

Analisa Performa Penandatanganan Digital Dokumen Secara Terdistribusi Pada Arsitektur Klaster Kubernetes

¹Sofyan Noor Arief, ²Arief Prasetyo, ³Irsyad Alif Mashudi, ⁴Chandrasena Setiadi
^{1,2,3,4} Politeknik Negeri Malang

sofyan@polinema.ac.id, arief.prasetyo@polinema.ac.id, irsyad.alif@polinema.ac.id,
chandrasena@polinema.ac.id

Submit : 19 Nov 2025 | Diterima : 28 Nov 2025 | Terbit : 30 Nov 2025

ABSTRAK

Dalam era digitalisasi yang berkembang, penandatanganan elektronik semakin vital dalam menanggapi kebutuhan bisnis dan gaya hidup modern. Proses ini memungkinkan transaksi dan persetujuan dokumen tanpa batasan geografis, mengatasi tantangan jarak jauh, menghemat waktu, dan mengurangi birokrasi. Regulasi yang semakin mengakui keabsahan tanda tangan elektronik mendorong organisasi beralih ke solusi ini, meningkatkan produktivitas dan memastikan keamanan transaksi. Penandatanganan dokumen elektronik menggunakan kriptografi, seperti algoritma RSA atau ECDSA, menciptakan tanda tangan digital yang unik. Proses ini mempercepat alur kerja, meminimalkan birokrasi, dan memungkinkan transaksi bisnis yang efisien. Standar PKI dan format dokumen seperti PDF/A menjamin keamanan dan interoperabilitas, sementara klaster Kubernetes memberikan layanan tanpa henti dengan otomatisasi penjadwalan, penskalaan, dan toleransi kesalahan. Implementasi ini dapat menjadi solusi efektif dan aman untuk kebutuhan penandatanganan dokumen digital di dunia bisnis dan komunikasi modern.

Kata Kunci: Kualitas Pelayanan dan Kepuasan Pelanggan

PENDAHULUAN

Dalam era digitalisasi yang terus berkembang, kebutuhan akan penandatanganan secara elektronik semakin meningkat sebagai respons terhadap tuntutan bisnis dan kehidupan modern. Proses ini menjadi penting karena memungkinkan pelaku bisnis, pemerintah, dan individu untuk melakukan transaksi, menyetujui dokumen, dan melibatkan pihak terlibat tanpa batasan geografis. Penandatanganan elektronik menyediakan solusi efisien dan aman untuk mengatasi tantangan jarak jauh, menghemat waktu, dan meminimalkan birokrasi. Di samping itu, dengan regulasi yang semakin mengakui keabsahan tanda tangan elektronik, organisasi cenderung beralih ke solusi ini untuk meningkatkan produktivitas, mengurangi biaya, dan memastikan keamanan transaksi di dunia digital yang terus berkembang.

Penandatanganan dokumen secara elektronik merujuk pada proses menggunakan sarana digital untuk memberikan persetujuan atau identifikasi pada suatu dokumen, menggantikan tanda tangan tradisional di atas kertas. Metode ini melibatkan teknologi kriptografi, seperti kunci privat-publik, dan menggunakan algoritma keamanan untuk membuat tanda tangan digital yang unik. Dengan demikian, penandatanganan elektronik memungkinkan individu atau organisasi untuk memberikan keabsahan dan keaslian terhadap dokumen tanpa perlu pertemuan fisik. Proses ini tidak hanya mempercepat alur kerja dan mengurangi birokrasi, tetapi juga menyediakan solusi efisien untuk melakukan transaksi bisnis, menandai persetujuan, atau memvalidasi kontrak di era digital yang terus berkembang. Keseluruhan, penandatanganan dokumen secara elektronik menjadi aspek integral dalam memfasilitasi interaksi dan pertukaran informasi di dunia digital saat ini.

Penandatanganan dokumen secara digital dilakukan melalui serangkaian langkah

menggunakan teknologi kriptografi untuk menciptakan tanda tangan elektronik yang sah dan aman. Pertama, pihak yang ingin menandatangani dokumen mengidentifikasi diri menggunakan kunci privat yang disimpan dengan aman. Setelah itu, dokumen yang akan ditandatangani diubah menjadi nilai hash menggunakan fungsi hash seperti SHA-256. Nilai hash ini kemudian ditandatangani dengan menggunakan kunci privat menggunakan algoritma kriptografi, seperti RSA atau ECDSA. Tanda tangan digital yang dihasilkan bersifat unik dan terkait erat dengan dokumen aslinya. Dokumen bersama dengan tanda tangan digital kemudian dapat dikirim ke pihak lain yang dapat memverifikasi tanda tangan tersebut menggunakan kunci publik yang sesuai. Proses ini memastikan integritas, otentikasi, dan keaslian dokumen elektronik tanpa perlu tanda tangan fisik atau kehadiran langsung. Dengan demikian, penandatanganan dokumen secara digital memberikan solusi yang efisien dan terjamin keamanannya dalam dunia bisnis dan komunikasi modern.

Penandatanganan dokumen secara elektronik diatur oleh berbagai standar yang menjamin keamanan, interoperabilitas, dan keabsahan tanda tangan digital. Salah satu standar yang umum digunakan adalah PKI (Public Key Infrastructure), yang melibatkan penggunaan pasangan kunci asimetris, seperti RSA atau ECDSA. Selain itu, format dokumen yang sering digunakan adalah PDF/A (ISO 19005) yang dirancang untuk penyimpanan arsip elektronik jangka panjang dan mendukung penandatanganan digital. Adopsi algoritma hash standar, seperti SHA-256, SHA-3, atau SHA-512, juga merupakan praktik umum untuk menghasilkan nilai hash dari dokumen sebelum tanda tangan. Selanjutnya, standar X.509 digunakan untuk sertifikat tanda tangan digital, yang mencakup informasi identitas dan kunci publik pemilik sertifikat. Sertifikat tersebut diterbitkan oleh Lembaga Sertifikasi Elektronik (LSE) yang terakreditasi oleh otoritas, seperti Kementerian Komunikasi dan Informatika. Adanya standar-standar ini membantu memastikan keamanan, keandalan, dan interoperabilitas dalam implementasi penandatanganan dokumen secara elektronik di berbagai platform dan aplikasi.

Klaster Kubernetes adalah infrastruktur yang digunakan untuk mengelola dan mendeploy aplikasi berbasis kontainer secara efisien dan skalabel. Kubernetes, sering disingkat sebagai K8s, memberikan lingkungan yang kokoh dan otomatis untuk mengelola siklus hidup aplikasi, mulai dari pembuatan hingga peningkatan dan penskalaan horizontal. Klaster ini terdiri dari beberapa mesin fisik atau virtual yang disebut sebagai "node," yang dikelompokkan bersama untuk membentuk suatu lingkungan yang terkelola secara terpusat. Setiap node dapat menjalankan satu atau lebih kontainer, yang diatur oleh Kubernetes untuk berkomunikasi dan beroperasi sebagai satu kesatuan. K8s menyediakan layanan seperti otomatisasi penjadwalan, manajemen daya, dan pemantauan kesehatan aplikasi. Dengan adanya konsep klaster Kubernetes, pengembang dan administrator dapat memanfaatkan keuntungan skalabilitas, toleransi kesalahan, dan manajemen sumber daya yang efisien, menjadikannya solusi yang populer untuk orkestrasi kontainer di lingkungan cloud atau on-premise.

JMeter merupakan software Open Source untuk pengujian. JMeter merupakan 100% aplikasi Java murni untuk pengujian beban dan kinerja. JMeter dirancang untuk melakukan beberapa kategori tes seperti beban, fungsional, kinerja, regresi, dll. JMeter sendiri itu membutuhkan JDK (Java Development Kits) 5 atau lebih tinggi agar bisa dijalankan pada sebuah komputer (Apache, 2014).

Aplikasi Jmeter merupakan aplikasi pengujian kinerja yang fleksibel yang dapat mensimulasikan banyak pengguna aplikasi secara bersamaan sehingga dapat dilakukan evaluasi terhadap kinerja aplikasi. Pengujian dengan jumlah pengguna yang dapat disesuaikan ini akan sangat berguna bagi organisasi karena dapat menggambarkan kondisi yang sebenarnya. (Wang and Wu, 2019). Pada pengujiannya, aplikasi seperti Jmeter dirancang untuk menangkap kinerja pada tingkat antar muka pengguna dan system itu sendiri. Menggunakan aplikasi seperti ini dapat membantu menghemat waktu dan biaya organisasi. (Agnihotri and Phalnikar, 2018)

Dengan menerapkan klaster Kubernetes dalam konteks penandatanganan dokumen secara digital, dapat diasumsikan bahwa ini akan memberikan layanan tanpa henti dengan beberapa cara. Pertama, kemampuan otomatisasi penjadwalan dan penskalaan horizontal Kubernetes dapat memberikan keandalan tinggi terhadap lonjakan permintaan untuk penandatanganan dokumen.

Klaster dapat secara dinamis menyesuaikan jumlah instance atau kontainer yang berjalan sesuai dengan beban kerja, memastikan ketersediaan layanan secara optimal. Selain itu, fitur toleransi kesalahan Kubernetes memungkinkan klaster untuk secara otomatis mendeteksi dan mengatasi kegagalan pada salah satu nodenya, sehingga layanan tetap berjalan tanpa terganggu. Manajemen sumber daya yang efisien dari Kubernetes juga dapat membantu mengoptimalkan penggunaan infrastruktur, memastikan bahwa penandatanganan dokumen berlangsung dengan cepat dan efektif. Dengan demikian, klaster Kubernetes dapat menjadi fondasi yang kuat untuk mendukung layanan penandatanganan dokumen digital tanpa henti, menjawab tantangan keandalan dan skalabilitas di lingkungan yang terus berkembang.

TINJAUAN PUSTAKA

Penandatanganan Dokumen Digital

Penandatanganan dokumen secara digital telah menjadi topik yang berkembang pesat dalam literatur akademis dan industri, menyajikan pandangan mendalam tentang keamanan, legalitas, dan efektivitas. Proses ini penting dalam era digitalisasi karena memungkinkan transaksi dan persetujuan dokumen tanpa batasan geografis, menghemat waktu, dan mengurangi birokrasi.

Secara hukum, tanda tangan digital memiliki dasar yang kuat di banyak yurisdiksi, termasuk pengakuan dari undang-undang seperti ESIGN Act di Amerika Serikat dan eIDAS di Uni Eropa (G. Di Crescenzo dan A. Persaud, 2020). Teknologi kriptografi modern dan Infrastruktur Kunci Publik (PKI) mendukung integritas dan otentikasi tanda tangan digital. Dalam implementasinya, berbagai aplikasi penandatanganan digital telah dieksplorasi di sektor-sektor seperti perbankan, kesehatan, dan administrasi publik (R. Kumar et al., 2019).

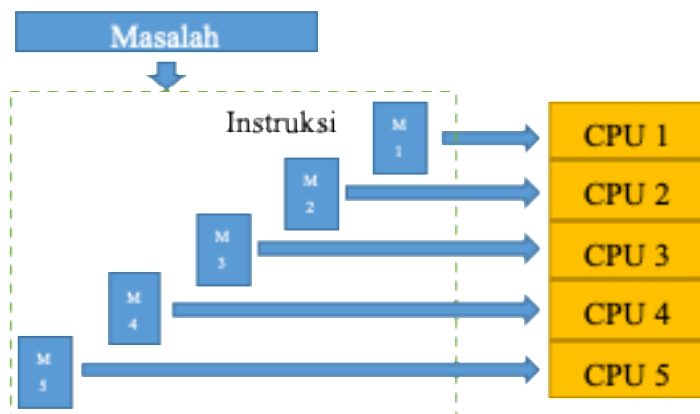
Algoritma RSA

Penelitian mengenai Algoritma RSA (Rivest, Shamir, dan Adleman) telah menjadi landasan utama dalam bidang kriptografi modern, dimulai dari karya seminal oleh penciptanya pada tahun 1978. Walaupun demikian, algoritma ini terus dievaluasi terkait serangan potensial, seperti yang diulas dalam penelitian "Twenty Years of Attacks on the RSA Cryptosystem" (Dan Boneh, 1999).

Dalam konteks penandatanganan dokumen digital, implementasi RSA diulas dalam kerangka hukum yang mendukung keabsahan tanda tangan (G. Di Crescenzo, A. Persaud, 2020). Penelitian ini secara spesifik membatasi masalahnya pada penggunaan algoritma asimetris RSA untuk platform penandatanganan.

Komputasi Terdistribusi (Distributed Computing)

Komputasi terdistribusi adalah bidang ilmu komputer yang mempelajari sistem terdistribusi, di mana beberapa komputer otonom berinteraksi melalui jaringan komputer untuk mencapai tujuan bersama (Andrews, G.R., 1999).

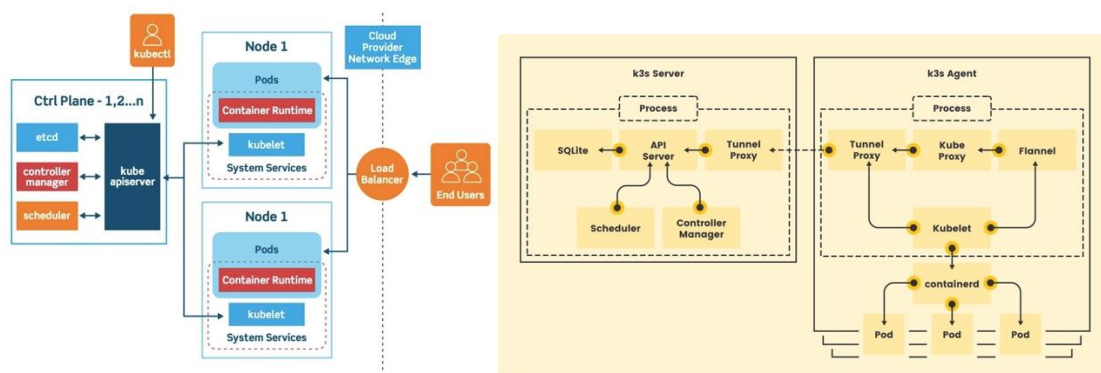


Gambar 1 Cara Kerja Komputasi Terdistribusi

Cara kerjanya adalah dengan mendistribusikan beban kerja ke komputer-komputer yang terhubung, di mana sistem operasi mengorganisasi kemampuan dan tugasnya. Ciri khas dari komputasi terdistribusi adalah heterogenitas (dalam perangkat keras, sistem operasi, dan bahasa pemrograman), di mana interoperabilitas menjadi kunci untuk menjembatani perbedaan-perbedaan yang muncul.

Kubernetes

Kubernetes (sering disingkat K8s) adalah platform open source portabel dan dapat diperluas yang digunakan untuk mengelola beban kerja dan layanan berbasis kontainer, memfasilitasi konfigurasi deklaratif dan otomatisasi (“Kubernetes Overview,” n.d.). Kluster Kubernetes terdiri dari mesin pekerja yang disebut node, yang menampung Pod—objek terkecil yang dapat dikontrol oleh Kubernetes dan berisi satu atau beberapa container.

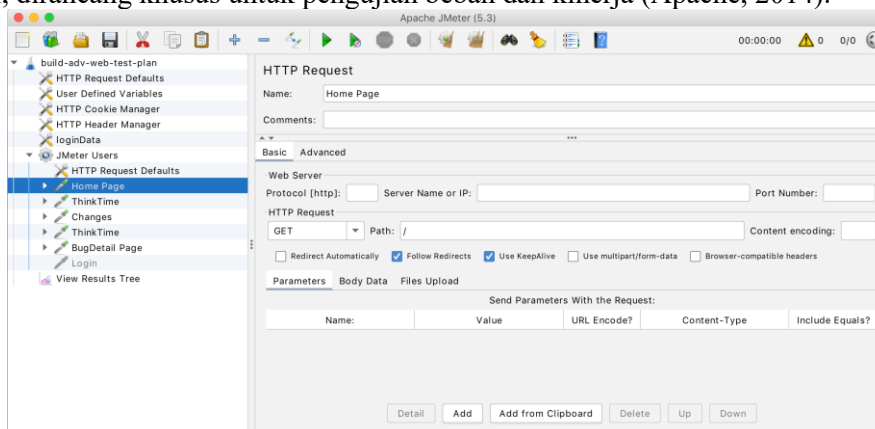


Gambar 2 Kubernetes

Kluster K8s menyediakan layanan seperti otomatisasi penjadwalan, manajemen daya, dan pemantauan kesehatan aplikasi. K8s dapat menjadi fondasi yang kuat untuk mendukung layanan penandatanganan dokumen digital tanpa henti karena kemampuan otomatisasi penjadwalan, penskalaan horizontal, dan toleransi kesalahan. Penelitian ini menggunakan Rancher Kubernetes.

Apache JMeter

JMeter merupakan perangkat lunak Open Source yang sepenuhnya merupakan aplikasi Java murni, dirancang khusus untuk pengujian beban dan kinerja (Apache, 2014).



Gambar 3 Tampilan Aplikasi JMeter

Aplikasi ini fleksibel karena dapat mensimulasikan banyak pengguna secara bersamaan, memungkinkan evaluasi kinerja aplikasi yang akurat, dan sangat berguna bagi organisasi karena dapat menggambarkan kondisi sebenarnya (Wang and Wu, 2019). Jenis tes utama yang dapat dilakukan oleh JMeter meliputi:

1. Uji Kinerja (Performance Test).
2. Uji Beban (Load Test).

3. Uji Tekanan (Stress Test).

Penggunaan JMeter dalam penelitian ini bertujuan untuk pengukuran performa dan pemberian simulasi pengguna pada platform penandatanganan dokumen digital.

METODE PENELITIAN

Populasi dan Sampel

1. Populasi

Populasi dalam penelitian ini adalah keseluruhan potensi proses penandatanganan dokumen digital secara terdistribusi yang dilaksanakan pada arsitektur kluster Kubernetes.

Populasi ini didefinisikan oleh lingkup teknologi yang digunakan:

- a. Platform Penandatanganan Digital Dokumen yang dibuat menggunakan algoritma asimetris RSA.
- b. Lingkungan orkestrasi kontainer, yaitu Arsitektur Kluster Kubernetes (khususnya Rancher Kubernetes) yang menyediakan layanan secara terpusat.
- c. Seluruh operasi penandatanganan yang melibatkan serangkaian langkah kriptografi seperti fungsi hash (misalnya SHA-256), penggunaan kunci privat, dan penyimpanan hasil PDF yang telah ditandatangani.

2. Sampel

Dalam penelitian ini, sample yang akan diujikan dapat dibagi menjadi 2 yaitu Sampel Beban Kerja (Simulasi Pengguna) dan Sample Dokumen. Sampel beban kerja merupakan variasi jumlah pengguna yang mensimulasikan permintaan penandatanganan dokumen secara bersamaan.

- a. Alat simulasi: Aplikasi pengujian Apache JMeter.
- b. Jumlah skenario pengujian: 37 skenario.
- c. Variasi pengguna (Number of Client): Ditingkatkan secara bertahap dari 1 pengguna hingga 475 pengguna.

Sampel dokumen merupakan bahan yang digunakan untuk pengujian yaitu berupa dokumen yang ditandatangani.

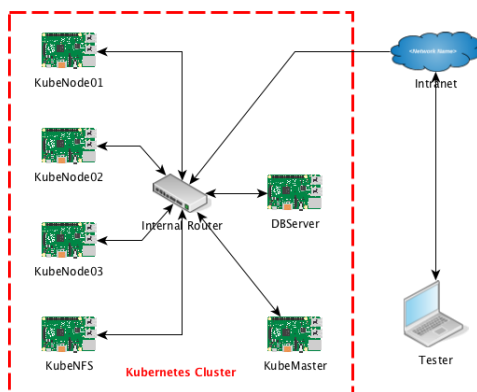
- a. Sebuah berkas dokumen sebesar 300KB diujikan.
- b. Batasan penelitian menyatakan bahwa jumlah dokumen yang digunakan divariasikan untuk mendapatkan kemampuan maksimal yang dapat dilayani.

Hipotesis

Dengan menerapkan kluster Kubernetes dalam konteks penandatanganan dokumen secara digital, dapat diasumsikan bahwa ini akan memberikan layanan tanpa henti dengan beberapa cara. Pertama, kemampuan otomatisasi penjadwalan dan penskalaan horizontal Kubernetes dapat memberikan keandalan tinggi terhadap lonjakan permintaan untuk penandatanganan dokumen. Kluster dapat secara dinamis menyesuaikan jumlah instance atau kontainer yang berjalan sesuai dengan beban kerja, memastikan ketersediaan layanan secara optimal. Selain itu, fitur toleransi kesalahan Kubernetes memungkinkan kluster untuk secara otomatis mendeteksi dan mengatasi kegagalan pada salah satu nodenya, sehingga layanan tetap berjalan tanpa terganggu. Manajemen sumber daya yang efisien dari Kubernetes juga dapat membantu mengoptimalkan penggunaan infrastruktur, memastikan bahwa penandatanganan dokumen berlangsung dengan cepat dan efektif.

Teknik Model Analisis

Penelitian ini menggunakan teknik Analisis Performa Empiris melalui pengujian beban dan tekanan (Load and Stress Testing) untuk mengevaluasi kinerja sistem diatas tesbed yang dibuat. Berikut adalah gambaran testbed yang dibuat.



Gambar 4 Model Analisis

A. Metode Pengujian:

- a. Pengujian Kinerja dan Beban: Pengukuran performa dilakukan menggunakan aplikasi Apache JMeter, yang berfungsi untuk mensimulasikan banyak pengguna aplikasi secara bersamaan dan melakukan evaluasi kinerja aplikasi.
- b. Skenario Variasi: Uji coba dilakukan sesuai skenario (1 hingga 475 pengguna).

B. Parameter Kinerja yang Dianalisis: Data-data yang didapatkan dari pengujian dianalisis untuk menentukan kemampuan platform. Parameter-parameter yang menjadi fokus analisis adalah:

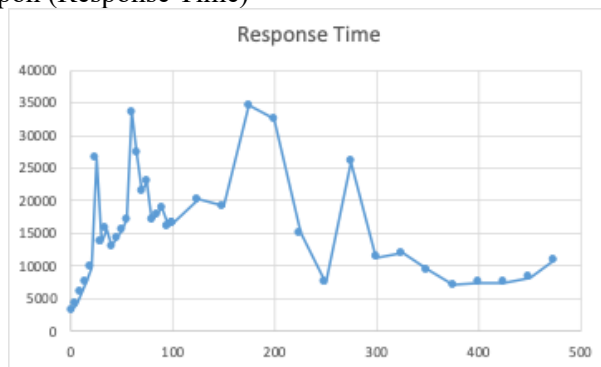
- a. Waktu Respon (Response Time).
- b. Tingkat Error Rate (jumlah error yang didapat).
- c. Throughput.

HASIL DAN PEMBAHASAN

Pengujian dilakukan menggunakan simulasi beban kerja dari 1 hingga 475 pengguna (Number of Client) melalui 37 skenario pengujian menggunakan Apache JMeter, dengan objek uji adalah berkas dokumen sebesar 300KB.

Berikut adalah gambaran hasil pengujian berdasarkan parameter utama yang dianalisis:

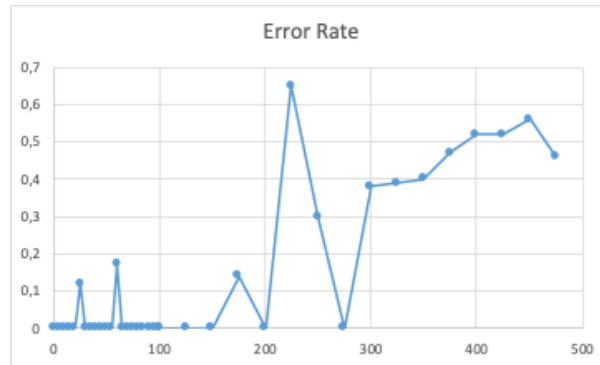
1. Analisis Waktu Respon (Response Time)



Gambar 5 Analisis Waktu Respon

- a. Secara umum, penerapan kluster Kubernetes untuk penandatanganan dokumen secara terdistribusi memiliki waktu respon yang cukup baik untuk jumlah pengguna di bawah 150.
- b. Namun, untuk jumlah pengguna di atas 150, waktu respon menjadi sangat fluktuatif atau tidak stabil. Ketidakstabilan ini diakibatkan oleh mulai terjadinya error pada sejumlah request yang dilakukan oleh pengguna.
- c. Data menunjukkan variasi yang signifikan; misalnya, Response Time melonjak dari 19105,74 ms pada 150 pengguna menjadi 34537,06 ms pada 175 pengguna, meskipun setelah itu nilainya kembali fluktuatif.

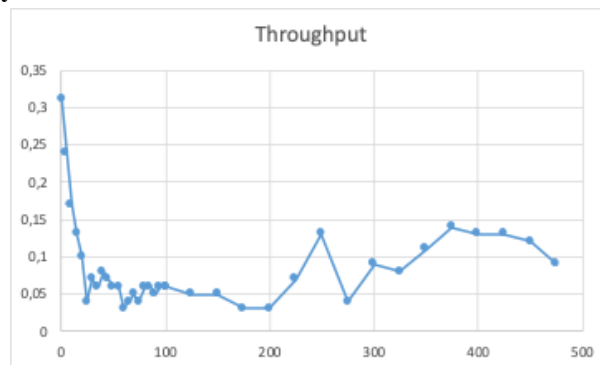
2. Analisis Tingkat Error Rate



Gambar 6 Analisis Tingkat Error

- Klaster Kubernetes yang diusulkan mampu melayani secara stabil untuk pengguna di bawah 150.
- Peningkatan error rate yang signifikan didapatkan setelah jumlah pengguna melebihi 150. Contohnya, error rate adalah 0% hingga 150 pengguna, namun melonjak menjadi 0,14% pada 175 pengguna, dan mencapai puncaknya di 0,65% pada 225 pengguna.
- Setelah 225 pengguna, error rate tetap tinggi, berkisar antara 0,3% hingga 0,56% hingga skenario pengujian terakhir (475 pengguna).

3. Analisis Throughput



Gambar 7 Analisis Throughput

- Data menunjukkan adanya penurunan throughput seiring dengan bertambahnya jumlah pengguna.
- Penurunan throughput tersebut berkorelasi langsung dengan peningkatan penggunaan sumber daya CPU dan Memory pada sistem klaster Kubernetes.

Penelitian ini berhasil mengidentifikasi keterbatasan sumber daya yang menyebabkan penurunan kinerja pada titik tertentu:

- Batas Stabilitas:** Ditemukan bahwa klaster testbed yang dibuat tepat digunakan untuk concurrent user sampai 150 dalam kasus penandatanganan dokumen digital.
- Penyebab Bottleneck:** Berdasarkan log monitoring klaster Kubernetes, klaster testbed mulai menggunakan sumber daya CPU secara penuh ketika jumlah pengguna sudah mencapai angka 175.
- Prioritas Sumber Daya:** Terdapat temuan menarik bahwa pemrosesan penandatanganan dokumen digital lebih memerlukan sumber daya CPU dibandingkan dengan RAM. Hal ini didukung oleh data di mana utilitas penggunaan CPU telah melebihi 100% (mencapai 177%), sementara rata-rata ketersediaan RAM sisa masih cukup banyak (33,3%).

KESIMPULAN

Berdasarkan pengujian yang telah dilakukan, dapat ditarik kesimpulan bahwa arsitektur kluster kubernetes yang dibuat sesuai testbed telah tepat digunakan untuk kasus penandatanganan dokumen secara digital dengan jumlah concurrent user dibawah 150. Perlu adanya penambahan sumber daya berupa core CPU agar dapat melayani penandatanganan dokumen secara digital dengan concurrent user diatas 150. Dalam kasus penandatanganan dokumen secara digital, komputasinya lebih membutuhkan CPU dibandingkan dengan RAM.

UCAPAN TERIMA KASIH

Terima kasih kepada tempat mengabdikan kami di Politeknik Negeri Malang yang sudah mendukung penelitian kami.

REFERENSI

- Agnihotri, J., & Phalnikar, R. (2018). Development of performance testing suite using apache JMeter. In *Advances in Intelligent Systems and Computing*. doi:10.1007/978-981-10-7245-1_32
- Andrews, G. R. (1999). *Foundations of multithreaded, parallel, and distributed programming* (1st ed.). Addison-Wesley..
- Apache. (2014). *Apache JMeter - Apache JMeterTM*. Apache JMeter..
- Boneh, D. (1999). Twenty years of attacks on the RSA cryptosystem.. (Catatan: Detail publikasi asli (jurnal/konferensi) tidak tersedia, hanya judul artikel yang dikutip di Tinjauan Pustaka).
- Di Crescenzo, G., & Persaud, A. (2020). Digital signatures: A survey of law and technology.. (Catatan: Detail publikasi asli (jurnal/konferensi) tidak tersedia, hanya judul artikel yang dikutip di Tinjauan Pustaka).
- Kumar, R., et al. (2019). A comprehensive study on digital signatures and its applications.. (Catatan: Detail publikasi asli (jurnal/konferensi) tidak tersedia, hanya judul artikel yang dikutip di Tinjauan Pustaka).
- Kubernetes Overview. (n.d.). Kubernetes. Retrieved from <https://kubernetes.io/docs/concepts/overview/>.
- Wang, J., & Wu, J. (2019). Research on performance automation testing technology based on JMeter. In *Proceedings - 2019 International Conference on Robots and Intelligent System, ICRIS 2019*.. doi:10.1109/ICRIS.2019.00023