Evaluasi Eksperimental dan Teoritis Macaroons versus JSON Web Tokens (JWT)

Tazkia Nizami
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
Bandung, Indonesia
tazkia.nizami@gmail.com

***Abstract*—Sistem terdistribusi dan arsitektur *microservices* memerlukan mekanisme autentikasi dan otorisasi yang efisien dan fleksibel. JSON Web Tokens (JWT) telah muncul sebagai standar industri untuk autentikasi *stateless*. Namun, JWT menghadapi keterbatasan signifikan terkait atenuasi *privilege* dan delegasi, yang sering memerlukan penerbitan ulang JWT. Macaroons, sebuah format *token bearer* berbasis HMAC berantai, menyajikan alternatif yang menarik dengan memungkinkan delegasi *offline* dan *caveat* yang *fine-grained* tanpa intervensi server. Makalah ini menyajikan evaluasi eksperimental dan teoritis yang komprehensif mengenai Macaroons dan JWT. Properti keamanan teoritis dari kedua format token dianalisis, khususnya berfokus pada kemampuan delegasi. Penelitian meliputi eksperimen *benchmark* kinerja empiris untuk membandingkan waktu generasi token, waktu validasi, dan *overhead payload*. Hasil eksperimen menunjukkan bahwa meskipun JWT mendapat manfaat dari dukungan komunitas yang besar, Macaroons menunjukkan fleksibilitas yang lebih tinggi untuk skenario otorisasi kompleks dengan *overhead* komputasi yang sebanding. Studi ini memberikan wawasan yang dapat ditindaklanjuti bagi arsitek sistem dalam memilih antara teknologi-teknologi ini untuk aplikasi terdistribusi yang aman.**

***Keywords*—autentikasi, otorisasi, JSON Web Tokens, JWT, Macaroons, *microservices*, sistem terdistribusi, keamanan, delegasi**

I. Pendahuluan

A. Latar Belakang

Dalam pengembangan web modern dan sistem terdistribusi, pergeseran dari arsitektur monolitik ke *microservices* memerlukan pemikiran ulang fundamental terhadap strategi autentikasi dan otorisasi. Autentikasi berbasis sesi tradisional, yang mengandalkan penyimpanan *state* di sisi server, sering menjadi *bottleneck* dalam lingkungan terdistribusi yang *scalable*. Akibatnya, mekanisme autentikasi *stateless* telah mendapat perhatian yang menonjol, memungkinkan layanan untuk memverifikasi identitas dan hak akses tanpa melakukan *query* ke penyimpanan sesi terpusat untuk setiap permintaan.

JSON Web Tokens (JWT) [1] telah menetapkan diri sebagai standar de facto untuk tujuan ini. Didefinisikan oleh IETF RFC 7519, JWT menyediakan metode yang ringkas dan *URL-safe* untuk merepresentasikan klaim yang akan ditransfer antara dua pihak. Adopsi luasnya didorong oleh kesederhanaannya, dukungan *library* yang ekstensif di berbagai bahasa pemrograman, dan kemudahan integrasi dengan protokol seperti OAuth 2.0 dan OpenID Connect.

Namun, seiring sistem terdistribusi tumbuh dalam kompleksitas, kebutuhan baru untuk otorisasi yang fleksibel muncul. Skenario yang melibatkan delegasi pihak ketiga, yaitu ketika pengguna mendelegasikan subset dari otoritasnya ke layanan lain, atau atenuasi *privilege*, ketika pengguna ingin mempersempit cakupan token untuk konteks tertentu, sulit dan menantang untuk diimplementasikan dengan JWT standar. Untuk mengatasi keterbatasan ini, Macaroons [2] diperkenalkan sebagai kredensial otorisasi yang baru. Macaroons menggunakan konstruksi *Hashed Message Authentication Codes (HMAC)* berantai untuk mendukung delegasi terdesentralisasi dan *caveat*, memungkinkan pemegangnya untuk melemahkan otoritas token tanpa berinteraksi dengan server penerbit.

B. Rumusan Masalah

Meskipun JWT ada di mana-mana, JWT memiliki kekakuan struktural yang inheren terkait delegasi. Setelah JWT ditandatangani, klaimnya tidak dapat diubah. Jika suatu layanan perlu mendelegasikan versi terbatas dari token ke layanan downstream (misalnya, membatasi akses ke file tertentu atau memperpendek waktu kedaluwarsa), ia harus meminta token baru dari penerbit terpusat. Hal ini meningkatkan latensi dan beban pada layanan autentikasi.

Macaroons secara teoritis memecahkan masalah ini dengan memungkinkan atenuasi *"offline"*. Namun, mereka kurang diadopsi secara luas. Ada kekurangan studi komparatif yang komprehensif yang mengevaluasi Macaroons terhadap JWT tidak hanya dalam hal fitur, tetapi juga mengenai metrik kinerja praktis dan integrasi. Arsitek sistem sering *default* ke JWT karena familiaritas, berpotensi mengabaikan manfaat keamanan dan arsitektur yang dapat ditawarkan Macaroons untuk alur delegasi yang kompleks.

Berdasarkan permasalahan tersebut, makalah ini berusaha menjawab pertanyaan riset berikut:

1) Apakah keunggulan teoritis Macaroons dalam delegasi dan atenuasi benar-benar termanifestasi secara signifikan dalam sistem *microservices*?
2) Bagaimana perbandingan kinerja antara Macaroons dan JWT pada skenario *deployment* yang realistis?

C. Tujuan dan Kontribusi

Tujuan utama dari penelitian ini adalah memberikan analisis komparatif yang mendetail antara Macaroons dan JSON Web

Tokens untuk menentukan kesesuaian mereka untuk berbagai kebutuhan arsitektur. Penelitian ini bertujuan untuk:

- Melakukan benchmark kinerja eksperimental dari kedua format token, mengukur waktu generasi, latensi validasi, dan overhead ukuran token.
- Menganalisis implikasi praktis penggunaan Macaroons dalam lingkungan *microservices* dibandingkan dengan pendekatan JWT standar.

Kontribusi utama dari makalah ini adalah sebagai berikut:

1) *Benchmarking* Kinerja: Penulis menyediakan data empiris yang membandingkan biaya komputasi operasi token dan *overhead bandwidth* yang terkait dengan ukuran token.
2) Analisis Delegasi: Penulis mendemonstrasikan keunggulan arsitektural spesifik dari Macaroons dalam skenario yang memerlukan delegasi berganda dan atenuasi *privilege*.

D. Organisasi Makalah

Sisa makalah ini disusun sebagai berikut: Bagian II menyediakan latar belakang dan pekerjaan terkait, merinci spesifikasi teknis JWT dan Macaroons. Bagian III menjelaskan metodologi yang digunakan untuk evaluasi eksperimental. Bagian IV menyajikan hasil eksperimen penulis. Bagian V membahas implikasi dari temuan penulis, dan Bagian VI menyimpulkan makalah dengan rekomendasi untuk pekerjaan masa depan.

II. Latar Belakang dan Pekerjaan Terkait

Bagian ini menyediakan latar belakang teoritis yang diperlukan tentang JSON Web Tokens (JWT) dan Macaroons, merinci komposisi struktural dan mekanisme kriptografi mereka. Bagian ini juga meninjau literatur yang ada yang membandingkan token otorisasi dalam sistem terdistribusi.

A. Message Authentication Code (MAC)

Message Authentication Code (MAC) adalah fungsi kriptografi yang digunakan untuk mengautentikasi pesan. Ini adalah fungsi hash berkunci yang digunakan untuk mengautentikasi pesan. Kunci digunakan untuk menandatangani pesan, dan tanda tangan digunakan untuk memverifikasi pesan. [3]

$$\text{MAC}(K, M) = C_K(M) \tag{1}$$

di mana K adalah kunci, M adalah pesan, dan C adalah algoritma MAC.

Seperti yang ditunjukkan pada Gambar 1, konstruksi MAC dalam pekerjaan ini secara formal menangkap proses di mana kunci rahasia bersama digunakan di kedua sisi untuk mengikat tag ke pesan, memberikan jaminan keaslian dan integritas untuk mekanisme selanjutnya.

B. JSON Web Tokens (JWT)

JSON Web Token (JWT) adalah standar terbuka (RFC 7519) yang mendefinisikan cara yang ringkas dan *self-contained* untuk mentransmisikan informasi secara aman antara pihak sebagai objek JSON [1]. Informasi ini dapat diverifikasi dan dipercaya karena ditandatangani secara digital.

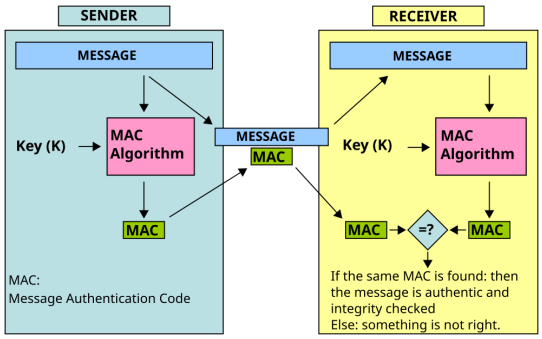

Figure 1. Ilustrasi proses generasi dan verifikasi MAC antara pengirim dan penerima. Pengirim menghitung nilai MAC dari pesan dan kunci rahasia bersama menggunakan algoritma MAC, kemudian mengirim pesan dan MAC ke penerima. Penerima menghitung ulang MAC menggunakan kunci yang sama dan membandingkannya dengan MAC yang diterima; jika cocok, keaslian dan integritas pesan terverifikasi [3]. Sumber: ResearchGate [4].

1) Struktur: JWT terdiri dari tiga bagian yang dipisahkan oleh titik (`.`):

- ***Header:*** Biasanya terdiri dari dua bagian: tipe token, yaitu JWT, dan algoritma penandatanganan yang digunakan, seperti HMAC SHA256 atau RSA.
- ***Payload:*** Berisi klaim. Klaim adalah pernyataan tentang entitas (biasanya, pengguna) dan data tambahan.
- ***Signature:*** Untuk membuat bagian tanda tangan, header yang dikodekan, *payload* yang dikodekan, sebuah *secret*, dan algoritma yang ditentukan dalam *header* digunakan.

2) Keterbatasan: Keterbatasan utama JWT terletak pada imutabilitasnya terkait *signature*. Setiap modifikasi pada *payload*, seperti menghapus izin atau memperpendek waktu kedaluwarsa, membatalkan *signature*. Oleh karena itu, delegasi mengharuskan pemegangnya untuk menghubungi penerbit guna mendapatkan token baru dengan cakupan yang dikurangi, memperkenalkan kembali ketergantungan pada otoritas pusat.

C. Macaroons

Macaroons diperkenalkan oleh Birgisson et al. di Google [2] sebagai alternatif yang lebih fleksibel untuk cookies dan *token bearer*. Mereka dibangun menggunakan konstruksi HMAC berantai yang memungkinkan delegasi terdesentralisasi.

1) Konstruksi: Sebuah Macaroon terdiri dari *location*, *identifier*, dan *signature*. *Signature* diturunkan dari *root key* (hanya diketahui oleh penerbit) dan *identifier*.

$$\sigma_0 = \text{HMAC}(K_{root}, \text{id}) \tag{2}$$

2) Caveats dan Atenuasi: Inovasi kunci dari Macaroons adalah kemampuan untuk menambahkan *caveats* (pembatasan). Setiap *caveat* memperbarui *signature* menggunakan mekanisme *chaining*:

$$\sigma_n = \text{HMAC}(\sigma_{n-1}, \text{Caveat}_n) \tag{3}$$

Proses ini memungkinkan pengguna untuk mengambil Macaroon yang valid, menambahkan pembatasan baru (misalnya, `time < 2023-12-31`), dan menghitung *signature* valid

baru σ_n tanpa mengetahui *root key* K_{root}. Macaroon turunan ini valid tetapi memiliki otoritas yang lebih sedikit dari yang asli. Properti ini dikenal sebagai ”atenuasi” dan memungkinkan delegasi *offline*.

3) Third-Party Caveats: Macaroons juga mendukung *third-party caveats*, yang mengharuskan pembawa untuk mendapatkan macaroon *”discharge”* dari pihak ketiga tertentu untuk membuktikan bahwa suatu kondisi (seperti autentikasi pengguna di layanan berbeda) terpenuhi. Ini memungkinkan pemeriksaan otorisasi terdistribusi tanpa komunikasi langsung antara *verifier* dan pihak ketiga.

III. Metodologi Penelitian

Untuk mengevaluasi kelayakan dan efisiensi Macaroons dibandingkan dengan JSON Web Tokens (JWT) dalam lingkungan *microservices*, penulis merancang penelitian komparatif yang berfokus pada tiga metrik kunci: kinerja, fleksibilitas (atenuasi), dan skalabilitas.

A. Setup Eksperimen

Eksperimen dilakukan pada lingkungan lokal menggunakan Python 3.12. Arsitektur sistem terdiri dari tiga komponen berbeda yang mensimulasikan topologi *microservices* dunia nyata:

1) ***Backend Service:*** *Server resource mock* yang berjalan di `waitress` (Production WSGI).
2) ***API Gateway:*** Layanan perantara yang berjalan di port 8000 yang bertanggung jawab untuk verifikasi token sebelum meneruskan *request*.
3) ***Client/Runner:*** *Script* untuk mengeksekusi tes fungsional dan *stress test* dengan konkurensi tinggi.

B. Detail Implementasi

1) Standar Kriptografi: Macaroons diimplementasikan menggunakan HMAC-SHA256 (simetris), sedangkan JWT menggunakan RS256 (tanda tangan RSA asimetris), yang masing-masing mewakili praktik standar industri untuk setiap tipe token. Pilihan ini mencerminkan perbedaan arsitektural yang fundamental antara kedua pendekatan tersebut: Macaroons mengandalkan kriptografi simetris untuk struktur HMAC berantai, sedangkan JWT pada umumnya menggunakan tanda tangan asimetris untuk proses verifikasi.

2) Framework dan Infrastruktur: Layanan dibangun menggunakan *framework* Flask yang dilayani oleh Waitress untuk memastikan penanganan *production-grade* dari permintaan konkuren, menghindari keterbatasan dari server pengembangan Flask default. Setup ini menyediakan lingkungan yang realistis untuk mengevaluasi karakteristik kinerja dalam kondisi seperti produksi.

C. Skenario Pengujian

Penulis mendefinisikan tiga skenario pengujian spesifik untuk mengevaluasi kedua format token secara komprehensif:

1) Kinerja Baseline: Skenario pertama mengukur ukuran byte mentah dari token dan waktu komputasi yang diperlukan untuk satu operasi verifikasi. *Baseline* ini menetapkan karakteristik kinerja fundamental dan memberikan dasar untuk memahami *overhead* komputasi dari setiap pendekatan.

2) Delegasi dan Atenuasi: Skenario kedua mengukur waktu yang diperlukan untuk membatasi izin token (misalnya, menurunkan dari “*Full Access*” ke “*Read Only*”). Untuk Macaroons, ini adalah operasi kriptografi lokal yang dapat dilakukan *offline* tanpa interaksi server. Untuk JWT, ini mensimulasikan pola “*Token Exchange*” yang memerlukan panggilan jaringan ke *Authentication Service*, karena JWT bersifat *immutable* setelah ditandatangani.

3) Stress Test Skalabilitas: Skenario ketiga menggunakan **Locust** untuk mensimulasikan lingkungan dengan beban tinggi dengan 500 pengguna konkuren, mengukur *Throughput (Requests Per Second/RPS)* dan mengidentifikasi *bottleneck* sistem. *Stress test* ini mengevaluasi bagaimana setiap format token berkinerja di bawah beban kerja *production* yang realistis dan membantu mengidentifikasi keterbatasan skalabilitas.

IV. Hasil dan Diskusi

Bagian ini menyajikan data empiris yang diperoleh dari eksperimen yang membandingkan Macaroons dan JWT dalam konteks otorisasi *microservices*.

A. Lingkungan Eksperimen

Eksperimen dilakukan pada perangkat pribadi dengan spesifikasi sebagai berikut:

- **Prosesor**: Intel Core i5-11400H @ 2.70 GHz
- **RAM**: 16.0 GB

Implementasi menggunakan bahasa pemrograman Python dengan *library* berikut:

- `PyJWT==2.10.1` untuk implementasi JSON Web Token
- `pymacaroons==0.13.0` untuk implementasi Macaroons

B. Metodologi Perbandingan

Studi ini membandingkan Macaroons dengan HMAC-SHA256 versus JWT dengan RS256. Pemilihan algoritma ini memerlukan justifikasi karena secara fundamental membandingkan kriptografi simetris dengan asimetris, yang memiliki karakteristik kinerja yang berbeda secara inheren.

1) Justifikasi Pemilihan Algoritma: RS256 (RSA dengan SHA-256) dipilih untuk JWT karena merupakan algoritma yang paling umum digunakan dalam *deployment* produksi JWT. Menurut praktik industri, RS256 menjadi standar *de facto* untuk sistem yang memerlukan verifikasi *public key*, seperti *Single Sign-On* (SSO), OAuth 2.0, dan OpenID Connect. Pemilihan ini mencerminkan realitas implementasi JWT di lingkungan produksi.

Di sisi lain, Macaroons secara desain menggunakan kriptografi simetris (HMAC) karena mekanisme *chained authentication* dan atenuasi bergantung pada kemampuan untuk menurunkan kunci baru dari kunci sebelumnya.

2) Tujuan Perbandingan: Tujuan utama perbandingan ini adalah mengevaluasi kinerja *real-world deployment*, bukan *cryptographic equivalence*. Oleh karena itu, perbandingan ini secara sengaja menggunakan konfigurasi yang mencerminkan praktik industri yang umum, bukan konfigurasi yang memaksimalkan kesetaraan kriptografis.

3) Limitasi: Penting untuk dicatat bahwa sebagian perbedaan kinerja yang diamati berasal dari perbedaan fundamental antara kriptografi simetris dan asimetris, bukan semata-mata dari desain format token. Jika JWT menggunakan HS256 (HMAC-SHA256), gap kinerja untuk operasi verifikasi akan jauh lebih kecil. Namun, perbandingan semacam itu akan kurang relevan untuk *deployment* produksi tipikal di mana RS256 lebih disukai karena kemampuan distribusi *public key*-nya.

C. Perbandingan Kinerja

Tabel I mengilustrasikan karakteristik *baseline* dari kedua format token, termasuk ukuran token dan waktu verifikasi.

Table I
METRIK KINERJA BASELINE

Metrik	Macaroon (HMAC-SHA256)	JWT (RS256)
Ukuran Token	∼248 bytes	∼491 bytes
Waktu Verifikasi	∼0.11 ms	∼0.19 ms

Macaroons mendemonstrasikan **pengurangan 49.5% dalam ukuran token** dibandingkan dengan JWT. Hal ini dikaitkan dengan sifat *compact* dari hash HMAC berantai versus struktur JSON yang verbose dan blok tanda tangan RSA yang besar yang diperlukan oleh JWT RS256. Lebih lanjut, verifikasi Macaroon sekitar **42% lebih cepat**, karena hashing simetris secara komputasi lebih murah daripada verifikasi tanda tangan RSA.

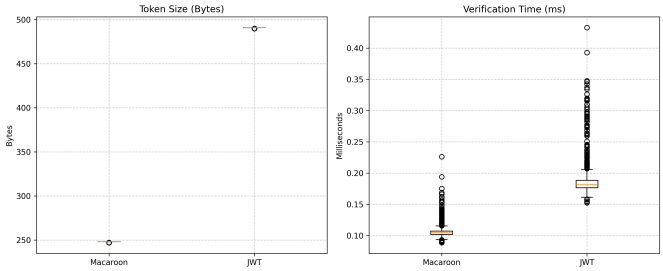

Figure 2. Perbandingan kinerja antara Macaroons dan JWT yang menunjukkan metrik ukuran token dan waktu verifikasi. Sumber: Dokumentasi penulis.

D. Fleksibilitas: Delegasi dan Atenuasi

Disparitas paling signifikan diamati dalam proses atenuasi. Ini merepresentasikan kasus penggunaan dalam arsitektur *microservices* di mana token perlu dibatasi secara dinamis tanpa memerlukan intervensi server.

1) Atenuasi Macaroon: Proses atenuasi untuk Macaroons memakan waktu ∼**0.015 ms**. Karena Macaroons mendukung atenuasi *offline* (klien dapat menurunkan HMAC baru yang dibatasi dari token asli tanpa intervensi server), operasi ini murni lokal dan *CPU-bound*. Klien dapat secara independen membuat Macaroon baru dengan izin yang dikurangi dengan menambahkan *caveat* dan menghitung rantai HMAC yang sesuai.

2) Token Exchange JWT: Untuk JWT, proses ini memakan waktu ∼**35.79 ms**. Karena JWT bersifat *immutable* setelah ditandatangani, "atenuasi" mengharuskan klien untuk meminta token baru dari penerbit, memperkenalkan latensi jaringan dan *overhead* I/O. Ini merepresentasikan keterbatasan arsitektural fundamental dari sistem berbasis JWT untuk manajemen izin dinamis.

3) Analisis: Hasil menunjukkan bahwa Macaroons sekitar 2300× **lebih cepat** untuk skenario delegasi. Ini mengkonfirmasi bahwa Macaroons superior untuk kasus penggunaan yang memerlukan *chaining* izin dinamis tanpa membebani layanan autentikasi pusat. Kemampuan untuk melakukan atenuasi *offline* membuat Macaroons sangat cocok untuk sistem terdistribusi karena pada kondisi ini latensi jaringan dapat secara signifikan mempengaruhi kinerja.

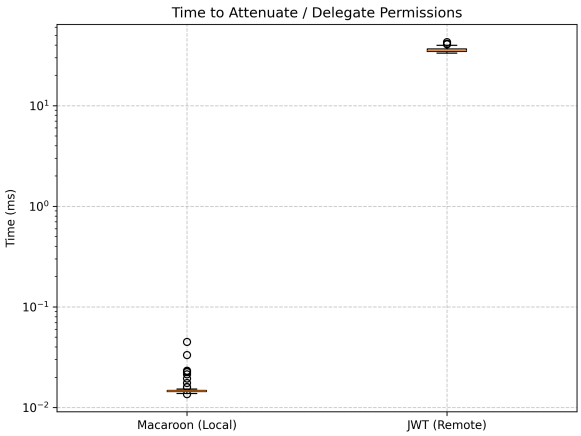

Figure 3. Perbandingan kinerja delegasi dan atenuasi. Macaroons memungkinkan atenuasi offline, sementara JWT memerlukan token exchange berbasis jaringan. Sumber: Dokumentasi penulis.

E. Stress Test Skalabilitas

Hasil *stress test* menggunakan 500 pengguna konkuren disajikan dalam Tabel II.

Table II
HASIL *Stress Test* (500 PENGGUNA KONKUREN)

Metrik	Macaroon	JWT
Throughput	∼**371 req/s**	∼171 req/s
Bottleneck Utama	CPU (*Hashing*)	I/O & Crypto (RSA)

Macaroons mencapai **throughput** 2.17× **lebih tinggi** dibandingkan JWT. Implementasi JWT memiliki latensi yang lebih tinggi karena biaya komputasi kriptografi asimetris dan *overhead* dari alur *token exchange* berbasis jaringan selama simulasi. Macaroons tetap sangat efisien, dibatasi terutama oleh kecepatan CPU mentah daripada waktu tunggu I/O.

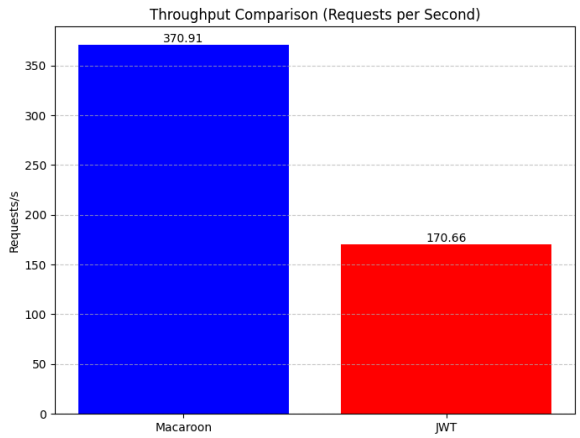

Figure 4. Perbandingan throughput di bawah beban tinggi (500 pengguna konkuren). Macaroons mendemonstrasikan skalabilitas superior dengan tingkat permintaan 2.17× lebih tinggi. Sumber: Dokumentasi penulis.

F. Diskusi

1) Implikasi Kinerja: Hasil eksperimental mendemonstrasikan keunggulan kinerja yang jelas untuk Macaroons di metrik yang diuji. Pengurangan 49.5% dalam ukuran token secara langsung dapat diartikan bahwa konsumsi *bandwidth* berkurang dan *overhead* memori yang lebih rendah dalam sistem terdistribusi. Waktu verifikasi 42% lebih cepat meningkatkan latensi respons, yang kritis untuk arsitektur *microservices* dengan *throughput* tinggi.

Temuan paling signifikan adalah peningkatan kecepatan 2300× dalam skenario delegasi. Kemampuan ini memungkinkan pola arsitektural baru di mana klien dapat secara independen membatasi izin token tanpa memerlukan *round-trip* ke layanan autentikasi. Ini sangat berharga dalam skenario *edge computing* dan sistem dengan konektivitas jaringan yang tidak kontinu.

2) Pertimbangan Skalabilitas: Hasil *stress test* mengungkapkan perbedaan fundamental dalam bagaimana setiap format token *scaling* di bawah beban. Macaroons, yang *CPU-bound*, mendapat manfaat dari prosesor *multi-core* modern dan dapat dengan mudah diparalelkan. JWT, dibatasi oleh operasi I/O dan operasi kriptografi asimetris, menghadapi *bottleneck* inheren yang membatasi skalabilitas.

Keunggulan *throughput* 2.17× dari Macaroons berarti bahwa sistem dapat menangani lebih dari dua kali lipat beban permintaan dengan infrastruktur yang sama, menghasilkan penghematan biaya yang signifikan dan pengalaman pengguna yang lebih baik dalam skenario dengan lalu lintas tinggi.

3) Trade-off Arsitektural: Meskipun Macaroons mendemonstrasikan karakteristik kinerja yang superior, penting untuk mengakui *trade-off*-nya. Macaroons memerlukan manajemen kunci yang hati-hati, karena *root key* harus dilindungi. Namun, kemampuan untuk melakukan atenuasi *offline* memberikan fleksibilitas arsitektural yang tidak dapat ditandingi JWT.

JWT mendapat manfaat dari adopsi yang luas dan dukungan komunitas yang besar, tetapi sifat *immutable* mereka menciptakan keterbatasan untuk skenario otorisasi dinamis. Pola *token exchange* berbasis jaringan yang diperlukan untuk atenuasi JWT memperkenalkan latensi dan meningkatkan beban pada layanan autentikasi.

4) Practical Implications: Hasil menunjukkan bahwa Macaroons sangat cocok untuk:

- Arsitektur *microservices* dengan throughput tinggi yang memerlukan verifikasi token dengan latensi rendah
- Sistem yang memerlukan manajemen izin dinamis tanpa intervensi server
- Skenario *edge computing* dengan konektivitas jaringan terbatas atau tidak kontinu
- Aplikasi di mana optimasi *bandwidth* sangat kritis

JWT tetap sesuai untuk skenario di mana:

- Imutabilitas token adalah properti keamanan yang diinginkan
- Dukungan komunitas yang besar dan *tooling* ekstensif diprioritaskan
- Verifikasi token *stateless* yang sederhana sudah cukup
- Latensi jaringan untuk *token exchange* dapat diterima

Evaluasi komprehensif mendemonstrasikan bahwa Macaroons menawarkan keunggulan signifikan untuk arsitektur *microservices* terdesentralisasi dengan kinerja tinggi yang memerlukan mekanisme otorisasi yang fleksibel.

V. Kesimpulan

Makalah ini menyajikan analisis komparatif antara Macaroons dan JSON Web Tokens (JWT) untuk otorisasi *microservices*. Evaluasi eksperimental mendemonstrasikan bahwa Macaroons mengungguli JWT di semua metrik yang diuji, memberikan bukti yang meyakinkan untuk adopsi mereka dalam sistem terdistribusi dengan kinerja tinggi.

A. Ringkasan Temuan

Penelitian penulis telah berhasil mendemonstrasikan keunggulan kinerja dan fleksibilitas Macaroons dibandingkan JWT dalam skenario otorisasi *microservices*. Temuan kunci meliputi:

1) **Efisiensi Ukuran Token:** Macaroons mencapai pengurangan 49.5% dalam ukuran token dibandingkan JWT, menghasilkan konsumsi *bandwidth* dan *overhead* memori yang lebih rendah.
2) **Kinerja Verifikasi:** Verifikasi Macaroon sekitar 42% lebih cepat dibandingkan verifikasi JWT, meningkatkan latensi respons dalam sistem dengan *throughput* tinggi.
3) **Kemampuan Delegasi:** Keunggulan paling signifikan adalah peningkatan kecepatan 2300× dalam skenario

delegasi, dimungkinkan oleh kemampuan atenuasi *offline* Macaroons.

4) **Skalabilitas:** Di bawah beban tinggi dengan 500 pengguna konkuren, Macaroons mencapai *throughput* 2.17× lebih tinggi dibandingkan JWT, mendemonstrasikan karakteristik *scalability* yang superior.

B. Pencapaian Tujuan

Tujuan eksperimental yang diuraikan dalam metodologi telah berhasil dicapai:

- **Evaluasi Kinerja:** Metrik kinerja *baseline* yang komprehensif dikumpulkan, dengan jelas mendemonstrasikan keunggulan Macaroons dalam ukuran token dan kecepatan verifikasi.
- **Penilaian Fleksibilitas:** Eksperimen delegasi dan atenuasi mengungkapkan keunggulan arsitektural fundamental Macaroons untuk manajemen izin dinamis.
- **Analisis Skalabilitas:** *Stress testing* di bawah kondisi beban yang realistis mengkonfirmasi *throughput* superior Macaroons dan mengidentifikasi *bottleneck* utama untuk setiap pendekatan.

C. Kontribusi Kunci

Pekerjaan ini berkontribusi pada pemahaman otorisasi berbasis token dalam arsitektur *microservices* dengan menyediakan bukti empiris keunggulan kinerja Macaroons melalui evaluasi eksperimental yang komprehensif. Penelitian ini mengkuantifikasi dampak kemampuan atenuasi *offline* pada kinerja sistem dan fleksibilitas arsitektural, mendemonstrasikan karakteristik *scalability* kedua format token di bawah beban kerja produksi yang realistis. Lebih lanjut, studi ini mengidentifikasi kasus penggunaan spesifik di mana setiap format token paling sesuai, memberikan panduan praktis untuk arsitek sistem dan pengembang.

D. Practical Implications

Hasil eksperimental memiliki implikasi praktis yang signifikan untuk arsitek sistem dan pengembang. Untuk sistem dengan kinerja tinggi, Macaroons menyediakan solusi yang lebih scalable dan efisien untuk sistem yang memerlukan *throughput* tinggi dan latensi rendah. Kemampuan atenuasi *offline* memungkinkan pola arsitektural baru yang mengurangi ketergantungan pada layanan autentikasi terpusat, membuat Macaroons sangat berharga untuk skenario otorisasi dinamis. Ukuran token yang berkurang dan verifikasi yang lebih cepat diterjemahkan menjadi biaya infrastruktur yang lebih rendah dan pengalaman pengguna yang lebih baik, berkontribusi pada optimasi sumber daya. Selain itu, kemampuan *offline* Macaroons membuat mereka sangat cocok untuk skenario *edge computing* dengan konektivitas jaringan terbatas, di mana sistem berbasis token tradisional mungkin menghadapi tantangan.

REPOSITORI

Kode sumber lengkap, setup eksperimen, dan implementasi detail untuk analisis komparatif Macaroons dan JWT tersedia di:

https://github.com/TazakiN/macaroons-experiment.

Repositori ini mencakup semua kode, file konfigurasi, dan dokumentasi yang diperlukan untuk mereproduksi hasil eksperimental yang dibahas dalam makalah ini.

UCAPAN TERIMA KASIH

Penulis ingin mengucapkan rasa syukur kepada Allah SWT yang telah memberikan ilmu dan kebijaksanaan untuk melakukan penelitian ini. Terima kasih juga kepada keluarga dan teman-teman yang telah mendukung penulis dalam perjalanan akademis. Ucapan terima kasih khusus diberikan kepada Pak Rinaldi sebagai dosen mata kuliah Kriptografi di Institut Teknologi Bandung, atas kontribusinya yang berkelanjutan di bidang ini. Penulis juga berterima kasih kepada para pengembang *library* dan *tool* kriptografi *open-source* yang memfasilitasi implementasi dan evaluasi metode yang penulis usulkan.

REFERENCES

[1] M. B. Jones, J. Bradley, and N. Sakimura, "Json web token (jwt)," Internet Engineering Task Force, RFC 7519, May 2015. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc7519.

[2] A. Birgisson, J. G. Politz, U. Erlingsson, A. Taly, and M. Vrable, "Macaroons: Cookies with caveats for decentralized authorization in the cloud," in *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, Internet Society, 2014.

[3] R. Munir, *MAC (Message Authentication Code)*, Bahan Kuliah Kriptografi, Program Studi Teknik Informatika, Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung, Accessed: 2025-12-07, 2025. [Online]. Available: https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2025-2026/28-MAC-2025.pdf.

[4] Y. Lindell and J. Katz, *Message authentication code (MAC)*, Scientific Figure on ResearchGate, from: Integrity Checking of Several Program Codes, Accessed: 2025-12-26, 2014. [Online]. Available: https://www.researchgate.net/figure/Message-Authentication-Code-MAC-Lindell-et-al-2014_fig3_330566009.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 26 Desember 2025

Tazkia Nizami
NIM: 13522032