Journal of Computation Science And Artificial Intelligence

Journal homepage:
https://jcsai.xjurnal.com/index.php/journal/index

PT. BERKAH DIGITAL TEKNOLOGI

e-ISSN: 3032-4653

Vol. 3, No. 1, 2026

PENERAPAN ALGORITMA KRIPTOGRAFI ELGAMAL DAN SHA-256 PADA APLIKASI PENGOLAHAN DATA ATP-WTP UNTUK PENGAMANAN DATA RESPONDEN

Muhammad Erwanto[1*], Fajriatus Sholihah [2], Sugeng Wahyudi [3]
[1,2,3]Sekolah Tinggi Ilmu Komputer (STIKOM) Poltek Cirebon
Email: [1*]muhammaderwanto@gmail.com, [2] fajriatus.aholihah@stikompoltekcirebon.ac.id,
[3]sugeng.wahyudi3209@gmail.com

Abstrak

Berdasarkan UU Nomor 16 Tahun 1997 Tentang Statistik, pada pasal 21 disebutkan bahwa penyelenggara kegiatan statistik wajib melindungi kerahasiaan data individu dari responden. BPS Kabupaten Cirebon sebagai penyelenggara kegiatan statistik wajib melindungi data individu responden. Kriptografi menyediakan layanan untuk mengamankan data dengan cara melakukan transformasi sehingga data tersebut tidak lagi dapat dibaca. Algoritma kriptografi yang digunakan dalam penelitian ini adalah Algoritma ElGamal, algoritma ini dipilih karena kehandalannya dalam menjaga plainteks. Namun hal tersebut tidak menutup kemungkinan terjadinya penyisipan pesan ke dalam *ciphertext* sehingga akan membuat rusak pesan pada saat dilakukan *deciphering*, namun hal tersebut dapat diatasi menggunakan *checksum.* SHA256 sebagai algoritma standar yang dibuat oleh NIST di gunakan untuk menjaga integritas file *ciphertext*. Adapun Data yang digunakan dalam penelitian ini adalah berupa data hasil pengolahan survei ATP-WTP (*ability to pay and willingness to pay*) yang diselenggarakan oleh Badan Pusat Statistik Kabupaten Cirebon.

Kata kunci: *ElGamal Public Key Algorithm; SHA256; ATP-WTP.*

IMPLEMENTATION OF ELGAMAL AND SHA-256 CRYPTOGRAPHIC ALGORITHMS IN THE ATP-WTP DATA PROCESSING APPLICATION FOR RESPONDENT DATA SECURITY

Abstract

According to Law Number 16 of 1997 concerning of Statistics, Article 21 states that the organizers of statistical activities are obliged to protect the confidentiality of individual data from respondents. BPS Cirebon Regency as the organizer of statistical activities is obliged to protect individual respondent data. Cryptography provides services to secure data by transforming it so that the data can no longer be read. The cryptographic algorithm used in this study is the ElGamal Algorithm, this algorithm was chosen because of its reliability in maintaining plaintext. However, this does not rule out the possibility of inserting messages into the ciphertext so that it will damage the message when deciphering is done, but this can be overcome using checksums. SHA256 as a standard algorithm created by NIST is used to maintain the integrity of ciphertext files. The data used in this study is in the form of data from the processing of the ATP-WTP survey (ability to pay and willingness to pay) organized by the Central Bureau of Statistics of Cirebon Regency.

Kata kunci: *ElGamal Public Key Algorithm; SHA256; ATP-WTP*
DOI: https://doi.org/10.58468/jcsai.v3i1.26

1. PENDAHULUAN

Penelitian ini dilakukan untuk menerapkan metode pengamanan data dengan menggunakan Algoritma Kriptografi ELGamal dan *checksum* dengan menggunakan SHA256, pengamanan data dimaksud adalah pada data mentah hasil survei yang dilakukan oleh BPS Kabupaten Cirebon sebagai Instansi pemerintah dengan tupoksi penyelenggara kegiatan statistik untuk statistik dasar. Untuk studi kasus pada penelitian ini, digunakan data hasil survei ATPWTP *(Ability to Pay and Willingness to Pay).* Tujuan penelitian ini adalah membangun dan mengimplementasikan sistem keamanan untuk melindungi data pribadi yang dikumpulkan dalam survei ATP-WTP yang dilakukan oleh Badan Pusat Statistik Kabupaten Cirebon. Sistem ini menggunakan algoritma kriptografi asimetris, yaitu algoritma ElGamal, untuk mengenkripsi data survei agar aman saat dikirimkan kepada pihak yang berwenang. Selain itu, sistem ini juga mengintegrasikan penggunaan algoritma SHA-256 untuk menjaga integritas file data yang telah terenkripsi. Penelitian ini bertujuan untuk memberikan solusi atas tantangan pengamanan data survei, sesuai dengan kewajiban yang diatur dalam Undang-Undang Nomor 16 Tahun 1997 tentang Statistik, yang mewajibkan perlindungan data pribadi responden dalam kegiatan statistik.. Berdasarkan pasal 21 UU No.16 Tahun 1997 tentang Statistik, penyelenggara statistik wajib melindungi data individu responden. Data yang dapat dipublish adalah berupa data agregat.

2. LITERATURE REVIEW

Menurut Menez (1996) Kriptografi adalah Ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan Informasi seperti kerahasiaan, Integritas data, serta otentikasi.(Munir, 2019)

Kriptografi juga dapat didefinisikan sebagai Ilmu dan Seni untuk menjaga keamanan pesan (Shneier, 1996).

Kriptografi dikatakan seni, karena pada awal-awal perkembangannya, setiap orang memiliki cara yang unik untuk menjaga kerahasiaan pesan, cara unik tersebut berbeda-beda tiap orang, oleh karena itulah memiliki estetika tersendiri, sehingga disebut seni. Namun seiring berjalannya waktu, kriptografi berkembang begitu pesat sehingga menjadi sedemikian kompleks, sejak perang dunia I dan datangnya era komputer, metode yang digunakan dalam kriptografi telah meningkat secara eksponensial. Kriptografi modern sangat didasari pada fungsi-fungsi matematis dan aplikasi komputer (Munir 2019).

Algoritma Kriptografi ElGamal merupakan algoritma kritografi kunci publik, yang ditemukan oleh Taher ElGamal yang diterbitkan melalui papernya pada tahun 1985.

Keamanan Algoritma Elgamal terletak pada kesulitan perhitungan logaritma diskrit pada modulo prima yang besar (*finite field)* (ElGamal, 1985) sehingga upaya untuk menyelesaikan masalah logaritma ini menjadi sulit untuk dipecahkan.

Beberapa serangan berhasil dilakukan terhadap algoritma kriptografi ElGamal dan RSA apabila pesan yang dienkripsi pendek dan tidak mengalami pemrosesan lebih lanjut. Serangan terhadap Algoritma ElGamal juga tergantung terhadap parameter tambahan yang digunakan untuk membuat *cryptosystem*. (Allen, 2008).

Secure Hash Algorithm (SHA) yang dikembangkan oleh Institut Nasional Standar dan Teknologi (NIST) juga dirancang dengan prinsip yang sama dengan MD4 dan diterbitkan sebagai Standar Pemrosesan Informasi Federal (FIPS 180) pada tahun 1993. Versi revisi dikeluarkan sebagai FIPS180- 1 pada tahun 1995 dan secara umum disebut sebagai

SHA-1. Ketika versi revisi SHA-1 diterbitkan, tidak ada rincian kelemahan yang ditemukan dalam SHA-0 (awalnya SHA) disediakan. SHA-1 menghasilkan nilai hash 160 bit. Pada tahun 2002, NIST menghasilkan versi revisi dari standar yang dikenal sebagai FIPS180-2 dan mendefinisikan tiga versi baru SHA dengan panjang intisari 256, 384 dan 512 dan masing-masing dikenal sebagai SHA-256, SHA-384, dan SHA-512 (Okeyinka, 2018).

Penelitian ini menggunakan algoritma kriptografi ElGamal, yang merupakan algoritma kriptografi kunci publik yang andal dalam menjaga kerahasiaan pesan. ElGamal dipilih karena ketahanannya terhadap serangan kriptografi, terutama serangan yang memanfaatkan kelemahan pada algoritma lain, seperti RSA. Selain itu, SHA-256, yang merupakan bagian dari keluarga algoritma Secure Hash Algorithm (SHA), digunakan untuk memastikan integritas data yang telah dienkripsi. SHA-256 menghasilkan nilai hash yang tidak dapat diubah, sehingga memungkinkan penerima data untuk memverifikasi keaslian data yang diterima. Sebagai tambahan, kami juga menambahkan contoh penerapan ElGamal dalam pengamanan data transaksi dan sistem komunikasi aman, untuk memperlihatkan aplikasinya dalam konteks yang lebih luas.

3. BAHAN DAN METODE

Menurut Menez (1996) Kriptografi adalah Ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan Informasi seperti kerahasiaan, Integritas data, serta otentikasi.(Munir, 2019)

Kriptografi juga dapat didefinisikan sebagai Ilmu dan Seni untuk menjaga keamanan pesan (Shneier, 1996).

Kriptografi dikatakan seni, karena pada awal-awal perkembangannya, setiap orang memiliki cara yang unik untuk menjaga kerahasiaan pesan, cara unik tersebut berbeda-beda tiap orang, oleh karena itulah memiliki estetika tersendiri, sehingga disebut seni. Namun seiring berjalannya waktu, kriptografi berkembang begitu pesat sehingga menjadi sedemikian kompleks, sejak perang dunia I dan datangnya era komputer, metode yang digunakan dalam kriptografi telah meningkat secara eksponensial. Kriptografi modern sangat didasari pada fungsi-fungsi matematis dan aplikasi komputer (Munir 2019).

Algoritma Kriptografi ElGamal merupakan algoritma kritografi kunci publik, yang ditemukan oleh Taher ElGamal yang diterbitkan melalui papernya pada tahun 1985.

Keamanan Algoritma Elgamal terletak pada kesulitan perhitungan logaritma diskrit pada modulo prima yang besar (*finite field)* (ElGamal, 1985) sehingga upaya untuk menyelesaikan masalah logaritma ini menjadi sulit untuk dipecahkan.

Beberapa serangan berhasil dilakukan terhadap algoritma kriptografi ElGamal dan RSA apabila pesan yang dienkripsi pendek dan tidak mengalami pemrosesan lebih lanjut. Serangan terhadap Algoritma ElGamal juga tergantung terhadap parameter tambahan yang digunakan untuk membuat *cryptosystem*. (Allen, 2008).

Secure Hash Algorithm (SHA) yang dikembangkan oleh Institut Nasional Standar dan Teknologi (NIST) juga dirancang dengan prinsip yang sama dengan MD4 dan diterbitkan sebagai Standar Pemrosesan Informasi Federal (FIPS 180) pada tahun 1993. Versi revisi dikeluarkan sebagai FIPS180- 1 pada tahun 1995 dan secara umum disebut sebagai SHA-1. Ketika versi revisi SHA-1 diterbitkan, tidak ada rincian kelemahan yang ditemukan dalam SHA-0 (awalnya SHA) disediakan. SHA-1 menghasilkan nilai hash 160 bit. Pada tahun 2002, NIST menghasilkan versi revisi dari standar yang dikenal sebagai FIPS180-2 dan mendefinisikan tiga versi baru SHA dengan panjang intisari 256, 384 dan 512 dan masing-masing dikenal sebagai SHA-256, SHA-384, dan SHA-512 (Okeyinka, 2018).

Penelitian ini menggunakan algoritma kriptografi ElGamal, yang merupakan

algoritma kriptografi kunci publik yang andal dalam menjaga kerahasiaan pesan. ElGamal dipilih karena ketahanannya terhadap serangan kriptografi, terutama serangan yang memanfaatkan kelemahan pada algoritma lain, seperti RSA. Selain itu, SHA-256, yang merupakan bagian dari keluarga algoritma Secure Hash Algorithm (SHA), digunakan untuk memastikan integritas data yang telah dienkripsi. SHA-256 menghasilkan nilai hash yang tidak dapat diubah, sehingga memungkinkan penerima data untuk memverifikasi keaslian data yang diterima. Sebagai tambahan, kami juga menambahkan contoh penerapan ElGamal dalam pengamanan data transaksi dan sistem komunikasi aman, untuk memperlihatkan aplikasinya dalam konteks yang lebih luas.

3. METODE YANG DIGUNAKAN

Analisis permasalahan dalam penelitian ini adalah kurangnya keamanan dalam hal transfer data hasil dari pengolahan survei yang dilakukan oleh kantor BPS Kabupaten Cirebon, oleh karena itu perlu dibuat suatu sistem yang mampu menangani celah keamanan tersebut.

Penelitian ini dilakukan untuk mengatasi keamanan tersebut, yaitu dengan menerapkan suatu metode penyandian atau kriptografi terhadap file data hasil pengolahan survei, metode yang digunakan adalah penggabungan metode kriptografi algoritma kunci publik ElGamal dengan SHA256. Algoritma ElGamal digunakan untuk mengenkripsi dan mendekripsi file data kemudian sebelum data yang terenkripsi dikirimkan dilakukan generate terlebih dahulu nilai hash dengan menggunakan algoritma SHA256. Setelah data yang terenkripsi diterima oleh pihak penerima, maka sebelum dilakukan proses dekripsi data, dilakukan penyamaan terlebih dahulu hash yang dikirimkan oleh pihak pengirim, dengan tujuan untuk mengetahui apakah data yang telah dienkripsi mengalami perubahan pada saat pengiriman ataukah merupakan data asli dari pengirim.

Karena apabila data yang diterima tidak memiliki kesamaan nilai *hash* hal tersebut dapat berarti bahwa data yang diterima (cipher) bukan merupakan data yang asli atau telah mengalami perubahan. Perubahan tersebut bisa saja terjadi karena malware atau karena *man in the middle attack*. Perlu ditekankan bahwa apabila cipher telah mengalami perubahan, maka hasil dekripsi juga akan berubah, apabila perubahan tersebut dilakukan secara sengaja oleh pihak yang tidak memiliki wewenang maka hal tersebut akan merubah data hasil dekripsi, yang artinya data yang diterima tidak sesuai dengan hasil survei yang telah dilakukan dan hal ini menjadi krusial mengingat file data digunakan untuk tabulasi maupun analisis terhadap survei yang telah dilakukan dan hal ini akan berpengaruh terhadap kebijakan pemerintah yang akan diambil berdasarkan analisa dari data mentah hasil entri.

3.1 Algoritma Kriptografi ElGamal

Algoritma ElGamal merupakan contoh algoritma dengan kunci asimetris yang didasarkan pada logaritma diskrit. (Anggreni, 2019).

Algoritma ElGamal terdiri dari 3 (tiga) proses, yaitu (1) Proses pembentukan kunci, (2) proses enkripsi dan (3) proses dekripsi. Skema algoritma ElGamal memerlukan sepasang kunci yang akan digunakan untuk proses enkripsi dan dekripsi.

Untuk proses enkripsi, dibangkitkan dari nilai *p, g* dan *y* (merupakan kunci publik) sedangkan kunci dekripsi terdiri dari nilai *x* (*private*) dan kunci publik *p.* Namun Waktu proses enkripsi lebih lama dibandingkan waktu dekripsi dikarenakan

perhitungan matematika dalam proses enkripsi (Nurul, 2018).

Pembangkitan Kunci

Adapun langkah-langkah dan persyaratan yang harus dipenuhi untuk pembangkitan kunci adalah sebagai berikut :
1.Pilih sembarang bilangan prima p
2.Pilih bilangan acak *g* dengan syarat $g < p$
3.Pilih bilangan acak *x* dengan syarat $1<x<p-2$
4.Hitung nilai :

$$y = g^x \bmod p \text{ ..... } (1)$$

Nilai *p, g, y* merupakan kunci publik bersifat tidak rahasia, sedangkan nilai *x* merupakan kunci *private* bersifat rahasia yang digunakan untuk mendekripsi pesan**.**

Enkripsi

Pada proses enkripsi digunakan triplet kunci publik (*p, g, y*). Langkah-langkah dalam melakukan enkripsi adalah sebagai berikut :

1. Potong plaintext menjadi blok-blok *m1, m2…mn* nilai setiap blok di dalam selang [*0, p – 1*].
2. Ubah nilai blok pesan ke dalam nilai ASCII (Desimal).
3. Pilih bilangan acak k, dengan syarat 1 < k > (p – 1) sebanyak *m*
4. Setiap blok m dienkripsi dengan *formula* sebagai berikut :
$$a = g^k \bmod p \text{..... } (2)$$
$$b = y^k m \bmod p \text{.....} (3)$$
5. Susun *chipertext* dengan urutan *a1, b1, a2, b2,…, an, bn.*

Pasangan *a* dan *b* adalah *chipertext* untuk blok pesan *m*. Hasil yang didapat dari proses enkripsi berupa pesan rahasia (*chipertext*), dari persamaan (2) dan (3) terlihat bahwa sebuah blok pesan *m* akan menjadi dua buah blok pesan *a* dan *b*. Karena hal inilah maka hasil dari *cipherteks* akan menjadi dua kali ukuran dari plainteks.

Dekripsi

Pada Algoritma ElGamal, untuk melakukan dekripsi pesan dilakukan dengan menggunakan kunci private (*x, p*). Berikut langkah-langkah dalam melakukan dekripsi :

1. Hitung Plainteks *m* dengan persamaan berikut :
$$m_i = b(a^p)^{-1-x} \bmod p \text{ ..... } (4)$$
2. Nilai m_i yang diperoleh dari persamaan diatas dalam bentuk ASCII tersebut, kemudian diubah dalam bentuk plainteks.
3. Susun plainteks ke dalam urutan *m1,m2,...,mn.*

Dengan menyusunnya kembali seperti susunan diatas maka akan didapat kembali plainteks.

***Secure Hash Algorithm* (SHA-256)**

Fungsi hash adalah algoritma Kriptografi yang memampatkan pesan (kompresi) pesan berukuran sembarang menjadi pesan ringkas (message digest) berukuran tetap (fixed). Beberapa fungsi hash yang, diataranya MD (Message Digest) 5, SHA-1, SHA-2, Keccak (SHA-3), RIPEMD, dan sebagainya. (Munir, 2019). SHA256 merupakan generasi ke-2 (dua) dari SHA. Pada tahun 2011 – 2015 SHA-1 ditetapkan sebagai algoritma utama dalam industri SSL (*Secure Socket Layer*) namun karena ditemukan collution maka pada tahun 2016 ditetapkan SHA-2 menjadi standar baru dalam penggunaan sertifikat SSL. *Collution* merupakan suatu celah keamanan dimana suatu pesan yang berbeda dapat memiliki hash yang sama. Fungsi hash bersifat satu arah, artinya sebuah pesan yang dikompres menjadi sebuah hash (ciphertext) maka pesan tersebut tidak dapat dikembalikan kembali menjadi pesan semula.

Fungsi hash h mengambil pesan m dengan panjang sembarang dan menghasilkan pesan inti *h (m)* dengan panjang tetap (Okeyinka, 2018)
Fungsi hash menjadi suatu bagian yang penting dalam kriptografi, dengan fungsi hash suatu keamanan pesan dapat diuji

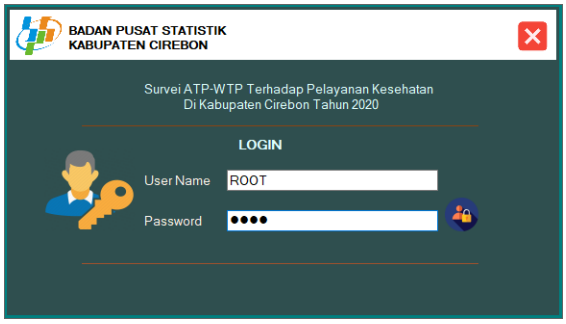
integritasnya, karena dua buah pesan yang berbeda (walaupun hanya berbeda satu buah karakter) maka akan menghasilkan pesan ringkas (*hash*) yang berbeda secara signifikan, perubahan pada sebuah pesan walaupun satu buah karakter maka akan merubah keseluruhan dari nilai hash. Dengan sifatnya yang tetap (*fixed*) dan sensitif terhadap perubahan, maka SHA256 dapat dijadikan untuk menguji keaslian sebuah pesan, apakah pesan tersebut telah mengalami perubahan atau tidak.

Algoritma ElGamal digunakan untuk mengenkripsi data survei yang sensitif, sementara SHA-256 digunakan untuk menghasilkan nilai hash dari data yang telah terenkripsi. Sebelum data terenkripsi dikirimkan ke penerima, nilai hash dihitung untuk memastikan bahwa data tidak berubah selama transmisi. Penerima akan menghitung kembali nilai hash dari data yang diterima dan membandingkannya dengan nilai hash yang dikirimkan oleh pengirim. Jika kedua nilai hash tersebut cocok, maka data dapat dipastikan tidak mengalami perubahan dan tetap utuh. Penjelasan lebih rinci tentang penerapan kedua algoritma ini secara bersamaan terdapat pada Bagian 3.1, pada sub-bab "Integrasi ElGamal dan SHA-256 untuk Keamanan Data".

4. HASIL DAN BAHASAN

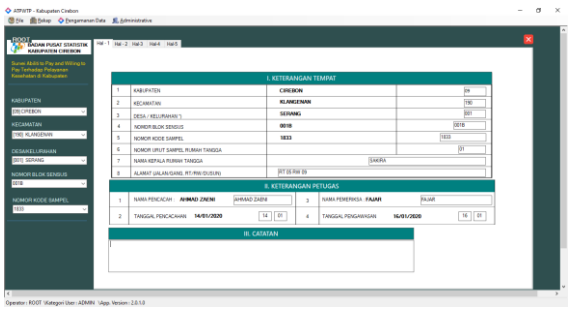
A. Pembuatan sistem input data

Pada tahap ini dilakukan pembuatan aplikasi untuk input data dari kegiatan ATP-WTP, aplikasi yang dibuat menggunakan Visual Basic.Net dengan DBMS menggunakan MySQL, tampilan antarmuka aplikasi entri data ATP-WTP adalah sebagai berikut:



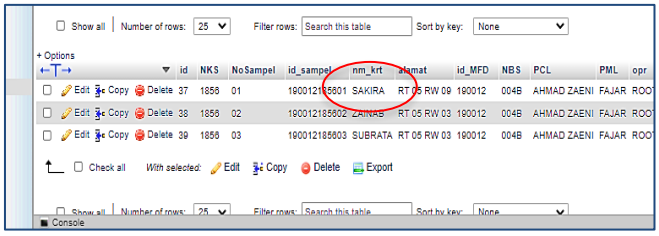
Gambar 1 User Interface Login

Pengguna harus login terlebih dahulu untuk dapat mengakses sistem. Hanya pengguna dengan tipe "operator" yang dapat melakukan entri data, pengguna dengan tipe "supervisor" dapat melakukan pengamanan data.



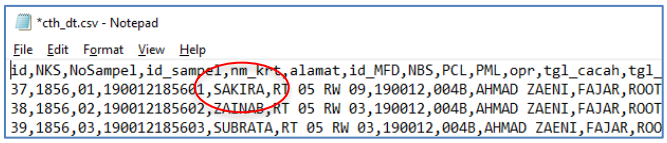
Gambar 2 Form Input Data

Form input data individu responden terdiri dari 5 halaman, menyesuaikan dengan kuesioner.



Gambar 3 Data di dalam Database MySQL

Setelah data responden di entri, maka untuk dapat menerapkan pengamanan data, data tersebut harus di ekspor terlebih dahulu ke dalam format *.csv



Gambar 4 Data Responden dalam bentuk *.csv

Pada penelitian ini, teknik penerapan algoritma kriptografi tidak dilakukan pada data yang terdapat didalam database, namun pada data hasil eksport, seperti pada gambar

3. Hal ini dikarenakan, data inilah yang rentan untuk disalah gunakan (menyalahi pasal 21 UU No.16 Tahun 1997 tentang Statistik).

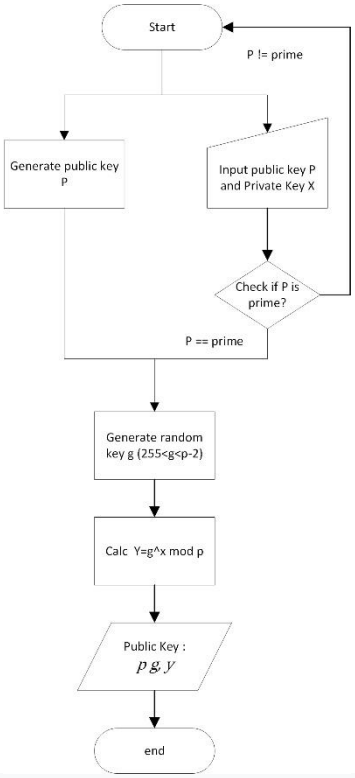
B. Implementasi Pengamanan Data

Implementasi algoritma Kriptografi kunci publik ElGamal dan algoritma fungsi hash menyatu dengan aplikasi input data.. Secara garis besar berikut ini adalah tahapan yang digunakan :

1. Pembangkitan Kunci Publik
2. Enkripsi dan Dekripsi data
3. Pembangkitan Hash
4. Pencocokan nilai hash cipher

Untuk memperkuat klaim keamanan sistem yang dibangun, kami telah melakukan pengujian terhadap ketahanan sistem terhadap beberapa jenis serangan yang umum, seperti serangan Man-in-the-Middle dan modifikasi data selama proses pengiriman. Kami menggunakan data uji untuk mensimulasikan serangan ini dan mengamati bagaimana sistem kriptografi ElGamal dan SHA-256 dapat mendeteksi dan mengatasi potensi perubahan data yang tidak sah. Hasil pengujian menunjukkan bahwa sistem dapat menjaga integritas data, dan hasil hash yang diverifikasi tidak berubah meskipun data mengalami transmisi melalui jaringan yang tidak aman. Detil hasil pengujian dapat dilihat pada Tabel 4 dan Gambar 5 di bagian ini.

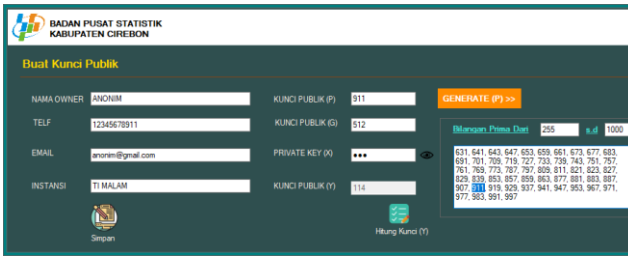
***Flowchart* Pembangkitan Kunci Publik**



Gambar 5. Pembangkitan kunci publik *p, g dan y*

Pembangkitan kunci publik *P* dilakukan secara manual, pengguna memilih bilangan prima *(p)>255*, kemudian pilih sebuah bilangan acak private X, dengan *x(1,p-2)* setelah kunci publik *P* dipilih, maka proses pembentukan kunci publik *G* dilakukan oleh sistem dengan cara memilih bilangan *random* dengan syarat *g<p* setelah kunci publik *g* terbentuk, maka proses selanjutnya sistem akan menghitung kunci publik *y*, dengan menggunakan persamaan (1). Pada contoh penelitian ini digunakan kunci publik sebagai berikut : *p=911 dan x=212*; berdasarkan kunci p, maka dibangkitkan kunci publik *g (random) < p*, dipilih bilangan *g=512*; maka pembangkitan untuk kunci y sebagai berikut :

$$y = g^{x} \bmod p$$
$$y = (512^{212}) \bmod 911$$
$$y = 114$$



Gambar 6 Form Pembangkitan Kunci Publik

Apabila triplet kunci publik sudah terbentuk, maka kunci publik ini dapat di berikan kepada pihak-pihak yang hendak melakukan enkripsi, kunci *private* hanya diketahui oleh pembuat kunci publik, digunakan untuk melakukan deciphering pada pesan yang terenkripsi.

Flowchart Enkripsi Data

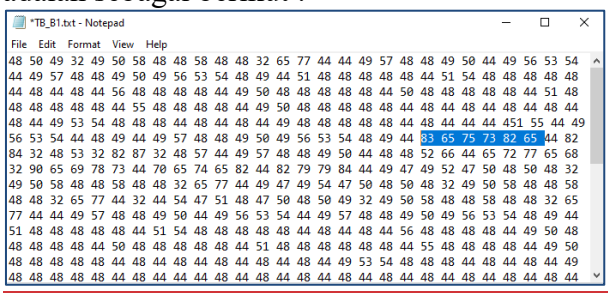
Gambar 7 Flowchart Enkripsi dan Generate String

pada contoh penghitungan ini dilakukan enkripsi terhadap file pada gambar 4.

Proses enkripsi file menggunakan Aplikasi Pengolahan Data ATP-WTP dilakukan tahapan sebagai berikut:

1. Load data dari database ATP-WTP (Gambar 3)
2. Eksport data ke dalam format *.csv (plainteks) (Gambar 4)
3. Setelah data dieksport, maka rubah plainteks tersebut ke dalam bentuk ASCII decimal (gambar 7). Untuk contoh penghitungan dilakukan terhadap satu buah kata SAKIRA, yaitu nama responden seperti pada gambar 4.

Antar muka aplikasi untuk proses enkripsi adalah sebagai berikut :



Gambar 8 hasil konversi plainteks ke ASCII Desimal

Plaintext	S	A	K	I	R	A
ASCII	83	65	75	73	82	65

Setelah mendapatkan nilai decimal ASCII, maka akan dihitung untuk setiap nilai pada blok plaintext akan dienkripsi menjadi 2 (dua) buah blok ciphertext dengan menggunakan persamaan (1) dan (2) berikut ini :

$$a = g^k mod\ p$$
$$b = y^k m\ mod\ p$$

Dengan p=911;g=512 dan y=114, dan diasumsikan pesan k di tentukan, maka untuk plainteks (**83 65 75 73 82 65**) diperoleh cipher sebagai berikut :

1. Untuk $m = 83$ (ASCII=S), generate $k = 435$

$a = g^k mod\ p$
$a = 512^{435} mod\ 911$
$\boldsymbol{a = 4}$
$b = y^k m\ mod\ p$
$b = 114^{435} 83\ mod\ 911$
$\boldsymbol{b = 448}$

Untuk 1 (satu) buah pesan ***m=83*** maka akan dienkripsi menjadi 2 (dua) buah blok pesan (***4, 448***).

2. Untuk m = 65 (ASCII=A), generate k = 783

$a = g^k mod\ p$
$a = 512^{783} mod\ 911$
$\boldsymbol{a = 225}$
$b = y^k m\ mod\ p$
$b = 114^{783} 65\ mod\ 911$
$\boldsymbol{b = 8}$

Untuk 1 (satu) buah pesan m=65 maka akan dienkripsi menjadi 2 (dua) buah blok pesan (225,8)

3. Untuk m = 75 (ASCII=K), generate k = 245

$a = g^k mod\ p$

$a = 512^{245} mod\ 911$

$\boldsymbol{a = 30}$

$b = y^k m\ mod\ p$

$b = 114^{245}75\ mod\ 911$

$\boldsymbol{b = 876}$

Untuk 1 (satu) buah pesan *m=75* maka akan dienkripsi menjadi 2 (dua) buah blok pesan *(30, 876).*

4. Untuk *m = 73 (*ASCII=I*),* generate *k = 854*

$a = g^k mod\ p$

$a = 512^{854} mod\ 911$

$\boldsymbol{a = 900}$

$b = y^k m\ mod\ p$

$b = 114^{854}73\ mod\ 911$

$\boldsymbol{b = 190}$

Untuk 1 (satu) buah pesan *m=73* maka akan dienkripsi menjadi 2 (dua) buah blok pesan *(900, 190).*

5. Untuk *m = 82 (ASCII=R),* generate *k = 396*

$a = g^k mod\ p$

$a = 512^{396} mod\ 911$

$\boldsymbol{a = 883}$

$b = y^k m\ mod\ p$

$b = 114^{396}82\ mod\ 911$

$\boldsymbol{b = 515}$

Untuk 1 (satu) buah pesan *m=73* maka akan dienkripsi menjadi 2 (dua) buah blok pesan *(883, 515).*

6. Untuk m = 65 (ASCII=A), generate k = 273

$a = g^k mod\ p$

$a = 512^{273} mod\ 911$

$\boldsymbol{a = 1}$

$b = y^k m\ mod\ p$

$b = 114^{273}65\ mod\ 911$

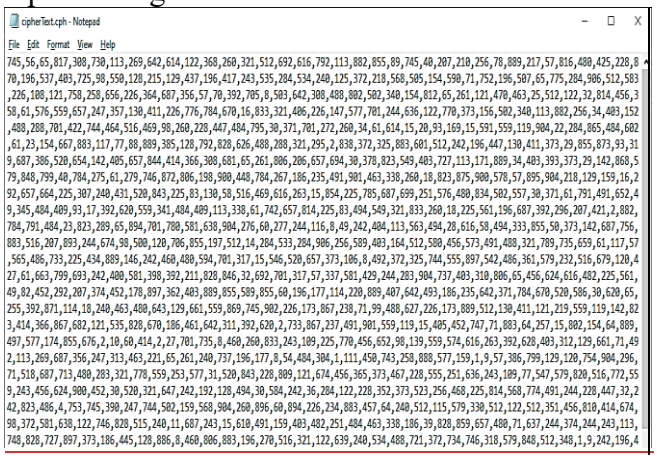
$\boldsymbol{b = 65}$

Untuk 1 (satu) buah pesan *m=73* maka akan dienkripsi menjadi 2 (dua) buah blok pesan *(1, 65).*

Plaintext	:	S	A	K	I	R	A

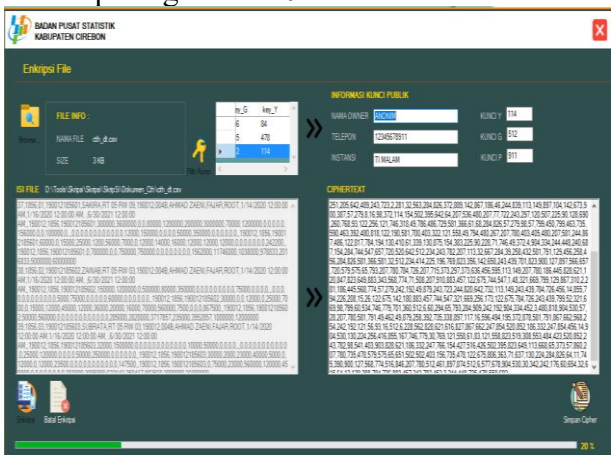
ASCII	:	83	65	75	73	82	65
Cipherte xt	:	444 8	22 5	8	30	87 6	90 0	19 0	88 3	51 5	1	65

Apabila keseluruhan plainteks pada gambar 7 di enkripsi, maka akan menghasilkan cipher sebagai berikut:



Gambar 9 Cipherteks

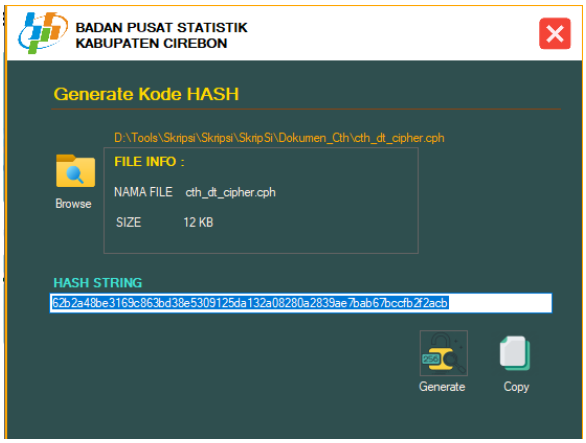
Cipherteks pada gambar 9 merupakan hasil enkripsi dari form pengamanan yang terdapat pada aplikasi olah data ATP-WTP. Cipherteks pada gambar 9 menunjukkan bahwa setelah data dienkripsi,maka data tidak dapat dibaca kecuali setelah dilakukan dekripsi. Form untuk enkripsi data dapat dilihat pada gambar 10 berikut ini :



Gambar 10 Form Enkripsi

Sebelum cipherteks dikirimkan ke pihak yang akan menerima data, maka terlebih dahulu harus di *generate hash* untuk memastikan bahwa file cipherteks tidak mengalami perubahan baik itu sebelum, saat maupun setelah data sampai di pihak penerima cipherteks. Apabila terdapat penyebarluasan cipherteks maka pihak yang

tidak memiliki otoritas tidak akan dapat membaca data karena data yang dikirimkan merupakan data yang sudah di enkripsi. Hanya pihak yang memiliki kunci *private X* saja yang dapat membuka data.



Gambar11 Proses Generate String Hash SHA256

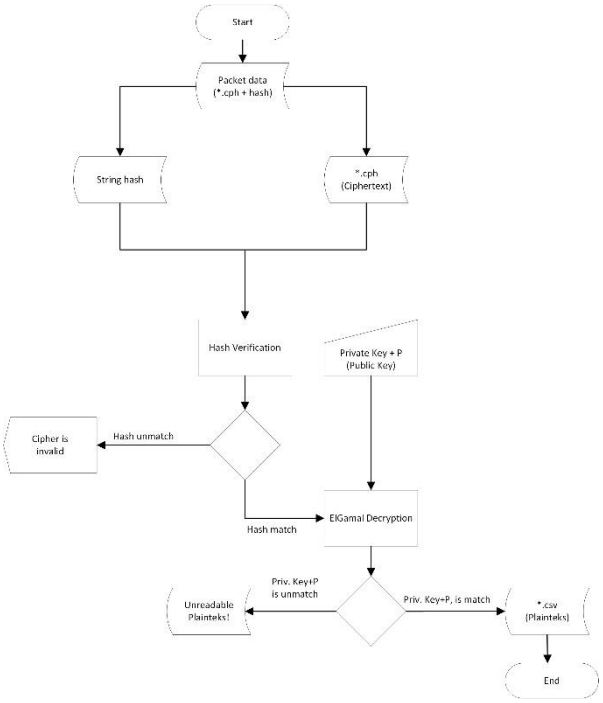
Flowchart Dekripsi data

Sistem menerima masukan berupa file cipherteks *.cph, namun sebelum dilakukan proses deciphering harus di lakukan *checksum* hash terlebih dahulu

Gambar 12 Verifikasi hash

Apabila hash yang diterima sesuai dengan file cipher, maka proses dekripsi dapat dilanjutkan. Dengan menggunakan kunci publik P dan kunci private X, maka proses dekripsi dilakukan dengan formula :

$$m_i = b(a^p)\text{-}^{1\text{-}x} \; mod \; p$$



Gambar 13 Flowmap Dekripsi

Contoh kasus untuk dekripsi adalah kata SAKIRA atau dengan Cipherteks : : **4 448 225 8 30 876 900 190 883 515 1** cipher akan dikelompokkan kedalam blok pesan sebagai berikut :

Blok Pesan	:	m_1		m_2		m_3		m_4		m_5		m_6	
Ciphertext	:	*4*	*448*	*225*	*8*	*30*	*876*	*900*	*190*	*883*	*515*	*1*	*65*

a. Untuk *ciphertext* $m_1(a,b) = (4, 448)$

$$(a^x)^{-1} = a^{p-1-x} \; mod \; p$$
$$(a^x)^{-1} = 4^{911-1-212} \; mod \; 911$$
$$(a^x)^{-1} = 4^{698} \; mod \; 911$$
$$(a^x)^{-1} = \quad 657$$
$$m_1 = b(a^x)^{-1} mod \; p$$
$$m_1$$
$$= 448 \times 657 \; mod \; 911$$
$$m_1 = 294336 \; mod \; 911$$
$$m_1 = \; 83$$

b. Untuk *ciphertext* $m_2 \; (a,b) = (225, 8)$

$$(a^x)^{-1} = a^{p-1-x} \; mod \; p$$
$$(a^x)^{-1} = 225^{911-1-212} \; mod \; 911$$
$$(a^x)^{-1} = 225^{698} \; mod \; 911$$
$$(a^x)^{-1} = 122$$
$$m_2 = b(a^x)^{-1} mod \; p$$
$$m_2 = 8 \times 122 \; mod \; 911$$
$$m_2 = 976 \; mod \; 911$$
$$m_2 = \; 65$$

c. Untuk *ciphertext* $m_3(a,b) = (30, 876)$

$$(a^x)^{-1} = a^{p-1-x} \; mod \; p$$

$(a^x)^{-1} = 30^{911-1-212} \, mod \, 911$
$(a^x)^{-1} = 30^{698} \, mod \, 911$
$(a^x)^{-1} = 128$
$m_3 = b(a^x)^{-1} mod \, p$
$m_3 = 876 \times 128 \, mod \, 911$
$m_3 = 112128 \, mod \, 911$
$m_3 = 75$

d. Untuk *ciphertext* $m_4(a,b) = (900, 190)$
$(a^x)^{-1} = a^{p-1-x} \, mod \, p$
$(a^x)^{-1} = 900^{911-1-212} \, mod \, 911$
$(a^x)^{-1} = 900^{698} \, mod \, 911$
$(a^x)^{-1} = 897$
$m_4 = b(a^x)^{-1} mod \, p$
$m_4 = 190 \times 897 \, mod \, 911$
$m_4 = 170430 \, mod \, 911$
$m_4 = 73$

e. Untuk *ciphertext* $m_5 \, (a,b) = (883, 515)$
$(a^x)^{-1} = a^{p-1-x} \, mod \, p$
$(a^x)^{-1} = 883^{911-1-212} \, mod \, 911$
$(a^x)^{-1} = 883^{698} \, mod \, 911$
$(a^x)^{-1} = 32$
$m_5 = b(a^x)^{-1} mod \, p$
$m_5 = 515 \times 32 \, mod \, 911$
$m_5 = 16480 \, mod \, 911$
$m_5 = 82$

f. Untuk *ciphertext* $m_6 \, (a,b) = (1, 65)$
$(a^x)^{-1} = a^{p-1-x} \, mod \, p$
$(a^x)^{-1} = 1^{911-1-212} \, mod \, 911$
$(a^x)^{-1} = 1^{698} \, mod \, 911$
$(a^x)^{-1} = 1$
$m_6 = b(a^x)^{-1} mod \, p$
$m_6 = 65 \times 1 \, mod \, 911$
$m_6 = 65 \, mod \, 911$
$m_6 = 65$

Susun kembali blok pesan (*m1-m6*) menjadi berikut:

Blok Pesan	:	m_1	m_2	m_3	m_4	m_5	m_6
Ciphertext	:	44	225	830	876	900	190	883	515	1	65
ASCII	:	83	65	75	73	82	65
Plaintext	:	S	A	K	I	R	A

Proses *deciphering* menggunakan Aplikasi pengolahan data ATPWTP menunjukkan bahwa cipherteks dapat di kembalikan menjadi plainteks tanpa merusak isi pesan.

Plainteks dapat disimpan kembali oleh penerima data dalam bentuk *.csv untuk dilakukan proses analisis lanjutan.

Gambar 14 Dekripsi Cipherteks

5. KESIMPULAN

1. Algoritma kriptografi ElGamal dan checksum SHA256 dapat diterapkan untuk mengamankan data hasil entri ATP-WTP
2. Ukuran file data hasil enkripsi (*cipher* data) lebih besar dari file semula.
3. Waktu yang digunakan untuk proses enkripsi lebih lama dari waktu yang digunakan untuk proses dekripsi. Karena perhitungan matematika untuk proses enkripsi lebih kompleks dibandingkan perhitungan untuk dekripsi.
4. Hardware dan ukuran data berpengaruh siginifikan terhadap waktu yang digunakan untuk proses enkripsi maupun dekripsi.
5. Untuk file yang relatif besar, maka disarankan untuk di pisah menjadi beberapa file plainteks, agar proses enkripsi dapat dilakukan secara parsial.

Dalam penelitian ini, kami juga melakukan pengujian terhadap waktu proses enkripsi dan dekripsi, serta ukuran data yang terenkripsi. Pengujian dilakukan dengan data dalam jumlah kecil hingga besar untuk mengukur skalabilitas sistem. Hasil pengujian menunjukkan bahwa meskipun ukuran file terenkripsi lebih besar dari file asli, proses dekripsi dapat dilakukan dengan waktu yang lebih singkat dibandingkan proses enkripsi. Untuk file yang lebih besar, kami menyarankan agar

data dibagi menjadi beberapa file kecil untuk mempermudah proses enkripsi tanpa mengorbankan keamanan.

6. DAFTAR PUSTAKA

Aditya Penerapan Algoritma Elgamal dan SSL Pada Aplikasi Group Chat. Generation Journal, 2(1), 48. https://doi.org/10.29407/gj.v2i1.12052

Anggreni, N. K. A. S., Linawati, L., & Sastra, N. P. (2019). Sistem Pengamanan Anonym dengan Menggunakan Algoritma Kriptografi ElGamal. Majalah Ilmiah Teknologi Elektro, 18(2). https://doi.org/10.24843/mite.2019.v18i02.p07

Arboleda, E. R. (2019). Secure and fast chaotic el gamal cryptosystem. International Journal of Engineering and Advanced Technology, 8(5), 1693–1699.

Chol, K. G., & Chol, L. S. (n.d.). A study on the fast ElGamal encryption. 1–13.

Efendy, Z. (2018). Normalisasi Dalam Desain Database. Jurnal CorelIT, 4(1), 34–43.

ElGamal, T. (1985). A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 196 LNCS(4), 10–18. https://doi.org/10.1007/3-540-39568-7_2

Fitri Ayu and Nia Permatasari. (2018). perancangan sistem informasi pengolahan data PKL pada divisi humas PT pegadaian. Jurnal Infra Tech, 2(2), 12–26. http://journal.amikmahaputra.ac.id/index.php/JIT/article/download/33/25

Hutomo, B. P. S., Wardana, H. K., & Yohanes, B. W. (2018). Pendeteksi Error dengan CRC32 dan Cek Integritas dengan SHA256 pada Aplikasi Pengunduh dan Transfer File. Techné : Jurnal Ilmiah Elektroteknika, 17(02), 109–114. https://doi.org/10.31358/techne.v17i02.178

Janner Simarmata, Sriadhi, R. R. (2019). Kriptografi: Teknik Keamanan Data dan Informasi (M. Kika (ed.); I). Penerbit ANDI.

Karima, A., & Saputro, A. (2016). Pembangkitan Kunci pada Algoritma Asimetris ElGamal untuk Meningkatkan Keamanan Data bertipe . docx Key Generation on ElGamal Asymmetric Algorithm To Enhance . docx Format Data Security. Sisfotenika, 6(2), 170–181. Retrieved from https://www.mendeley.com/catalogue/pembangkitan-kunci-pada-algoritma-asimetris-elgamal-untuk-meningkatkan-keamanan-data-bertipe-docx/

Kurniawan, T. A. (2018). Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik. Jurnal Teknologi Informasi Dan Ilmu Komputer, 5(1), 77. https://doi.org/10.25126/jtiik.201851610

Muhammad Tesar Sandikapura, & Eko Maulana Sukendar. (2018). Sub Sistem Informasi Pembayaran Uang Semester di Sekolah Tinggi Kesehatan, Ilmu Kencana, Mitra Tasikmalaya, Kampus Sandikapura. Jurnal Teknik Informatika, 6(2), 41–50. Retrieved from http://jurnal.stmik-dci.ac.id/index.php/jutekin/article/download/332/406

Munawar, Z., Fudsyi, M. I., & Musadad, D. Z. (2019). Perancangan Interface Aplikasi Pencatatan Persediaan Barang Di Kios Buku Palasari Bandung Dengan Metode User Centered Design Menggunakan Balsamiq Mockups. Jurnal Informatika, 6(2), 10–20.

Munir, R. (2019). Kriptografi. Penerbit Informatika.

Neamah, I. (2018). A Modification of ElGamal Cryptosystem using Statistical Methods. 133(May), 20–25.

Okeyinka, A. E., Alao, O., Gbadamosi, B., & Ogundokun, R. O. (2018). Application of SHA-256 in Formulation of Digital Signatures of RSA and Elgamal Cryptosystems. Operations Research and Information Engineering, 1(October 2008), 61–66.

Putera Utama Siahaan, A., Elviwani, E., & Oktaviana, B. (2018). Comparative Analysis of RSA and ElGamal Cryptographic Public-key Algorithms. https://doi.org/10.4108/eai.23-4-2018.2277584

Rachmawati, D., Tarigan, J. T., & Ginting, A. B. C. (2018). A comparative study of Message Digest 5(MD5) and SHA256 algorithm. Journal of Physics: Conference Series, 978(1). https://doi.org/10.1088/1742-6596/978/1/012116

Rusdiana, L. (2018). Dynamic Systems Development Method dalam membangun Aplikasi Data Kependudukan Pada Kelurahan Rantau Pulut. Jurnal Transformatika, 16(1), 84. https://doi.org/10.26623/transformatika.v16i1.859

Saputra, I., & Nasution, S. D. (2019). Analisa Algoritma SHA-256 Untuk Mendeteksi Orisinalitas Citra Digital. Prosiding Seminar Nasional Riset Information Science (SENARIS), 1(September), 164. https://doi.org/10.30645/senaris.v1i0.20

Sarwandy, M. H. A., Arliansyah, J., & Fitriani, H. (2019). The Analysis of Ability to Pay (ATP) and Willingness to Pay (WTP) on Light Rail Transit (LRT) Tariff in Palembang. Journal of Physics: Conference Series, 1198(8). https://doi.org/10.1088/1742-6596/1198/8/082023

Schneier, B. (2007). Schneier's Cryptography Classics Library: Applied Cryptography, Secrets and Lies, and Practical Cryptography.

Sembiring, H., Utara, S., Manik, F. Y., Utara, S., & Utara, S. (2019). Penerapan Algoritma Secure Hash Algorithm (SHA) Keamanan Pada

Citra. 4(1), 33–36.

Suendri. (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan). Jurnal Ilmu Komputer Dan Informatika,3(1),1–9. Retrieved from http://jurnal.uinsu.ac.id/index.php/algoritma/article/download/3148/1871

Warnilah, A. I., & Nugraha, S. N. (2018). Komparasi Algoritma Kriptografi Elgamal Dan Caesar Cipher Untuk Enkripsi Dan Dekripsi Pesan. 3(2), 243–252.

Yalisa, N., Arhami, M., & Azhar. (2018). Algoritma Elgamal dengan Pertukaran Kunci Diffie Hellman pada Aplikasi Keamanan Citra Sidik Jari Berbasis Android. 2(1), 2598–3954.