritma CLEFIA 128 Bit Jenis Block Cipher Untuk Pengamanan Teks," *SAINTEK (Jurnal Sains dan Teknol.*, vol. 1, no. 2, pp. 35–38, 2020.

[3] M. P. Sari, "Analisis Algoritma SHA-3 Keamanan pada Data Pribadi," *TECNOSCIENZA*, vol. 5, no. 2, 2021.

[4] A. Fauzi and R. P. Rahayu, "Pemanfaatan USB Flashdisk Sebagai Kunci Pada Keamanan Data Dengan Penerapan Algoritma ElGamal," *J. Tek. Inform. Kaputama*, vol. 4, no. 2, pp. 178–186, 2020.

[5] A. Lorien and T. Wellem, "Implementasi Sistem Otentikasi Dokumen Berbasis Quick Response (QR) Code dan Digital Signature," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 4, pp. 663–671, 2021, doi: 10.29207/resti.v5i4.3316.

[6] Sugiyatno and P. D. Atika, "Digital Signature Dengan Algoritma Sha-1 Dan Rsa Sebagai Autentikasi," *J. Cendikia*, vol. 16, no. 2, pp. 74–83, 2018.

[7] B. K. Hutasuhut, *Analisis Rancangan Model Digital Signature dengan Kombinasi Algoritma MD5, Algoritma Elgamal, dan Algoritma RSA untuk Menguji Keaslian Data dengan Akurat*. 2019.

[8] N. Indahwati and A. Prihanto, "Penerapan Algoritma Kriptografi Asimetris Elgamal dengan Modifikasi Pembangkit Kunci terhadap Enkripsi dan Dekripsi Gambar Warna," *J. Informatics Comput. Sci.*, vol. 1, no. 02, pp. 97–103, 2020, doi: 10.26740/jinacs.v1n02.p97-103.

[9] A. D. Dias, "PENGAMANAN FILE PDF MENGGUNAKAN ALGORITMA RSA DIGITAL SIGNATURE DAN FUNGSI HASH SHA-3 BERBASIS WEB," 2021.

[10] N. Gunawan, "Implementasi Tanda Tangan Digital Menggunakan Algoritma SHA-3 dan ED25519," 2020.

[11] F. Kurniawan, "Analisis Dan Implementasi Algoritma Sha-1 Dan Sha-3 Pada Sistem Autentikasi Garuda Training Cost," 2017.