Jurnal TICOM: Technology of Information and Communication
Volume 14, Nomor 3, Mei 2026 Hal: 183-187 E-ISSN: 2962-7982

Implementasi Algoritma AES-256 dan Arsitektur *Zero-Knowledge* pada Sistem Catatan Rahasia Berbasis Web

Manda Fatimah Azaziah[1], Febriana Nur Aini[2], Muhammad Rifki Iqbal G[3], Muhammad Dava Khoirur[4], Azis Suroni[5]

[1,2,3]Fakultas Teknik, Program Studi Teknik Informatika, Universitas Negeri Surabaya, Surabaya, Indonesia
Jl. Ketintang, Kel. Ketintang, Kec. Gayungan, Kota Surabaya, Jawa Timur 60231
E-mail: [1]25051204310@mhs.unesa.ac.id, [2]25051204381@mhs.unesa.ac.id, [3]25051204277@mhs.unesa.ac.id,
[4]25051204255@mhs.unesa.ac.id, [5]azissuroni@unesa.ac.id
(*: corresponding author)

Abstrak— **Penyimpanan informasi sensitif pada aplikasi catatan digital menimbulkan tantangan terkait keamanan dan privasi data pengguna. Sebagian besar sistem penyimpanan konvensional masih memberikan akses terhadap data yang disimpan pada sisi *backend*, sehingga meningkatkan risiko kebocoran informasi apabila terjadi kompromi sistem. Penelitian ini bertujuan untuk merancang dan mengimplementasikan aplikasi web Secret Ink dengan mengintegrasikan algoritma *Advanced Encryption Standard* (AES) 256-bit dan prinsip *Zero-Knowledge* sebagai mekanisme perlindungan data. Metode penelitian yang digunakan meliputi analisis kebutuhan, perancangan arsitektur keamanan, implementasi sistem menggunakan teknologi berbasis JavaScript, serta pengujian fungsionalitas dan keamanan aplikasi. Hasil penelitian menunjukkan bahwa proses enkripsi dan dekripsi dapat dilakukan pada sisi pengguna, sementara *backend* hanya menerima dan menyimpan data dalam bentuk *ciphertext*. Pengujian keamanan juga menunjukkan bahwa sistem mampu memitigasi ancaman umum aplikasi web, seperti *Cross-Site Request Forgery* (CSRF), *Cross-Site Scripting* (XSS), dan *SQL Injection*. Dengan demikian, Secret Ink berhasil menyediakan mekanisme penyimpanan catatan digital yang mampu menjaga kerahasiaan dan privasi data pengguna melalui penerapan AES-256 dan arsitektur *Zero-Knowledge*.**

Kata Kunci—**Keamanan Web, Kriptografi, AES-256, Arsitektur Zero-Knowledge, Catatan Digital**

Abstract— ***Storing sensitive information in digital note-taking applications poses challenges related to the security and privacy of user data. Most conventional storage systems still provide access to data stored on the server side, increasing the risk of information leakage in the event of a system compromise. This research aims to design and implement the Secret Ink web application by integrating the Advanced Encryption Standard (AES) 256-bit algorithm and the Zero-Knowledge principle as a data protection mechanism. The research methods used include requirements analysis, security architecture design, system implementation using JavaScript-based technology, and testing the application's functionality and security. The results show that the encryption and decryption processes can be performed on the user side, while the backend only receives and stores data in ciphertext form. Security testing also shows that the system is able to mitigate common web application threats, such as Cross-Site Request Forgery (CSRF), Cross-Site Scripting (XSS), and SQL Injection. Thus, Secret Ink successfully provides a digital note-taking mechanism that is able to maintain the confidentiality and privacy of user data through the implementation of AES-256 and the Zero-Knowledge architecture.***

Keywords—***Web Security, Cryptography, AES-256, Zero-Knowledge Architecture, Digital Notes***

I. PENDAHULUAN

Perkembangan teknologi digital telah mengubah paradigma pengelolaan informasi, di mana pengguna secara masif menyimpan data sensitif seperti kata sandi, catatan pribadi, dan dokumen rahasia pada platform digital. Meskipun memberikan kemudahan akses, transisi ini memunculkan kerentanan signifikan terhadap ancaman siber dan kebocoran data pribadi pada ruang penyimpanan berbasis internet [1]. Sebagian besar platform penyimpanan konvensional beroperasi dengan arsitektur yang memungkinkan penyedia layanan memiliki akses langsung terhadap data pengguna. Kondisi tersebut menimbulkan risiko yang tinggi, apabila *backend* (server) mengalami peretasan, informasi sensitif pengguna dapat terekspos dan dieksploitasi. Oleh karena itu, diperlukan suatu mekanisme perlindungan data yang komprehensif guna menjamin kerahasiaan informasi secara mutlak.

Dalam upaya memitigasi risiko keamanan tersebut, kriptografi menjadi pendekatan utama untuk melindungi data digital. Algoritma *Advanced Encryption Standard* (AES) dengan panjang kunci 256-bit merupakan standar enkripsi simetris yang terbukti memiliki tingkat keamanan yang tangguh terhadap upaya peretasan [2], [3]. Namun, implementasi enkripsi semata belum memberikan jaminan privasi yang optimal apabila pihak penyedia layanan masih memiliki kendali atas kunci dekripsi. Untuk mengatasi celah keamanan tersebut, penerapan arsitektur *Zero-Knowledge* menjadi sangat esensial [4]. Melalui arsitektur ini, proses enkripsi dan dekripsi dilakukan sepenuhnya secara mandiri di sisi *user* (pengguna). *Backend* hanya menerima serta menyimpan data dalam format tersandi (*ciphertext*), sehingga penyedia layanan tidak memiliki akses maupun pengetahuan terhadap data asli (*plaintext*).

Berdasarkan permasalahan tersebut, penelitian ini mengusulkan pengembangan Secret Ink, yakni sebuah sistem catatan rahasia berbasis *web* yang mengintegrasikan algoritma

enkripsi AES-256 dengan arsitektur *Zero-Knowledge*. Selain pengamanan pada level data, perlindungan terhadap ekosistem aplikasi web secara keseluruhan juga merupakan aspek yang krusial, mengingat celah seperti XSS maupun injeksi basis data kini semakin bervariasi dan dapat dieksploitasi dari berbagai sisi aplikasi [5], [6]. Oleh sebab itu, pengembangan Secret Ink juga difokuskan pada penguatan mekanisme keamanan aplikasi *web* terhadap berbagai ancaman eksternal.

Penelitian ini bertujuan untuk merancang dan mengembangkan aplikasi web Secret Ink sebagai media penyimpanan catatan digital yang mengutamakan privasi dan keamanan data pengguna. Sistem dibangun dengan menerapkan algoritma kriptografi *Advanced Encryption Standard* (AES) 256-bit serta prinsip *Zero-Knowledge* untuk memastikan kerahasiaan informasi yang disimpan. Selain itu, penelitian ini juga mengevaluasi kinerja dan keamanan sistem guna memastikan bahwa mekanisme perlindungan yang diterapkan mampu mendukung penyimpanan data secara aman pada lingkungan aplikasi web. Hasil penelitian diharapkan dapat memberikan kontribusi berupa model aplikasi catatan digital yang mampu menjaga kerahasiaan data pengguna sekaligus meningkatkan ketahanan sistem terhadap berbagai ancaman keamanan siber.

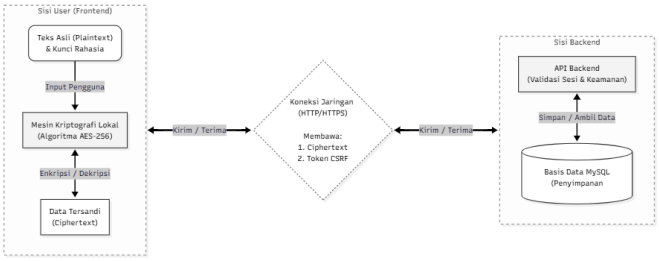
II. METODE PENELITIAN

Penelitian ini difokuskan pada perancangan dan pengembangan aplikasi catatan rahasia berbasis *web* yang diberi nama Secret Ink. Pengembangan sistem menggunakan metode *Waterfall* [7]. Fase pengembangannya meliputi tahapan analisis kebutuhan, perancangan sistem, implementasi, dan pengujian.

A. Spesifikasi Perangkat Lunak dan Kriptografi

Sebagai bentuk implementasi dari tahap analisis kebutuhan dan perancangan sistem, aplikasi web Secret Ink dikembangkan menggunakan ekosistem teknologi JavaScript yang terintegrasi. Pengelolaan logika aplikasi pada sisi *backend* serta layanan API dan sesi pengguna diproteksi menggunakan integrasi token *JSON Web Token* (JWT) tersandi [8]. Pada area *backend* ini, perlindungan akses terhadap *Application Programming Interface* (API) diimplementasikan melalui mekanisme *token* autentikasi personal yang dikombinasikan dengan standar enkripsi kunci simetris yang kuat, guna mencegah intervensi pihak luar terhadap sesi komunikasi system [9].

Perlindungan privasi data pada aplikasi web ini diterapkan melalui algoritma kriptografi simetris *Advanced Encryption Standard* (AES) 256-bit. Implementasi algoritma tersebut dikombinasikan dengan prinsip *Zero-Knowledge*, yang dirancang untuk memastikan bahwa data sensitif tidak dapat diakses maupun diketahui oleh *backend* pusat [4]. Untuk mendukung tujuan tersebut, sistem menerapkan pemisahan batas keamanan (*security boundary*) sehingga proses enkripsi dan dekripsi dilakukan tanpa menyebabkan kebocoran informasi selama transmisi data. Arsitektur perlindungan data yang diterapkan pada aplikasi web Secret Ink ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur Sistem Secret Ink

Berdasarkan Gambar 1, arsitektur keamanan sistem dirancang dengan pemisahan ke dalam tiga zona utama. Pada sisi pengguna (*client-side*), teks asli (*plaintext*) dan kunci rahasia diproses secara lokal melalui mekanisme enkripsi AES-256 untuk menghasilkan data tersandi (*ciphertext*), sehingga kunci enkripsi tidak pernah dikirimkan ke luar peramban pengguna. Pada zona komunikasi jaringan, sistem memastikan bahwa data yang ditransmisikan melalui protokol HTTP/HTTPS hanya berupa *ciphertext* beserta token keamanan yang diperlukan, seperti *Cross-Site Request Forgery* (CSRF) Token. Selanjutnya, pada sisi *backend* (*server-side*), sistem melakukan proses autentikasi dan penyimpanan *ciphertext* ke dalam basis data MySQL tanpa memiliki akses terhadap isi data asli maupun kunci enkripsi pengguna. Proses dekripsi dilakukan dengan alur yang berlawanan, di mana *ciphertext* yang diterima dari *backend* didekripsi secara lokal pada perangkat pengguna untuk merekonstruksi kembali teks asli (*plaintext*).

B. Pengujian Sistem

Untuk memvalidasi fungsionalitas dan keamanan sistem, evaluasi operasional dilakukan secara menyeluruh. Pengujian antarmuka dan fungsi sistem dievaluasi menggunakan *Black Box Testing* [10], [11]. Proses pengujian berfokus pada eksekusi program melalui pengisian data pada setiap formulir sistem untuk memverifikasi apakah fungsionalitas mampu menerima atau menolak masukan sesuai ekspektasi pengembang.

Pengujian keamanan dilakukan dengan memverifikasi implementasi mekanisme perlindungan terhadap serangan *Cross-Site Request Forgery* (CSRF) [12], deteksi peretasan *Cross-Site Scripting* (XSS) [5], [6], dan ketahanan terhadap *SQL Injection* [13], [14]. Verifikasi dilakukan melalui simulasi berbagai skenario serangan untuk memastikan bahwa sistem mampu menolak permintaan tidak sah, memfilter masukan berbahaya, serta mencegah manipulasi kueri basis data.

III. HASIL DAN PEMBAHASAN

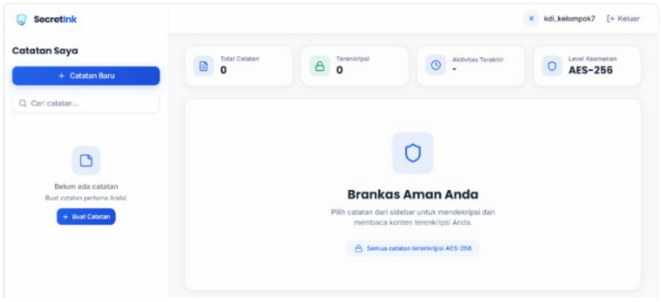
Bab ini menyajikan hasil implementasi dan evaluasi aplikasi web Secret Ink yang dikembangkan untuk mendukung penyimpanan catatan digital secara aman. Pembahasan diawali dengan implementasi fitur-fitur utama sistem yang berperan dalam proses pengelolaan dan perlindungan data pengguna. Selanjutnya, dilakukan pengujian fungsionalitas dan keamanan untuk mengevaluasi efektivitas mekanisme perlindungan yang diterapkan pada sistem.

A. Implementasi Antarmuka Sistem

Implementasi sistem difokuskan pada fitur-fitur utama yang mendukung proses penyimpanan dan perlindungan catatan digital. Antarmuka dirancang secara sederhana agar pengguna dapat mengelola catatan dengan mudah tanpa mengurangi aspek keamanan dan privasi. Implementasi fitur-fitur utama sistem ditunjukkan melalui beberapa tampilan berikut.

1. *Dashboard* Utama

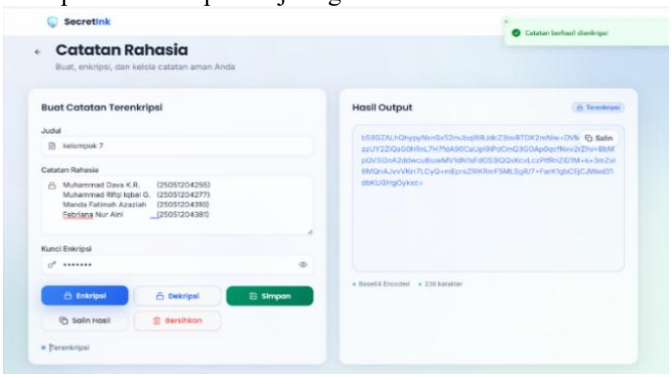
Pada Gambar 2 yang merupakan halaman berfungsi sebagai pusat navigasi dan kendali utama setelah pengguna berhasil melakukan autentikasi. Pada antarmuka ini, sistem menampilkan panel ringkasan informasi di bagian atas. Pada sisi kiri (sidebar), pengguna diberikan akses untuk mencari catatan spesifik atau membuat catatan baru. Sementara itu, area konten utama didesain untuk memilih catatan dari sidebar terlebih dahulu sebelum proses dekripsi dapat dilakukan.



Gambar 2. Antarmuka *Dashboard*

2. Pembuatan Catatan Terenkripsi

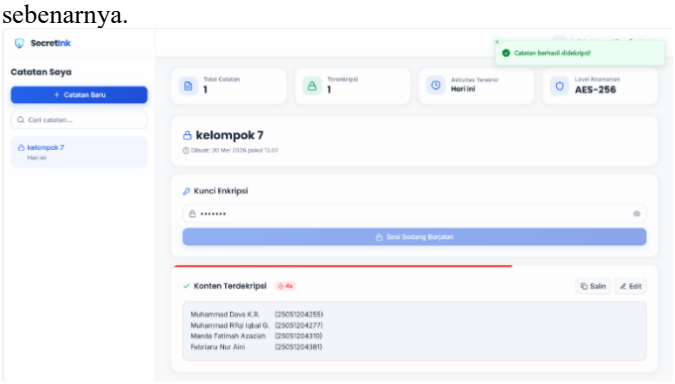
Antarmuka pada Gambar 3 membuktikan bagaimana proses enkripsi beroperasi secara *real-time* sebelum data ditransmisikan. Halaman ini dirancang menggunakan tata letak dua panel utama. Pada panel kiri ("Buat Catatan Terenkripsi"), pengguna memasukkan *metadata* (Judul), teks asli (*plaintext*) pada kolom catatan rahasia, serta menetapkan kunci enkripsi. Panel kanan ("Hasil Output") dalam wujud teks acak (*ciphertext*) berformat *Base64 Encoded*. Melalui mekanisme ini, ketika pengguna menekan tombol "Simpan", sistem secara eksklusif hanya akan mengirimkan *ciphertext* dari panel kanan menuju basis data, menjamin bahwa teks asli dan kunci rahasia tidak pernah terekspos ke jaringan lalu lintas data.



Gambar 3. Antarmuka Pembuatan Catatan

3. Detail dan Dekripsi Catatan

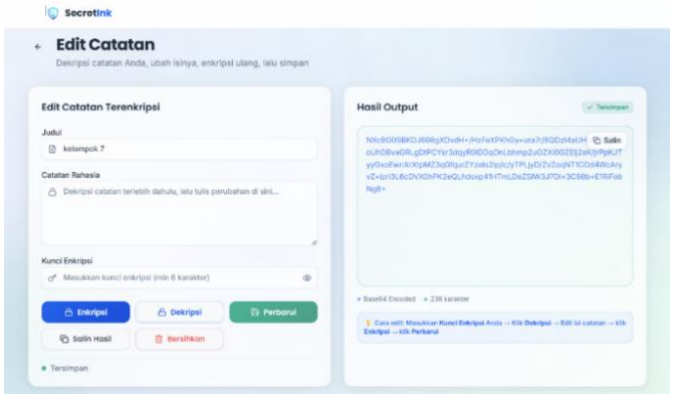
Antarmuka pada Gambar 4 menunjukkan proses dekripsi catatan yang dilakukan pada perangkat pengguna. Ketika catatan dibuka, *backend* hanya mengirimkan data dalam bentuk *ciphertext* tanpa menyertakan kunci enkripsi. Untuk memperoleh kembali isi catatan, pengguna harus memasukkan kunci yang sesuai pada kolom yang telah disediakan. Setelah kunci diverifikasi, sistem menjalankan proses dekripsi pada sisi *user* dan menampilkan hasilnya dalam bentuk *plaintext* secara langsung pada antarmuka. Hasil ini menunjukkan keberhasilan implementasi *Zero-Knowledge*, di mana *backend* terbukti tidak memiliki akses terhadap kunci dekripsi maupun isi pesan yang sebenarnya.



Gambar 4. Antarmuka Detail Catatan

4. Edit Catatan

Fitur edit catatan memungkinkan pengguna memperbarui informasi yang telah tersimpan pada sistem. Untuk melakukan perubahan, pengguna terlebih dahulu membuka catatan yang dipilih dan memasukkan kunci enkripsi yang sesuai agar isi catatan dapat ditampilkan. Setelah proses pengeditan selesai, sistem akan memproses kembali data yang diperbarui sebelum menyimpannya ke basis data. Mekanisme ini memastikan bahwa perubahan data dapat dilakukan tanpa mengurangi tingkat keamanan dan kerahasiaan informasi yang tersimpan pada aplikasi.



Gambar 5. Antarmuka Edit Catatan

B. Evaluasi dan Pengujian Keamanan Sistem

1) Proteksi *Cross-Site Request Forgery* (CSRF)

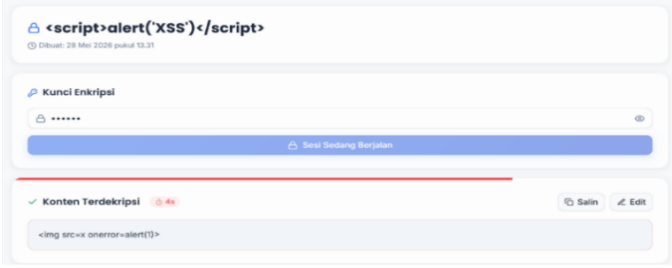
Pengujian dilakukan melalui *header request* HTTP pada metode POST yang mengarah ke *endpoint* API penyimpanan. Analisis jaringan menunjukkan bahwa setiap permintaan mutasi data secara presisi menyertakan token pencegahan CSRF secara dinamis, yang secara efektif memvalidasi keabsahan *request* dan memblokir upaya eksekusi pemalsuan pihak ketiga [12].

TABEL I
HASIL PENGUJIAN PROTEKTIF CSRF

CSRF protection aktif	Session terenkripsi	Same-origin policy
next-auth.csrf-token=c99764a52c	next-auth.session-token=eyJhbGciOi.	Sec-Fetch-Site same-origin

2) Mitigasi *Cross-Site Scripting* (XSS)

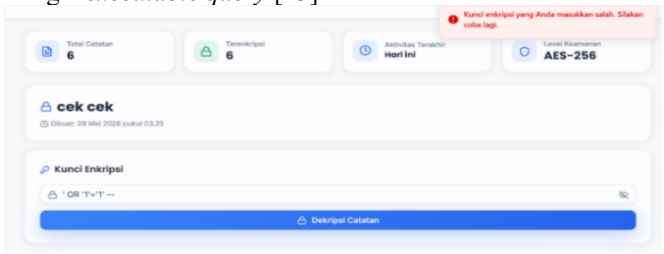
Uji terhadap celah XSS dilakukan dengan menginjeksikan *payload* skrip berbahaya, seperti <script>alert('XSS')</script> dan <img src=x onerror=alert(1)>, ke dalam kolom *input* judul dan isi catatan [5]. Hasil pengujian memvalidasi bahwa *payload* tersebut dirender secara pasif sebagai teks biasa (*plaintext*) oleh peramban, mengonfirmasi bahwa mekanisme *HTML escaping* beroperasi secara optimal dalam mencegah eksekusi manipulasi *Document Object Model* (DOM) [6].



Gambar 6. Hasil Pengujian Mitigasi XSS

3) Ketahanan terhadap *SQL Injection*

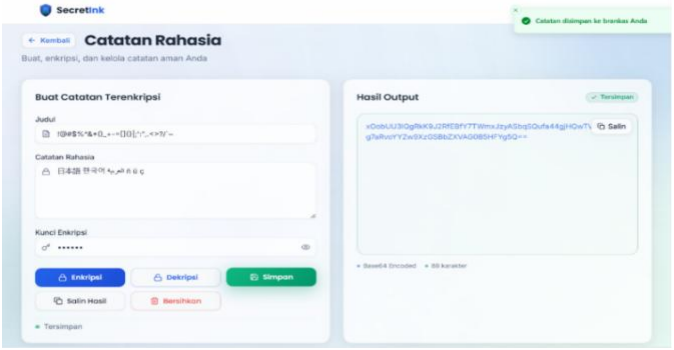
Evaluasi keamanan basis data dilakukan dengan injeksi perintah SQL pada kolom input kunci enkripsi saat proses dekripsi (contoh *payload*: ' OR '1'='1' --) [14]. Sistem merespons manipulasi ini dengan menampilkan pesan penolakan yang mengindikasikan kegagalan dekripsi. Hal ini membuktikan bahwa implementasi *Object-Relational Mapping* (ORM) Prisma pada *backend* secara otomatis menggunakan *parameterized query*, sehingga *input* pengguna tidak diproses sebagai *executable query* [13].



Gambar 7. Hasil Pengujian Ketahanan terhadap *SQL Injection*

4) Sanitasi *Input*

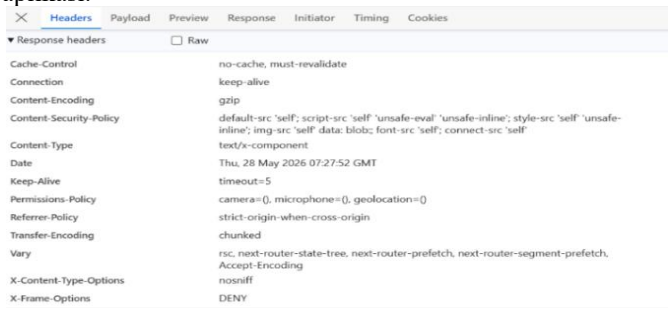
Pengujian batas toleransi karakter diverifikasi menggunakan kombinasi karakter spesial (!@#$%^&*()_+-=[]{};':",.<>?/~) serta karakter (*日本語 한국어 العربية ñ ü ç*). Sistem terbukti mampu memproses, mengenkripsi, dan menyimpan format data *non-standard* tersebut dengan presisi tanpa memicu anomali (*error*), sehingga menjaga integritas data tanpa mengorbankan fungsionalitas.



Gambar 8. Hasil Pengujian Sanitasi *Input*

5) Konfigurasi *Security Headers* HTTP

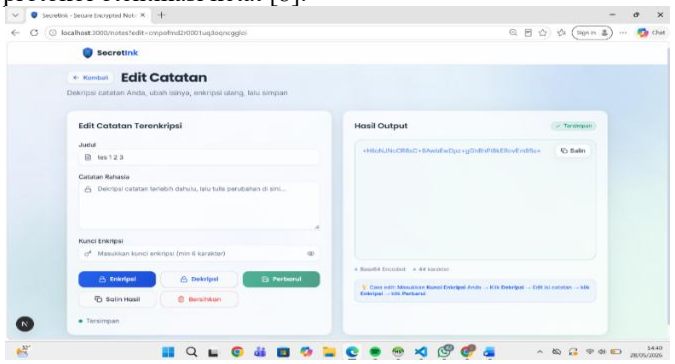
Pemeriksaan terhadap respons *backend* mengonfirmasi pengaktifan serangkaian *security headers* secara komprehensif, mencakup *Content-Security-Policy* (CSP), X-Frame-Options: DENY, X-Content-Type-Options: nosniff, dan Referrer-Policy: strict-origin-when-cross-origin. Konfigurasi ini secara signifikan mempersempit area permukaan serangan pada aplikasi.



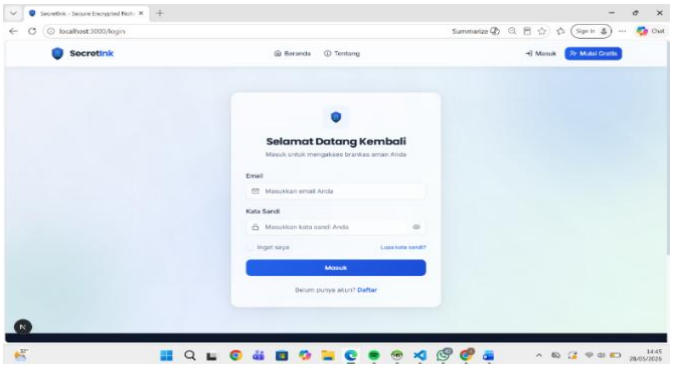
Gambar 9. Hasil Pengujian Konfigurasi Security Headers

6) Manajemen Sesi Waktu

Pengujian kontrol sesi dilakukan dengan membiarkan aplikasi dalam status tidak aktif (*idle*) selama lima menit. Aplikasi secara presisi mengeksekusi pemutusan sesi otomatis. Pemantauan membuktikan bahwa token format *JSON Web Token* (JWT) yang dienkripsi dengan standar AES-256-GCM menjaga transmisi data tetap stabil dan aman melalui jalur protokol otentikasi ketat [8].



Gambar 10. Kondisi Sesi Sebelum *Session Timeout*



Gambar 11. Hasil Pengujian Session Timeout

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan, aplikasi web Secret Ink mampu menjalankan seluruh fungsi utama sesuai dengan kebutuhan yang telah ditetapkan. Implementasi algoritma AES-256 berhasil melindungi data pengguna melalui mekanisme enkripsi dan dekripsi yang berjalan pada sisi *user*, sedangkan penerapan prinsip *Zero-Knowledge* memastikan bahwa backend hanya berperan sebagai media penyimpanan *ciphertext* tanpa memiliki akses terhadap *plaintext* maupun kunci enkripsi pengguna. Selain itu, hasil pengujian keamanan menunjukkan bahwa sistem mampu memberikan perlindungan terhadap berbagai ancaman umum pada aplikasi web, seperti CSRF, XSS, dan *SQL Injection*. Temuan ini menunjukkan bahwa mekanisme keamanan yang diterapkan telah berfungsi sesuai dengan rancangan dan mendukung terciptanya lingkungan penyimpanan catatan digital yang aman dan menjaga privasi pengguna.

IV. PENUTUP

Penelitian ini berhasil merancang dan mengimplementasikan aplikasi web Secret Ink sebagai media penyimpanan catatan digitaldan privasi data pengguna. Sistem dibangun dengan mengintegrasikan algoritma *Advanced Encryption Standard* (AES) 256-bit dan prinsip *Zero-Knowledge* sehingga proses enkripsi dan dekripsi dapat dilakukan pada sisi pengguna tanpa memberikan akses terhadap *plaintext* maupun kunci enkripsi kepada *backend*.

Hasil pengujian menunjukkan bahwa seluruh fungsi utama sistem dapat berjalan sesuai dengan kebutuhan yang telah ditetapkan. Selain itu, pengujian keamanan membuktikan bahwa mekanisme perlindungan yang diterapkan mampu memitigasi berbagai ancaman umum pada aplikasi web, termasuk *Cross-Site Request Forgery* (CSRF), *Cross-Site Scripting* (XSS), dan *SQL Injection* [15]. Dengan demikian, Secret Ink dapat menjadi alternatif aplikasi catatan digital yang mampu menjaga kerahasiaan data pengguna melalui kombinasi mekanisme kriptografi dan arsitektur keamanan yang diterapkan.

REFERENSI

[1] M. B. Yel and M. K. M. Nasution, "Keamanan Informasi Data Pribadi Pada Media Sosial," *Jurnal Informatika Kaputama (JIK)*, vol. 6, no. 1, pp. 92-101, 2022, doi: 10.59697/jik.v6i1.144.

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

[3] N. I. Khairunnisak, et al., "Implementasi Pengamanan Data Menggunakan Teknik Bcrypt Hashing Password dan Algoritma Advanced Encryption Standard (AES)," *JUSTIN (Jurnal Sistem dan Teknologi Informasi)*, vol. 13, pp. 1–10, 2025, doi: 10.26418/justin.v13i1.84997.

[4] N. Anugraha, M. Riswanto, L. Lindawati, and A. Asrul, "Sistem E-Voting berbasis Blockchain dengan Autentikasi Biometrik Sidik Jari dan Protokol Zero-Knowledge Proof Untuk Pengaman Privasi," *Jurnal Minfo Polgan*, vol. 14, no. 2, pp. 2756–2767, 2025, doi: 10.33395/jmp.v14i2.15536.

[5] H. Hartono, K. Khotimah, and A. Wibowo, "Deteksi Serangan Remote Code Execution Dan Cross Site Scripting Menggunakan Machine Learning," *Jurnal Informatika*, vol. 23, no. 2, pp. 229-242, 2023, doi: 10.30873/ji.v23i2.3931.

[6] N. A. Azizah, G. Arna, J. Saskara, and G. S. Santyadiputra, "Penetration Testing for Cross-Site Scripting (XSS) Detection Using Penetration Testing Execution Standard (PTES) Methodology: A Case Study of Information Systems in Institution X," *KARMAPATI (Kumpulan Artikel Mahasiswa Pendidikan Teknik Informatika)*, vol. 15, no. 2, 2026, doi: 10.23887/karmapati.v15i2.114689

[7] S. Supiyandi, M. Zen, C. Rizal, and M. Eka, "Perancangan Sistem Informasi Desa Tomuan Holbung Menggunakan Metode Waterfall," *JURIKOM (Jurnal Riset Komputer)*, vol. 9, no. 2, p. 274, Apr. 2022, doi: 10.30865/jurikom.v9i2.3986.

[8] F. C. Ramdani, A. Rahmatulloh, R. N. Shofa, "Implementasi JSON Web Token pada Authentication dengan Algoritma HMAC SHA-256" *SISTEMASI: Jurnal Sistem Informasi*, vol. 12, no. 1, pp 194-205, 2023, doi: 10.32520/stmsi.v12i1.2450.

[9] F. Shofyan and R. T. Shita, "Implementasi Web Service Restful API dengan Autentikasi Personal Access Tokens dan Algoritma AES 256," *Jurnal Ticom: Technology of Information and Communication*, vol. 12, no. 3, pp. 108-144, 2024, doi: /10.70309/ticom.v12i3.130.

[10] M. Putri, A. Ginting, and A. S. Lubis, "Pengujian Aplikasi Berbasis Web Data SKA Menggunakan Metode Black Box Testing," *FEBRUARI*, vol. 2, no. 1, pp. 41–48, 2024, doi: 10.55537/cosmic.

[11] M. Pangri, D. A. Faroek, and S. R. M Weyai, "Perancangan Sistem Informasi Akademik TK Berbasis Web: Studi pada TK Ekklesia," *Jurnal Media Informatika [JUMIN]*, vol. 6, no. 4, pp. 2025, doi: 10.55338/jumin.v6i4.6647.

[12] A. F. Akbar, M. Data, and M. A. Fauzi, "Deteksi Kerentanan Cross-Site Request Forgery (CSRF) pada Plugin WordPress dengan Menggunakan Analisis Nonce," *Sarjana thesis, Universitas Brawijaya*, 2025. [Online]. Available: https://repository.ub.ac.id/id/eprint/249361/

[13] N. Christina Sari *et al.*, "Deteksi Kerentanan SQL Injection pada Website Menggunakan Vulnerability Assessment Info Artikel," vol. 2, no. 1, pp. 9–17, 2024, doi: 10.26714/jodi.

[14] R. D. Irawan, H. D. Apriyanto, I. R. Hendrawan, and H. A. Id, "Analisis Vulnerability Scanning pada Alat Pemindaian Kerentanan Web: Studi Kasus Pemanfaatan Probely dalam Aplikasi Web," Prosiding Seminar Nasional Amikom Surakarta (SEMNASA) 2025, vol. 3, pp. 902-913, [Online] Available At: https://ojs.amikomsolo.ac.id/index.php/semnasa/article/view/1008/316

[15] D. Rohmaniah, W. M. Ashari, and A. D. Putra, "Enhancing Website Security Using Vulnerability Assessment and Penetration Testing (VAPT) Based on OWASP Top Ten," *Journal of Applied Informatics and Computing*, vol 9, no. 2, 2025, doi: 10.30871/jaic.v9i2.9069