Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

Implementasi AES ECB dan Hashing MD5/SHA-256 Pada Aplikasi Penyuratan Android

Fajar Febriyadi, Fitra Kurnia*, Nazaruddin Safaat Harahap, Febi Yanto, Pizaini

Fakultas Sains dan Teknologi, Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru, Indonesia
Email: [1]1950115043@students.uin-suska.ac.id, [2,*]fitra.kurnia.hasbi@gmail.com, [3]nazruddin.safaat@uin-suska.ac.id, [4]febiyanto@uin-suska.ac.id, [5]pizaini@uin-suska.ac.id
Email Penulis Korespondensi: fitra.kurnia.hasbi@gmail.com
Submitted: 30/10/2023; Accepted: 30/11/2023; Published: 30/11/2023

Abstrak−Kantor Wilayah Kementerian Agama Riau masih mengarsipkan surat tugas dan surat perjalanan dinas dengan cara manual. Staf yang mengurus bagian penyuratan yaitu staf unit kepegawaian dan hukum tidak memiliki aplikasi yang memfasilitasi kegiatan surat tugas dan surat perjalanan dinas untuk menyederhanakan pengarsipan dan data yang berisi informasi tertentu terdapat dalam surat yang meliputi surat tugas dan surat perjalanan dinas. Keamanan penting karena berhubungan dengan data. Oleh karena itu, dibuatlah aplikasi penyuratan untuk mendukung kegiatan penyuratan Kanwil Kemenag Riau dan memudahkan staf bagian unit Kepegawaian dan Hukum untuk mengurus surat tugas dan surat perjalanan dinas serta buku kontrol dengan baik. Pengembangan aplikasi android menggunakan metode *waterfall* dan algoritma AES mode ECB (*Electronic Code Book*) dan *hashing* MD5/SHA-256 untuk keamanannya. Dengan dibangunnya aplikasi ini memudahkan pimpinan dan pegawai dalam bertukar surat dan informasi yang sifatnya rahasia terjamin keamanannya dan aplikasi yang dibangun bisa digunakan oleh pengguna secara mudah. Hasil pengujian *Black Box* yang dilakukan pada aplikasi menghasilkan keluaran yang diharapkan dan pengujian UAT di dapatkan nilai 89 %. Pengujian aplikasi pada teks, *file* Jpg, Png dan Pdf memiliki tingkat keamanan yang cukup tinggi dengan menggunakan metode analisis statistik yaitu pengujian frekuensi bit, autokorelasi, distribusi bit 0/1, entropi.

Kata Kunci: Penyuratan; Aplikasi; AES; Hash; Kemenag

Abstract−The Riau Ministry of Religion Regional Office is still archiving assignment letters and official travel letters manually. The staff who take care of the correspondence section, namely personnel and legal unit staff, do not have an application that facilitates the activities of assignment letters and official travel letters to simplify filing and data containing certain information contained in letters which include assignment letters and official travel letters. Security is important because it relates to data. Therefore, a correspondence application was created to support the correspondence activities of the Riau Ministry of Religion Regional Office and make it easier for staff in the Civil Service and Legal unit to properly manage assignment letters and official travel letters as well as control books. Android application development uses the waterfall method and the ECB (Electronic Code Book) mode AES algorithm and MD5/SHA-256 hashing for security. By building this application, it will be easier for leaders and employees to exchange letters and confidential information, guaranteed security and the application built can be used by users easily. The results of the Black Box testing carried out on the application produced the expected output and the UAT test obtained a score of 89%. Application testing on sentences, Jpg, Png and PDF files has a fairly high level of security using statistical analysis methods, namely bit frequency testing, autocorrelation, 0/1 bit distribution, entropy.

Keywords: Lettering; Application; AES; Hash; Ministry of Religion

1. PENDAHULUAN

Kantor Wilayah Kementerian Agama Provinsi Riau adalah lembaga vertikal yang tidak memiliki otonomi dan harus bertanggung jawab secara langsung kepada Menteri Agama. Kantor Wilayah Kementerian Agama Provinsi Riau yang berada di Pekanbaru, ibu kota provinsi yang mengemban tugas dan fungsi Kementerian Agama di wilayah provinsi berdasarkan kebijakan, peraturan, dan undang-undang yang dikeluarkan oleh Menteri Agama yaitu peraturan Menteri Agama Republik Indonesia Nomor 13 tahun 2012. Kantor Wilayah Kementerian Agama Provinsi Riau memiliki 12 Kantor Kementerian Agama Kabupaten/Kota sebagai mitra kerjanya. Kantor Wilayah Kementerian Agama Provinsi Riau harus membuat keputusan yang sesuai dengan arah dan jalur yang ditentukan oleh Menteri Agama dan juga harus memberikan laporan kepada Menteri Agama. Visi, misi dan kebijakan teknis Kanwil Kemenag adalah untuk melayani dan membimbing masyarakat di Provinsi Riau dalam hal kehidupan beragama, Haji dan Umrah, pendidikan madrasah, pendidikan agama dan keagamaan, dan kerukunan umat beragama. Kanwil Kemenag juga mengatur kebijakan teknis untuk mengurus administrasi dan informasi. Kanwil Kemenag juga bertanggung jawab untuk mengkoordinasikan perencanaan, pengendalian, pengawasan, dan evaluasi program.

Kantor Wilayah Kementerian Agama Provinsi Riau melakukan penyuratan sebagai salah satu kegiatan kementerian yang penting. Surat berperan sebagai alat komunikasi dan alat bukti fakta. Kantor Kementerian memiliki bagian staf khusus yang disebut kepegawaian dan hukum yang mengurus kegiatan kepegawaian dan hukum kantor, termasuk kegiatan surat tugas dan surat perjalanan dinas yang diikuti oleh pegawai. Pengarsipan surat sangat penting bagi kegiatan Kanwil Kemenag Riau karena ada banyak jenis surat, termasuk surat tugas dan surat perjalanan dinas, yang harus dijaga dengan baik agar tidak terjadi kesalahan, kerusakan, atau bahaya seperti terbakar, robek, atau basah. Kanwil Kemenag Riau masih mengarsipkan surat tugas dan surat perjalanan dinas dengan cara manual, yaitu dengan membuat pembukuan besar. Hal ini menimbulkan masalah ketika harus

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

mencatat nomor surat, tanggal, asal surat, perihal surat, dan bagian surat lainnya yang sangat banyak. Akibatnya, pembukuan nomor surat menjadi semakin banyak dan pengarsipan surat menjadi tidak rapi. Penyimpanan yang tidak rapi dan penggunaan binder untuk menyimpan surat juga menyulitkan dalam mencari kembali surat yang sudah lama dan berisiko kehilangan surat yang sebenarnya ada. Oleh karena itu, diperlukan aplikasi yang dapat mengurus surat tugas dan surat perjalanan dinas, termasuk buku kontrol di dalamnya.

Staf unit kepegawaian dan hukum tidak memiliki sistem atau aplikasi yang memfasilitasi kegiatan surat tugas dan surat perjalanan dinas untuk menyederhanakan pengarsipan. Surat tugas dan surat perjalanan dinas, termasuk buku kontrol, yang ditangani oleh kantor wilayah Kementerian Agama Provinsi Riau khususnya divisi bagian Kepegawaian dan Hukum tentu semakin banyak setiap tahunnya. Oleh karena itu, dibuatlah Aplikasi Penyuratan kantor wilayah Kementerian Agama Provinsi Riau untuk mendukung kegiatan surat menyurat kantor wilayah Kementerian Agama Provinsi Riau dengan lebih baik dan memudahkan staf bagian unit Kepegawaian dan Hukum untuk mengurus surat tugas dan surat perjalanan dinas serta buku kontrol dengan lebih baik. Dalam mengembangkan aplikasi ini keamanan merupakan salah satu aspek penting yang seharusnya dipenuhi dari suatu informasi atau data. Data yang berisi informasi tertentu terdapat dalam surat yang meliputi surat tugas dan surat perjalanan dinas. Keamanan sangat penting karena berhubungan dengan data. Teknik kriptografi adalah teknik yang digunakan untuk mengamankan data dengan enkripsi yang membuat data sulit dibaca atau dibuka oleh orang yang tidak berwenang yang dapat menyalahgunakan informasi yang didapat karena tidak memiliki kunci untuk dekripsi. Teknik ini dapat menyelesaikan beberapa poin dalam keamanan data, yaitu *authentication*, *confidentiality*/*privacy*, *integrity* dan *non-repudiation*.

Penelitian oleh Pasaribu et al. [1] membahas aplikasi enkripsi *file* dokumen pada *smartphone* Android dengan menggunakan algoritma AES-256 dan *hashing* MD5 untuk meningkatkan keamanan data dari serangan atau *bug*. Penelitian ini menunjukkan bahwa proses enkripsi lebih cepat dari dekripsi, dan kecepatan berbeda tergantung pada jenis *file* dokumen yang dienkripsi. Pada penelitian oleh P.U.K et al. [2] tentang mengimplementasikan sistem pengenkripsian dokumen di LPSE dengan menggunakan algoritma AES dan MD5 untuk mendukung transparansi dan akuntabilitas pengadaan barang/jasa pemerintah. Penelitian ini menunjukkan bahwa ukuran *file* dan waktu proses enkripsi dipengaruhi oleh ukuran *file* asli, bukan jenis format *file*. Penelitian yang dilakukan Fathurrozi dan Selviyani [3] yaitu mengembangkan aplikasi enkripsi dan dekripsi *file* berbasis Windows dengan menggunakan algoritma AES mode CBC dan SHA untuk mengamankan data lembaga. Penelitian ini menunjukkan bahwa aplikasi dapat mengenkripsi dan dekripsi *file* dengan berbagai ekstensi dengan kunci yang sama, dan waktu proses dan ukuran *file* dipengaruhi oleh ukuran *file* asli. Penelitian yang dilakukan Sulastri dan Putri [4] yaitu mengembangkan metode kolaborasi dan modifikasi MD5 dan SHA-256 untuk meningkatkan keamanan enkripsi *password*. Dengan perangkat lunak penyerang, metode ini dapat menahan serangan *brute force* karena memiliki tingkat pengacakan yang tinggi. Hal ini ditunjukkan oleh hasil penelitian ini.

Penelitian oleh Abdullah et al. [5] yaitu aplikasi *web* ini dikembangkan untuk mengenkripsi *file* dokumen dengan menggunakan algoritma AES dan OTP. Dengan demikian, data dapat lebih aman dari risiko penyadapan dan pencurian. Aplikasi ini dapat mengamankan *file* Docx, Xlsx, dan Pdf dengan kunci yang unik dan waktu enkripsi yang cepat. Penelitian oleh Nuraeni et al. [6] yaitu menguji keamanan data pajak bumi dan bangunan dengan menggunakan algoritma Caesar Cipher dan AES-128 CBC. Hasilnya menunjukkan bahwa algoritma ini memiliki korelasi rendah dan entropi tinggi, tetapi *avalanche effect* rendah. Oleh karena itu, penelitian ini mengusulkan *Super* Enkripsi sebagai solusi untuk meningkatkan keamanan data. Penelitian yang dilakukan Pramudito dan Kusumaningsih [7] Aplikasi ini menggunakan algoritma AES 128 dan RC4 untuk mengamankan *email* dengan kriptografi. Aplikasi ini dapat melakukan enkripsi dan dekripsi dengan mudah dan aman pada pesan dan *file* lampiran. Aplikasi ini juga memiliki kecepatan yang berbanding lurus dengan ukuran data yang diproses. Penelitian yang dilakukan Hermawan dan Ujianto [8] yaitu mengkombinasikan algoritma AES dan RSA untuk enkripsi data yang lebih aman. Algoritma AES digunakan untuk enkripsi data, sedangkan algoritma RSA digunakan untuk enkripsi kunci AES. Penelitian ini menunjukkan bahwa kombinasi ini membutuhkan dua kali proses dekripsi dan kunci pribadi untuk mendapatkan pesan rahasia. Penelitian ini juga mengukur waktu yang dibutuhkan untuk enkripsi dan dekripsi dengan berbagai ukuran kunci.

Penelitian oleh Kurnia et al. [9] yaitu melakukan penelitian di unit kepegawaian Kanwil Kemenag Riau untuk membangun sistem informasi penyuratan dengan menggunakan metode *waterfall*. Nilai UAT (*User Acceptance Test*) yang tinggi didapat oleh sistem informasi ini setelah berhasil diuji dengan metode *black box*. Data surat kepegawaian dapat dikelola dengan baik dengan menggunakan sistem informasi ini. Penelitian yang dilakukan Ravida dan Santoso [10] yaitu mengaplikasikan algoritma AES untuk keamanan data IoT (*Internet of Things*) tanaman hidroponik. Kunci yang sama digunakan oleh algoritma AES untuk mengenkripsi dan mendekripsi data. Algoritma AES dapat bekerja secara lintas *platform* dan menyembunyikan data saat pengiriman dari *server* ke Android. Penelitian ini masih memiliki kekurangan yaitu nilai entropi yang belum sempurna. Penelitian oleh Hulu et al. [11] yaitu mengimplementasikan algoritma AES untuk keamanan *file* hasil radiologi di RSU Imelda Medan. Algoritma AES dapat mengirim *file* Jpg dan Png yang telah dienkripsi dengan kunci yang sama ke dokter. Algoritma AES berbasis *web* dan menggunakan metode SDLC (*Systems Development Life Cycle)* untuk pengembangan. Kecepatan enkripsi dan dekripsi tergantung ukuran *file*. Penelitian oleh Putra et al. [12] yaitu mengamankan dokumen dengan menggunakan algoritma AES. Algoritma

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

AES dapat mengubah *file* dokumen Docx dan Pdf menjadi *file* yang tidak bisa dibaca tanpa kunci yang sama. Algoritma AES dipengaruhi oleh ukuran *file* dalam waktu enkripsi dan dekripsi. Penelitian ini menunjukkan bahwa *file* yang dienkripsi dan didekripsi dengan AES kembali seperti *file* asal.

Penelitian oleh Handoyo dan Subakti [13] tentang mengimplementasikan algoritma AES Rijndael untuk keamanan data pada aplikasi android. Algoritma AES Rijndael dapat mengacak *bytes* dari *file* asli dengan kunci yang berbeda. Algoritma AES Rijndael dapat mengamankan *file* digital untuk berbagai jenis ekstensi dengan ukuran maksimal 30 MB. Penelitian ini menguji kompleksitas waktu, perubahan bit, dan ketahanan terhadap serangan *brute force* dari algoritma AES Rijndael. Penelitian ini menunjukkan bahwa algoritma AES Rijndael memiliki notasi *O(n)*, nilai *Avalanche Effect* 45%-60%, dan waktu pemecahan kunci $3{,}8 \times 10^8$ tahun. Penelitian yang dilakukan Kirana dan Sugianto [14] tentang menerapkan algoritma AES dan konversi bahasa Khek untuk keamanan SMS berbasis android. Kunci yang sama digunakan oleh algoritma AES untuk mengenkripsi dan mendekripsi SMS. Konversi bahasa Khek dapat menyembunyikan makna SMS dari orang yang tidak mengerti bahasa tersebut. Penelitian ini menunjukkan bahwa SMS menjadi lebih aman dengan algoritma AES dan konversi bahasa Khek. Penelitian oleh Permana dan Nurnaningsih [15] tentang merancang aplikasi pengamanan data dengan algoritma AES Rijndael untuk enkripsi dan dekripsi *file* dan teks. Algoritma AES Rijndael merupakan algoritma yang sulit dipecahkan karena memiliki pola acak dan kecepatan komputasi yang tinggi. Aplikasi ini dibuat dengan menggunakan Microsoft Visual Studio 2010 dan mendukung perkembangan zaman yang semakin canggih.

Penelitian yang dilakukan oleh Fitriani dan Utomo [16] yaitu mengimplementasikan algoritma AES untuk keamanan SMS desa dengan menggunakan layanan SMS *Gateway*. SMS dapat dienkripsi dan didekripsi dengan kunci yang sama oleh algoritma AES yang tidak dapat ditembus oleh perangkat lunak penyerang. Algoritma AES memiliki nilai *avalanche effect* yang tinggi dan menunjukkan tingkat keamanan yang baik. Penelitian ini juga menguji kelayakan sistem layanan SMS desa dan mendapatkan persentase yang sangat tinggi. Penelitian ini menunjukkan bahwa sistem ini dapat meningkatkan pelayanan pemerintahan desa terhadap warga. Penelitian oleh Putri et al. [17] yaitu menggunakan AES dan EOF (*End of File*) untuk mengamankan *file* dokumen dan gambar. *File* dienkripsi dan disisipkan pesan rahasia dengan kunci yang sama. *File* tidak dapat dibobol, dicuri, atau dimanipulasi. *File* didekripsi dengan kunci asli akan kembali seperti semula. Penelitian yang dilakukan Ahyuna dan Hozeng [18] yaitu membuat aplikasi enkripsi *email* dengan AES berbasis android. Aplikasi ini dapat mengenkripsi *email* dengan kunci yang sama antara pengirim dan penerima. Aplikasi ini diuji dengan metode *Black Box* dan tidak ditemukan kesalahan fungsionalitas. Aplikasi ini dapat meningkatkan keamanan pengiriman *email*.

Penelitian-penelitian sebelumnya yang menerapkan algoritma AES-ECB dengan MD5 atau SHA-256 untuk enkripsi data, baik *file* maupun pesan teks, belum banyak membahas aplikasi keamanan penyuratan Android dengan mengukur dan membandingkan 4 model keamanan yaitu AES ECB-128, AES ECB-256, AES ECB-128 kunci di *Hashing* MD5 dan AES ECB-256 kunci di *Hashing* SHA-256 pada frekuensi bit, autokorelasi, entropi, distribusi bit, perubahan bit dan menguji kecepatan enkripsi dan dekripsi dari data yang dienkripsi serta menguji keamanan dari data yang dienkripsi terhadap serangan *known-plaintext* dan *brute force* dengan fungsi *cost* entropi yang dapat mengancam kerahasiaan dan integritas dari data yang dienkripsi. Aplikasi penyuratan Android memiliki tantangan dalam hal keamanan data dan dengan parameter-parameter yang disebutkan sebelumnya dapat menunjukkan seberapa acak, seimbang, dan sensitif data yang dienkripsi terhadap perubahan bit pada *plainteks* atau kunci serta seberapa efisiensi dan efektivitas dari algoritma enkripsi yang digunakan. Oleh karena itu, perlu adanya penelitian yang mengembangkan aplikasi penyuratan Android yang aman dengan menggunakan algoritma AES-ECB dengan MD5 atau SHA-256 dengan melakukan analisis statistik dan pengukuran kecepatan enkripsi dan dekripsi terhadap data yang di enkripsi disertai simulasi keamanan dari data yang dienkripsi dengan serangan *known-plaintext* dan *brute force* dengan fungsi *cost* entropi.

Ketika pengiriman data ke *database* maka harus diamankan agar ketika orang yang tidak diinginkan tidak bisa mengetahui data tersebut ketika proses pengiriman. Algoritma kriptografi AES dipilih untuk menjaga keamanan data pada pembuatan aplikasi ini karena AES adalah *cipher* yang berarah kepada bit, lalu AES mode ECB (*Electronic Code Book*) menjadi algoritma yang efisien untuk diterapkan ke *software* dan *hardware*. AES mode operasi ECB sederhana serta mudah untuk diimplementasikan. AES cocok untuk digunakan dalam mengamankan data pada sistem/*database* dan algoritma AES adalah algoritma yang tergolong sulit untuk dipecahkan karena belum ada serangan yang berhasil dengan analisis matematika secara efektif dan efisien karena pola yang dibentuk cukup acak dengan kekuatan hasil enkripsi didapatkan bergantung kepada semakin panjang dan unik kunci yang dipakai. Kompleksnya pada aplikasi yang dibangun kunci yang digunakan sebelum melakukan enkripsi akan dilakukan *hashing* MD5/SHA-256. Pada penelitian ini membangun aplikasi serta menganalisis 4 model keamanan yang diimplemetasikan pada aplikasi yaitu dengan AES ECB 128, AES ECB 256, AES ECB 128 dengan kunci di *Hashing* MD5 dan AES ECB 256 dengan kunci di *hashing* SHA-256. Rancang bangun aplikasi ini memasukkan ekstensi *file* berupa Jpg, Png dan Pdf. Kompleksnya lagi untuk dapat melihat dokumen tersebut harus menggunakan aplikasi yang dikembangkan karena proses enkripsi dan dekripsi harus berjalan di dalam aplikasi kemudian dilihat seberapa kuat enkripsi terhadap serangan pada aplikasi dengan metode analisis statistik. Dengan dibangunnya aplikasi ini bisa memudahkan pimpinan dan pegawai dalam berturkar surat dengan repositori dan informasi yang sifatnya rahasia terjamin keamanannya dan aplikasi yang

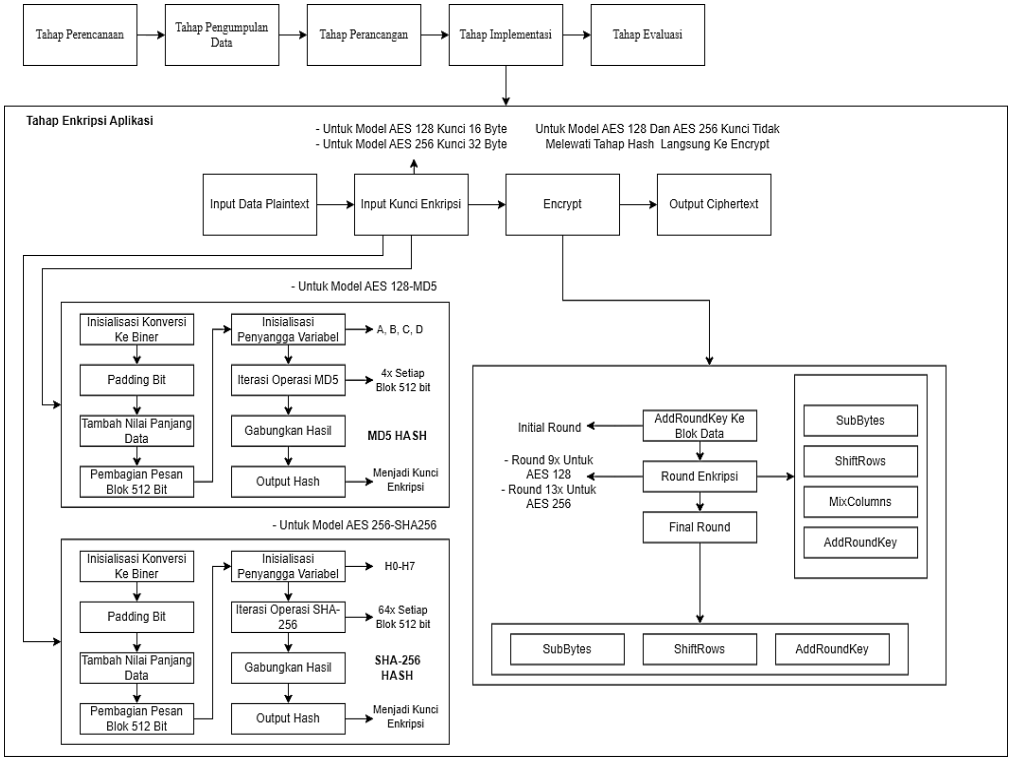
Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

dibangun bisa digunakan oleh pengguna secara mudah. Sesuai dengan semua ketentuan penilaian di atas menjadi dasar untuk membuat rancang bangun aplikasi repositori untuk keamanan penyuratan menggunakan algoritma AES ECB dan *hashing* MD5/SHA-256 pada kantor wilayah Kementerian Agama Provinsi Riau berbasis android.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Berdasarkan gambar 1 terdapat proses metodologi penelitian ini terdapat tahapan-tahapan yang mana ada 5 tahapan prosesnya meliputi tahap perencanaan, tahap pengumpulan data, tahap perancangan, tahap implementasi, dan tahap evaluasi.



Gambar 1. Tahapan Metodologi Penelitian

Flowchart pada gambar 1 tersebut, berdasarkan dari metode *waterfall* yaitu model proses pengembangan perangkat lunak yang paling sederhana dan tertua. Model ini membagi proses menjadi beberapa fase yang berurutan dan tidak saling tumpang tindih, seperti aliran air terjun. Setiap fase harus selesai sebelum fase berikutnya dapat dimulai [19]. *Flowchart* adalah skema yang menunjukkan aliran logika dalam suatu program. *Flowchart* menggunakan notasi-notasi tertentu untuk menggambarkan algoritma dalam bentuk visual. *Flowchart* membantu para pengembang untuk merancang, menganalisa, dan menguji program [20]. Model *Waterfall* adalah model proses pengembangan sistem informasi yang terdiri dari lima tahap, yaitu investigasi, analisis, desain, implementasi dan perawatan. Model ini cocok untuk proyek-proyek yang memiliki kebutuhan yang jelas dan stabil [20]. Aplikasi dibangun dengan mengimlementasi dan menganalisis 4 jenis yaitu dengan AES ECB 128, AES ECB 256, AES ECB 128 dengan hashing MD5 dan AES ECB 256 dengan *hashing* SHA-256. Untuk AES ECB 128/256 yang di *hashing* dengan MD5/SHA-256 dikenal dengan KDF (*Key Derivation Functions*) yang mana kunci pada AES akan diturunkan sebelum di enkripsi. Pada tahapan implementasi enkripsi data, data diinput terlebih dahulu kemudian kunci di masukkan dan proses enkripsi menggunakan algoritma AES dilakukan dengan operasi *AddRound*, *SubBytes*, *ShiftRows*, *MixColumns* dan mengeluarkan *output ciphertext*. Jika kunci 16 karakter maka model AES ECB 128 digunakan dan jika kunci 32 karakter maka model AES ECB 256 digunakan. Sedangkan untuk model AES ECB 128-MD5 dan AES ECB 256-SHA 256, kunci akan diproses *hashing* terlebih dahulu dengan algoritma MD5 untuk kunci 16 karakter dan dengan algoritma SHA-256 untuk kunci 32 karakter sebelum digunakan pada proses enkripsi. KDF adalah fungsi kriptografi yang menghasilkan kunci dari kunci rahasia yang mana KDF dapat digunakan untuk mendapatkan kunci yang kuat dari input lain. Dengan kata lain, KDF yang ditentukan di sini memberikan kemampuan perluasan kunci dan KDF penting untuk sistem kriptografi. Kekuatan keamanan KDF adalah usaha yang diperlukan untuk membedakan keluaran KDF dari *string* bit acak, dengan asumsi kunci derivasi kunci tidak diketahui [31].

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

2.2 Tahap Perencanaan

Pada perencanaan dibatasi masalah hanya berupa penyuratan surat tugas dan surat perjalanan dinas dan algoritma yang digunakan AES mode ECB dan *hashing* MD5/SHA-256 untuk keamanan serta metode yang digunakan adalah metode *waterfall* dalam membangun aplikasi android.

2.3 Tahap Pengumpulan Data

Dalam tahap ini, peneliti menggunakan teknik wawancara untuk menanyakan kepada pegawai unit kepegawaian dan hukum Kanwil Kemenag Riau yang mengurus surat tugas dan surat perjalanan dinas. Tujuannya adalah untuk memvalidasi data observasi dan mengidentifikasi kebutuhan dan permintaan yang dibutuhkan.

2.4 Tahap Perancangan

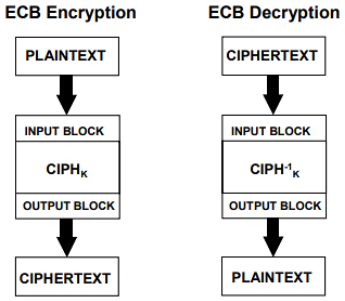
Tahap dengan menggunakan metode OOA (*object-oriented-analysis*), untuk membuat uraian sistem dari sistem yang sedang berjalan sebagai analisis. Dengan menggunakan metode OOD (*object-oriented-design*), untuk membuat diagram–diagram UML (*Unified-Modeling-Language*) sebagai rancangan sistem. OOAD (*object-oriented-analysis-and-design)* adalah pendekatan berorientasi objek untuk analisis dan desain sistem perangkat lunak. OOAD menjelaskan kebutuhan aktor, objek, kelas dalam lingkup sistem. OOAD menggunakan UML sebagai notasi visual dan *use case* sebagai pemahaman masalah dan perencanaan proyek [21].

2.5 Tahap Implementasi

Dalam membangun aplikasi ini menerapkan aplikasi android dengan menggunakan Java serta algoritma kriptografi untuk implementasi keamanan informasi dengan aplikasi menyimpan data surat dengan repositori. Repositori adalah layanan untuk mengelola dan menyebarluaskan materi digital. Repositori melestarikan, mengorganisasi, dan mendistribusikan materi digital secara tepat. Repositori digital adalah repositori yang dapat diakses dari luar institusi, yang bisa berdasarkan subjek, kelembagaan, atau komersial [22]. Kriptografi adalah ilmu dan keahlian tentang komunikasi yang aman dari pihak ketiga yang tidak berhak. Kriptografi modern memadukan ilmu matematika, ilmu komputer, dan teknik elektro untuk menciptakan dan menganalisis protokol komunikasi yang terproteksi. Kriptografi juga menangani kerahasiaan data, integritas data, autentikasi, dan non-repudansi. Kriptografi digunakan dalam ATM, *password* komputer, dan *E-commerce* [23]. Algoritma kriptografi yang digunakan dalam aplikasi yang dibangun adalah AES yang termasuk bagian algoritma simteris dan MD5 serta SHA-256 yang termasuk bagian algoritma *hashing*.

Kriptografi adalah ilmu yang mempelajari cara menyamarkan data asli (*plaintext*) menjadi data teracak (*ciphertext*) dengan menggunakan algoritma dan kunci tertentu. Tujuannya adalah untuk melindungi data dari pihak yang tidak berhak. Algoritma dan kunci adalah rahasia yang hanya diketahui oleh pengirim dan penerima data. *Ciphertext* dapat dikembalikan menjadi *plaintext* dengan proses dekripsi [24]. Enkripsi simetris menggunakan kunci yang sama untuk mengubah dan mengembalikan data, sedangkan enkripsi asimetris menggunakan kunci yang berbeda [25]. *Cryptographically secure hashing* adalah cara membuat sidik jari digital untuk naskah dengan ukuran berapa pun. Sidik jari ini unik, kecil, dan sulit untuk dipalsukan. Algoritma yang berbeda dapat menghasilkan sidik jari dengan panjang yang berbeda [25]

Algoritma kriptografi AES dipilih untuk menjaga keamanan data pada aplikasi ini karena AES adalah *cipher* yang berbasis bit dan memiliki variasi ukuran kunci yang baik, yaitu 128, 192 dan 256 bit. AES cocok untuk digunakan dalam mengamankan data pada sistem/*database* dan algoritma AES adalah algoritma yang tahan terhadap serangan kriptanalisis dan belum ada serangan yang berhasil dengan analisis matematika secara efektif dan efisien karena pola yang dibuat cukup acak dan kualitas enkripsi ditentukan oleh panjang dan keunikan kunci yang dipakai. Selanjutnya AES mode ECB (*Electronic Code Book*) yang terlihat pada gambar 2, dipilih karena mudah diimplementasikan baik pada perangkat keras maupun perangkat lunak dan memiliki kecepatan enkripsi dan dekripsi yang cukup tinggi dengan efisiensi memori yang baik serta ingin melakukan perbandingan antara mode AES ECB 128, AES ECB 256, AES ECB 128 kunci dihash MD5 dan AES ECB 256 kunci dihash SHA-256. Pada gambar 4, MD5 adalah fungsi *hash* yang mengubah pesan menjadi sidik jari 128-bit yang unik dan aman.



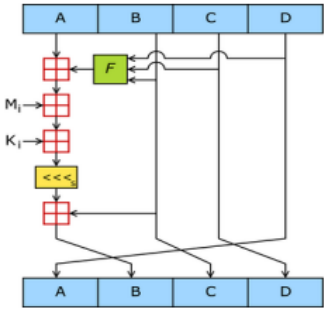
Gambar 2. Enkripsi/Dekripsi ECB

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

$$Ch(x,y,z) = (x \wedge y) \oplus (\neg x \wedge z)$$
$$Maj(x,y,z) = (x \wedge y) \oplus (x \wedge z) \oplus (y \wedge z)$$

$$\sum_0^{\{256\}}(x) = ROTR^{2}(x) \oplus ROTR^{13}(x) \oplus ROTR^{22}(x)$$
$$\sum_1^{\{256\}}(x) = ROTR^{6}(x) \oplus ROTR^{11}(x) \oplus ROTR^{25}(x)$$
$$\sigma_0^{\{256\}}(x) = ROTR^{7}(x) \oplus ROTR^{18}(x) \oplus SHR^{3}(x)$$
$$\sigma_1^{\{256\}}(x) = ROTR^{17}(x) \oplus ROTR^{19}(x) \oplus SHR^{10}(x)$$

Gambar 3. SHA-256 Formula



Gambar 4. Model MD5

AES adalah teknik enkripsi standar yang mengamankan data dengan kunci besar (128, 192, atau 256 bit) dan beberapa transformasi terhadap naskah asli. AES termasuk *block cipher*, yang mengenkripsi data dalam blok-blok 128 bit. AES menggunakan *S-boxes* untuk substitusi, dan operasi-operasi lain seperti *ShiftRows*, *MixColumns*, dan *AddRoundKey*. Jumlah putaran AES tergantung pada besar kunci. AES adalah teknik enkripsi yang kuat dan efisien [26]. Enkripsi ECB adalah mode enkripsi yang mengubah setiap blok teks biasa menjadi blok *ciphertext* dengan menggunakan fungsi sandi yang sama. Enkripsi ECB dapat dilakukan secara paralel dan sederhana, tetapi blok teks biasa yang sama akan menghasilkan blok *ciphertext* yang sama. Enkripsi ECB mirip dengan buku kode yang memiliki pasangan kata kode tetap untuk setiap kata [27]. CBC (*Cipher Block Chaining*) adalah mode yang melakukan enkripsi pada setiap blok data dengan menggunakan hasil enkripsi dari blok sebelumnya. Dengan demikian, blok-blok data yang identik akan menghasilkan blok-blok *ciphertext* yang berlainan. Mode ini memerlukan sebuah inisialisasi vektor (IV) yang acak dan unik untuk tiap enkripsi [25]. MD5 (*Message-Digest algorithm* 5) bekerja dengan membagi pesan menjadi blok 32-bit dan melakukan empat putaran operasi dari 16 operasi (64 operasi) *nonlinear* pada setiap blok. MD5 menggunakan konstanta 32-bit yang berbeda untuk setiap operasi [28]. SHA-256 (*Secure Hash Algorithm* 256-bit) keduanya menggunakan enam fungsi logis, di mana setiap fungsi beroperasi pada kata-kata 32-bit, yang direpresentasikan sebagai x, y, dan z. Hasil dari setiap fungsi adalah kata 32-bit baru. Hasil akhir dari SHA-256 adalah intisari pesan 256-bit [29].

2.6 Tahap Evaluasi

Pada tahap evaluasi menggunakan *black box testing* untuk validasi pengujian aplikasi sudah berjalan dengan tepat. Dilanjutkan dengan pengujian UAT (*User Acceptance Test*) yaitu menguji kelayakan untuk digunakan pengguna aplikasi yang dibangun. Pada bagian keamanan yang diimplementasikan pada aplikasi menggunakan metode analisis statistik. *Black Box Testing* adalah metode pengujian untuk mengecek fungsi masukan dan keluaran sesuai spesifikasi. *Tester* tahu apa yang harus dilakukan program tetapi tidak tahu bagaimana caranya [30]. UAT adalah pengujian oleh *end-user* untuk memastikan sistem sesuai dengan kebutuhan. UAT dilakukan setelah sistem *testing* dan menunjukkan sistem memenuhi persyaratan [31]. Untuk mengukur keacakan barisan *biner*, lima uji statistik digunakan untuk mengevaluasi beberapa karakteristik spesifik yang biasanya dimiliki oleh urutan yang sepenuhnya acak. Namun, hasil dari setiap uji adalah berdasarkan probabilitas. Tidak ada jaminan bahwa barisan *biner* benar-benar acak meskipun lulus kelima uji. Berikut 5 uji statistik menurut [32]:

a. Uji frekuensi bit mengukur apakah barisan bit memiliki proporsi bit “0” dan “1” yang seimbang. Evaluasi distribusi bit *ciphertext* untuk mengetahui tingkat keacakan. Distribusi bit yang seragam dan mendekati 50% menandakan tidak adanya pola dalam *ciphertext*. Dengan persamaan rumus (1) sebagai berikut:

$$X_1 = \frac{(n_0 - n_1)^2}{n} \quad (1)$$

b. Uji serial menghitung jumlah sub barisan bit 00, 01, 10 dan 11 yang muncul dengan frekuensi yang sama. Dengan persamaan rumus (2) sebagai berikut:

$$X_2 = \frac{4}{n-1}(n_{00}^2 + n_{01}^2 + n_{10}^2 + n_{11}^2) - \frac{2}{n}(n_0^2 + n_1^2) + 1 \quad (2)$$

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

c. Uji *poker* mengamati jumlah barisan bit dengan panjang tertentu yang sesuai dengan harapan. Dengan persamaan rumus (3) sebagai berikut:

$$X_3 = \frac{2^m}{k}\left(\sum_{i=1}^{2^m} n_i^2\right) - k \quad (3)$$

d. Uji *run* mengecek jumlah *run* dalam barisan bit dengan berbagai ukuran yang memenuhi kriteria keacakan. Dengan persamaan rumus (4) sebagai berikut:

$$X_4 = \sum_{i=1}^{k} \frac{(B_i - e_i)^2}{e_i} + \sum_{i=1}^{k} \frac{(G_i - e_i)^2}{e_i} \quad (4)$$

e. Uji autokorelasi menilai kemungkinan keterkaitan antara barisan bit dan versi gesernya. Menurut kriteria keacakan nilai autokorelasi adalah konstan meskipun barisan bit digeser berkali-kali. Tidak ada hubungan antara bit-bit *ciphertext* jika nilai korelasi mendekati nol. Dengan persamaan rumus (5) sebagai berikut:

$$X_5 = 2\ (A(d) - \frac{n-d}{2})/\sqrt{n-d} \quad (5)$$

Selain 5 uji diatas terdapat pengujian lainnya yaitu pengujian entropi. Entropi mengukur ketidakpastian dalam variabel acak X [33]. Entropi dari variabel acak X yang memiliki alfabet X dan distribusi probabilitas PX. Jika nilai entropi mendekati 8 menunjukkan hasil *ciphertext* yang acak dengan didefinisikan sebagai berikut menurut [34]. Dengan persamaan rumus (6) sebagai berikut:

$$H(X) = -\sum_{x \in X} P_x(x) \log P_x(x) \quad (6)$$

Persamaan matematika yang disebut rumus chi-kuadrat digunakan untuk menguji apakah suatu hipotesis sesuai dengan data yang ada. Persamaan ini dapat diterapkan untuk berbagai macam uji, misalnya uji kecocokan, uji ketergantungan, atau uji keseragaman. Persamaan ini dapat digunakan untuk pengujian distribusi bit 0 dan 1. Distribusi bit *ciphertext* acak jika frekuensi bit 0 dan 1 sama-sama mendekati 50% [35].

$$x^2 = \sum_{i=0}^{k} \frac{(O_i - E_i)^2}{E_i} \quad (7)$$

Jadi, pada pengujian analisis statistik nanti, yang digunakan dalam penelitian ini hanya uji frekuensi bit, uji distribusi bit 0 dan 1, uji autokorelasi, uji entropi, uji kecepatan enkripsi dan dekripsi, serta uji perubahan bit yang terjadi pada hasil enkripsi. Selain menganalisis *ciphertext*, pengujian ini juga akan mencoba menemukan kunci dengan metode *brute force* dengan menggunakan fungsi entropi dan *known plaintext attack*. Alat yang digunakan dalam pengujian ini adalah CrypTool. *Known plaintext attack* adalah proses untuk menentukan kunci enkripsi dengan menggunakan pasangan naskah asli-naskah acak yang ada. Tingkat kemudahan *known-plaintext attack* ditentukan oleh rumus enkripsi yang digunakan. Rumus enkripsi yang lebih canggih akan mengurangi peluang *known-plaintext attack* [25]. CrypTool adalah perangkat lunak yang menghadirkan berbagai sumber daya untuk mengimplementasikan dan mempelajari enkripsi dan dekripsi serta mengetahui sejarah kriptografi dari masa lampau hingga saat ini. Beberapa fitur kriptografi yang disediakan antara lain caesar, vigenere, hill, playfair, transposisi, permutasi, DES, AES dan RSA. Beberapa fitur kriptoanalisis yang ada antara lain entropi, histogram, *brute force*, frekuensi mengambang, analisis n-gram, korelasi otomatis, periodisitas, faktorisasi modul RSA, serangan berbasis kisi dan saluran samping [36].

3. HASIL DAN PEMBAHASAN

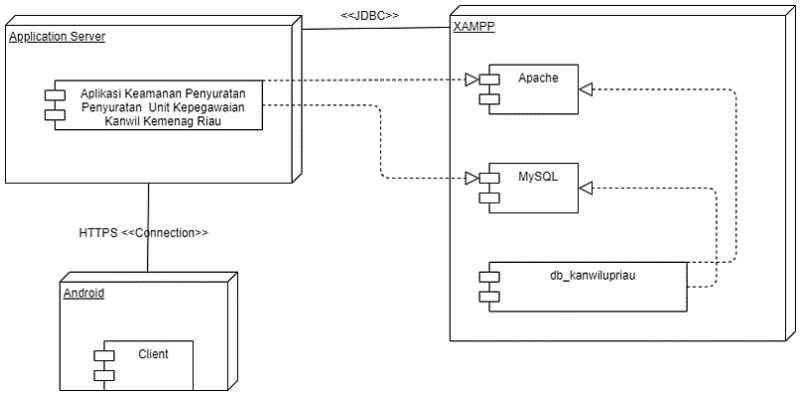
3.1 Analisa dan Perancangan

Tahap analisa dan perancangan terdapat analisis sistem lama dan sistem baru untuk aplikasi keamanan penyuratan kantor wilayah Kementerian Agama Riau bagian kepegawaian yang mana dengan memperoleh data dengan cara wawancara dan observasi. Sistem lama memiliki permasalahan dalam penyampaian informasi dan pelayanan yang masih manual, sedangkan sistem baru memanfaatkan *gadget* untuk memudahkan pencarian, pengamanan, dan pengarsipan informasi surat-surat. Sistem baru terdiri dari menu beranda, buku kontrol, buku surat perjalanan dinas, buku surat tugas, dan pengelolaan administrator yang dapat melihat, merubah, dan menghapus data.

3.2 Implementasi

Pada lingkungan pengembangan aplikasi dengan menggunakan *hardware processor* Ryzen 3 3200U dan Ram 8 GB. Pada bagian *software* menggunakan sistem operasi Windows 10 *Home* 64-bit dengan menggunakan Android Studio Java sebagai pembuatan aplikasi. Pada bagian perangkat android menggunakan *processor* Qualcomm SDM636, Ram 3GB dengan Android 9.0 *Pie*. Gambar 5 dibawah merupakan *deploy* diagram hubungan antara perangkat android *client* dengan *application server* dan *database*.

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505



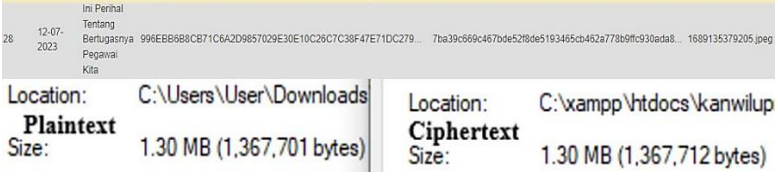
Gambar 5. Deploy Diagram

Berdasarkan analisa dan perancangan sistem yang telah dibuat sebelumnya, sistem diimplementasikan pada tahap ini. Berikut ini adalah gambar 6 merupakan tampilan input buku kontrol dari aplikasi keamanan penyuratan unit kepegawaian Kanwil Kemenag Riau yang berfungsi untuk menambah data kontrol baru dengan mengisi *form*, kunci enkripsi dan *file* lalu mengenkripsi sebelum menyimpan ke basis data.

Gambar 6. Tambah Data Kontrol

```
public String AESEncrypt(String string, String kunci) throws NoSuchAlgorithmException {
    try {
        cipher = Cipher.getInstance( transformation: "AES/ECB/PKCS5Padding");
    } catch (NoSuchAlgorithmException e) {
        e.printStackTrace();
    } catch (NoSuchPaddingException e) {
        e.printStackTrace();
    }
    String inputString = kunci;
    MessageDigest sha = MessageDigest.getInstance( algorithm: "SHA-256");
    byte[] byteArrray = sha.digest(inputString.getBytes(StandardCharsets.UTF_8));

    secretKeySpec = new SecretKeySpec(Arrays.copyOf(byteArrray,  newLength: 32),  algorithm: "AES");
    byte[] stringByte = string.getBytes();
    byte[] encryptedByte = new byte[stringByte.length];
    try {
        cipher.init(Cipher.ENCRYPT_MODE, secretKeySpec);
        encryptedByte = cipher.doFinal(stringByte);
```



Gambar 7. Kode Enkripsi

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

Setelah data dimasukkan, maka akan kembali ke halaman utama buku kontrol dengan daftar data yang sudah ditambah. Selain fitur kontrol, juga ada fitur surat tugas, surat perjalanan dinas dan user. Gambar 7 diatas adalah potongan kode pada implementasi enkripsi ke aplikasi di android studio. Potongan kode diatas menjelaskan proses enkripsi untuk data teks. Perbedaannya dengan *file* hanya pada tipe data, jika teks maka akan diterima sebagai *String* kemudian diubah ke *bytes array* sedangkan *file* langsung menggunakan *bytes array*. Setelah data dan kunci diterima maka kunci di *hashing* dengan SHA-256 dalam bentuk *bytes array* kemudian data dienkripsi dengan *cipher* AES-ECB dengan menggunakan kunci *hash* tersebut. Untuk model AES-ECB 128 dan AES-ECB 256 proses kunci akan langsung diubah ke *bytes array* dan langsung digunakan untuk proses enkripsi. Tampilan diatas juga menunjukkan ukuran *file* asli dan *file* setelah dienkripsi serta tampilan inputan sebelumnya setelah enkripsi di basis data yang menampilkan *output* enkripsi *ciphertext*.

3.3 Evaluasi

Pada hasil pengujian *Black Box* yang dilakukan pada aplikasi menghasilkan keluaran yaitu berjalannya aplikasi sesuai harapan dari tiap-tiap fungsi masukan sesuai dengan keluaran yang ditentukan. Pada hasil pengujian UAT di dapatkan nilai akhir sebesar 89 %. Total responden yang dilakukan sebanyak 5 orang dengan daftar pertanyaan yang terlihat pada tabel 1 sebagai berikut:

Tabel 1. Daftar Pertanyaan UAT

No	Pertanyaan
1	Tampilan sistem secara keseluruhan telah baik
2	Sistem memiliki menu yang mudah digunakan.
3	Sistem mudah dipahami
4	Fitur-fitur pada sistem berjalan dengan semestinya
5	Fungsi keamanan sistem berjalan dengan semestinya
6	Surat tugas dan surat perjalanan dinas menjadi lebih mudah disurati oleh pegawai dengan menggunakan sistem.

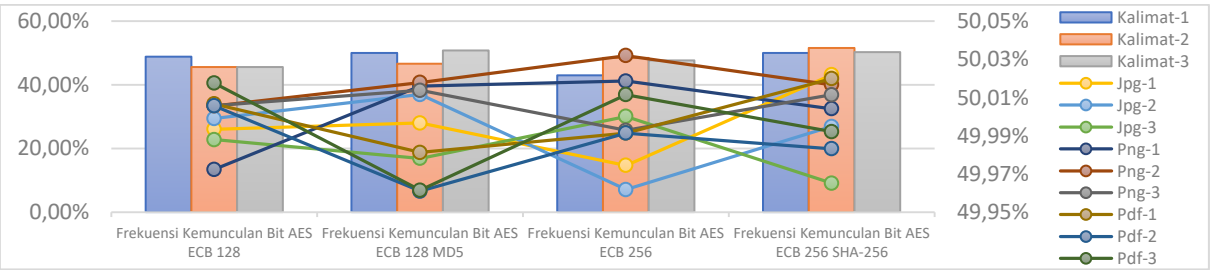
Berdasarkan pengujian *black box* dan UAT aplikasi penyuratan yang dibangun sudah layak untuk digunakan. Pengujian pada analisis statistik terdapat 12 data dengan 4 kategori yaitu berupa data teks, *file* gambar format Jpg, *file* gambar format Png dan *file* dokumen format Pdf. Berikut tabel 2 merupakan tabel keterangan dari data uji.

Tabel 2. Keterangan Data Uji

Kategori	1	2	3
Kalimat	Ini Perihal Tentang Bertugasnya (31 *Bytes*)	Ini Perihal Tentang Bertugasnya Pegawai (38 *Bytes*)	Ini Perihal Tentang Bertugasnya Pegawai Kita (44 *Bytes*)
Jpg	735 *Kilobytes*	0.97 *Megabytes*	1.30 *Megabytes*
Png	734 *Kilobytes*	0.97 *Megabytes*	1.30 *Megabytes*
Pdf	734 *Kilobytes*	0.97 *Megabytes*	1.30 *Megabytes*

Kunci AES 128/AES 128 Dengan MD5: KitaHarusTetap16
Kunci AES 256/AES 256 Dengan SHA-256: ApapunMasalahnyaJanganMenyerah32

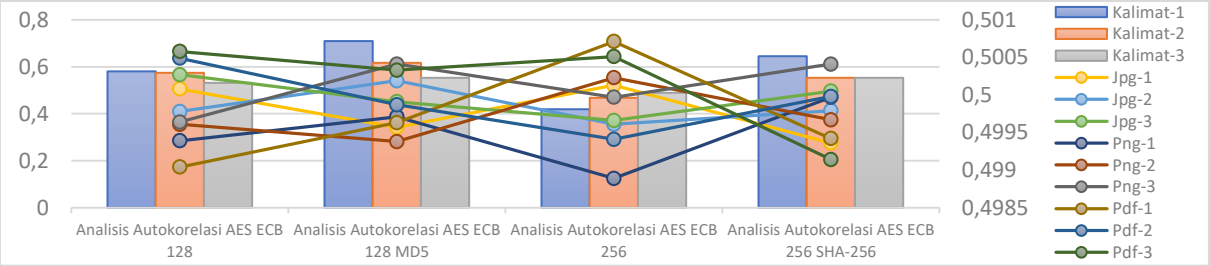
Pengujian analisis statistik yang dilakukan melibatkan frekuensi kemunculan bit, autokorelasi, distribusi bit 0 dan 1, entropi serta pengujian kecepatan enkripsi dan dekripsi dan perubahan bit menghasilkan grafik di bawah. Gambar 8 berikut menggambarkan grafik uji frekuensi kemunculan bit dari 4 model aplikasi yang dikembangkan yaitu:



Gambar 8. Frekuensi Kemunculan Bit

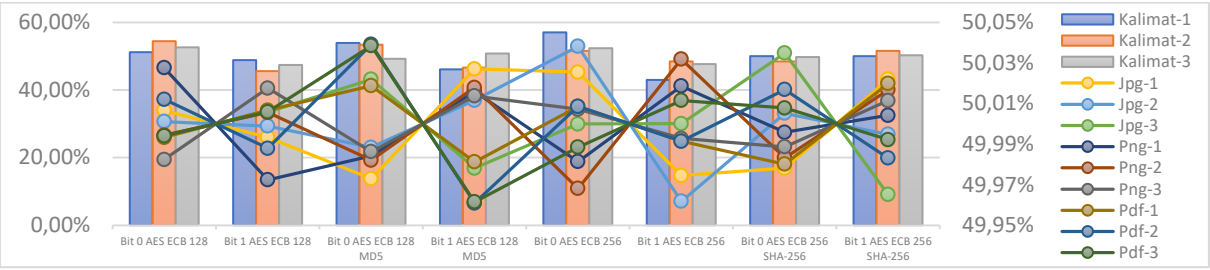
Frekuensi kemunculan bit dari berbagai jenis data dan variasi AES ECB berada dalam rentang 42.97% - 52% bit 0 dan 48% - 57.03% bit 1. Data teks memiliki frekuensi kemunculan bit yang lebih rendah dibandingkan dengan data *file*. Variasi fungsi *hash* dan kunci 256-bit memiliki frekuensi kemunculan bit yang lebih tinggi. Gambar 9 berikut menggambarkan grafik uji autokorelasi dari 4 model aplikasi yang dikembangkan yaitu:

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505



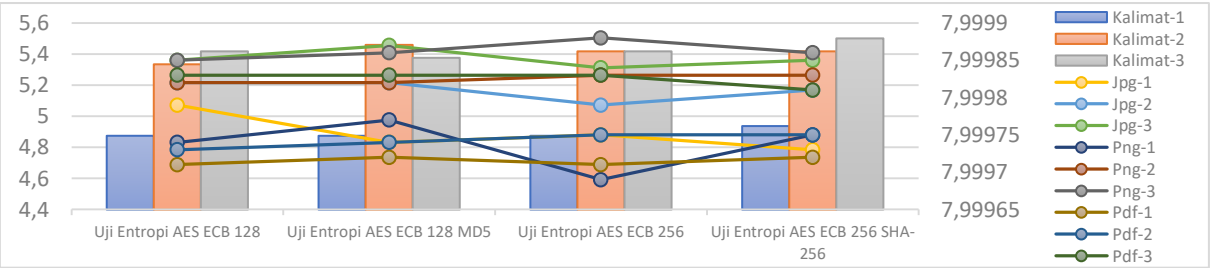
Gambar 9. Analisis Autokorelasi

Data terkompresi berupa *file* memiliki hubungan lemah antara nilai-nilai data yang berurutan, sedangkan data teks memiliki hubungan yang kuat. Variasi AES ECB tidak begitu mempengaruhi hasil autokorelasi karena dari hasil semua variasi memiliki nilai yang hampir serupa. Gambar 10 menggambarkan grafik uji analisis distribusi bit 0/1 dari 4 model aplikasi yang dikembangkan yaitu:



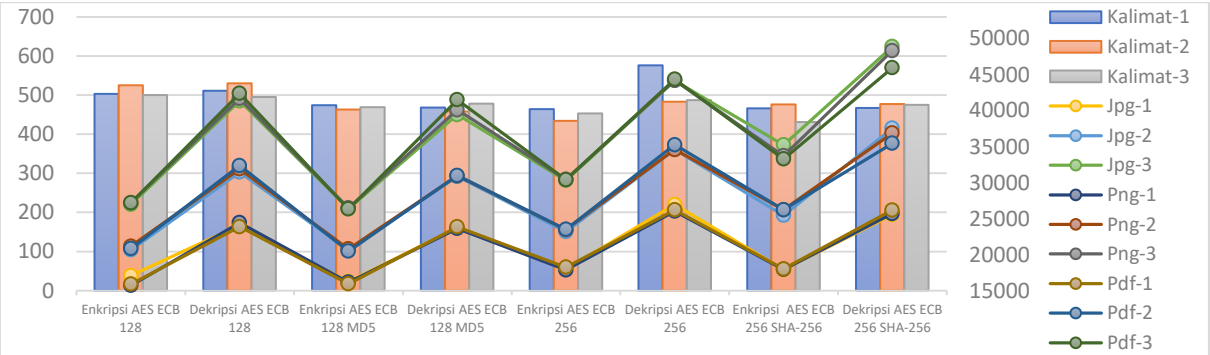
Gambar 10. Analisis Distribusi Bit 0/1

Data teks memiliki distribusi bit yang cukup tidak seimbang, berada dalam rentang 42.97% - 57.03% untuk bit 0 dan 42.97% - 57.03% untuk bit 1. Data *file* memiliki distribusi bit yang sangat seimbang, mendekati nilai ideal 50% untuk bit 0 dan bit 1. Variasi AES-ECB cukup mempengaruhi hasil data teks. Gambar 11 menggambarkan grafik uji entropi dari 4 model aplikasi yang dikembangkan yaitu:



Gambar 11. Uji Entropi

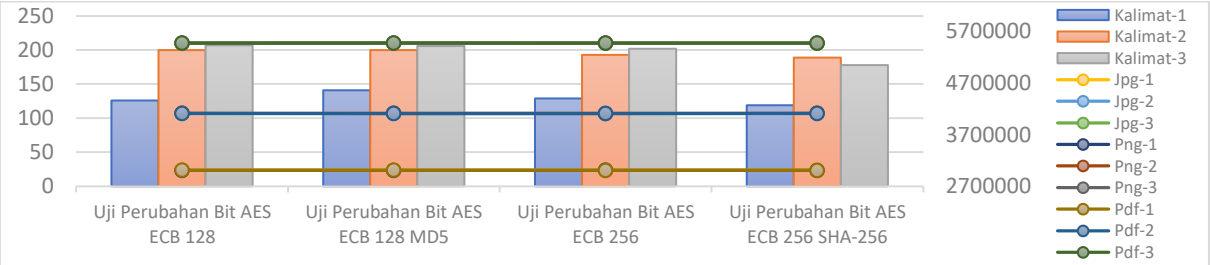
Data teks menghasilkan entropi yang cukup rendah, berada dalam rentang 4.875 sampai 5.5016. Data *file* menghasilkan entropi yang sangat tinggi, hampir mencapai nilai maksimum 8. Data *file* memiliki entropi tinggi, sedangkan teks memiliki entropi yang lebih rendah. Variasi AES-ECB tidak mempengaruhi hasil entropi. Gambar 12 menggambarkan grafik uji kecepatan enkripsi/dekripsi dari 4 model aplikasi yang dikembangkan yaitu:



Gambar 12. Uji Kecepatan Enkripsi/Dekripsi

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

Uji kecepatan menunjukkan data berukuran besar dan kompleks membutuhkan waktu enkripsi dan dekripsi yang lebih lama dibandingkan data berukuran kecil dan kurang kompleks. Variasi AES ECB mempengaruhi kecepatan perbedaan kunci 256 dan melalui proses *hashing*. Selain itu, proses dekripsi lebih lama daripada enkripsi. Gambar 13 menggambarkan grafik uji perubahan bit dari 4 model aplikasi yang dikembangkan yaitu:



Gambar 13. Uji Perubahan Bit

Perbandingan antara data teks dan data *file* yang tinggi menunjukkan bahwa data teks memiliki perubahan bit yang lebih rendah. Data *file* yang telah dikompresi lebih sensitif terhadap perubahan bit dibandingkan dengan data teks. Hasil uji perubahan bit tidak dipengaruhi oleh variasi AES ECB yang digunakan. Selain itu, pengujian pencarian kunci dilakukan dengan menggunakan *tools* CrypTool untuk menyerang empat model yang dikembangkan berdasarkan data kalimat ketiga dan *file* gambar jpg ketiga. Simulasi penyerangan dengan menggunakan Cryptool dapat digambarkan pada gambar 14 sebagai berikut:

Gambar 14. Pengujian Cryptool

Setelah memasukkan data dan kunci, lalu mengenkripsi sesuai dengan model-model yang dikembangkan, *ciphertext* yang didapat akan diserang untuk mencari kunci dengan menggunakan CrypTool. Penyerangan menggunakan dua jenis fungsi, yaitu fungsi entropi dan *known plaintext attack*. Fungsi entropi akan mencari kunci dengan metode *brute force* dengan menghitung entropi dari *plaintext* yang di dekripsi. *Plaintext* dengan entropi terendah akan berada di urutan pertama. Fungsi *known plaintext attack* akan menggunakan ekspresi regular untuk mencari kunci dengan asumsi ada kata yang diketahui. Berikut adalah tabel 3 merupakan kunci dalam bentuk heksadesimal. Kunci yang memiliki 16 karakter dan 32 karakter diasumsikan memiliki tiga komponen subruang dari *keyspace* yang kosong.

Tabel 3. Kunci Bentuk Heksadesimal

Kunci 16 Bytes	4B-69-74-61-48-61-72-75-73-54-65-74-61-**-**-**
Kunci 32 Bytes	41-70-61-70-75-6E-4D-61-73-61-6C-61-68-6E-79-61-4A-61-6E-67-61-6E-4D-65-6E-79-65-72-61-**-**-**

Hasil penyerangan pencarian kunci yang dilakukan adalah sebagai berikut. Fungsi entropi, data teks dan *file* yang menggunakan model AES 128 ECB dan AES 256 ECB berhasil menemukan kunci, tetapi model AES 128 ECB-MD5 dan AES 256 ECB-SHA 256 tidak berhasil. Fungsi *regex* (*known plaintext attack*), data teks yang menggunakan model AES 128 ECB dan AES 256 ECB berhasil menemukan kunci, tetapi model lainnya tidak berhasil. Dengan fungsi *regex*, data *file* untuk semua model tidak berhasil menemukan kunci. Tabel 4 berikut terlihat bentuk hasil dari beberapa fungsi tersebut.

Journal of Computer System and Informatics (JoSYC)
ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

Tabel 4. Hasil Penyerangan

Fungsi	Keterangan	Model
Fungsi Entropi (*Text*)	*Blocksize*: 1	AES 128 ECB: *Success Find Key*
	Bytes to use: 64	AES 256 ECB: *Success Find Key*
	Bytes offset: 0	AES 128 ECB-MD5: *Failed Find Key*
	Entropi *Function*: *External*	AES 256 ECB-SHA 256: *Failed Find Key*
Fungsi Entropi (*File* Jpg)	*Blocksize*: 1	AES 128 ECB: *Success Find Key*
	Bytes to use: 64	AES 256 ECB: *Success Find Key*
	Bytes offset: 0	AES 128 ECB-MD5: *Failed Find Key*
	Entropi *Function*: *External*	AES 256 ECB-SHA 256: *Failed Find Key*
Fungsi *Regex Text* (*Known plaintext attack*)	*Bytes to use*: 64	AES 128 ECB: *Success Find Key*
	Bytes offset: 0	AES 256 ECB: *Success Find Key*
	Regular Expression In Hex:	AES 128 ECB-MD5: *Failed Find Key*
	2E2A5065726968616C2E2A	AES 256 ECB-SHA 256: *Failed Find Key*
Fungsi *Regex File* Jpg (*Known plaintext attack*)	*Bytes to use*: 64	AES 128 ECB: *Failed Find Key*
	Bytes offset: 0	AES 256 ECB: *Failed Find Key*
	Regular Expression In Hex:	AES 128 ECB-MD5: *Failed Find Key*
	FFD8FFE000104A464946000101010060	AES 256 ECB-SHA 256: *Failed Find Key*

4. KESIMPULAN

Berdasarkan Dari hasil penelitian dapat disimpulkan bahwa aplikasi penyuratan yang dibangun telah diuji dengan *black box* dan UAT (*User Acceptance Test*) dan menunjukkan hasil yang sesuai dengan harapan. Pada pengujian UAT diperoleh nilai akhir yaitu 89 %. Pengujian aplikasi juga memiliki tingkat keamanan yang cukup tinggi berdasarkan analisis statistik yang dilakukan pada empat jenis keamanan aplikasi yang dikembangkan. Pada pengujian frekuensi bit diperoleh hasil hampir rata-rata 50%. Uji autokorelasi didapat hampir mendekati 0. Analisis distribusi bit 0 dan 1 diperoleh dengan masing-masing hampir mencapai rata-rata 50%. Uji entropi diperoleh hasil hampir mendekati rata-rata 8. Jenis format *file* berupa format Jpg, Png dan Pdf tidak begitu berbeda hasil-hasil yang didapatkan pada pengujian yang dilakukan. Kunci AES yang di *hashing* tidak berpengaruh signifikan terhadap hasil enkripsi dan kecepatan aplikasi serta perubahan *size* bit tidak begitu besar terjadi dari *size* awal, tetapi kunci AES yang di *hashing* dapat meningkatkan keamanan itu sendiri. Data pengujian juga didapatkan bahwa rata-rata kecepatan enkripsi lebih unggul daripada kecepatan dekripsi. Hasil analisis statistik menunjukkan bahwa kunci AES yang di hashing MD5/SHA-256 tidak memberikan perlindungan tambahan untuk enkripsi AES, namun konsep KDF dapat memberikan keamanan tambahan dengan cara mengubah kunci secara acak. Dengan menggunakan CrypTool, semua Model Hashing tidak dapat mencari kunci tersebut. Pada penelitian ini terdapat batasan yang dilakukan yaitu aplikasi yang dikembangkan bersistem operasi android dan implementasi digunakan hanya metode *waterfall* serta pengujian dilakukan hanya menggunakan satu perangkat android berupa pengujian analisis statistik, entropi serta kecepatan enkripsi dan dekripsi. Pada 5 pengujian analisis statistik tidak dilakukan pengujian serial, *poker* dan *run*.

REFERENCES

[1] H. Pasaribu, D. Sitanggang, R. R. Damanik, dan A. C. R. Sitompul, "Combination Of Advanced Encryption Standard 256 Bits With MD5 to Secure Documents On Android Smartphone," *IOP Conf. Series: Journal of Physics*, vol. 1007, no. 1, hlm. 1–8, 2018, doi: 10.1088/1742-6596/1007/1/012014.

[2] G. G. P.U.K, Ernawati, dan A. Erlanshari, "Implementasi Metode Advanced Encryption Standard (AES) Dan Message Digest 5 (MD5) Pada Enkripsi Dokumen (Studi Kasus LPSE UNIB)," *Jurnal Rekursif*, vol. 4, no. 3, hlm. 277–287, 2016, doi: 10.33369/rekursif.v4i3.864.

[3] A. Fathurrozi dan Selviyani, "Penerapan Algoritma Advanced Encryption Standard (AES256) Dengan Mode CBC DanSecure Hash Algorithm (SHA-256) Untuk Pengamanan Data File," *Journal of Information and Information Security (JIFORTY)*, vol. 2, no. 2, hlm. 227–238, 2021, Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://ejurnal.ubharajaya.ac.id/index.php/jiforty/article/view/879

[4] S. Sulastri dan R. D. M. Putri, "Implementasi Enkripsi Data Secure Hash Algorithm (SHA-256) dan Message Digest Algorithm (MD5) pada Proses Pengamanan Kata Sandi Sistem Penjadwalan Karyawan," *Jurnal Teknik Elektro*, vol. 10, no. 2, hlm. 70–74, 2018, doi: 10.15294/jte.v10i2.18628.

[5] I. N. Abdullah, D. Kusumaningsih, dan M. Alawy, "Aplikasi Enkripsi File Dokumen Menggunakan Metode Algoritma AES (Advanced Encryption Standard) Dan OTP (One Time Pad) Berbasis Web Pada PT. MNC Sky Vision," *Telematika MKOM*, vol. 10, no. 1, hlm. 11–16, 2018, doi: 10.36080/telematikamkom.654.

ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

[6] F. Nuraeni, Y. H. Agustin, dan A. E. Purnama, "Implementasi Caesar Cipher And Advanced Encryption Standard (AES) Pada Pengamanan Data Pajak Bumi Bangunan," *Jurnal Ilmiah Matrik*, vol. 22, no. 2, hlm. 188–194, 2020, doi: 10.33557/jurnalmatrik.v22i2.949.

[7] A. G. Pramudito dan D. Kusumaningsih, "Implementasi Algoritma AES 128 Dan RC4 Untuk Pengamanan Email Pada PT. Dinamika Hydro Engineering," *SKANIKA*, vol. 1, no. 3, hlm. 869–876, 2018, Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://jom.fti.budiluhur.ac.id/index.php/SKANIKA/article/view/2499

[8] A. Hermawan dan E. I. H. Ujianto, "Implementasi Enkripsi Data Menggunakan Kombinasi AES Dan RSA," *Jurnal Nasional Informatika Dan Teknologi Jaringan*, vol. 5, no. 2, hlm. 325–330, 2021, doi: 10.30743/infotekjar.v5i2.3585.

[9] F. Kurnia, M. Fikry, dan F. Febriyadi, "Rancang Bangun Sistem Informasi Penyuratan Unit Kepegawaian Kantor Wilayah Kementerian Agama Provinsi Riau," *Jurnal Ilmiah Rekayasa Dan Manajemen Sistem Informasi*, vol. 8, no. 2, hlm. 180–188, 2022, doi: 10.24014/rmsi.v8i2.18208.

[10] R. Ravida dan H. A. Santoso, "Advanced Encryption Standard (AES) 128 Bit untuk Keamanan Data Internet of Things (IoT) Tanaman Hidroponik," *Jurnal Rekayasa Sistem dan Teknologi Informasi (RESTI)*, vol. 4, no. 6, hlm. 1157–1164, 2019, doi: 10.29207/resti.v4i6.2478.

[11] D. Hulu, B. Nadeak, dan S. Aripin, "Implementasi Algoritma AES (Advanced Encryption Standard) Untuk Keamanan File Hasil Radiologi di RSU Imelda Medan," *Konferensi Nasional Teknologi Informasi Dan Komputer*, vol. 4, no. 1, hlm. 78–86, 2020, doi: 10.30865/komik.v4i1.2645.

[12] A. P. Putra, Herfina, S. Maryana, dan A. Setiawan, "Implementasi Algoritma AES (Advanced Encryption Standard) Rijndael Pada Aplikasi Keamanan Data," *Jurnal Ilmiah Penelitian Teknologi Informasi Dan Komputer*, vol. 1, no. 2, hlm. 46–51, 2020, Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://journal.upgris.ac.id/index.php/jipetik/article/view/KAI

[13] J. Handoyo dan Y. M. Subakti, "Keamanan Dokumen Menggunakan Algoritma Advanced Encryption Standard (AES)," *Jurnal Sistem Informasi Dan Teknologi*, vol. 3, no. 2, hlm. 143–152, 2020, doi: 10.24176/sitech.v3i2.5865.

[14] C. Kirana dan E. Sugianto, "Penerapan Algoritma AES dan Konversi SMS Ke Dalam Bahasa Khek pada Aplikasi Enkripsi Berbasis Mobile Application," *Jurnal Ilmu Komputer Dan Informatika*, vol. 5, no. 1, hlm. 68–77, 2019, doi: 10.23917/khif.v5i1.7453.

[15] A. A. Permana dan D. Nurnaningsih, "Rancangan Aplikasi Pengamanan Data Dengan Algoritma Advanced Encyption Standard (AES)," *Jurnal Teknik Informatika*, vol. 11, no. 2, hlm. 177–186, 2018, doi: 10.15408/jti.v11i2.7811.

[16] I. Fitriani dan A. B. Utomo, "Implementasi Algoritma Advanced Encryption Standard (AES) Pada Layanan SMS Desa," *JISKa*, vol. 5, no. 3, hlm. 153–163, 2020, doi: 10.14421/jiska.2020.53-03.

[17] A. E. Putri, A. Kartikadewi, dan L. A. A. Rosyid, "Implementasi Kriptografi Dengan Algoritma Advanced Encryption Standard (AES) 128 Bit Dan Steganografi Menggunakan Metode End Of File (EOF) Berbasis Java Desktop Pada Dinas Pendidikan Kabupaten Tangerang," *Applied Information Systems and Management (AISM)*, vol. 3, no. 2, hlm. 69–78, 2020, doi: 10.15408/aism.v3i2.14722.

[18] Ahyuna dan S. Hozeng, "Perancangan Aplikasi Enkripsi Menggunakan Algoritma AES Berbasis Android," dalam *Prosiding Seminar Nasional Komunikasi Dan Informatika #3*, 2019, hlm. 130–135. Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://jurnal.kominfo.go.id/index.php/snki/article/view/2569

[19] E. Ali, *Rekayasa Perangkat Lunak*. Yogyakarta: MFA, 2019.

[20] A. R. Mulyanto, *Rekayasa Perangkat Lunak Jilid 1*. Jakarta: Direktorat Pembinaan Sekolah Menengah Kejuruan, 2008. Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: http://opac.salatigakota.go.id/ucs/index.php?p=show_detail&id=33903

[21] Widiyawati *dkk.*, *Rekayasa Perangkat Lunak*. Bandung: Widina Bhakti Persada Bandung, 2022. Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://repository.penerbitwidina.com/publications/410361/rekayasa-perangkat-lunak

[22] M. Armstrong, "Institutional Repository Management Models That Support Faculty Research Dissemination," *Albertsons Library*, vol. 30, no. 1, hlm. 1–3, 2013, doi: 10.1108/OCLC-07-2013-0028.

[23] M. C. Sinaga, *Kriptografi Python*. Medan: Academia, 2017. doi: 10.31227/osf.io/6su2h.

[24] B. Rahardjo, *Keamanan Informasi*. Bandung: Insan Infonesia, 2017. Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://budi.rahardjo.id/files/keamanan.pdf

[25] S. Kromodimoeljo, *Teori dan Aplikasi Kriptografi*. Jakarta: SPK IT Consulting, 2009. Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://lmsspada.kemdikbud.go.id/pluginfile.php/548262/mod_resource/content/1/crypto-book-complete.pdf

[26] J. Daemen dan V. Rijmen, *The Design of Rijndael The Advanced Encryption Standard (AES) Second Edition*. Berlin: Springer, 2002. doi: 10.1007/978-3-662-60769-5.

[27] NIST, "NIST Special Publication 800-38A," Maryland, 2001. doi: 10.6028/NIST.SP.800-38A.

ISSN 2714-8912 (media online), ISSN 2714-7150 (media cetak)
Volume 5, No. 1, November 2023, Page 113-126
https://ejurnal.seminar-id.com/index.php/josyc
DOI 10.47065/josyc.v5i1.4505

[28] E. Z. A. Kashogi, "Algoritma Message Digest 5 (MD5)," *Institut Teknologi Bandung Makalah0607-116*, hlm. 1–11, 2021, Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2006-2007/Makalah/Makalah0607-116.pdf

[29] NIST, "Federal Information Processing Standards (FIPS) Publication 180-4," Maryland, 2015. doi: 10.6028/NIST.FIPS.180-4.

[30] F. N. Hasanah dan R. S. Untar, *Rekayasa Perangkat Lunak*. Sidoarjo: Umsida Press, 2020. doi: 10.21070/2020/978-623-6833-89-6.

[31] W. E. Perry, *Effective Methods for Software Testing*. Indianapolis: Wiley Publishing, 2006. Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://www.academia.edu/8858693/Effective_Methods_for_Software_Testing_Third_Edition_Effective_Methods_for_Software_Testing_Third_Edition

[32] A. J. Menezes, P. C. van Oorschot, dan S. A. Vanstone, *Handbook of Applied Cryptography*. Cambridge: Massachusetts Institute of Technology, 1996. Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://cacr.uwaterloo.ca/hac/

[33] NIST, "The Testing of Entropy Sources for Cryptography," Maryland, 2014. Diakses: 16 Oktober 2023. [Daring]. Tersedia pada: https://www.sarahscheffler.net/2014/scheffler_entropy.pdf

[34] C. Cachin, "Entropy Measures and Unconditional Security in Cryptography," Swiss Federal Institute Of Technology Zurich, Swiss, 1997. doi: 10.3929/ethz-a-001806220.

[35] A. Anwar, *Statistika Untuk Penelitian Pendidikan Dan Aplikasinya Dengan SPSS Dan Excel*. Kediri: IAIT Press, 2009. Diakses: 24 Oktober 2023. [Daring]. Tersedia pada: http://repository.iainkediri.ac.id/25/1/Ali%20Anwar_Statistika%20untuk%20Penelitian%20Pendidikan.pdf

[36] R. Yang, L. Wallace, dan I. Burchett, "Teaching Cryptology At All Levels Using CrypTool," dalam *Proceedings of the 15 Colloquium for Information Systems Security Education*, Ohio: CISSE, 2011, hlm. 22–28. [Daring]. Tersedia pada: https://www.cryptool.org/assets/ctp/documents/teachingcryptool.pdf