BAB II

TINJAUAN LITERATUR

2.1 Penelitian Terkait

Pustaka merupakan acuan yang digunakan peneliti untuk memahami dan memperdalam temuan-temuan dari penelitian sebelumnya yang memiliki relevansi. Dengan merujuk pada karya ilmiah terdahulu, peneliti dapat mengutip pendapat maupun hasil penelitian yang mendukung studi yang sedang dilakukan. Langkah ini penting sebagai landasan teoritis dan metodologis, serta menunjukkan adanya hubungan antara penelitian sekarang dengan studi-studi sebelumnya.

1.) Dalam penelitian oleh (Aryanto et al., 2023)mereka mengkaji penerapan algoritma AES-128 untuk enkripsi dan dekripsi file pada aplikasi Android. Fokus penelitian ini adalah pada keamanan data yang disimpan dan ditransmisikan melalui perangkat mobile, dengan tujuan untuk meningkatkan perlindungan data sensitif. Penelitian ini relevan bagi aplikasi mobile yang menangani data penting, seperti transaksi atau komunikasi pribadi, karena penggunaan AES-128 dapat meningkatkan tingkat keamanan terhadap potensi kebocoran data(Aryanto et al., 2023)

2.) Penelitian yang dilakukan oleh Putra Utama et al. (2023) mengusulkan penggunaan algoritma AES 256 CBC, Base64, dan SHA 256 dalam sistem ujian online untuk memastikan keamanan dan integritas data ujian. Mereka

mengimplementasikan teknik enkripsi untuk menghindari kecurangan serta memastikan bahwa data ujian yang dikirimkan melalui jaringan tetap aman. Penelitian ini relevan bagi sistem pendidikan dan ujian online, karena dapat memperkuat perlindungan terhadap data ujian dan memvalidasi integritasnya (Putra Utama et al., 2023)

3.) Dalam studi oleh Taufiqur Rohman & Romli (2021), mereka mengkaji penggunaan algoritma Base64 untuk encoding gambar dalam aplikasi kriptografi berbasis Android. Penelitian ini bertujuan untuk mengamankan data visual yang dikirimkan melalui perangkat mobile, dengan menjamin bahwa gambar terenkripsi tetap dapat ditransmisikan dalam format yang kompatibel. Penelitian ini sangat relevan untuk aplikasi yang mengirimkan gambar atau file media lainnya, seperti aplikasi pesan instan atau berbagi foto, yang membutuhkan lapisan keamanan tambahan (Taufiqur Rohman & Romli, 2021)

4.) Penelitian oleh (Winata et al., 2024) membahas implementasi algoritma AES-128 untuk pengamanan data pada sistem berbasis web di restoran McDonald's Cabang T.B. Simatupang. Mereka mengimplementasikan enkripsi untuk melindungi data pelanggan dan transaksi yang bersifat sensitif. Penelitian ini relevan bagi industri restoran atau bisnis berbasis web yang menangani data pribadi pelanggan, karena dapat memberikan solusi untuk meningkatkan tingkat keamanan data transaksi yang diproses(Winata et al., 2024)

5.) Dalam penelitian yang dilakukan oleh (Destriyani & Painem, 2023) mereka mengembangkan aplikasi Android yang mengimplementasikan algoritma AES-128 dan *Blowfish* untuk fitur chat satu lawan satu di aplikasi Blucampus.

Penelitian ini bertujuan untuk meningkatkan keamanan komunikasi antar pengguna dengan menggunakan enkripsi yang kuat. Penelitian ini relevan untuk aplikasi komunikasi seperti chat dan pesan instan, karena dapat menyediakan solusi keamanan yang lebih baik untuk melindungi percakapan pengguna dari potensi ancaman atau penyadapan (Destriyani & Painem, 2023).

2.2 Algoritma Aes-128

Penelitian berjudul "*Securing Messages Using AES Algorithm and Blockchain Technology on Mobile Devices*" oleh Al Farissi, Pradata, dan Miraswan (Al Farissi; Arya Pradata; Kanda Miraswan, 2020) membahas solusi keamanan pesan pada perangkat mobile dengan menggabungkan algoritma kriptografi AES-128 dan teknologi blockchain. AES-128 digunakan untuk mengenkripsi isi pesan agar tidak dapat dibaca oleh pihak yang tidak berwenang, sedangkan blockchain dimanfaatkan untuk mencatat dan memverifikasi integritas pesan tersebut. Kombinasi ini dirancang untuk menjaga kerahasiaan dan keaslian pesan tanpa bergantung pada pihak ketiga, serta tetap efisien dijalankan pada perangkat dengan keterbatasan sumber daya.

Hasil pengujian menunjukkan bahwa proses enkripsi menggunakan AES-128 berlangsung sangat cepat, dengan waktu rata-rata hanya sekitar 33,6 milidetik. Selain itu, pengujian efek avalanche menunjukkan perubahan ciphertext yang signifikan meski hanya ada sedikit perubahan pada plaintext, menandakan sistem memiliki tingkat kerahasiaan yang baik. Penelitian ini menunjukkan bahwa integrasi AES-128 dan blockchain merupakan pendekatan yang efektif dan praktis

dalam meningkatkan keamanan pesan digital, terutama untuk kebutuhan komunikasi rahasia di era mobile yang rentan terhadap serangan siber.

2.3 Algoritma Base64

Algoritma Base64 merupakan metode encoding yang banyak digunakan untuk mengamankan data berbasis teks, terutama dalam proses pengiriman pesan melalui media yang tidak aman. Dalam penelitian berjudul "*Implementation of the Base64 Algorithm for Text Encryption And Decryption Using the Python Programming Language*", Caroko Aji Pamungkas dan tim (2025) mengembangkan sebuah aplikasi sederhana yang memungkinkan pengguna untuk mengenkripsi dan mendekripsi pesan teks menggunakan algoritma Base64. Penelitian ini menekankan bahwa meskipun Base64 bukan metode kriptografi yang kompleks, namun dapat menjadi lapisan perlindungan awal dalam menjaga kerahasiaan data teks sebelum dikirim ke penerima.(Febitri et al., 2023; Witriyono & Fernandez, 2021)

Penelitian ini dibuat dengan tujuan edukatif dan implementatif, menggunakan bahasa pemrograman Python untuk mempermudah proses encoding dan decoding teks. Aplikasi yang dibuat bersifat open source dan dapat digunakan untuk kebutuhan dasar seperti pengiriman pesan pribadi atau penyimpanan data ringan yang tidak boleh langsung terlihat. Hasilnya menunjukkan bahwa metode ini sangat cepat dijalankan dan memiliki kompatibilitas yang tinggi dengan berbagai format teks. Meskipun algoritma ini tidak cukup kuat untuk keamanan tingkat tinggi, penggunaannya dalam sistem lapis awal atau bersama algoritma lain seperti AES dapat meningkatkan keamanan pesan secara keseluruhan.

2.4 Hashing Bcrypt

Algoritma bcrypt adalah fungsi hashing berbasis password yang dirancang untuk meningkatkan keamanan penyimpanan kata sandi melalui penggunaan salt acak dan parameter cost yang dapat disesuaikan. Salt mencegah serangan *rainbow-table*, sementara cost meningkatkan jumlah iterasi hashing sehingga memperlambat proses pencarian kata sandi oleh penyerang . Karena bcrypt adaptif, seiring peningkatan kekuatan komputasi, sistem dapat menaikkan cost untuk menjaga keamanan tetap tangguh terhadap serangan *brute-force* masa depan.

Beberapa studi di tahun-tahun terakhir menunjukkan bahwa bcrypt mampu menghadang serangan brute-force secara efektif, terutama untuk kombinasi karakter campuran dengan panjang tertentu. Misalnya, menurut penelitian oleh Batubara dkk. (2021), *plaintext* lima karakter alfanumerik tidak berhasil di-crack dalam waktu 5 hari pada tingkat cost tertentu, sementara plaintext campuran tujuh karakter masih belum berhasil dipulihkan dalam periode tersebut .Ini menegaskan bahwa bcrypt tetap menjadi pilihan yang kuat untuk penyimpanan kredensial pengguna, meskipun algoritma password hashing modern seperti Argon2 dan scrypt kini menawarkan keamanan yang lebih tinggi untuk penggunaan memori-hard.

2.5 Bahasa Vue Js

Vue.js adalah framework JavaScript progresif yang dirancang untuk membangun antarmuka pengguna (*user interface*) yang interaktif dan responsif. Vue bersifat ringan, fleksibel, dan mudah diintegrasikan dengan berbagai teknologi backend seperti Node.js, Java, atau PHP. Dengan konsep komponen, binding data,

dan reaktivitas, Vue sangat cocok untuk membangun aplikasi web modern, termasuk yang memiliki fitur keamanan seperti otentikasi, enkripsi, dan proteksi data pengguna.

Dalam aplikasi Vue.js, integrasi dengan algoritma kriptografi seperti AES-128 sering dilakukan melalui *library CryptoJS*, yang memungkinkan enkripsi dan dekripsi data langsung di sisi klien. Artikel oleh Jha Pratik (2023) menunjukkan bagaimana Vue.js memanggil fungsi *CryptoJS.AES.encrypt* dan *decrypt* dalam metode *encrypteData* dan *decrypteData*, menggunakan teknik derivasi kunci PBKDF2 dan mode CBC, yang memungkinkan pesan rahasia dienkripsi sebelum dikirim ke backend atau disimpan dalam *local storage.*

Sedangkan untuk hashing password dengan bcrypt dalam konteks *Vue.js*, panduan oleh (Crudu & Ana, 2025)merekomendasikan penggunaan bcrypt di backend ketika membangun sistem autentikasi dengan *Passport.js. Vue.js* digunakan di frontend untuk mengelola login form dan sesi pengguna (*Vuex),* sementara hashing dilakukan secara aman di server menggunakan *bcrypt.hashSync(),* yang mengurangi risiko kebocoran kredensial dan kompatibilitas dengan standar keamanan modern.

2.6 Pesan Rahasia

Pesan rahasia merupakan inti dari keamanan informasi dalam era digital saat ini, di mana data pribadi, transaksi, maupun komunikasi sensitif harus dijaga agar tidak jatuh ke tangan yang salah. Berbagai algoritma kriptografi seperti AES-128, Base64, dan bcrypt telah banyak digunakan untuk melindungi data dari potensi

serangan. Salah satu penelitian yang relevan adalah yang dilakukan oleh Al Farissi et al. (2023), yang menggabungkan algoritma AES dengan teknologi blockchain untuk mengamankan pesan dalam perangkat mobile. Penelitian ini membuktikan bahwa kombinasi teknologi tersebut mampu menjaga kerahasiaan dan integritas pesan dengan tingkat efisiensi yang baik. Selain itu, penggunaan teknik encoding Base64 juga dibahas dalam jurnal oleh (Kurniawan & Hidayatullah, 2022) sebagai pelengkap dalam pengolahan string terenkripsi.

Untuk mendukung implementasi frontend dari sistem keamanan ini, Vue.js digunakan sebagai kerangka kerja antarmuka yang interaktif dan ringan. Sebuah artikel teknis oleh Crudu (2025) menguraikan bagaimana Vue.js dapat diintegrasikan dengan pustaka seperti CryptoJS untuk mengenkripsi pesan secara langsung di sisi klien menggunakan AES, kemudian mengirimkannya ke backend yang memverifikasi dengan hash bcrypt. Pendekatan ini sangat relevan dalam pengembangan aplikasi modern yang mengutamakan keamanan dari sisi pengguna hingga server. Penggunaan Vue.js juga memberikan fleksibilitas dalam pengelolaan data terenkripsi sebelum dikirim ke backend melalui API yang aman.