Jurnal Teknik Informatika (JUTIF)
Vol. 5, No. 3, June 2024, pp. 855-864

DOI: https://doi.org/10.52436/1.jutif.2024.5.3.2038
p-ISSN: 2723-3863
e-ISSN: 2723-3871

ANALYSIS AND IMPLEMENTATION OF AES-128 ALGORITHM IN SUKAHARJA KARAWANG VILLAGE SERVICE SYSTEM

Fariz Duta Nugraha[*1], Kiki Ahmad Baihaqi[2], Hilda Yulia Novita[3], Amril Mutoi Siregar[4]

[1,2,3,4]Information Engineering, Faculty of Computer Science, Universitas Buana Perjuangan Karawang, Indonesia
Email: [1]if20.fariznugraha@mhs.ubpkarawang.ac.id, [2]kikiahmad@ubpkarawang.ac.id,
[3]hilda.yulia@ubpkarawang.ac.id, [4]amrilmutoi@ubpkarawang.ac.id

(Article received: May 03, 2024; Revision: May 12, 2024; published: June 05, 2024)

Abstract

Data security in databases is needed in the industrial era 4.0 to prevent attacks and unwanted things from happening, one of the biggest cases that has been widely reported is data leakage, in this study aims to implement and analyze the Advanced Encryption Standard Algorithm, one of the data security algorithms with a block chiper type that has 4 transformations (SubByte, ShiftColumn, MixColumn, AddRoundKey), or what we usually call the Cryptography method. Cryptography is a method that is often used to secure important data in databases, in this article the Advanced Encryption Standard Algorithm is used to secure citizen data and family card data in the Sukaharja Karawang Village service system. The method in this research is the observation method, the data is obtained from each head of the neighborhood in Sukaharja Karawang Village with the permission of the head of Sukaharja Karawang Village. Citizen data and family cards were encrypted and analyzed for resource requirements in storing encryption results and time in returning and displaying original data. The results of the analysis obtained the amount of resources required 1.5MB to store family card data, which before encryption required 352KB. Citizen data requires a resource of 6.5MB, before encryption it takes 1.5MB. As for the AES resilience test stage using the Bruteforce attack method with the help of Hashcat software version 6.2.5 with 4 trial processes, One encrypted address data was taken for this test, but out of 4 attempts none of them showed that the data could be cracked.

Keywords: *Advanced Ecryption Standard, Bruteforce Attack, Cyber Security, Cryptography, Data Security.*

ANALISIS DAN IMPLEMENTASI ALGORITMA AES-128 PADA SISTEM PELAYANAN DESA SUKAHARJA KARAWANG

Abstrak

Pengamanan data pada basis data diperlukan dalam era industri 4.0 untuk mencegah serangan dan hal yang tidak diinginkan terjadi, salah satu kasus terbesar yang sudah banyak diberitakan yaitu kebocoran data, dalam penelitian ini bertujuan dalam implementasi dan analisis Algoritma *Advanced Encryption Standard*, salah satu Algoritma pengamanan data dengan tipe *chiper block* yang mempunyai 4 transformasi (*SubByte, ShiftColumn, MixColumn, AddRoundKey*), atau biasa akita sebut dengan metode Kriptografi. Kriptografi merupakan metode yang sering dipakai untuk mengamankan data penting pada basis data, dalam artikel ini digunakan Algoritma *Advanced Encryption Standard* untuk mengamankan data warga dan data kartu keluarga di dalam sistem pelayanan Desa Sukaharja Karawang. Metode dalam penelitian ini merupaka metode observasi, data didapatkan dari setiap ketua rukun tetangga di Desa Sukaharja Karawang atas izin kepala Desa Sukaharja Karawang. Data warga dan kartu keluarga dienkripsi dan dianalisis kebutuhan *resource* dalam menyimpan hasil enkripsi dan waktu dalam mengembalikan lalu menampilkan data asli. Hasil analisis didapatkan besaran *resource* yang dibutuhkan 1,5MB untuk menyimpan data kartu keluarga, dimana sebelum dienkripsi dibutuhkan 352KB. Pada data warga dibutuhkan *resource* sebesar 6.5MB, sebelum dienkripsi dibutuhkan 1.5MB. Adapun dalam tahapan uji ketahan AES menggunakan metode serangan *Bruteforce* dengan bantuan *software* Hashcat versi 6.2.5 dengan 4 proses percobaan, satu data alamat terenkripsi diambil untuk pengujian ini, namun dari 4 proses percobaan tidak ada menunjukan data tersebut bisa dipecahkan.

Kata kunci: *Advanced Encryption Standard, Keamanan Data, Keamanan Siber, Kriptografi, Serangan Bruteforce.*

1. PENDAHULUAN

Pada era industri 4.0 saat ini, teknologi diterapkan hampir di setiap bidang baik itu pemerintahan, pariwisata, bisnis dan aktivitas sehari-hari, semua kalangan menikmati manfaat dari teknologi. Namun, data diri harus didaftarkan pada setiap layanan yang diberikan pada internet, data ataupun informasi yang didaftarkan mungkin saja tidak aman, data ataupun informasi pribadi kita mungkin saja dicuri oleh pihak tidak bertanggung jawab lalu disalah gunakan [1]. Menurut penelitian yang dilakukan oleh Samya Al Busafi, data yang diperoleh dari surel ataupun dari *web* tidak ada jaminan bahwa data tersebut asli. Maka dari itu, teknik Kriptografi merupakan upaya dalam mengamankan data atau informasi yang digunakan saat pertukaran data [2]. Pengamanan data menggunakan teknik Kriptografi perlu dilakukan dalam menjaga data pada *database* dari berbagai serangan, pada penelitian Dhirendra dan kawan-kawan menyebutkan dalam *Cloud Computing* perlindungan data merupakan masalah utama, maka dari itu diperlukan pengamanan data [3]. Upaya pengamanan data pada *database* dilakukan dalam penelitian Tawfiq dan kawan-kawan, dalam penelitiannya dilakukan pengamanan menggunakan Algoritma *Advanced Encryption* Standard (AES) untuk pengamanan di lingkungan sistem *E-commerce* [4]. Pada penelitian Faturrahmad, Algoritma AES digunakan dalam pengamanan data pada *website* dengan hasil kecepatan Algoritma tersebut memiliki kecepatan yang baik [5]. Algoritma AES juga dilakukan dalam tokenisasi untuk pencegahan serangan *SQL Injection* dari protocol HTTP yang dilakukan dalam penelitian Farras dan kawan-kawan [6]. Dalam penelitian Mohammad Al-Mashhadani disebutkan bahwa Algoritma AES sudah dipercaya dan dijadikan standar oleh pemerintahan *United States* dan banyak institusi lainnya, terlebih Algoritma AES juga disebutkan dalam penelitian tersebut enam kali lebih cepat dibandingkan Algoritma TripleDES dan banyak diterima oleh standarisasi enkripsi di dunia [7]. Algoritma AES memiliki tingkat keamanan yang tinggi dengan kompleksitas waktu yang lebih rendah dibandingkan dengan Algoritma RSA, sifat Algoritma AES yang fleksibel, mudah diimplementasikan dan kebutuhan memori yang sedikit merupakan kelebihan dari Algoritma AES yang disebutkan dalam penelitian Sana Fatima [8]. Terlepas dari keunggulan Algoritma AES, diperlukan penanganan masalah kebocoran data pribadi warga di Indonesia, hal tersebut merupakan urgensi, disebutkan dalam penelitian Hezkiel [9]. Pada bulan Mei 2021, masyarakat Indonesia dikejutkan oleh kabar yang menyebutkan dugaan kebocoran data BPJS. Sebanyak 279 juta data pribadi, termasuk nomor kartu keluarga, tanggungan, dan status pembayaran, diduga bocor dengan harga jual sekitar 80 juta rupiah [10]. Pada bulan April tanggal 17 tahun 2020, terdapat berita kebocoran data yang dialami oleh marketplace Tokopedia, dikabarkan bahwa 15 juta akun telah diretas, namun setelah ditelusuri lebih lanjut, total akun teretas menjadi 91 juta pengguna, ditambah dengan 7 juta akun *merchant* [11]. Pada penelitian Berita Estu Widodo menyebutkan bahwa kemanan maupun kerahasiaan dokumen menjadi salah satu aspek penting dalam dunia informasi, terlebih di dalam instansi pemerintahan [12]. Mengingat dalam penelitian ini tertuju pada lingkungan pelayanan Desa, maka pelaksanaan *e-government* harus memperhatikan dari segi aspek keamanan data pribadi atau *privacy* yang bersifat sensitif, siapa saja bisa menyalah gunakan data tersebut yang berakibat merugikan orang lain, maka dari itu, jaminan perlindungan data pribadi harus memiliki kelayakan seperti yang disebutkan dalam penelitian Bunga [13]. Perlindungan data pribadi merupakan hak atas setiap warga seperti yang dirujuk pada Pasal 28 G Ayat (1), setiap orang berhak dalam memutuskan kepada siapa saja mereka membagikan data mereka [14]. Merujuk pada penelitian Gatot, minat seseorang dalam menggunakan layanan teknologi salah satunya kepercayaan, oleh karena itu, pengamanan data pribadi perlu diperhatikan [15]. Dari uraian di atas maka, penulis melakukan penelitian ini dengan analisis dan implementasi Algoritma *Advanced Encryption Standard* (AES) pada sistem pelayanan Desa Sukaharja Karawang, diharapkan dari implementasi pengamanan data, data warga menjadi lebih aman dari serangan *hacker* dan hasil dari analisis implementasi AES-128 pada sistem pelayanan Desa Sukaharja Karawang dapat memberikan informasi yang berguna bagi penulis maupun peneliti lainnya.

2. METODE PENELITIAN

A. Objek Penelitian

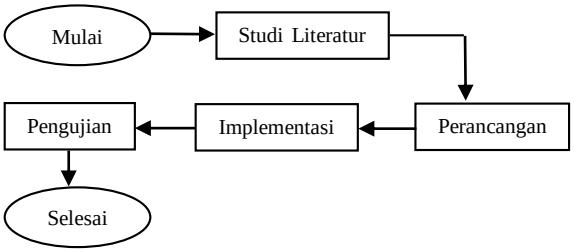
Data warga Desa Sukaharja Karawang digunakan dalam penelitian ini, data tersebut dienkripsi menggunakan Algoritma AES-128, lalu dimasukan ke dalam *database*. Data tersebut merupakan salinan dari kartu keluarga berdomisili Sukaharja Karawang yang didapatkan dengan observasi ke setiap ketua rukun tetangga Desa Sukaharja atas izin dan sepengetahuan kepala Desa Sukaharja Karawang. Dari data yang didapatkan, penelitian ini bertujuan dalam implementasi Algoritma AES-128 untuk pengamanan data tersebut, lalu menguji hasil enkripsi menggunakan metode *Bruteforce Attack* dengan perangkat lunak Hashcat versi 6.2.5 serta menguji efisiensi Algoritma AES-128 menggunakan Chrome DevTools yang disediakan oleh Google. Berikut aktivitas, lokasi dan waktu penelitian yang dilampirkan pada tabel 1.

Tabel 1 Lokasi dan Waktu Penelitian

Aktivitas	Lokasi	Waktu
Studi Literatur	Lab UBP	1 Juni 2023 - 1 Juli 2023
Analisis Kebutuhan	Lab UBP	02 Juli 2023 - 9 Juli 2023
Pengumpulan Data	Desa Sukaharja Karawang	10 Juli 2023 - 10 Agustus 2023

B. Prosedur Penelitian

Dalam penelitian ini, terdapat lima langkah prosedur, diawali dengan studi literatur terhadap jurnal yang membahas tentang pengamanan data. Setelah itu dilakukan pengumpulan data, kemudian perancangan dan dilanjut ke prosedur implementasi. Dalam penelitian ini juga dilakukan prosedur pegujian, adapun gambaran prosedur dalam penelitian ini dilampirkan dalam bentuk *flowchart* pada Gambar 1.



Gambar 1 Prosedur Penelitian

C. Studi Literatur

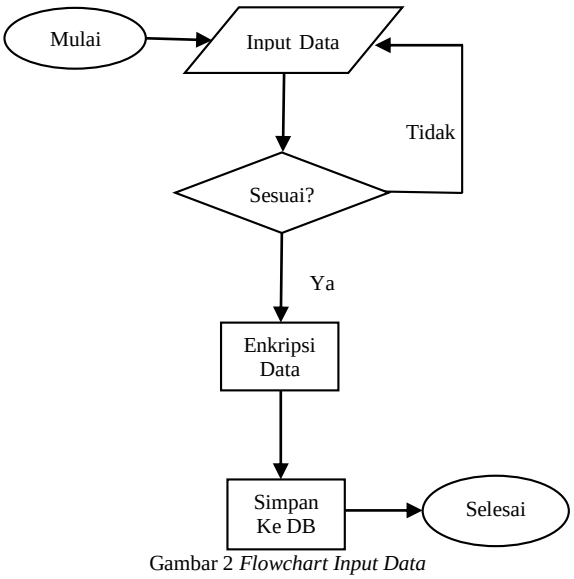
Mencari dan memilah jurnal penelitian tentang pengamanan data yang didapatkan dari salah satu layanan Google yaitu Google Scholar merupakan tahapan yang dilakukan dalam prosedur penelitian ini. Hasil dari studi literatur didapatkan beberapa jurnal yang sesuai dengan penelitian ini berupa tentang pengamanan data pada basis data, dan juga pengujian hasil enkripsi menggunakan teknik *Bruteforce Attack.*

D. Pengumpulan Data

Pengumpulan data dilakukan dengan observasi langsung ke Desa Sukaharja Karawang. Dalam observasi tersebut, penghimpunan data berupa salinan kartu keluarga berdomisili Sukaharja Karawang yang diperoleh dari ketua rukun tetangga setempat atas izin dan sepengetahuan kepala Desa Sukaharja Karawang. Terdapat 1.950 total data berupa salinan kartu keluarga berdomisili Sukaharja Karawang, dari total data tersebut didapatkan data pribadi warga berdomisili Sukaharja Karawang dengan total 6.952.

E. Perancangan

Dalam prosedur perancangan, dibuat rancangan untuk implementasi data berupa *plain text* menjadi *chiper text*, lalu *chiper text* disimpan ke dalam basis data sistem pelayanan Desa Sukaharja Karawang berbasis *web* jika data tersebut memenuhi validasi dari sistem. Perancangan juga dilakukan dalam menentukan tipe data pada kolom disetiap tabel, baik tabel tersebut untuk menyimpan data hasil enkripsi kartu keluarga maupun data warga Desa Sukaharja Karawang yang terenkripsi. Perancangan pada basis data dilakukan karena mengingat hasil enkripsi dari Algoritma AES-128 lebih panjang daripada data asli atau *plain text*. Perancangan alur pengamanan dilampirkan pada Gambar 2, Gambar 3 dan Gambar 4 merupakan perancangan tabel pada basis data.



Gambar 2 *Flowchart Input Data*

Gambar 3 *Class Diagram* Kartu Keluarga

Gambar 4 *Class Diagram* Warga

F. Implementasi

Implementasi dilakukan setelah tahap perancangan selesai, dalam implementasi kali ini mengikuti *environment* sistem pelayanan Desa Sukaharja Karawang berbasis *web*, sistem tersebut menggunakan kerangka kerja Laravel versi 8 dengan Bahasa program PHP, dan basis data menggunakan MySQL. Sebelum data dienkripsi menggunakan Algoritma AES-128, data divalidasi terlebih dahulu oleh sistem, jika data tervalidasi maka akan dilakukan enkripsi, lalu disimpan kedalam basis data. Apabila data tidak tervalidasi, maka sistem akan mengembalikan ke halaman sebelumnya dan data tidak dienkripsi maupun disimpan ke dalam basis data, implementasi dilampirkan pada Gambar 5 dan Gambar 6.

```
$request->validate([
 'no_kk' =>'required|unique:kartu_keluarga',
'kepalaKeluarga' => 'required',
'alamat' => 'required',
 'rt' => 'required',
 'rw' => 'required',
 'desa' => 'required',
 'kecamatan' => 'required',
  …]);
 KartuKeluarga::create(['no_kk'
=>Crypt::encrypt($request-
>no_kk),'nama_kepala_keluarga' => $request-
>kepalaKeluarga,'alamat' => Crypt::encrypt($request-
>alamat),'rt' => $request->rt,'rw' => $request-
>rw,'desa' => $request->desa,'kecamatan'
=>$request->kecamatan,'kabupaten' => $request-
>kabupaten,'pos' => $request->pos,'provinsi' =>
$request->provinsi,]);
```

Gambar 5 Menyimpan Data Kartu Keluarga

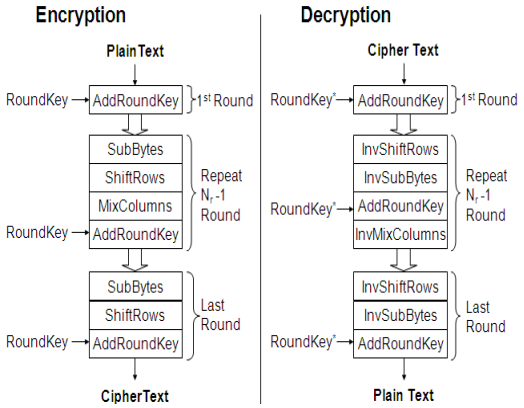
Pada Gambar 5 dan Gambar 6 terlihat data yang dienkripsi adalah data nomor kartu keluarga, alamat, nomor induk kependudukan, nama ayah, dan nama ibu. Hal tersebut dikarenakan dalam penelitian Benedict Ell Nino menyebutkan bahwa Nomor Induk Kependudukan atau NIK merupakan salah satu data penting dan bersifat rahasia [16]. Penelitian yang dilakukan oleh Diah Puspitasari dan kawan-kawan menyebutkan bahwa alamat dan kartu keluarga merupakan data identitas yang bilamana data tersebut bocor, bisa memungkinkan data tersebut digunakan oleh pihak tak bertanggung jawab dalam melakukan penipuan dan disalah gunakan [17]. Kasus pelanggaran lainnya juga disebutkan dalam penelitian Rachma, pengaksesan data tanpa izin di dalam *smartphone* lalu menyebarluaskan tanpa izin dari pemilik data tersebut, aksi itu dilakukan oleh beberapa layanan pinjaman *online* di Indonesia [18]. dan merujuk pada UU Nomor 27 Tahun 2022 pasal 5 ayat (2) huruf e dan pasal 5 ayat (3) huruf f tentang jenis data pribadi, pada kedua pasal tersebut mengenai data anak dan data pribadi yang dikombinasikan untuk mengidentifikasi seseorang.

```
$request->validate(['nama' => 'required','nik' =>
'required|unique:profiles|min:16|max:16','jk' =>
'required','kartu_keluarga_id' =>'required','tempatLahir'
=>'required','tanggalLahir' =>'required','agama' =>
'required','status_perkawinan' => 'required',
'status_hubungan_dalam_keluarga' =>
'required','nama_ayah' => 'required','nama_ibu' =>
'required',]);
$profilesId = Profiles::create(['kartu_keluarga_id' =>
$request->kartu_keluarga_id,'nama' =>$request-
>nama,'nik' => Crypt::encrypt($request->nik),'jk' =>
$request->jk,'tempat_lahir' => $request-
>tempatLahir,'tanggal_lahir' =>$request-
>tanggalLahir,'agama' => $request-
>agama,'pendidikan' => $request-
>pendidikan,'jenis_pekerjaan' => $request-
>jenisPekerjaan'status_perkawinan' => $request-
>status_perkawinan,
'status_hubungan_dalam_keluarga' => $request-
>status_hubungan_dalam_keluarga,'nama_ayah' =>
Crypt::encrypt($request->nama_ayah),'nama_ibu' =>
Crypt::encrypt($request->nama_ibu),])->id;
```

Gambar 6 Menyimpan Data Warga

G. Advanced Encryption Standard

Advanced Encryption Standard merupakan Algortima dari teknik Kriptografi, berperan dalam menggantikan Algoritma DES (*Data Encryption Standard*) yang dinilai sudah tidak aman menurut NIST (*National Institute of Standards and Technology*) pada tahun 2001. Panjang 128 bit dengan 10 putaran, 192 bit dengan 12 putaran, dan 256 bit dengan 14 putaran merupakan 3 jenis panjang ukuran dan banyaknya proses putaran dari Algoritma AES. Proses pengamanan dalam Algortima AES terdapat 4 jenis transformasi, diantaranya *Sub Bytes*, *Shift Rows*, *Mix Columns* dan *Add Round Key* [19]. Algoritma AES berbentuk *Chiper Block* dengan ukuran 4x4, ukuran *Chiper Block* tersebut tetap sama walaupun panjang ukuran dan banyakanya putaran yang digunakan dalam Algoritma AES berbeda [20]. Algoritma AES dibuat oleh Vincent Rijmen dan Joan Daemen, mereka berdua memenangkan kompetisi yang diselenggarakan oleh NIST (*National Institute of Standards and Technology*), kompetisi tersebut bertujuan mencari Algoritma *Block Encryption* yang paling cocok dalam standar Rijndael [21]. Alur pemrosesan Algoritma *Advanced Encryption Standard* dapat dilihat pada Gambar 7.



Gambar 7 Proses Enkripsi dan Dekripsi AES
(sumber: www.researchgate.net)

Sub Bytes merupakan proses transformasi *byte* di Algoritma AES, setiap *byte* pada *state* akan disubtitusikan menggunakan S-Box (tabel subtitusi Algoritma AES). Subtitusi ini dilakukan dengan cara dimana sebuah *byte* tidak pernah digantikan oleh dirinya sendiri dan juga tidak digantikan dengan *byte* lain yang merupakan komplemen dari *byte* sekarang [22]. Tahapan setelah *Sub Bytes* dalam transformasi Algoritma AES yaitu *Shift Rows*. Proses perpindahan setiap *byte* sesuai dengan nilai dari posisi baris, perpindahan pada proses *Shift Rows* dilakukan dengan menggeser ke arah kiri setiap *Byte*. Pada baris pertama, tidak terjadi pergeseran, pada baris kedua, setiap *byte* bergeser ke arah kiri sebanyak 1 *step*, pada baris ketiga, *byte* bergeser ke arah kiri sebanya 2 *step*, dan dilakukan pergeseran *byte* kea rah kiri sebanyak 3 *step* pada baris keempat [23]. Pada proses *Mix Columns* dilakukan perkalian antar matriks setelah proses *Shift Rows* selesai. *Mix Columns* melakukan perkalian antara matriks hasil proses *Shift Rows* dengan Matriks Galois Field [24].

Setelah proses *Mix Columns*, dilakukan proses *Add Round Key* yang bertujuan dalam pembangkitan kunci putaran selanjutnya. Pada penelitian Kareem, proses *Add Round Key* dalam Algoritma AES ini dilakukan dengan menggunakan fungsi XOR [25]. Dalam penelitian lain memberikan penjelasan dalam proses *Add Round Key*, dimana *Secret Key* asli atau beberapa *Secret Key* disusun sebagai *state* atau matriks 4x4, hasil dari dari proses *Add Round Key* ini adalah *chiper text* [26].

H. Pengujian

Merujuk pada penelitian Hasanudin dalam jurnalnya tentang pengamanan data menggunakan AES, dengan pengujian menggunakan metode *Bruteforce Attack* menggunakan *tool Crack Station* [27]. Maka dalam penelitian ini menggunakan *tool* yang berbeda, tahapan pengujian yang dilakukan pada penelitian ini menggunakan metode *Bruteforce Attack* dengan dukungan perangkat lunak Hashcat versi 6.2.5, digunakan dua buah model untuk menguji hasil dari enkripsi data menggunakan Algoritma AES-128, yaitu model AES 128 dengan nomor model 26401 dan model AES CRYPT dengan nomor model 22400 pada Hashcat versi 6.2.5. Dilakukan juga pengujian waktu dalam menampilkan data hasil dekripsi dan pengujian waktu proses enkripsi dalam penelitian ini. Untuk mengukur waktu kecepatan menampilkan data hasil enkripsi dan proses dekripsi menggunakan ekstensi DevTools yang sudah disediakan di dalam Chrome oleh Google. Pengujian menggunakan *Bruteforce Attack* dilakukan 4 kali percobaan, percobaan pertama menggunakan model AES 128 tanpa *wordlist* atau kumpulan kandidat nilai asli yang disimpan ke dalam *file* bernama wordlistSukaharja.txt. Data percobaan dengan *Bruteforce Attack* diperoleh dari data alamat dari data kartu keluarga yang sudah dienkripsi menggunakan Algoritma AES-128, dan disimpan ke dalam *file* sampleData.txt. Percobaan kedua menggunakan model AES 128 dengan wordlist, dalam percobaan ketiga, sama seperti percobaan pertama, namun menggunakan model AES CRYPT. Pada percobaan keempat menggunakan *Bruteforce Attack*, dilakukan dengan model AES CRYPT ditambah menggunakan *wordlist*. Berikut kode perintah untuk percobaan kesatu hingga keempat menggunakan *Bruteforce Attack* dengan dukungan Hashcat versi 6.2.5:

1. hashcat -a 0 -m 26041 sampleData.txt
2. hashcat -a 0 -m 26041 sampleData.txt wordlistSukaharja.txt
3. hashcat -a 0 -m 22400 sampleData.txt
4. hashcat -a 0 -m 22400 sampleData.txt wordlistSukaharja.txt

Penjelasan dari kode perintah tersebut bahwa haschat merupakan perintah dalam menjalankan program Hashcat, -a merupakan jenis serangan, karena pada penelitian ini menggunakan *Bruteforce Attack*, maka 0 merupakan perwakilan dari jenis serangan yang digunakan. Kode perintah -m merupakan perwakilan dari jenis model *chiper text* yang akan diserang, pada penelitian ini menggunakan Algoritma AES-128, maka digunakan model AES 128 dan AES CRYPT dengan kode model 26041 dan 22400. Pada pengujian waktu, cukup membuka Chrome lalu klik kanan, pilih Inspect dan klik Network, atau bisa menggunakan *shortcut* dengan menekan CTRL + Shift + j lalu klik Network.

3. HASIL DAN PEMBAHASAN

Pada bagian ini dapat diuraikan mengenai hasil dari penelitian beserta pengujian yang telah dilakukan. Selain itu, disampaikan juga mengenai pembahasan dari penelitian maupun pengujian yang telah dilakukan

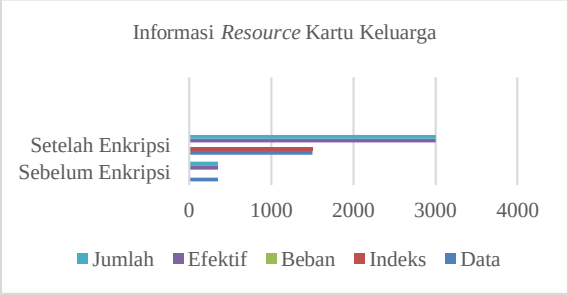
Hasil dari penelitian ini telah diterapkan pada sistem pelayanan Desa Sukaharja Karawang berbasis *web*, dalam upaya pengamanan data Warga Desa Sukaharja di dalam basis data sistem pelayanan Desa. Proses enkripsi berjalan dengan baik dan jumlah karakter pada *chiper text* sebanyak 228 karakter, proses dekripsi pun mengembalikan nilai yang sesuai. Berikut hasil enkripsi dilampirkan pada Gambar 8.

Gambar 8 Hasil Enkripsi Data Kartu Keluarga

nama	nik
SARIAH	eyJpdiI6ImU0cG1Zb201ZnpXU3JFOEs5WUJUcGc9PSIsInZhbH...
NYOMAN NURTE, S.Pd	eyJpdiI6IlVIT1IPUERta0w1MFV4UXBSM2FRQXc9PSIsInZhbH...
NANIK DWI UTAMI	eyJpdiI6Ill2bUxFY0l4YjUrVHlEbko3Z1pDa0E9PSIsInZhbH...
I.W.DHARMA NURTA PRATAMA	eyJpdiI6IlpQWFFyQXQwNEhESlRnRnNMQ1JROXc9PSIsInZhbH...
NI MADEAYU GAYA TRI PRATIWI	eyJpdiI6IjBPREdpUDh0TkdKL0Jqa1d5dGh6TlE9PSIsInZhbH...
SUDARSONO, S.Pd	eyJpdiI6IjNnY2Q4TE5xNVlROUFZWUZsWjVkMVE9PSIsInZhbH...

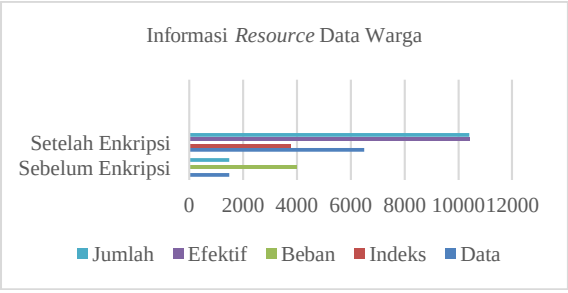
Gambar 9 Hasil Enkripsi Data Warga

Informasi dari table kartu keluarga sebelum dienkripsi, menunjukan informasi data sebesar 352.0 KB, indeks sebesar 0 B, beban sebesar 0B, efektif sebesar 352.0 KB dan jumlah sebesar 352.0 KB. Setelah dienkripsi, informasi beban yang ditampung oleh table kartu keluarga mengalami kenaikan dengan data sebesar 1.5 MB, indeks sebesar 1.5 MB, beban sebesar 0 B, efektif sebesar 3.0 MB dan jumlah sebesar 3.0 MB, data tersebut dilampirkan agar lebih jelas pada Gambar 10 dengan satuan *Kilo Byte*.



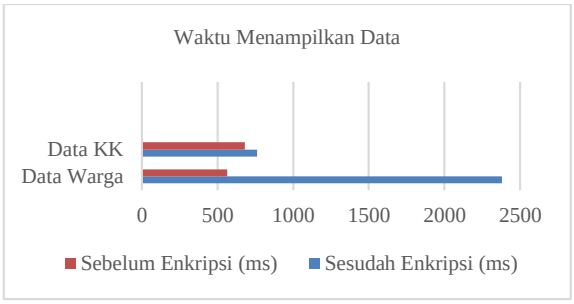
Gambar 10 Informasi *Resource* Data Kartu Keluarga

Untuk informasi dari tabel yang menampung data warga sebelum dienkripsi, menunjukan informasi data sebesar 1.5 MB, dengan indeks sebesar 0 B, beban sebesar 4.0 MB, efektif sebesar 0 dan jumlah sebesar 1.5 MB, setelah dienkripsi tabel informasi menunjukan kenaikan nilai dengan data sebesar 6.5 MB, indeks sebesar 3.8 MB, beban sebesar 0 B, efektif sebesar 10.4 MB dan jumlah sebesar 10.4 MB. Agar lebih jelas, data tersebut dilampirkan pada Gambar 11.



Gambar 11 Informasi Resource Data Warga

Dalam proses dekripsi lalu menampilkan data kartu keluarga dan data warga pada sistem pelayanan desa tidak memerlukan waktu yang lama. Pengambilan data pada sistem pelayanan Desa menggunakan AJAX tetapi pada proses dekripsi menggunakan PHP. Dibutuhkan waktu 2.38 detik dalam menampilkan 100 data pertama pada data warga. Untuk proses dekripsi lalu menampilkan data kartu keluarga pada sistem pelayanan desa, dibutuhkan waktu 761.02 *millisecond* dalam menampilkan 100 data pertama, sedangkan sebelum data dienkripsi dibutuhkan waktu sebesar 462.97 *millisecond* untuk data warga, dan 522 *millisecond* untuk data kartu keluarga, penjelasan lebih lanjut bisa dilihat pada Gambar 12.



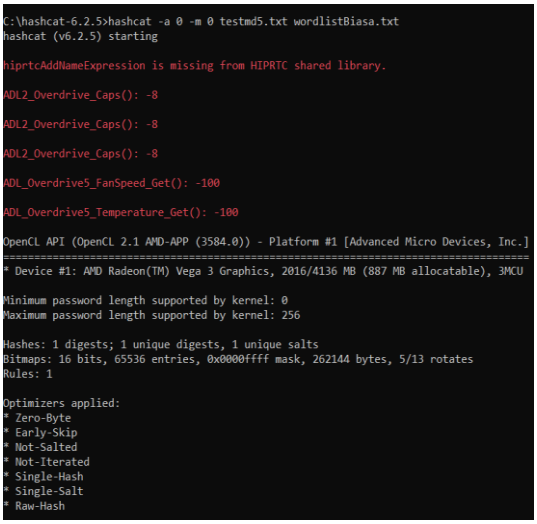
Gambar 12 Waktu Dekripsi dan Menampilkan Data

Data waktu yang dibutuhkan dalam menampilkan 100 data pertama diperoleh dari ekstensi DevTools di dalam perangkat lunak Google Chrome dan sudah disediakan oleh Google. Selanjutnya merupakan pengujian menggunakan metode *Bruteforce Attack* dengan menggunakan bantuan perangkat lunak Hashcat versi 6.2.5.

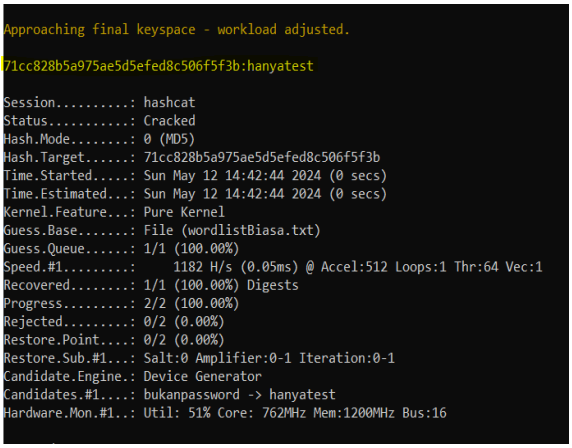
Sebelum melakukan pengujian terhadap *sample data*, pengujian pertama dilakukan dengan data palsu hasil dari proses *hash* MD5 sebegai berikut:

- *Plain Text:* hanyatest
- *ChiperText:*71cc828b5a975ae5d5efed8c506f5f3b

Untuk urutan kode perintah masih sama, namun modelnya dirubah menjadi 0, karena mewakili model MD5 pada perangkat lunak Hashcat versi 6.2.5, pengujian ini dilakukan sekali dengan menggunakan wordlist, untuk lebih jelasnya proses dan hasil tersebut dilampirkan pada Gambar 13 dan Gambar 14.



Gambar 13 Uji Coba MD5



Gambar 14 Hasil Uji Coba MD5

Terlihat pada Gambar 14, Hashcat berhasil dalam memecahkan *chiper text* MD5. *File* testmd5.txt merupakan *file* berisi *chiper text* MD5, dan *file* wordlistBiasa.txt merupakan kumpulan kandidat nilai asli dari *chiper text* pada *file* testmd5.txt, dalam *file* wordlistBiasa.txt berisi 2 kandidat, yaitu "bukanpassword" dan "hanyatest". Setelah pengujian terhadap data palsu untuk uji coba perangkat lunak, maka dilanjut untuk pengujian data asli dari hasil enkripsi AES-128. Data pengujian kali ini diambil secara acak dari data kartu keluarga, data tersebut berupa alamat saja. Data pengujian disimpan kedalam *file* sampleData.txt dengan nilai asli "DUSUN ULEKAN", berikut hasil dari pengujian tanpa *wordlist* menggunakan model AES 128 dan AES CRYPT dilampirkan pada Gambar 15 dan Gambar 16.

```
C:\hashcat-6.2.5>hashcat -a 3 -m 26401
sampleData.txt
hashcat (v6.2.5) starting
OpenCL API (OpenCL 2.1 AMD-APP (3584.0)) -
Platform #1 [Advanced Micro Devices, Inc.]
============================================
============================================
===
* Device #1: AMD Radeon(TM) Vega 3 Graphics,
2016/4136 MB (887 MB allocatable), 3MCU

./OpenCL/m26401_a3-optimized.cl: Pure kernel not
found, falling back to optimized kernel
Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 16

Hashfile 'sampleData.txt' on line 1
(eyJpdi...NzY2MzIwNzJjN2VjliwidGFnIjoiIn0=):
Separator unmatched
No hashes loaded.

Started: Sun May 12 14:50:38 2024
Stopped: Sun May 12 14:50:43 2024
```

Gambar 15 Pengujian Dengan Model AES-128

```
C:\hashcat-6.2.5>hashcat -a 3 -m 22400
sampleData.txt
hashcat (v6.2.5) starting

OpenCL API (OpenCL 2.1 AMD-APP (3584.0)) -
Platform #1 [Advanced Micro Devices, Inc.]
============================================
============================================
=
* Device #1: AMD Radeon(TM) Vega 3 Graphics,
2016/4136 MB (887 MB allocatable), 3MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 128

Hashfile 'sampleData.txt' on line 1
(eyJpdi...NzY2MzIwNzJjN2VjliwidGFnIjoiIn0=):
Separator unmatched
No hashes loaded.

Started: Sun May 12 14:59:13 2024
Stopped: Sun May 12 14:59:21 2024
```

Gambar 16 Pengujian Dengan Model AES CRYPT

Hashcat tidak dapat memecahkan *chiper text* dalam pengujian ini. Selanjutnya pengujian menggunakan *wordlist* yang ditampung pada *file* wordlistSukaharja.txt, di dalam *file* tersebut terdapat 3 kandidat nilai yaitu, "DSN ULEKAN", "ULEKAN" dan kandidat terakhir merupakan nilai asli dari *chiper text* yaitu "DUSUN ULEKAN". Hasil dari pengujian ini dilampirkan pada Gambar 17 dan Gambar 18.

```
C:\hashcat-6.2.5>hashcat -a 3 -m 22400
sampleData.txt wordlistSukaharja.txt
hashcat (v6.2.5) starting

OpenCL API (OpenCL 2.1 AMD-APP (3584.0)) -
Platform #1 [Advanced Micro Devices, Inc.]
============================================
============================================
===
* Device #1: AMD Radeon(TM) Vega 3 Graphics,
2016/4136 MB (887 MB allocatable), 3MCU
Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 128
Hashfile 'sampleData.txt' on line 1
(eyJpdi...NzY2MzIwNzJjN2VjliwidGFnIjoiIn0=):
Separator unmatched
No hashes loaded.
Started: Sun May 12 15:03:12 2024
Stopped: Sun May 12 15:03:17 2024
```

Gambar 17 Pengujian Model AES CRYPT Dengan *Wordlist*

```
C:\hashcat-6.2.5>hashcat -a 3 -m 26401
sampleData.txt wordlistSukaharja.txt
hashcat (v6.2.5) starting

OpenCL API (OpenCL 2.1 AMD-APP (3584.0)) -
Platform #1 [Advanced Micro Devices, Inc.]
==========================================
==========================================
===
* Device #1: AMD Radeon(TM) Vega 3 Graphics,
2016/4136 MB (887 MB allocatable), 3MCU

./OpenCL/m26401_a3-optimized.cl: Pure kernel not
found, falling back to optimized kernel
Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 16
Hashfile 'sampleData.txt' on line 1
(eyJpdi...NzY2MzIwNzJjN2VjIiwidGFnIjoiIn0=):
Separator unmatched
No hashes loaded.
Started: Sun May 12 15:04:56 2024
Stopped: Sun May 12 15:05:01 2024
```

Gambar 18 Pengujian Model AES-128 Dengan *Wordlist*

Hasil dari pengujian yang dilampirkan pada Gambar 17 dan Gambar 18 Hashcat tidak dapat memecahkan enkripsi AES-128 bahkan dengan bantuan *wordlist* sekalipun..

4. DISKUSI

Hasil pada penelitian ini didapatkan hasil dari analisis waktu, ketahanan dan besarnya *resource* yang diperlukan dari Algoritma AES-128. Waktu yang diperlukan untuk mengembalikan nilai asli untuk 100 data pertama menggunakan AJAX 2.38 detik pada data warga, hal tersebut dikarenakan banyak jenis data yang harus dikembalikan nilai aslinya, pada pengembalian nilai asli 100 data kartu keluarga pertama dibutuhkan waktu 761ms, lebih cepat dari data warga karena pada data kartu keluarga melakukan dekripsi tidak sebanyak data warga. Sebelum diimplementasikan Algoritma AES-128 pada sistem pelayanan desa, diperlukan waktu sebesar 462.97ms untuk menampilkan 100 data warga pertama, lalu dalam menampilkan 100 data kartu keluarga pertama dibutuhkan waktu 522ms. Terlihat terjadi kenaikan yang signifikan pada proses penampilan data warga, namun hal tersebut tidak membebani sistem pelayanan desa, hal tersebut dibandingkan dengan hasil penelitian dari Kurnia Nurhareza menggunakan AES-256 didapatkan rata-rata waktu dekripsi sebesar 1181000ms untuk data dokumen sebesar 2496069,1 *byte* dan 1167000 untuk dokumen sebesar 2496128 *byte*. Ketahanan diuji menggunakan serangan *Bruteforce* dengan alat bantu Hashcat versi 6.2.5 dan diterapkan teknik menggunakan *wordlist*, dimana pada peneltian yang dilakukan oleh Hasanudin dan Dasaprawira dalam sistem tabungan santri hanya menggunakan *online tools* yaitu CrackStation, pengujian tersebut tidak menggunakan tambahan *wordlist* , karena itu pada penelitian ini diterapkan *wordlist* dalam serangan *bruteforce* untuk menguji lebih dalam, namun hasilnya tetap sama, tidak bisa dipecahkan. Adapun besaran *resource* mengalami kenaikan pada tabel untuk menyimpan data warga dan tabel untuk menyimpan data kartu keluarga. *Resource* mengalami kenaikan yang signifikan seperti yang terlihat pada Gambar 10 dan Gambar 11, hal tersebut dikarenakan adanya penambahan karakter dari data pada tiap kolom yang dienkripsi, namun hal tersebut tidak mengganggu kinerja sistem maupun *server*. Data merupakan dari salinan kartu keluarga yang langsung didapatkan dari ketua rukun tetangga Desa Sukaharja Karawang atas izin Kepala Desa. Data tersebut dimasukan ke dalam sistem pelayanan Desa dengan validasi sistem, jika data tersebut memenuhi validasi maka data akan dienkripsi lalu dimasukan ke dalam *database*, dalam pengujian ketahanan Algoritma AES, menggunakan teknik serangan *Bruteforce* dengan ataupun tidak menggunakan *wordlist*, hasil dari pengujian tersebut, Hashcat atau perangkat lunak yang digunakan dalam pengujian ini tidak dapat memecahkan, bahkan tidak dapat mengenali *chiper text* meskipun sudah menggunakan model yang sesuai dengan Algoritma yang digunakan. Maka dari itu, penulis beranggapan bahwa data warga Desa Sukaharja lebih aman dari serangan *Bruteforce* baik itu menggunakan *wordlist* dengan isi yang sesuai kandidat nilai asli, maupun tidak.

5. KESIMPULAN

Upaya pengamanan dalam sistem pelayanan Desa Sukaharja Karawang berbasis *web* menggunakan Algoritma AES-128 merupakan langkah yang baik dalam meminimalisir terjadinya kebocoran data. Proses enkripsi dan dekripsi yang dilakukan tidak memakan *resource* besar, bahkan dalam menampilkan 100 data pertama juga tidak membutuhkan waktu lama. Ketahanan Algoritma AES-128 terbukti dalam pengujiannya menggunakan teknik *Bruteforce* dengan bantuan perangkat lunak Hashcat versi 6.2.5, Algoritma AES-128 tahan dari serangan tersebut. Algoritma AES yang diterapkan tidak mengganggu kinerja sistem pelayanan desa, seperti hasil yang dilampirkan di atas dalam menampilkan 100 data pertama pada data kartu keluarga, dibutuhkan 761.02 *millisecond* setelah data terenkripsi dan dibutuhkan waktu 522 *millisecond* sebelum data terenkripsi. Sedangkan dalam menampilkan 100 data pertama data warga, dibutuhkan waktu 2.38 detik setelah data terenkripsi, dan dibutuhkan waktu 462.97 *millisecond* sebelum data terenkripsi. Pada basis data walaupun kenaikan *resource* terlihat jauh berbeda ketika data sudah dienkripsi, namun hal tersebut tidak mempengaruhi kepada kinerja *server* yang digunakan. Algoritma *Advanced Encryption Standard* dengan panjang 128 bit sangat direkomendasikan dalam upaya pengamanan data pada basis data berdasarkan hasil dari analisis yang telah dilakukan dan dilampirkan pada penelitian ini.

DAFTAR PUSTAKA

[1] A. Susilo, Y. Irawan, A. R. Pratama, and R. Antono, "Journal of Sisfotek Global RC4 Cryptography Implementation Analysis on Text Data ARTICLE HISTORY," *Issn*, vol. 11, no. 2, pp. 115–120, 2021, [Online]. Available: http://journal.stmikglobal.ac.id/index.php/sisfotek

[2] S. Al Busafi and B. Kumar, "Review and Analysis of Cryptography Techniques," *Int. Conf. Syst. Model. Adv. Res. Trends SMART(SMART)*, pp. 2–6, 2020, doi: 10.1109.

[3] D. Kr, V. K. R. Dwivedi, and M. C. Trivedi, "Encryption algorithm in cloud computing," *ScienceDirect*, vol. 37, no. 2, pp. 1869–1875, 2021, doi: 10.1016/j.matpr.2020.07.452.

[4] F. Tawfiq, A. Hussien, A. M. S. Rahma, H. Bahjat, and A. Wahab, "A Secure Environment Using a New Lightweight AES Algorithm for E-Commerce Websites," *Secur. Commun. Networks*, vol. 2021, p. 15, 2023, doi: https://doi.org/10.1155/2021/9961172.

[5] F. Fathurrahmad, "Development And Implementation Of The Rijndael Algorithm And Base-64 Advanced Encryption Standard (AES) For Website Data Security," *Int. J. Sci. Technol. Res.*, vol. 9, no. November, pp. 7–11, 2020.

[6] M. Farras, I. J. Volume, M. F. Muttaqin, S. Dian, and H. Permana, "Implementation of AES-128 and Token-Base64 to Prevent SQL Injection Attacks via HTTP," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 9, pp. 2876–2882, 2020, doi: https://doi.org/10.30534/ijatcse/2020/60932020.

[7] M. Al-Mashhadani and M. Shujaa, "IoT Security Using AES Encryption Technology based ESP32 Platform," *Int. Arab J. Inf. Technol.*, vol. 19, no. 2, pp. 214–223, 2022, doi: 10.34028/iajit/19/2/8.

[8] S. Fatima, T. Rehman, M. Fatima, S. Khan, and M. A. Ali, "Comparative Analysis of Aes and Rsa Algorithms for Data Security in Cloud Computing †," *Eng. Proc.*, vol. 20, no. 1, 2022, doi: 10.3390/engproc2022020014.

[9] H. B. Setiawan and F. U. Najicha, "Perlindungan Data Pribadi Warga Negara Indonesia Terkait Dengan Kebocoran Data," *J. Kewarganegaraan*, vol. 6, no. 1, pp. 976–982, 2022.

[10] O. Maulida and H. Utomo, "Pertanggungjawaban Badan Penyelenggara Jaminan Sosial (BPJS) Kesehatan Atas Kebocoran Data Pribadi Pengguna dalam Perspektif Hukum Pidana," *Indones. J. Law Justice*, vol. 1, no. 2, p. 10, 2023, doi: 10.47134/ijlj.v1i2.2011.

[11] Muhammad Raihan, "Perlindungan Data Diri Konsumen Dan Tanggungjawab Marketplace Terhadap Data Diri Konsumen (Studi Kasus: Kebocoran Data 91 Juta Akun Tokopedia)," *J. Inov. Penelit.*, vol. 3, no. 10, pp. 7847–7856, 2023.

[12] B. E. Widodo and A. S. Purnomo, "Implementasi Advanced Encryption Standard Pada Enkripsi Dan Dekripsi Dokumen Rahasia Ditintelkam Polda Diy," *J. Tek. Inform.*, vol. 1, no. 2, pp. 69–77, 2020, doi: 10.20884/1.jutif.2020.1.2.21.

[13] B. A. Iswandari, "Jaminan Atas Pemenuhan Hak Keamanan Data Pribadi Dalam Penyelenggaraan E-Government Guna Mewujudkan Good Governance," *J. Huk. Ius Quia Iustum*, vol. 28, no. 1, pp. 115–138, 2021, doi: 10.20885/iustum.vol28.iss1.art6.

[14] R. Luthfi, "Perlindungan Data Pribadi sebagai Perwujudan Perlindungan Hak Asasi Manusia," *J. Sos. Teknol.*, vol. 2, no. 5, pp. 431–436, 2022, doi: 10.59188/jurnalsostech.v2i5.336.

[15] Gatot Efrianto and Nia Tresnawaty, "Pengaruh Privasi, Keamanan, Kepercayaan Dan Pengalaman Terhadap Penggunaan Fintech Di Kalangan Masyarakat Kabupaten Tangerang Banten," *J. Liabilitas*, vol. 6, no. 1, pp. 53–72, 2021, doi: 10.54964/liabilitas.v6i1.71.

[16] B. E. Nino, "Perbandingan Performa Algoritma AES dan Twofish Menggunakan Metode Strict Avalanche Criterion pada Nomor Induk Kependudukan Indonesia," *J. Teknol. Inf.*, vol. 9, no. 1, pp. 19–29, 2023, doi: 10.52643/jti.v9i1.2994.

[17] D. Puspitasari, I. Izzatusholekha, and S. K. Haniandaresta, "Urgensi Undang-Undang Perlindungan Data Pribadi Dalam Mengatasi Masalah Keamanan Data Penduduk," *J. Adm. Sos. Sci.*, vol. 4, no. 2, pp. 195–205, 2023, doi: https://doi.org/10.55606/jass.v4i2.403.

[18] R. F. Anggitafani, "Perlindungan Hukum Data Pribadi Peminjam Pinjaman Online Perspektif Pojk No. 1/Pojk.07/2013 tentang Perlindungan Konsumen Sektor Keuangan dan Aspek Kemaslahatan," *J. Islam. Bus. Law*, vol. Vol. 2, no. No. 2, pp. 56–72, 2021.

[19] A. Abdalrahman, "A Cloud Database based on AES 256 GCM Encryption Through Devolving Web application of Accounting Information System," *Int. J. Recent Technol. Eng.*, vol. 9, no. 5, pp. 216–221, 2021, doi: 10.35940/ijrte.e5269.019521.

[20] A. Fadlil, I. Riadi, and A. Nugrahantoro,

"Data Security for School Service Top-Up Transactions Based on AES Combination Blockchain Technology," *Lontar Komput. J. Ilm. Teknol. Inf.*, vol. 11, no. 3, p. 155, 2020, doi: 10.24843/lkjiti.2020.v11.i03.p04.

[21] E. Fernando, H. Rohayani, and D. Irsan, Muhammad, Dine, Agustin, Sujana, "Performance Comparison of Symmetries Encryption Algorithm AES and DES With Raspberry Pi," *IEEE*, pp. 353–357, 2019, doi: 10.1109/SIET48054.2019.8986122.

[22] P. Gaur, "AES Image Encryption (Advanced Encryption Standard)," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, no. 12, pp. 1357–1363, 2021, doi: 10.22214/ijraset.2021.39542.

[23] J. Kaur, S. Lamba, and P. Saini, "Advanced Encryption Standard: Attacks and Current Research Trends," *2021 Int. Conf. Adv. Comput. Innov. Technol. Eng. ICACITE 2021*, vol. 7, pp. 112–116, 2021, doi: 10.1109/ICACITE51222.2021.9404716.

[24] K. Shahbazi, S. Ko, and S. Member, "Area-Efficient Nano-AES Implementation for Internet-of-Things Devices," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 29, pp. 136–148, 2020, doi: 10.1109/TVLSI.2020.3033928.

[25] S. M. Kareem and A. M. S. Rahma, "New method for improving add round key in the advanced encryption standard algorithm," *Inf. Secur. J.*, vol. 30, no. 6, pp. 371–383, 2021, doi: 10.1080/19393555.2020.1859654.

[26] T. M. Kumar, K. S. Reddy, S. Rinaldi, B. D. Parameshachari, and K. Arunachalam, "A low area high speed fpga implementation of aes architecture for cryptography application," *Electron.*, vol. 10, no. 16, 2021, doi: 10.3390/electronics10162023.

[27] M. Hasanudin and M. N. Dasaprawira, "Pengujian aplikasi tabungan santri berbasis web dengan menggunakan algoritma kriptografi advance encryption standard (aes) 256.," vol. 1, no. 1, pp. 11–18, 2022.