Smart Comp: Jurnalnya Orang Pintar Komputer
Volume 15, Nomor 3, Juli 2026, hlm. 778-786
Terakreditasi Sinta, Peringkat 4, SK No. 295/C/C3/KPT/2026
DOI : 10.30591/smartcomp.v15i1.4272

P-ISSN : 2089-676X
E-ISSN : 2549-0796

Analisis Keamanan Pertukaran Data Arsitektur Client-Server Menggunakan Protokol JSON Web Token dan Enkripsi AES

Mohammad Romdlon*[1], Bambang Irawan[2], Nur Ariesanto Ramdhan [3]
[1,2,3)]Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhadi Setiabudi
Email: *[1]romsesawwdoni@gmail.com, [2]bambangumus@gmail.com, [3]ariesantoramdhan@gmail.com

(Naskah masuk: *21 Februari 2026*, diterima untuk diterbitkan: *7 Juli 2026*)

Abstrak: *Pertukaran data pada arsitektur client-server menggunakan protokol HTTP standar sangat rentan terhadap serangan packet sniffing, di mana pihak tidak bertanggung jawab dapat menyadap dan membaca data sensitif secara langsung. Penelitian ini bertujuan untuk meningkatkan keamanan komunikasi data dengan mengimplementasikan metode autentikasi JSON Web Token (JWT) yang dikombinasikan dengan algoritma kriptografi Advanced Encryption Standard (AES). Metode penelitian dilakukan secara eksperimental di laboratorium dengan menguji fungsionalitas aplikasi, menguji ketahanan jaringan dari serangan sniffing menggunakan Wireshark, serta mengukur lonjakan waktu komputasi (overhead response time) menggunakan Postman. Hasil penelitian menunjukkan bahwa sistem berhasil mengamankan jalur komunikasi; data yang disadap melalui jaringan terbukti telah berubah menjadi ciphertext yang tidak dapat dibaca. Selain itu, penambahan lapisan keamanan kriptografi ini terbukti sangat efisien, di mana rata-rata lonjakan waktu respons yang dihasilkan hanya sebesar 4,87 milidetik (dari 9,97 ms menjadi 14,83 ms). Kesimpulannya, kombinasi metode JWT dan enkripsi AES sangat efektif dan direkomendasikan untuk mengamankan pertukaran data pada sistem client-server tanpa mengorbankan performa jaringan secara signifikan.*

Kata Kunci – *Client-Server; Enkripsi AES; JSON Web Token; Keamanan Jaringan; Sniffing*

Security Analysis of Data Exchange in Client-Server Architecture Using JSON Web Token and AES Encryption

Abstract: *Data exchange in a client-server architecture using standard HTTP protocols is highly vulnerable to packet sniffing attacks, where unauthorized parties can intercept and read sensitive data directly. This study aims to improve data communication security by implementing the JSON Web Token (JWT) authentication method combined with the Advanced Encryption Standard (AES) cryptographic algorithm. The research method was carried out experimentally in the laboratory by testing application functionality, testing network resilience from sniffing attacks using Wireshark, and measuring computational time spikes (overhead response time) using Postman. The results showed that the system successfully secured the communication line; the data intercepted over the network was proven to have been transformed into unreadable ciphertext. In addition, the addition of this cryptographic security layer proved to be very efficient, where the average response time spike generated was only 4.87 milliseconds (from 9.97 ms to 14.83 ms). In conclusion, the combination of JWT and AES encryption is highly effective and recommended for securing data exchange in client-server systems without significantly sacrificing network performance.*

Keywords –*Advanced Encryption Standard; Client-Server; JSON Web Token; Network Security; Sniffing*

1. PENDAHULUAN

Perkembangan teknologi informasi saat ini mendorong peralihan perangkat lunak dari aplikasi monolitik menuju arsitektur *client-server* yang berinteraksi melalui *Application Programming Interface* (API). Pertukaran data pada arsitektur ini umumnya memanfaatkan protokol *Hypertext Transfer Protocol* (HTTP) yang mentransmisikan data dalam bentuk *plaintext* atau teks terang [1]. Kelemahan utama dari transmisi *plaintext* di dalam jaringan komputer adalah rentannya lalu lintas data terhadap serangan *packet sniffing* atau penyadapan, serta ancaman *Man-in-the-Middle* (MitM)

[2], [3]. Melalui teknik ini, pihak ketiga yang berada dalam satu jaringan lokal (LAN) maupun publik dapat membaca, merekam, bahkan memanipulasi informasi sensitif yang sedang dikirimkan antara *client* dan *server* [2]. Oleh karena itu, diperlukan sebuah mekanisme perlindungan data berlapis yang mencakup autentikasi entitas pemohon data dan kerahasiaan isi muatan data (*payload*).

Penelitian terkait keamanan sistem *client-server* telah banyak dilakukan sebelumnya dan terus berkembang. Purwono dkk. (2023) dalam jurnal *Smart Comp* melakukan pengembangan keamanan sistem rekam medis menggunakan teknologi *blockchain* dan *smart contract* untuk memvalidasi hak akses data. Meskipun tingkat keamanannya sangat tinggi, metode desentralisasi ini membutuhkan beban komputasi yang cukup berat dan infrastruktur yang kompleks untuk aplikasi berskala menengah [4]. Pada pendekatan kriptografi konvensional, penelitian oleh Ramadhan dkk. (2022) membuktikan bahwa algoritma *Advanced Encryption Standard* (AES) 256-bit sangat tangguh dalam mengamankan teks rekam medis berbasis web dari serangan dekripsi paksa, namun penelitian tersebut tidak menguji dampaknya terhadap metrik kecepatan jaringan [5]. Sementara itu, dalam hal autentikasi jalur (*stateless*), *JSON Web Token* (JWT) kini telah menjadi standar industri yang terbukti lebih ringan, aman, dan efisien dibandingkan metode sesi (*session*) tradisional untuk mengamankan *endpoint* API [6].

Meskipun penggunaan token JWT dan algoritma enkripsi AES telah diimplementasikan secara luas pada berbagai penelitian secara terpisah, kajian yang secara khusus mengevaluasi performa jaringan dari kombinasi kedua metode ini secara bersamaan masih cukup terbatas. Menerapkan JWT pada HTTP *header* sebagai lapisan autentikasi, sekaligus melakukan enkripsi *End-to-End* pada seluruh isi *payload* menggunakan AES, tentu akan meningkatkan standar keamanan jaringan ke level yang lebih tinggi [7]. Namun, penggabungan lapisan keamanan ini berisiko menimbulkan *overhead* komputasi (beban pemrosesan tambahan) yang dapat memperpanjang *response time* (waktu respons API), sehingga berpotensi menurunkan kenyamanan pengguna.

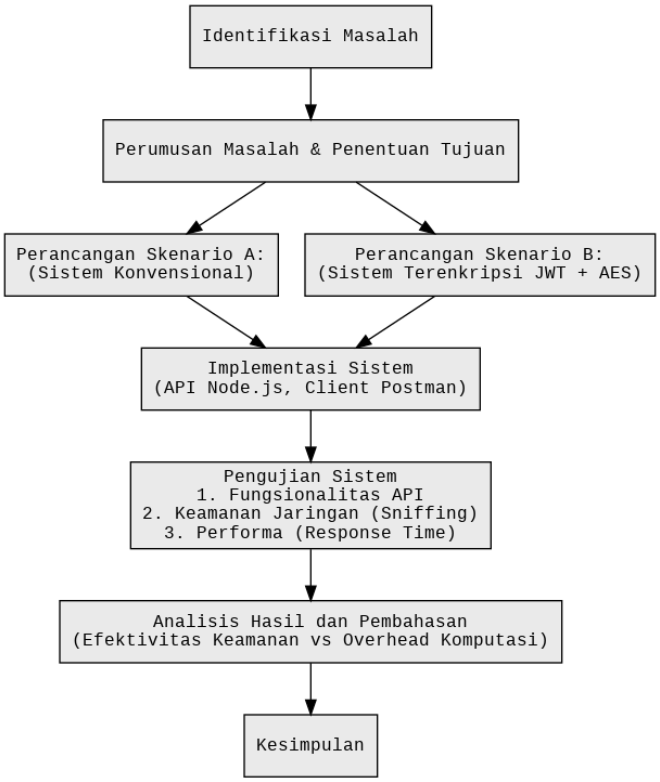
Berdasarkan *state of the art* dan celah penelitian tersebut, maka penelitian ini bertujuan untuk mengimplementasikan dan menganalisis protokol JWT yang dikombinasikan dengan enkripsi AES pada komunikasi *client-server*. Kebaruan (kontribusi) utama dari penelitian ini terletak pada pengujian empiris yang membandingkan efektivitas keamanan terhadap serangan *sniffing* secara langsung berhadapan dengan metrik penurunan performa (peningkatan *response time*). Hasil penelitian ini diharapkan dapat memberikan panduan kuantitatif bagi pengembang sistem komputer dalam menyeimbangkan antara tingkat keamanan maksimal dan beban kinerja optimal pada jaringan lokal.

2. METODE PENELITIAN

Penelitian ini menggunakan metode eksperimental terapan di laboratorium komputer dengan mengintegrasikan konsep dari tiga bidang keilmuan, yaitu Jaringan Komputer, Pemrograman *Client-Server*, dan Keamanan Komputer. Pendekatan eksperimental dipilih karena metode ini memungkinkan pengamatan kausalitas secara langsung terhadap manipulasi variabel keamanan di dalam sebuah ekosistem jaringan [8]. Pembahasan pada bagian ini menguraikan tahapan penelitian secara terstruktur mulai dari identifikasi masalah hingga tahapan pengujian sistem.

2.1. Tahapan Penelitian

Tahapan penelitian yang dilakukan dalam studi ini digambarkan dalam bentuk alur penelitian. Secara umum, penelitian ini terdiri dari beberapa fase yang berurutan untuk memastikan penyelesaian masalah dilakukan secara sistematis.



Gambar 1. Alur Penelitian

Berdasarkan alur penelitian pada Gambar 1, berikut adalah rincian dari masing-masing tahapan di fase awal :

1.1.1. Identifikasi Masalah

Langkah awal dilakukan dengan mengidentifikasi celah keamanan pada arsitektur *client-server* yang berjalan pada jaringan lokal. Proses identifikasi ini mencakup:

1. Menganalisis kerentanan transmisi *plaintext* pada protokol HTTP standar yang rentan terhadap penyadapan (*sniffing*).
2. Mengidentifikasi kebutuhan akan metode pengamanan data yang tangguh (autentikasi dan enkripsi) namun tidak membebani performa jaringan (*response time*) secara berlebihan.

1.1.2. Perumusan Masalah

Berdasarkan hasil identifikasi masalah yang telah diuraikan, maka perumusan masalah dalam penelitian ini ditetapkan sebagai berikut:

1. Bagaimana mengimplementasikan protokol JSON Web Token (JWT) dan algoritma enkripsi *Advanced Encryption Standard* (AES) secara End-to-End pada arsitektur komunikasi client-server?
2. Bagaimana tingkat keamanan data dari serangan *packet sniffing* serta perbandingan performa transmisi (*response time*) antara sistem standar dan sistem yang telah diamankan?

1.1.3. Penentuan Tujuan

Penentuan tujuan penelitian ini diarahkan untuk menjawab rumusan masalah yang telah ditetapkan, yaitu:

1. Mengimplementasikan lapisan keamanan ganda menggunakan JWT untuk autentikasi dan AES untuk enkripsi payload pada jaringan client-server.

2. Menguji dan menganalisis secara empiris efektivitas keamanan sistem terhadap serangan *sniffing*, sekaligus mengukur lonjakan waktu komputasi (*overhead response time*) yang dihasilkan.

1.2. Tahapan Pengujian

Fase pengujian dan pengumpulan data dilakukan di laboratorium secara eksperimental. Pengujian ini melibatkan mahasiswa secara langsung sebagai agen penguji untuk melakukan simulasi *request* data dan penyadapan lalu lintas jaringan. Skenario pengujian dibagi ke dalam dua langkah utama:

1.2.1. Skenario A (Sistem Konvensional)

1. Mahasiswa melakukan request ke *Application Programming Interface* (API) server menggunakan metode HTTP biasa (tanpa enkripsi).
2. Komputer client lain di jaringan yang sama menjalankan perangkat lunak Wireshark untuk melakukan *sniffing* dan menguji apakah paket data (*plaintext*) dapat dicegat dan dibaca
3. Mencatat rata-rata *response time* (dalam milidetik) menggunakan perangkat lunak Postman. Penggunaan alat pengujian API otomatis seperti Postman terbukti valid dalam mengukur metrik performa transmisi secara presisi [9].

1.2.2. Skenario B (Sistem Terenkripsi JWT dan AES)

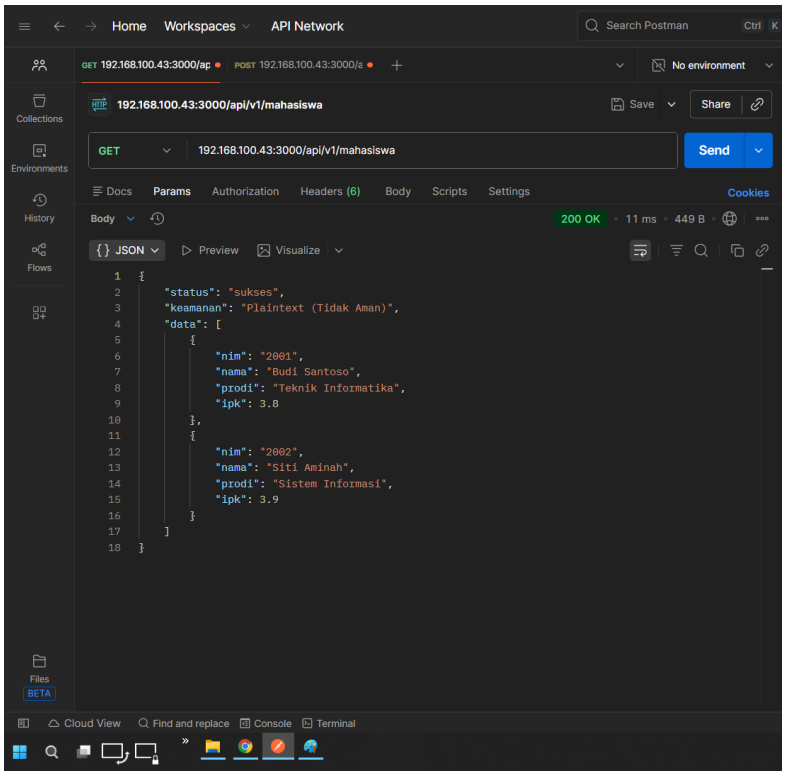
1. Mahasiswa melakukan request ke API dengan menyisipkan token JWT yang valid pada HTTP Header, di mana data balasan dari server dikirim dalam bentuk ciphertext (terenkripsi AES).
2. Simulasi *sniffing* kembali dilakukan menggunakan Wireshark, yang merupakan network protocol analyzer standar industri untuk mengaudit kerentanan paket data [10], guna membuktikan bahwa data yang dicegat tidak dapat dibaca atau dimanipulasi (*data integrity* terjaga).
3. Mencatat kembali rata-rata response time untuk dibandingkan dengan hasil pada Skenario A.

2. HASIL DAN PEMBAHASAN

Pembahasan terhadap hasil penelitian dan pengujian yang diperoleh disajikan dalam bentuk uraian teoritik, baik secara kualitatif maupun kuantitatif. Pengujian sistem terbagi menjadi tiga tahapan utama, yaitu pengujian fungsionalitas API, pengujian keamanan jaringan menggunakan teknik *sniffing*, dan pengujian performa waktu respons (*response time*).

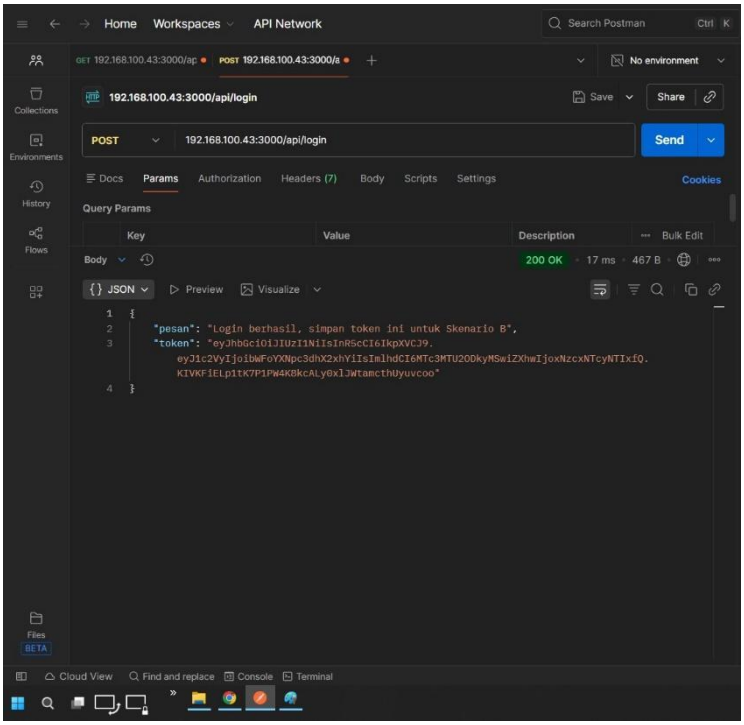
3.1 Pengujian Fungsionalitas API

Pengujian fungsionalitas dilakukan menggunakan aplikasi penguji *Application Programming Interface* (API) yaitu Postman. Pengujian ini bertujuan untuk memastikan bahwa server Node.js dapat merespons permintaan dari client dengan baik. Pada Skenario A (tanpa keamanan), client melakukan request dengan metode GET dan server langsung mengembalikan balasan dengan status 200 OK beserta payload berupa data JSON mahasiswa yang dapat terbaca secara jelas (*plaintext*). Hasil pengujian Skenario A dapat dilihat pada Gambar 2.



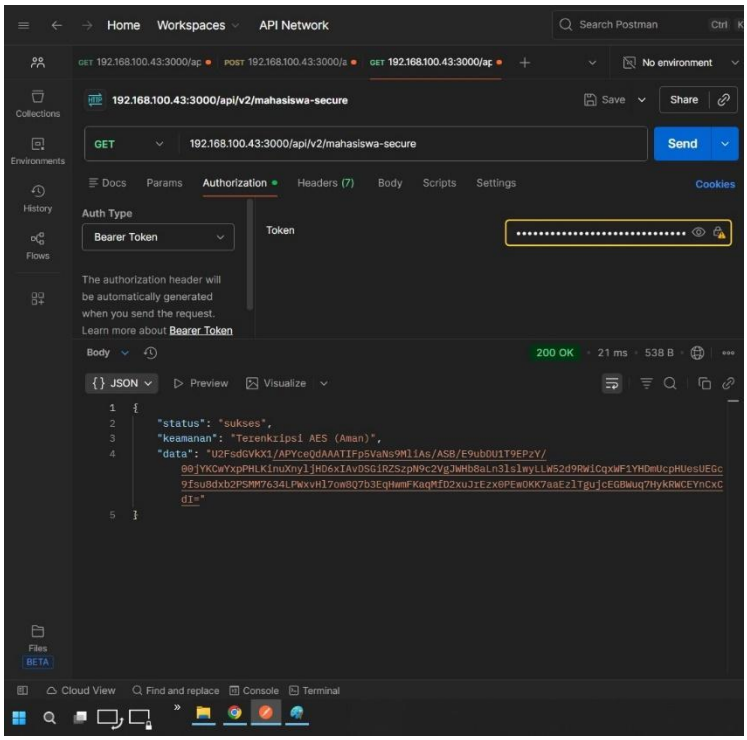
Gambar 2. Hasil Pengujian Fungsionalitas API Skenario A (Menampilkan *Plaintext*)

Berbeda dengan Skenario A, Skenario B menerapkan algoritma keamanan JSON *Web Token* (JWT) dan *Advanced Encryption Standard* (AES). *Client* diwajibkan melakukan *login* terlebih dahulu melalui *endpoint* autentikasi untuk mendapatkan token JWT. Hasil *generate* token JWT ditunjukkan pada Gambar 3.



Gambar 3. Proses *Request* Autentikasi untuk Mendapatkan Token JWT

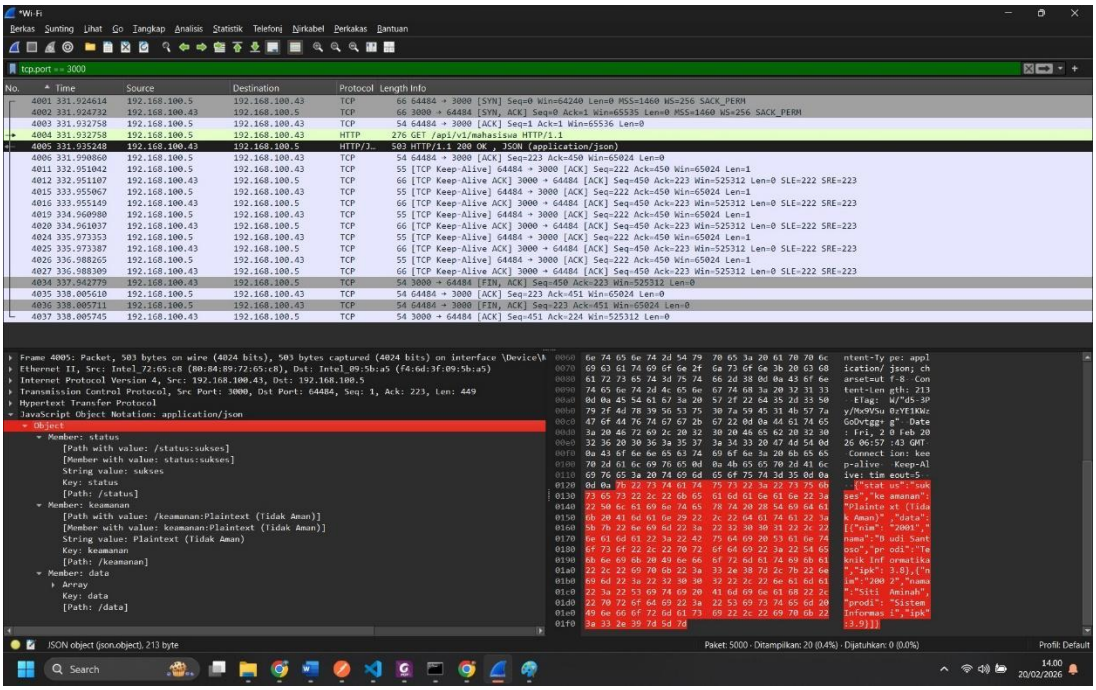
Setelah mendapatkan token, *client* menyematkan token tersebut pada *header Authorization* (tipe *Bearer*) untuk mengakses data. *Server* memvalidasi token tersebut, lalu melakukan enkripsi AES pada *payload* data mahasiswa sebelum dikirimkan. Gambar 4 menunjukkan bahwa fungsionalitas berjalan lancar dengan balasan status 200 OK, namun data *payload* yang diterima *client* sudah berbentuk karakter acak (*ciphertext*).



Gambar 4. Hasil Pengujian Fungsionalitas API Skenario B (Menampilkan *Ciphertext*)

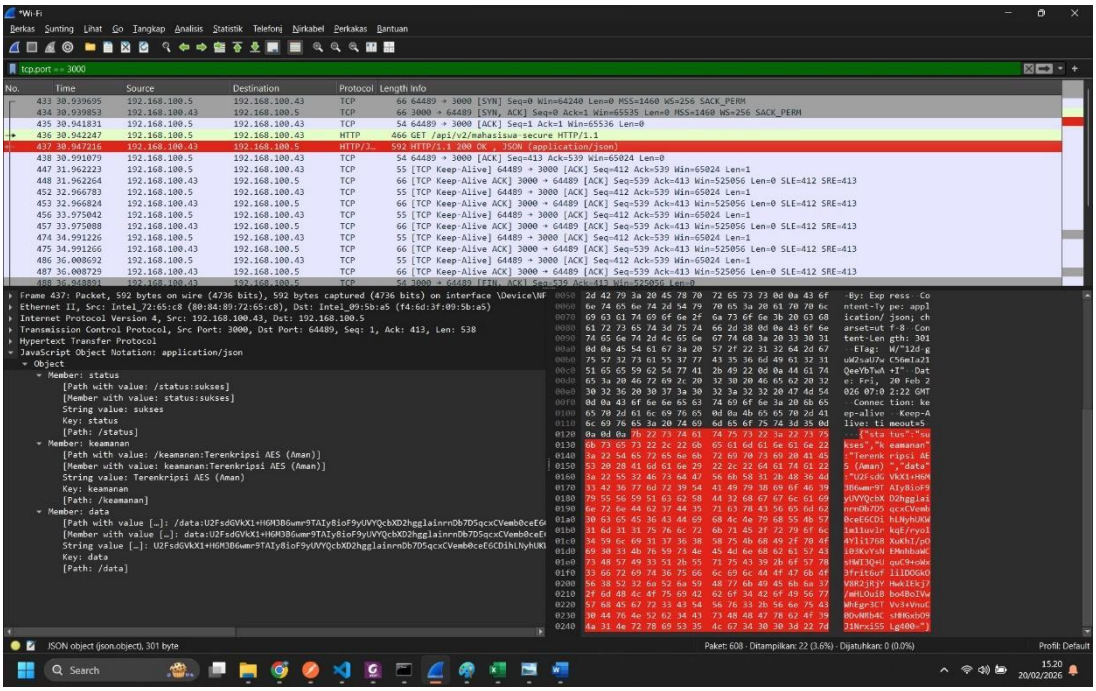
3.2 Pengujian Keamanan Jaringan (Sniffing)

Pengujian ini mensimulasikan serangan *Man-in-the-Middle* (MitM) di mana penyerang merekam lalu lintas jaringan (*sniffing*) menggunakan perangkat lunak Wireshark. Pada pengujian Skenario A (Gambar 5), penyerang menangkap paket HTTP/TCP pada *port* 3000. Saat dilakukan inspeksi pada bagian *JavaScript Object Notation* (JSON), data sensitif seperti Nomor Induk Mahasiswa (NIM), Nama, dan Program Studi terlihat sangat jelas karena tidak dilindungi enkripsi.



Gambar 5. Hasil Penyadapan (*Sniffing*) Skenario A Menampilkan Data *Plaintext*

Sebaliknya, pada pengujian Skenario B (Gambar 6), meskipun penyerang berhasil menangkap lalu lintas jaringan yang sama di *port* 3000, data asli tidak dapat diekstraksi. Inspeksi pada *payload* JSON hanya menampilkan susunan karakter acak (U2FsdGVk...). Hal ini membuktikan bahwa kombinasi token JWT untuk pembatasan akses dan enkripsi AES untuk perlindungan *payload* terbukti berhasil mengamankan pertukaran data dari ancaman pencurian informasi.



Gambar 6. Hasil Penyadapan (*Sniffing*) Skenario B Menampilkan Data Terenkripsi

3.3 Pengujian Performa Waktu Respons

Penerapan algoritma kriptografi tentunya membutuhkan proses komputasi yang berdampak pada waktu respons (*response time*). Untuk mengukur beban (*overhead*) tersebut, dilakukan 30 kali percobaan pengiriman *request* secara berulang untuk membandingkan waktu respons Skenario A dan Skenario B. Ringkasan hasil rata-rata pengujian disajikan pada Tabel 1.

Parameter Pengujian	**Rata-Rata Waktu (ms)**
Waktu Respons Skenario A (*Plaintext*)	9,97 ms
Waktu Respons Skenario B (JWT + AES)	14,83 ms
Selisih Waktu (*Overhead*)	4,87 ms

Tabel 1. Rata-rata Waktu Respons Skenario A dan B

Berdasarkan Tabel 1, dapat dilihat bahwa rata-rata waktu yang dibutuhkan untuk mengeksekusi Skenario A adalah 9,97 ms, sedangkan Skenario B membutuhkan waktu 14,83 ms. Proses validasi JWT dan enkripsi dekripsi AES menambahkan *overhead* waktu respons rata-rata sebesar 4,87 ms. Mengingat selisih waktu tersebut sangat kecil (di bawah 5 milidetik), penambahan beban komputasi ini dinilai sangat efisien. Keterlambatan tersebut tidak dapat dirasakan oleh pengguna manusia, namun memberikan tingkat keamanan yang sangat tinggi pada lapisan transmisi data.

Secara keseluruhan, hasil pengujian sistem ini sejalan dengan berbagai praktik terbaik, di mana pencegahan manipulasi informasi sensitif merupakan prioritas utama [11]. Penggunaan standar JSON Web Token (JWT) [12] terbukti andal bila dikombinasikan dengan prinsip keamanan siber yang komprehensif [13]. Selain itu, algoritma Advanced Encryption Standard (AES) [14] secara konsisten tetap menjadi standar enkripsi yang paling direkomendasikan untuk diterapkan pada arsitektur komunikasi *client-server* saat ini [15]

3. KESIMPULAN

Berdasarkan hasil pengujian dan pembahasan yang telah dilakukan terhadap penerapan JSON Web Token (JWT) dan algoritma Advanced Encryption Standard (AES) pada arsitektur komunikasi client-server, implementasi lapisan keamanan ganda ini berhasil dilakukan dengan baik. Sistem terbukti hanya mengizinkan client yang memiliki token valid untuk melakukan request data. Selain itu, pengujian keamanan jaringan menggunakan teknik sniffing membuktikan bahwa enkripsi AES secara End-to-End terbukti tangguh dalam melindungi kerahasiaan data. Paket data yang berhasil disadap oleh pihak ketiga hanya berupa ciphertext (karakter acak) yang tidak memiliki makna, sehingga ancaman manipulasi dan pencurian informasi sensitif dapat dicegah secara optimal. Penambahan algoritma kriptografi pada komunikasi data ini juga terbukti memberikan beban komputasi (overhead) yang sangat minimal. Perbandingan waktu respons (response time) menunjukkan selisih rata-rata hanya sebesar 4,87 ms (dari 9,97 ms menjadi 14,83 ms). Keterlambatan ini sangat efisien dan tidak mengganggu kenyamanan pengguna dalam menggunakan aplikasi secara real-time.

Sebagai saran untuk penelitian selanjutnya, metode pengamanan ini dapat dikembangkan lebih lanjut dengan memadukan algoritma enkripsi asimetris (seperti RSA) pada proses distribusi kunci (key exchange) agar tingkat keamanannya menjadi lebih kompleks terhadap serangan yang lebih masif.

UCAPAN TERIMA KASIH

Terima kasih kepada pihak-pihak yang telah berkontribusi dalam penelitian, Civitas Akademika Universitas Muhadi SetiaBudi, Partisipasi Mahasiswa Semester 4 TA 2025/2026 Program Studi Teknik Informatika.

DAFTAR PUSTAKA

[1] J. Kurose and K. Ross, Computer Networking: A Top-Down Approach, 8th, Ed., Pearson, 2021.
[2] A. P. P. R. Syahputra, "Analisis Keamanan Jaringan Komputer Terhadap Serangan Sniffing Menggunakan Wireshark," *Jurnal Teknik Informatika,* vol. 6, no. 1, pp. 45-52, 2021.
[3] M. A. Faisal, "Deteksi dan Mitigasi Serangan Man-in-the-Middle (MitM) pada Arsitektur Jaringan Lokal," *Jurnal Keamanan Siber dan Sandi,* vol. 4, no. 2, pp. 88-95, 2022.
[4] P. Purwono, P. Dewi and S. D. Kurniawan, "Pengembangan Keamanan Sistem Rekam Medis Berbasis Blockchain dengan Smart Contract," *Smart Comp: Jurnalnya Orang Pintar Komputer,* vol. 12, no. 2, pp. 410-420, 2023.
[5] R. Ramadhan and S. W. Irianandа, "Implementasi Algoritma Kriptografi AES 256 pada," *Jurnal Komputer dan Keamanan,* vol. 8, no. 2, pp. 112-118, 2022.
[6] R. S. Pressman and B. R. Maxim, Software Engineering: A Practitioner's Approach, 9 ed., McGrawHill, 2020.
[7] F. A. Pratama, "Evaluasi Kinerja RESTful API dengan Autentikasi JSON Web Token (JWT)," *Jurnal Rekayasa Sistem,* vol. 5, no. 3, pp. 201-209, 2021.
[8] A. Setiawan and Y. Yuliana, "Pendekatan Metode Eksperimental dalam Evaluasi Performa Infrastruktur Jaringan," *Prosiding Seminar Nasional Ilmu Komputer (SNIK),* pp. 120-125, 2020.
[9] D. T. Santoso, "Analisis Response Time dan Throughput pada REST API Menggunakan Postman," *Jurnal Informatika dan Sistem Informasi,* vol. 9, no. 1, pp. 33-40, 2023.
[10] B. Hariyanto, "Audit Keamanan Jaringan Komputer Menggunakan Network Protocol Analyzer Wireshark," *Jurnal Teknologi Informasi dan Komunikasi,* vol. 11, no. 2, pp. 77-84, 2019.
[11] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 8th ed., Pearson, 2020.
[12] M. Jones, J. Bradley, and N. Sakimura, "JSON Web Token (JWT)," *Internet Engineering Task Force (IETF) RFC 7519*, 2015.
[13] O. W. Purbo, *Keamanan Jaringan Internet*, Yogyakarta: Andi Publisher, 2021.
[14] National Institute of Standards and Technology (NIST), "Advanced Encryption Standard (AES)," *FIPS PUB 197*, 2001.
[15] I. Sommerville, *Software Engineering*, 11th ed., Pearson, 2020.