Available *online* at : http://bit.ly/InfoTekJar

InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan

ISSN (Print) 2540-7597 | ISSN (Online) 2540-7600

Kriptografi

Pengamanan E-Dokumen Berbasis Steganografi Dengan Kombinasi *Advanced Encryption Standard* (AES) 128 Bit

Mohammad Imron, Aditiya Pratama

Program Studi Informatika Fakultas Ilmu Komputer Universitas Amikom Purwokerto, Jawa Tengah, Indonesia 53123

KEYWORDS

Kriptografi, Steganografi, *Advanced Encryption Standard* (AES), *Rapid Application Development* (RAD), E-Dokumen

CORRESPONDENCE

Phone: +62 81229672129

E-mail: imron@amikompurwokerto.ac.id

A B S T R A C T

Keamanan data informasi menjadi permasalahan yang cukup serius, informasi yang dianggap rahasia memerlukan keamanan yang tinggi sehingga apabila terdapat pihak ketiga yang dapat mengakses data penyimpanan tanpa seizin pemilik. Kasus yang terjadi terhadap data pribadi sebanyak 1207 di tahun 2016 dengan *crime clearance* 669 sehingga memiliki persentasi 57% kasus yang terjadi. Tujuan penelitian membuat sistem keamanan data dengan mengimplementasikan steganografi kombinasi enkripsi Advanced Encryption Standard (AES) 128 bit. Pada penelitian ini menggunakan tahapan metode pengembangan sistem Rapid Application Development (RAD). Dari hasil penelitian ini telah dilakukan pengujian dengan berbagai pengujian untuk mengenkripsikan file dokumen yang dienkripsi dan disembunyikan pada sebuah *image* sehingga keamanan data informasi dapat terjaga keamanannya dengan baik.

INTRODUCTION

Keamanan e-dokumen merupakan permasalahan yang cukup serius dengan seiring meningkatnya perkembangan komunikasi, informasi yang dianggap rahasia memerlukan keamanan yang tinggi sehingga apabila terdapat pihak ketiga yang dapat mengakses penyimpanan data tersebut tanpa seizin pemiliki [1].

Rekaman informasi yang tersimpan dalam dokumen merupakan hal penting bagi pemilik informasi tersebut dan juga hal yang penting bagi orang yang menginginkan sebuah informasi, mengingat pentingnya hal tersebut saat ini informasi telah menjadi target serangan oleh para pencuri informasi. Karenanya keamanan suatu informasi menjadi sesuatu yang harus dijaga dengan baik, pengamanan informasi pada prinsipnya berfungsi untuk melindungi informasi agar siapapun yang tidak berhak tidak dapat membaca, mengubahnya atau menghapus informasi tersebut. Model keamanan data yang dapat dilakukan menggunakan salah satu teknik penyandian data dalam bentuk enkripsi dan deskripsi [2]. Teknik penyandian tersebut dikenal dengan teknik kriptografi yang memiliki peranan sangat besar dalam dunia keamanan data [3].

Berdasarkan data dari Laporan Kinerja Instansi Pemerintah (LKIP) Ditreskrimsus Polda Metro Jaya dari tahun 2014 hingga tahun 2016 mengalami perbedaan dalam jumlah kasus kejahatan *cyber crime* termasuk didalamnya kasus pencurian terhadap data pribadi seperti *file e-document* yang mana kasus ditahun 2014 *crime total* sebanyak 1225 dengan *crime clearance* 790 sehingga memiliki persentasi 64%, di tahun 2015 sendiri *crime total* sebesar 1569 untuk *crime clearance* sebesar 851 dengan persentasi 54 % kasus, dan di tahun 2016 *crime total* sebesar 1207, sedang untuk *crime clearance* sebanyak 699 dengan persentasi 57%, dengan demikian bahwa kasus *cybercrime* dari data tersebut keamanan informasi menjadi sangat penting di era teknologi yang semakin meningkat. (Sumber: Bagbinopsal Dit Reskrimsus Polda Metro Jaya).

Salah satu solusi yang dapat digunakan yaitu dengan menggunakan teknik steganografi, steganografi sendiri merupakan seni komunikasi rahasia antara dua pihak dimana pesan disembunyikan dalam sebuah objek yang tampak polos seperti gambar, audio, video dan teks. Tujuan lain dari steganografi juga dapat menyembunyikan data penting didalam file lain, sehingga pihak yang dimaksudkan untuk menerima pesan dapat mengetahui keberadaan pesan rahasia tersebut. Agar pesan tersebut lebih aman lagi maka teknik steganografi dapat dikombinasikan dengan algoritma enkripsi, salah satu algoritma enkripsi yang paling umum digunakan adalah *Advanced Eccryption Standar* (*AES*) [4]. Menurut [5] AES menjadi kunci standar enkripsi data saat ini sebagai cara untuk melindungi data dari bentuk serangan yang dapat berguna untuk mengamankan data dan tingkat kesulitan hingga kriptografi AES menghasilkan cipher 16 bit

https://doi.org/10.30743/infotekjar.v6i2.4346

[6].

LANDASAN TEORI

A. Kriptografi

Kriptografi merupakan teknik dan juga ilmu untuk menjaga kerahasian dari pesan dengan cara menyamarkanya menjadi bentuk yang tersandi yang tidak memiliki makna, teknik sandi tersebut hanya berhak dibaca oleh yang berhak saja untuk dapat membacanya [7]. Pesan asli yang belum disamarkan disebut *plaintext* sedangkan pesan yang telah disamarkan disebut *chipertext*, sehingga proses penyamarannya disebut dengan enkripsi dan untuk proses membuka penyamarannya disebut deskripsi [8],.

Teknik enkripsi yang diterapkan pada data diacak menggunakan kunci yang dienkripsi menjadi sesuatu yang sulit dibaca orang yang tidak memiliki kunci dekripsi, [9] untuk memastikan kerahasiaan transmisi dapat menggunakan sebuah algoritma atau sandi untuk mengekripsi data menjadi pesan jelas, sehingga seseorang yang tidak tahu algoritma ini akan menemukan data transmisi tidak bermakna atau tidak mengetahui akan pesan tersbut

B. Steganografi

Steganografi merupakan seni untuk menyembunyikan pesan di dalam media digital sedemikian rupa agar orang lain tidak dapat menyadarinya bahwa ada suatu pesan di dalam media tersebut. Kata steganografi sendiri berasal dari yunani *steganos* yang berarti tersembunyi dan *grephien* sendiri artinya menulis sehingga kurang lebih arti dari steganografi menulis pesan yang terselubung [10], [11] yang diimplementasikan dengan teknik tunggal ataupun gabungan seperti steganografi.

C. *Advanced Encryption Standard* (*AES*)

Advanced Encryption Standard (*AES*) merupakan standar enkripsi kunci simetris yang pada awalnya diterbitkan dengan algoritma *Rijindael*, algoritma tersebut dikembangkan oleh dua kriptografer Belgia yaitu Joan Daemen dan Vincent Rijmen. Algoritma AES dipublikasikan oleh NIST (*National Institute of Standard and Technology*) pada tahun 2001 yang merupakan blok kode simetris untuk menggantikan DES (*Data Encryption Standard*), algoritma AES sendiri memiliki tiga pilihan kunci yang tipenya bervariasi yaitu tipe AES-128, AES-192, dan AES-256 [12].

Pada penelitian [13] kriptografi AES 128 dengan penerapan kriptografi AES 128 berupa URL yang telah diimplementasikan dapat mengamankan bentuk serangan *SQL Injection*. Kasus tersebut sebuah tindakan memasukan kode khusus pada URL website yang dapat merubah isi database.

E. Proses Enkripsi

Proses enkripsi algoritma AES 128 terdiri dari 4 jeis transformasi *bytes*, yaitu *SubBytes, ShiftRows, MixColumns*, dan *AddRoundkey*. Dari proses awal enkripsi *state* akan mengalami sebuah transformasi *byte AddRoundKey* dan transformasi akan diikuti *SubBytes, ShiftRows, MixCoulums,* dan *AddRoundKey* secara berulang sebanyak Nr (Nilai *round*). Proses tersebut dikenal dengan *round function*. Pada *round* yang terakhir dilakukan berbeda dari sebelumnya, dimana *state* tidak mengalami transformasi *MixColumns* dan *source code* ekripsi dapat dilihat pada gambar dibawah ini.

```
FileStream fsInput = new FileStream(sInputFilename, FileMode.Open, FileAccess.Read);

FileStream fsEncrypted = new FileStream(sOutputFilename, FileMode.Create, FileAccess.Write);
DESCryptoServiceProvider DES = new DESCryptoServiceProvider();
DES.Key = ASCIIEncoding.ASCII.GetBytes(sKey);
DES.IV = ASCIIEncoding.ASCII.GetBytes(sKey);
ICryptoTransform desencrypt = DES.CreateEncryptor();
CryptoStream cryptostream = new CryptoStream(fsEncrypted, desencrypt, CryptoStreamMode.Write);

byte[] bytearrayinput = new byte[fsInput.Length];
fsInput.Read(bytearrayinput, 0, bytearrayinput.Length);
cryptostream.Write(bytearrayinput, 0, bytearrayinput.Length);
cryptostream.Close();
fsInput.Close();
fsEncrypted.Close();
```

Gambar 1. *Source Code* Enkripsi

F. Proses Dekripsi

Pada proses dekripsi AES-128 diperlukan transformasi *chiper* dengan cara dibalik sehingga dapat menghasilkan *inverse cipher* dengan tahapan *InvShiftRows, InvSubBytes, InvMixColumns*, dan *AddRoundKey*, *source code* Deskripsidapat dilihat pada gambar dibawah ini.

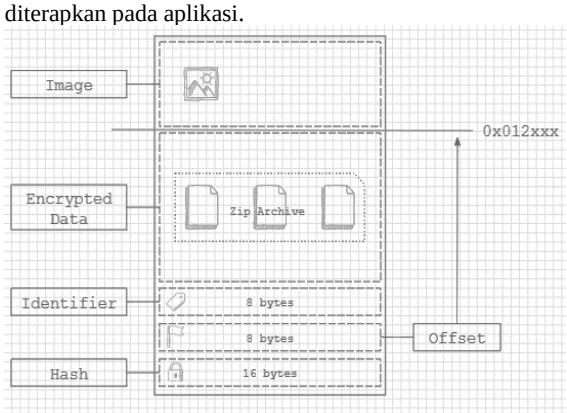
```
using (var DES = new DESCryptoServiceProvider())
{
    DES.Key = Encoding.Default.GetBytes(sKey);
    DES.IV = Encoding.Default.GetBytes(sKey);
    using (var desdecrypt = DES.CreateDecryptor())
    {
        using (var fsread = new FileStream(sInputFilename, FileMode.Open, FileAccess.Read))
        {
            using (var cryptostreamDecr = new CryptoStream(fsread, desdecrypt, CryptoStreamMode.Read))
            {
                using (var fswrite = new FileStream(sOutputFilename,
                    FileMode.Create, FileAccess.Write, FileShare.Write))
                {

                    cryptostreamDecr.CopyTo(fswrite);
                }
            }
        }
    }
}
```

Gambar 2. *Source Code* Dekripsi

METODOLOGI PENELITIAN

Pada penelitian ini penulis menggunakan konsep yang telah penulis kembangkan dari hasil teknik steganografi *end of file* yang dikombinasikan dengan enkripsi AES 128 bit yang diterapkan pada aplikasi.



Gambar 3. Konsep Aplikasi

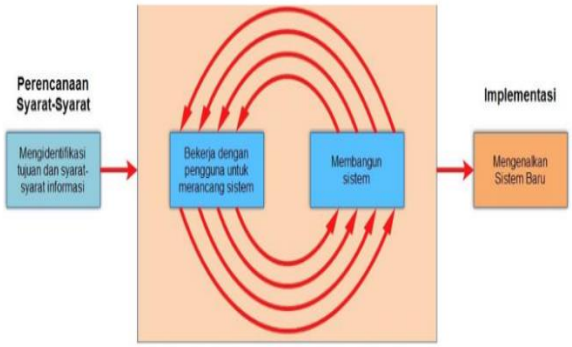
Pada konsept yang tergambar pada gambar 2.1 terdapat beberapa blok data atau informasi yang dijelaskan sebagai berikut:

a. *Image*, merupakan barisan data dari gambar yang akan digunakan sebagai media menyembunyikan data.
b. *Encrypted Data*, berisi data yang merupakan *Zip Archive* yang dienkripsi menggunakan AES 128 bit [14].

 https://doi.org/10.30743/infotekjar.v6i2.4346

c. *Idenfitier*, merupakan data berukuran 8 *bytes* yang berisikan identitas aplikasi yang digunakan untuk melakukan pengecekan validasi file.
d. *Offset*, merupakan data berukuran 8 *bytes* yang berisikan lokasi *end of file* dari *image*yang digunakan untuk mendapatkan posisi awal dari *encrypted Data* pada proses dekripsi.
e. *Hash*, [15],[14] merupakan data berukuran 16 *bytes* yang berisikan MD5 *hash* dari *password* yang dikombinasikan dengan *salt*, digunakan untuk memverifikasi *password* yang diinputkan pada saat proses dekripsi.

Sedangkan metode pengembangan sistem yang digunakan adalah RAD (*Rapid Application Development*) yang meimiliki fase-fase melakukan perencanaan syarat kebutuhan sistem, yang melibatkan pengguna untuk merancang sistem dan membangun sistem secra berulang-ulang sehingga tahap terakhirnya dapat diimplementasikan. Metode RAD sangat mementingkan keterlibatan pengguna dalam proses analisis dan perancangannya yang dapat memenuhi kebutuhan pengguna dengan baik yang dapat menikatkan tingkat kepuasan pengguna sistem secara keseluruhan [16]. Model proses pembangunan perangkat lunak yang tergolong dalam teknik *easy values*, *typical realistic values*, *extreme values* dan *illegal values* [17].



Gambar 4. Metode RAD [17]

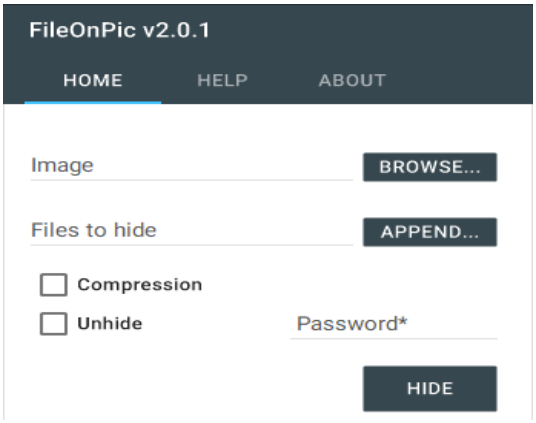
HASIL DAN PEMBAHASAN

Penelitian ini bertujuan untuk menghasilkan sebuah aplikasi yang berguna untuk melakukan pengamanan terhadap *file* e-dokumen, proses pengaman yang digunakan dalam penelitian ini menggunakan steganografi kombinasi enkripsi AES (*Advanced Encryption Standard*). File e-dokumen dalam penelitian ini berektensi *doc, docx, ppt, pptx, pdf, xls, xlsx, jpg, png, gif*, dan *file* yang digunakan dalam proses steganografi.

A. Aplikasi File On Pic

Aplikasi FileOnPic v2.0.1 merupakan sebuah aplikasi yang diharapkan akan dikembangkan dengan konsep menambahkan metode yang lebih baik sesuai dengan perkembangan teknologi informasi yang dibutuhkan. Aplikasi ini untuk mengamankan data rahasia dengan mengimplementasikan teknik steganografi yang dikombinasikan dengan algoritma enkripsi AES 128 bit sehingga data yang disembunyikan memiliki tingkat keamanan yang tinggi.

Teknik steganografi yang digunakan adalah steganografi *end of file* sehingga memungkinkan untuk mengamankan banyak data atau file sekaligus, fitur *password* pada aplikasi sendiri digunakan untuk meningkatkan tingkat keamanan dan privasi pemilik data. Dan pada aplikasi juga terdapat fitur *compression* yang digunakan untuk meminimalkan penambahan ukuran file hasil steganografi sehingga memperkecil tingkat kecurigaan pihak ketiga atau penyusup, berikut tampilan aplikasi:



Gambar 5. Antar Muka Tampilan Aplikasi

B. Pengujian Enkripsi

Pengujian ini dilakukan untuk mengetahui apakah data yang telah dikompresi dapat menyisipkan ke dalam *image* sehingga berjalan dengan semestinya seperti apa yang diharapkan oleh penulis.

TABEL I. PENGUJIAN *EMBEDDING*

No	Pengujian *Embedding*			
	Nama Gambar	Ukuran Gambar	Ukuran File	Status
1.	Pengujian 1	800 x 533	4,91 Mb	Berhasil
2.	Pengujian 2	700 x 550	228 Kb	Berhasil
3.	Pengujian 3	640 x 420	230 Kb	Berhasil
4.	Pengujian 4	3000 x 2000	411 Kb	Berhasil
5.	Pengujian 5	640 x 385	2,10 Mb	Berhasil
6.	Pengujian 6	640 x 455	1,56 Mb	Berhasil
7.	Pengujian 7	720 x 480	1,68 Mb	Berhasil
8.	Pengujian 8	700 x 393	3,89 Mb	Berhasil
9.	Pengujian 9	1000 x 667	1,59 Mb	Berhasil
10.	Pengujian 10	600 x 450	1,16 Mb	Berhasil

Dari hasil pengujian 1 hingga pengujian 10 yang telah dilakukan menunjukan bahwa penyisipan data ke dalam *image* berhasil dilakukan dengan baik, dari pengujian tersebut dapat disimpulkan bahwa data yang disisipkan menyesuaikan resolusi *image* dari citra itu sendiri.

C. Pengujian Kompresi

Hasil pengujian ini bertujuan untuk mengetahui seberapa besar kompresi data itu berhasil, pengujian enkripsi dengan *compression* dapat dilihat bahwa hasil dari stego image yang disisipkan berbagai ukuran file maka menghasilkan *ciphertext* dengan ukuran gambar yang lebih besar dari aslinya, akan tetapi gambar hasil ekstrasi tidak mengalami perubahan dari sisi tampilan. Hasil pengujian tersebut dapat dilihat pada tabel dibawah ini.

TABEL 2. PENGUJIAN *COMPRESSION*

No	Pengujian *Compression*		
	Ukuran Gambar	Ukuran Asli (Kb)	Hasil Kompresi (Kb)
1.	800 x 533	5.184	5.128
2.	700 x 550	294	282
3.	640 x 420	299	287
4.	3000 x 2000	1.171	1.088
5.	640 x 385	2.190	1.950
6.	640 x 455	1.664	1.638
7.	720 x 480	1.780	1.616
8.	700 x 393	4.045	3.854
9.	1000 x 667	1.821	1.535
10.	600 x 450	1.244	1.074

Sedangkan dengan pengujian *compression* dari beberapa pengujian bahwa file setelah di ekstrasi dari stego gambar menghasilkan gambar lebih besar dibandingkan dengan tanpa menggunakan *compression*, dari hasil pengujian juga sama bahwa tampilan gambar tidak ada perubahan dari sisi tampilan dan hanya ukuran gambar menjadi besar dari ukuran aslinya.

D. Pengujian Deskripsi

Pengujian ini dilakukan untuk mengetahui apakah data yang disembunyikan pada *image* dapat dikembalikan kembali atau tidak. Pengujian ini merupakan ekstraksi dari file yang tersembunyi di dalam *image*, berikut hasil dari pengujian deskripsi yang telah dilakukan.

TABEL 3 PENGUJIAN *DESKRIPSI*

No	Pengujian *Compression*		
	Ukuran Gambar	Ukuran Asli (Kb)	Hasil Kompresi (Kb)
1.	800 x 533	5.184	5.128
2.	700 x 550	294	282
3.	640 x 420	299	287
4.	3000 x 2000	1.171	1.088
5.	640 x 385	2.190	1.950
6.	640 x 455	1.664	1.638
7.	720 x 480	1.780	1.616
8.	700 x 393	4.045	3.854
9.	1000 x 667	1.821	1.535
10.	600 x 450	1.244	1.074

Sedangkan dengan pengujian *compression* dari beberapa pengujian bahwa file setelah di ekstrasi dari stego gambar menghasilkan gambar lebih besar dibandingkan dengan tanpa menggunakan *compression*, dari hasil pengujian juga sama bahwa tampilan gambar tidak ada perubahan dari sisi tampilan dan hanya ukuran gambar menjadi besar dari ukuran aslinya.

KESIMPULAN DAN SARAN

Dari hasil percobaan yang telah dilakukan maka dapat disimpulkan bahwa implementasi steganografi dengan kombinasi enkripsi AES 128 bit telah berhasil dan berjalan dengan baik, pengujian terhadap beberapa sample membuktikan bahwa metode keamanan mengkombinasikan steganografi dengan enkripsi AES 128 bit dalam proses *encode* menjadi file gambar dengan ukuran file yang lebih besar namun tidak merubah dari komposisi fisik gambar dengan hasil pemisahan file dalam bentuk file terkompres dalam gambar, mengingat secara kasat mata bahwa file dalam bentuk file gambar yang tidak diketahui bahwa didalamnya tersimpan pesan rahasia. Sehingga efektivitas steganografi dengan kombinasi AES 128 bit dalam pengiriman pesan file rahasia cukup baik dalam keamanan sistem informasi.

DAFTAR PUSTAKA

[1] Sumarno, "Analisis Kinerja Kombinasi Algoritma Message-Digest Algorithm 5 (MD5), Rivest Shamir Adleman (RSA) dan Rivest Cipher 4 (RC4) pada Keamanan E-Dokumen," *JUSIKOM PRIMA (Jurnal Sist. Inf. Ilmu Komput. Prima)*, vol. 2, no. 1, pp. 41–48, 2018.

[2] S. P. Gochhayat *et al.*, "Reliable and secure data transfer in IoT networks," *Wirel. Networks*, vol. 26, no. 8, pp. 5689–5702, 2020, doi: 10.1007/s11276-019-02036-0.

[3] M. K. Harahap, "Analisis Perbandingan Algoritma Kriptografi Klasik Vigenere Cipher Dan One Time Pad," *InfoTekJar (Jurnal Nas. Inform. dan Teknol. Jaringan)*, vol. 1, no. 1, pp. 61–64, 2016, doi: 10.30743/infotekjar.v1i1.43.

[4] N. Aleisa, "A comparison of the 3DES and AES encryption standards," *Int. J. Secur. its Appl.*, vol. 9, no. 7, pp. 241–246, 2015, doi: 10.14257/ijsia.2015.9.7.21.

[5] V. Chari, "International Journal of Innovative Technology and Exploring Engineering (IJITEE) | Enhanced Reader," *Int. J. Innov. Technol. Explor. Eng.*, vol. 8, no. 12, 2019.

[6] M. Imron, I. Ardiansyah, and D. Suhartono, "Implementasi Pengamanan Data Koperasi Menggunakan Algoritma Advanced Encryption Standard (Aes)," *CITISEE 2016 Proc.*, pp. 37–40, 2016, [Online]. Available: https://citisee.amikompurwokerto.ac.id/assets/proceedings/paper/8__Amikom_Purwokerto_Implementasi_Pengamanan_Data_Koperasi_Menggunakan_Algoritma_Advanced_Encryption_Standard_(Aes).pdf.

[7] M. H. Arif and A. Z. Fanani, "Kriptografi Hill Cipher Dan Least Significant Bit Untuk Keamanan Pesan Pada Citra," *CSRID (Computer Sci. Res. Its Dev. Journal)*, vol. 8, no. 1, p. 60, 2016, doi: 10.22303/csrid.8.1.2016.60-72.

[8] S. M. Ankita Verma, Paramita Guha, "Comparative study of different cryptographic algorithms," *Int. J. Emerg. Trends Technol. Comput. Sci.*, vol. 5, no. 2, pp. 58–63, 2016, [Online]. Available: https://www.researchgate.net/publication/306286725_Comparative_Study_of_Different_Cryptographic_Algorithms/link/57b6ca9208ae2fc031fd6cbc/download.

[9] Sentot Kromodimoeljo, *Teori & Aplikasi Kriptografi*. 2010.

[10] D. Darwis, "Implementasi Teknik Steganografi Least Significant Bit (LSB) Dan Kompresi Untuk Pengamanan Data Pengiriman Surat Elektronik," *J. Teknoinfo*, vol. 10, no. 2, p. 32, 2016, doi: 10.33365/jti.v10i2.8.

 https://doi.org/10.30743/infotekjar.v6i2.4346

[11] E. H. Rachmawanto, K. Prasetyo, C. A. Sari, D. R. I. M. Setiadi, and N. Rijati, "Secured PVD video steganography method based on AES and linear congruential generator," *2018 Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2018*, no. November, pp. 163–167, 2018, doi: 10.1109/ISRITI.2018.8864466.

[12] Y. Asimi, A. Asimi, A. Guezzaz, Z. Tbatou, and Y. Sadqi, "Unpredictable cryptographic primitives for the Robust Wireless Network Security," *Procedia Comput. Sci.*, vol. 134, no. February 2020, pp. 316–321, 2018, doi: 10.1016/j.procs.2018.07.178.

[13] H. Wijaya, "Implementasi Kriptografi AES-128 Untuk Mengamankan URL (Uniform Resource Locator) dari SQL Injection," *Akad. J.*, vol. 17, no. 1, pp. 8–13, 2020.

[14] P. Wijaya, M. Damanik, P. Hartati, and I. Gunawan, "Implementasi Enkripsi Dan Deskripsi Data Siak (Sistem Informasi Administrasi Kependudukan) Menggunakan Algoritma DES, AES, dan MD5," *TECHSI*, vol. 12, 2016.

[15] D. P. Precilia and A. Izzuddin, "Aplikasi Tanda Tangan Digital (Digital Signature) Menggunakan Algoritma Message Digest 5 (MD5)," *Energy*, vol. 5, no. 1, pp. 14–19, 2016.

[16] I. Sommerville, *Software Engineering (9th ed.; Boston, Ed.). Massachusetts: Pearson Education.* 2011.

[17] G. B. S. H. J. Rosenblatt, *Systems Analysis and Design*, vol. 148. 2012.