

Analisis *Quality of Service* Protokol MQTT, HTTP, dan CoAP dalam Pengiriman Data ke Thingsboard

I Kadek Yogi Kurniawan *, I Gede Andika **, I Gede Made Yudi Antara ***, I Gusti Made Ngurah Desnanjaya ****, Anak Agung Gde Ekayana *****

Program Studi Sistem Komputer, Fakultas Teknologi dan Informatika, Institut Bisnis dan Teknologi Indonesia (INSTIKI)

* ykurniawan716@gmail.com, ** gdandika@instiki.ac.id, ***yudi.antara@instiki.ac.id,

****ngurah.desnanjaya@instiki.ac.id, *****gungekayana@instiki.ac.id

ABSTRACT

The Internet of Things allows a sensor node to collect data from a place and send it to the Internet of Things platform. In its application, the Internet of Things has several protocols for sending data such as Message Queue Telemetry Transport (MQTT), Hyper Text Transport Protocol (HTTP), and Constrained Application Protocol (CoAP). These three protocols have their own characteristics and differences. For this reason, it is necessary to analyze the performance of these three protocols. This research focuses on analyzing the comparison of the Quality of Service of these three protocols in terms of packet loss, delay, and jitter parameters with the object used is sending temperature data to Thingsboard. The result shows that MQTT excels in sending data with short time intervals and CoAP excels in sending data with long time intervals.

Keyword: sensor node, IoT, Protocol, Thingsboard

1. Pendahuluan

Perkembangan teknologi saat ini membuat seluruh perangkat yang ada disekitar kita dapat terkoneksi dan berkomunikasi satu sama lain hanya dengan memanfaatkan koneksi internet. Adalah teknologi *Internet of Things* (IoT) yang memungkinkan hal tersebut terjadi. Dengan *Internet of Things*, sebuah *node sensor* dapat melakukan pengambilan data dari suatu tempat dan mengirimkannya ke platform *Internet of Things* [1].

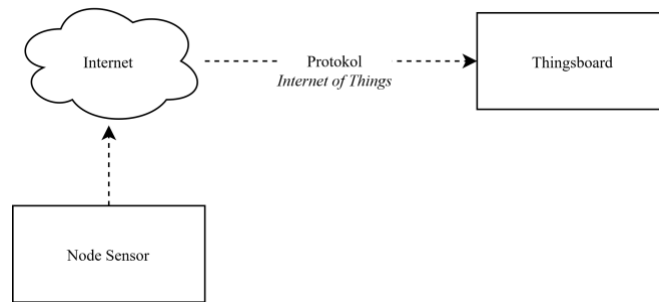
Dalam penerapannya, *Internet of Things* memiliki beberapa protokol dalam melakukan pengiriman data seperti *Message Queue Telemetry Transport* (MQTT), *Hyper Text Transport Protocol* (HTTP) dan *Constrained Application Protocol* (CoAP). Ketiga protokol tersebut memiliki karakteristik dan perbedaannya masing-masing seperti protokol MQTT menggunakan konsep *publish* dan *subscribe* [2] sedangkan protokol HTTP dan CoAP menggunakan konsep *post* dan *request*. Selain itu protokol MQTT dan HTTP berjalan di atas *Transmission Control Protocol* (TCP) sedangkan protokol CoAP beroperasi di atas *User Data Protocol* (UDP) untuk menghindari *congestion control* yang kompleks [3].

Dalam penelitian yang membandingkan protokol *Internet of Things* yang dilakukan oleh Bagas [4] mendapatkan hasil protokol MQTT memiliki keunggulan dibandingkan protokol HTTP dari parameter *latency*. Penelitian selanjutnya yang dilakukan oleh Pratama [5] mendapatkan hasil protokol MQTT jauh lebih baik digunakan dalam jaringan *wireless sensor* dengan keunggulan pada parameter *throughput*, *packet received* dan *power consumption* dibandingkan dengan protokol CoAP yang unggul dalam parameter *delay* saja. Penelitian lainnya yang dilakukan oleh Farid [6] mendapatkan hasil protokol HTTP memiliki hasil yang lebih baik dibandingkan protokol MQTT dari segi parameter *delay*, *jitter* dan *packet loss*. Dari segi parameter *throughput*, protokol MQTT jauh lebih baik dibandingkan protokol HTTP.

Dalam penelitian ini akan dilakukan analisis *Quality of Service* terhadap perbandingan performansi dari segi parameter *packet loss*, *delay* dan *jitter* terhadap protokol MQTT, HTTP dan CoAP. Objek yang digunakan adalah pengiriman data suhu dari *node sensor* ke Thingsboard sebagai platform *Internet of Things*. Pengujian akan menggunakan jaringan publik dengan bantuan aplikasi Wireshark untuk *capture* lalu lintas data.

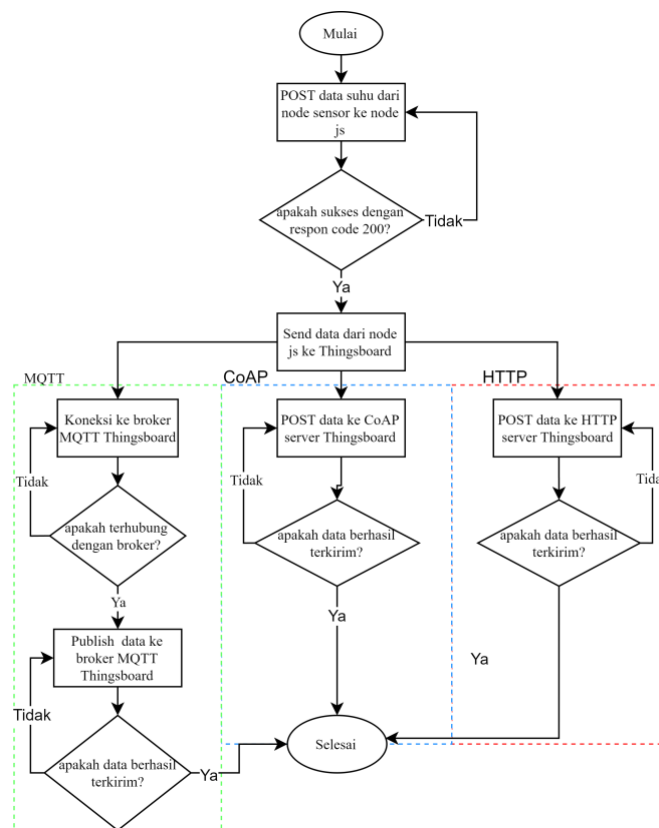
2. Metode Penelitian

Penelitian ini dilakukan dengan melakukan perancangan perangkat keras menggunakan *node sensor* yang terdiri dari ESP32 dan sensor DHT22 dalam mengukur suhu ruangan dan mengirimkannya ke Thingsboard. Arsitektur sistem secara keseluruhan digambarkan dalam diagram blok pada Gambar 1. *Node sensor* akan terhubung dengan jaringan internet untuk mengirimkan data suhu ke Thingsboard dengan menggunakan protokol *Internet of Things*.



Gambar 1. Arsitektur Sistem

Dalam proses pengujian protokol *Internet of Things*, *node sensor* akan mengirimkan data suhu ke *nodejs* yang akan dijalankan pada laptop. *Node sensor* dan *nodejs* akan terhubung dengan jaringan internet yang sama. Setelah data suhu diterima oleh *nodejs*, maka data tersebut akan dikirim ke Thingsboard menggunakan tiga protokol secara bersamaan yaitu MQTT, HTTP dan CoAP. Hal ini bertujuan untuk mendapatkan hasil pengujian yang lebih akurat serta efisiensi waktu pengujian. Dalam pengiriman data dengan protokol MQTT, data tidak langsung dikirim ke server, akan tetapi sistem akan melakukan koneksi ke broker MQTT terlebih dahulu. Jika koneksi berhasil, maka sistem akan melakukan *publish* data ke broker MQTT. Sedangkan untuk protokol HTTP dan CoAP, sistem akan langsung mengirimkan data ke Thingsboard. Diagram alir dari proses pengujian protokol *Internet of Things* digambarkan pada Gambar 2.



Gambar 2. Diagram Alir Pengujian Protokol *Internet of Tings*

Pada Thingsboard, akan dibuat tiga *device profile* dengan konfigurasi *transport type* yang digunakan disesuaikan dengan protokol *Internet of Things* yang akan diuji. *Nodejs* dalam mengirim data suhu akan menggunakan format JSON ke Thingsboard. Data tersebut akan mengarah ke *device profile* yang telah dibuat sebelumnya dengan *endpoint* yang berbeda pada setiap *device profile*. Sehingga data yang dikirim oleh *nodejs* ke web server Thingsboard dapat sesuai dengan protokol *Internet of Things* yang digunakan. Selama proses pengujian, laptop yang menjalankan *nodejs* juga akan menjalankan aplikasi Wireshark untuk *capture* lalu lintas data. Data yang sudah di *capture* kemudian akan diolah menggunakan Microsoft Excel untuk

kemudian dilakukan filter data berdasarkan jenis protokol *Internet of Things* yang akan dianalisa dan menghitung nilai *packet loss*, *delay* dan *jitter*. Sehingga dalam penelitian ini, analisa yang dilakukan murni terhadap protokol *Internet of Things* tanpa melakukan analisa terhadap jaringan secara keseluruhan. Adapun rumus dan kategori dari *packet loss*, *delay* dan *jitter* adalah sebagai berikut.

a. *Packet loss*

$$Packet\ Loss = \frac{(Paket\ dikirim - paket\ diterima) \times 100\%}{Paket\ dikirim} \tag{1}$$

Tabel 1. Kategori *Packet Loss*

Kategori Packet Loss	Packet Loss (%)	Indeks
Sangat Bagus	0-2%	4
Bagus	3-14%	3
Sedang	15-24%	2
Jelek	>25%	1

b. *Delay*

$$Delay = \frac{Total\ delay}{Jumlah\ total\ paket} \tag{2}$$

Tabel 2. Kategori *Delay*

Kategori Delay	Delay (ms)	Indeks
Sangat Bagus	<150ms	4
Bagus	150-300ms	3
Sedang	300-450ms	2
Jelek	>450ms	1

c. *Jitter*

$$Jitter = \frac{Total\ variasi\ delay}{Jumlah\ total\ paket} \tag{3}$$

Tabel 3. Kategori *Jitter*

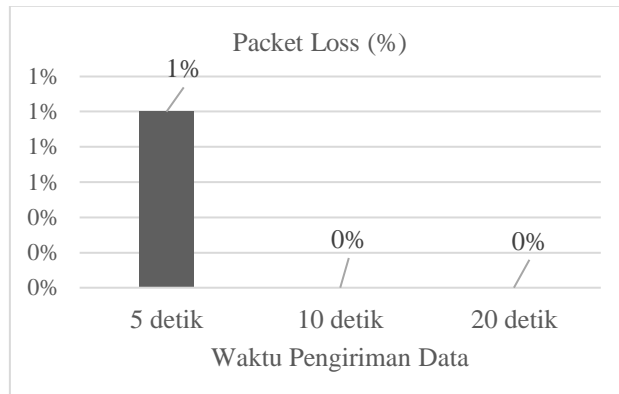
Kategori Jitter	Jitter (ms)	Indeks
Sangat Bagus	0ms	4
Bagus	0-75ms	3
Sedang	75-125ms	2
Jelek	125-225ms	1

3. Hasil dan Pembahasan

Pengujian dilakukan dalam 3 tahap yaitu pengujian dengan pengiriman data setiap 5 detik, 10 detik dan 20 detik dengan melakukan pengiriman data sebanyak 100 kali. Pengujian dengan perbedaan pengiriman data ini dimaksudkan untuk mencari tau apakah interval waktu pengiriman data yang singkat dapat mempengaruhi terjadinya *packet loss* atau tidak dibandingkan dengan pengiriman data dengan interval waktu yang lama. Sedangkan pengiriman data sebanyak 100 kali dimaksudkan untuk lebih memaksimalkan pengujian untuk mencari *packet loss* sehingga diharapkan data yang didapatkan lebih valid.

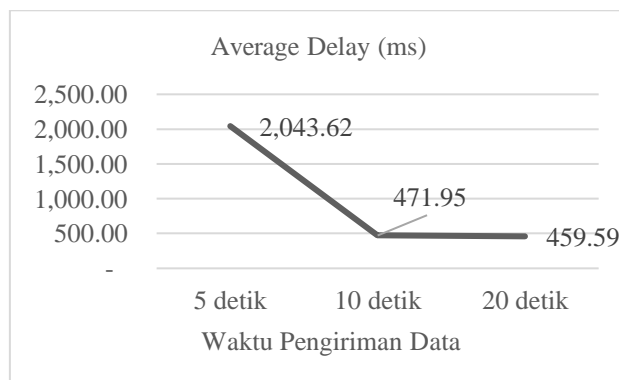
3.1. Hasil Pengujian Protokol MQTT

Gambar 3 menunjukkan hasil pengujian protokol MQTT dengan parameter *packet loss*. Nilai *packet loss* tertinggi terjadi pada pengiriman data setiap 5 detik yang mencapai 1%. Kemudian nilai *packet loss* berhasil mendapatkan nilai 0% pada pengiriman data setiap 10 detik dan 20 detik.



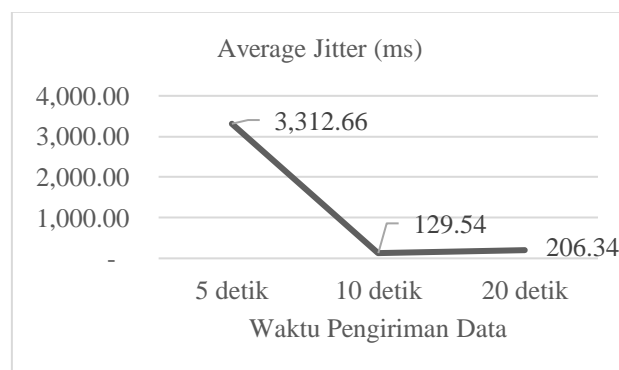
Gambar 3. Grafik *packet loss* protokol MQTT

Gambar 4 Menunjukkan hasil pengujian *Message Queuing Telemetry Transport* dengan parameter *delay*. Nilai rata-rata *delay* terbesar terjadi pada pengiriman data setiap 5 detik yang mencapai 2,043.62ms. Kemudian nilai rata-rata *delay* menurun pada pengiriman data setiap 10 detik sebesar 471.95ms dan 20 detik sebesar 459.59ms.



Gambar 4. Grafik *delay* protokol MQTT

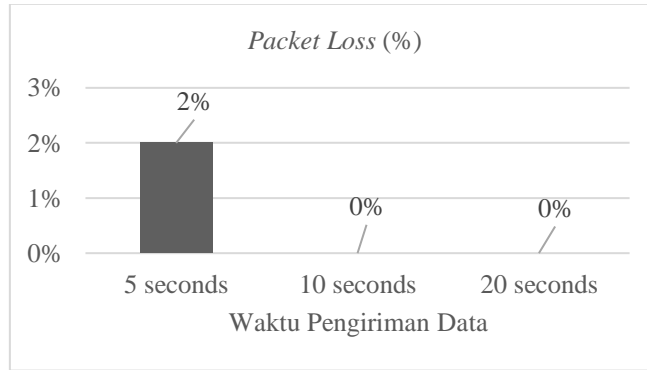
Gambar 5 menunjukkan hasil pengujian protokol MQTT dengan parameter *jitter*. Nilai rata-rata *jitter* terbesar terjadi pada pengiriman data setiap 5 detik sebesar 3,312.66ms. Nilai rata-rata *jitter* kemudian menurun pada pengiriman data setiap 10 detik sebesar 129.54ms dan naik naik kembali pada pengiriman data setiap 20 detik sebesar 206.34ms.



Gambar 5. Grafik *jitter* protokol MQTT

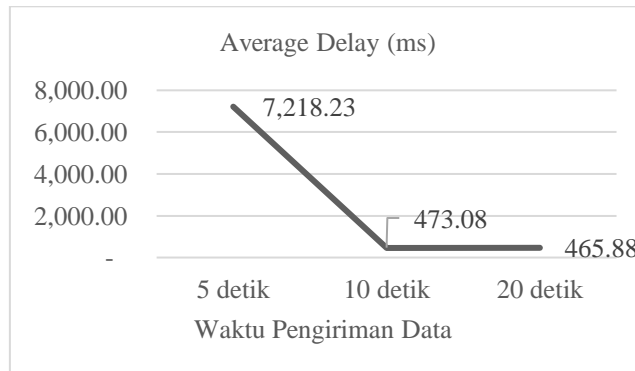
3.2. Hasil Pengujian Protokol HTTP

Gambar 6 menunjukkan hasil pengujian protokol HTTP dengan parameter *packet loss*. Nilai *packet loss* tertinggi terjadi pada pengiriman data setiap 5 detik mencapai 2%. Sedangkan nilai *packet loss* terendah sebesar 0% didapatkan pada pengiriman data setiap 10 detik dan 20 detik.



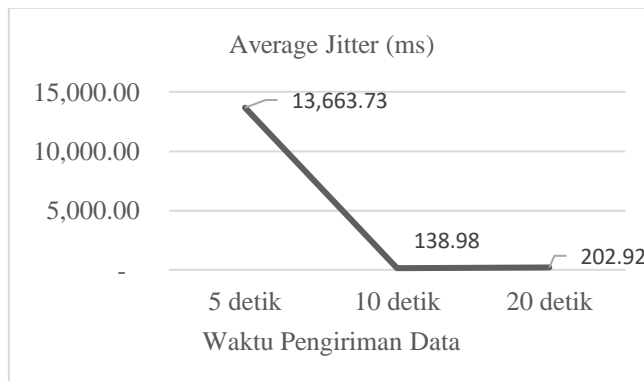
Gambar 6. Grafik *packet loss* protokol HTTP

Gambar 7 Menunjukkan hasil pengujian protokol HTTP dengan parameter *delay*. Nilai rata-rata *delay* tertinggi terjadi pada pengiriman data setiap 5 detik sebesar 7,218.23ms. Kemudian nilai rata-rata *delay* menurun pada pengiriman data setiap 10 detik sebesar 473.08ms dan pengiriman data setiap 20 detik sebesar 465.88ms.



Gambar 7. Grafik *delay* protokol HTTP

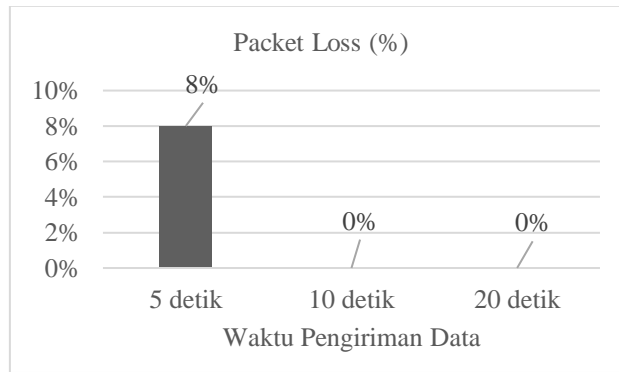
Gambar 8 menunjukkan hasil pengujian *Hypertext Transfer Protokol* dengan parameter *jitter*. Nilai rata-rata *jitter* terbesar terjadi pada pengiriman data setiap 5 detik sebesar 13,663.73ms. Nilai rata-rata *jitter* kemudian menurun pada pengiriman data setiap 10 detik sebesar 138.98ms dan naik kembali pada pengiriman data setiap 20 detik sebesar 202.92ms.



Gambar 8. Grafik *jitter* protokol HTTP

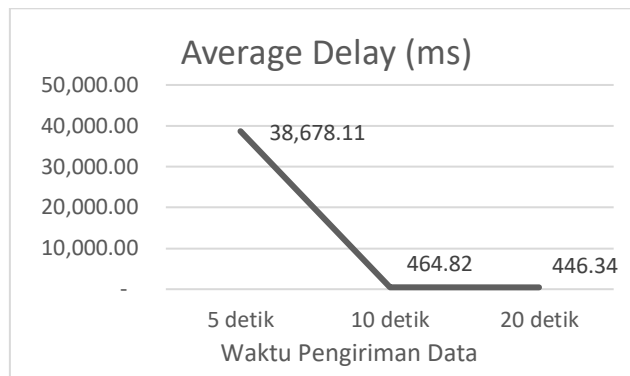
3.3. Hasil Pengujian Protokol CoAP

Gambar 9 menunjukkan hasil pengujian protokol CoAP dengan parameter *packet loss*. Nilai *packet loss* tertinggi terjadi pada pengiriman data setiap 5 detik mencapai 8%. Sedangkan nilai *packet loss* terendah sebesar 0% didapatkan pada pengiriman data setiap 10 detik dan 20 detik.



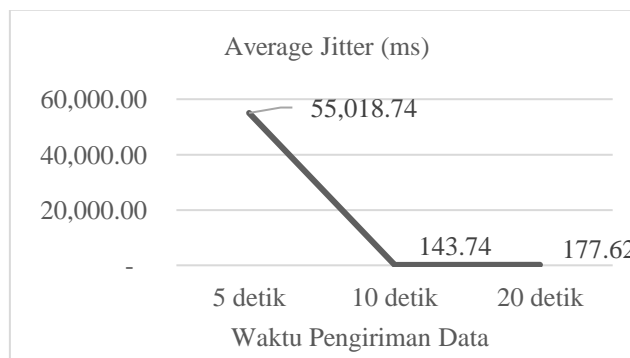
Gambar 9. Grafik *packet loss* protokol CoAP

Gambar 10 Menunjukkan hasil pengujian protokol CoAP dengan parameter *delay*. Nilai rata-rata *delay* tertinggi terjadi pada pengiriman data setiap 5 detik sebesar 38,678.11ms. Kemudian nilai rata-rata *delay* menurun pada pengiriman data setiap 10 detik sebesar 464.82ms dan pengiriman data setiap 20 detik sebesar 446.34ms.



Gambar 10. Grafik *delay* protokol CoAP

Gambar 11 menunjukkan hasil pengujian protokol CoAP dengan parameter *jitter*. Nilai rata-rata *jitter* terbesar terjadi pada pengiriman data setiap 5 detik sebesar 55,018.74ms. Nilai rata-rata *jitter* kemudian menurun pada pengiriman data setiap 10 detik sebesar 143.74ms dan naik kembali pada pengiriman data setiap 20 detik sebesar 177.62ms.



Gambar 11. Grafik *jitter* protokol CoAP

3.3. Perbandingan Pengujian Protokol *Internet of Things*

Tabel 4 menunjukkan perbandingan hasil pengujian *Quality of Service* protokol *Internet of Things* dengan parameter *packet loss*. Nilai *packet loss* tertinggi didapatkan pada pengiriman data dengan interval waktu setiap 5 detik dimana protokol CoAP memiliki nilai *packet loss* tertinggi sebesar 8%, sedangkan protokol HTTP sebesar 2% dan protokol MQTT sebesar 1%. Interval waktu pengiriman data yang lebih lama yakni setiap 10 detik dan 20 detik mendapatkan nilai *packet loss* 0% pada ketiga protokol.

Tabel 4. Perbandingan Parameter *Packet Loss*

waktu pengiriman	<i>Packet loss (%)</i>		
	MQTT	HTTP	CoAP
5 detik	1%	2%	8%
10 detik	0%	0%	0%
20 detik	0%	0%	0%

Tabel 5 menunjukkan hasil pengujian *Quality of Service* protokol *Internet of Things* dengan parameter *delay*. Nilai rata-rata *delay* tertinggi didapatkan pada pengiriman data dengan interval waktu setiap 5 detik pada ketiga protokol. Hal ini disebabkan karena pada pengiriman data setiap 5 detik, seluruh protokol mengalami *packet loss* yang mana data yang dikirim oleh *node sensor* tidak berhasil diterima oleh Thingsboard. Nilai *delay* juga cenderung menurun setiap interval pengiriman data yang semakin lama pada ketiga protokol. Pengiriman data dengan interval waktu yang semakin lama tidak menyebabkan adanya antrian di jaringan dan *Congestion Control* yang dapat dipicu jika data dikirim dengan interval waktu yang singkat sehingga terjadi penundaan lebih lama untuk mengatasi kemungkinan *overload* jaringan. Protokol MQTT memiliki nilai *delay* yang paling rendah saat pengiriman data dengan interval waktu yang singkat dibandingkan dengan protokol HTTP dan protokol CoAP. Namun jika pengiriman data dengan interval waktu yang lebih lama, justru protokol CoAP yang memiliki nilai *delay* paling rendah dibandingkan kedua protokol lainnya.

Tabel 5. Perbandingan Parameter *Delay*

waktu pengiriman	<i>Average Delay (ms)</i>		
	MQTT	HTTP	CoAP
5 detik	2,043.62	7,218.23	13,663.73
10 detik	471.95	473.08	464.82
20 detik	459.59	465.88	446.34

Tabel 5 menunjukkan hasil pengujian *Quality of Service* protokol *Internet of Things* dengan parameter *jitter*. Nilai rata-rata *jitter* tertinggi terjadi pada pengiriman data dengan interval waktu setiap 5 detik pada ketiga protokol dimana protokol CoAP memiliki nilai rata-rata *jitter* yang paling tinggi dibandingkan protokol lainnya. Nilai rata-rata *jitter* juga turun pada pengiriman data dengan interval waktu setiap 10 detik dan naik kembali pada pengiriman data dengan interval waktu setiap 20 detik yang terjadi pada ketiga protokol.

Tabel 6. Perbandingan Parameter *Jitter*

waktu pengiriman	<i>Average Jitter (ms)</i>		
	MQTT	HTTP	CoAP
5 detik	3,312.66	13,663.73	55,018.74
10 detik	129.54	138.98	143.74
20 detik	206.34	202.92	177.62

4. Kesimpulan

Penggunaan protokol MQTT lebih cocok diterapkan jika ingin melakukan pengiriman data ke Thingsboard dengan interval waktu yang singkat karena protokol ini mampu memperkecil terjadinya *packet loss* sehingga menghasilkan nilai *delay* dan *jitter* yang rendah. Sementara penggunaan protokol CoAP lebih cocok diterapkan jika ingin melakukan pengiriman data ke Thingsboard dengan interval waktu yang tidak terlalu singkat karena dari hasil penelitian yang dilakukan, protokol ini memiliki nilai *packet loss* yang sama dibandingkan protokol lainnya dan nilai *delay* dan *jitter* yang lebih rendah dibandingkan protokol MQTT dan HTTP. [7]

Daftar Pustaka

- [1] F. Vinola and Rakhman, "Sistem Monitoring dan Controlling Suhu Ruang Berbasis Internet of Things," *Jurnal Teknik dan Komputer*, vol. 9, no. 2, pp. 117–126, 2020, doi: 10.35793/jtek.v9i2.29698.
- [2] L. Nurfiqin, Z. Sari, and F. D. S. Sumadi, "Analisis Quality of Service (QoS) Protokol MQTT dan HTTP Pada Sistem Smart Metering Arus Listrik," *REPOSITOR*, vol. 3, no. 1, pp. 121–130, 2021, doi: 10.22219/repositor.v3i1.1084.
- [3] Y. Febri Wiryawan, D. P. Kartikasari, and M. Data, "Implementasi Constrained Application Protocol (CoAP) pada Sistem Pengamatan Kelembaban Tanah," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 8, pp. 2480–2487, 2018.
- [4] S. Bagas Bhaskoro, H. Supriyanto, B. B. Aji, and B. Pamungkas, "Perbandingan Performansi Latency Protokol Komunikasi HTTP Dan MQTT Pada Internet of Things," *Jurnal Teknologi Terapan*, vol. 8, no. 2, pp. 82–89, 2022, doi: 10.31884/jtt.v8i2.309.
- [5] S. B. Pratama, R. Munadi, and A. Syauqi, "Analisis Performansi Protokol CoAP Dan MQTT-SN Pada Sistem Smarthome Dengan Cooja Network Simulator," *e-Proceeding of Engineering*, vol. 5, no. 2, pp. 1982–1991, 2018.
- [6] M. Farid Ali Safii, S. Raharjo, and U. Lestari, "Analisis Quality of Service Protokol MQTT Dan HTTP Pada Penerapan Sistem Monitoring Suhu Berbasis Nodemcu (Studi Kasus Ruang Server Kampus 3 IST Akprind Yogyakarta)," *Jurnal JARKOM*, vol. 7, no. 1, pp. 11–19, 2019, doi: 10.22219/repositor.v3i1.1084.
- [7] J. Prayudha, A. Pranata, and A. Al Hafiz, "Implementasi Metode Fuzzy Logic Untuk Sistem Pengukuran Kualitas Udara di Kota Medan Berbasis Internet of Things (IoT)," *JURTEKSI*, vol. IV, no. 2, pp. 141–148, 2018, doi: 10.33330/jurteksi.v4i2.57.
- [8] I. Kassem and A. Sleit, "Elapsed Time of IoT Application Protocol for ECG: A Comparative Study Between CoAP and MQTT," *International Conference on Electrical, Communication and Computer Engineering*, pp. 1–6, 2020, doi: 10.1109/ICECCE49384.2020.9179435.
- [9] G. Hergika, Siswanto, and Sutarti, "Perancangan Internet of Things (IoT) Sebagai Kontrol Infrastruktur Dan Peralatan Toll Pada PT. Astra Infratoll Road," *Jurnal PROSISKO*, vol. 8, no. 2, pp. 86–98, 2021, doi: 10.30656/prosisko.v8i2.3862.
- [10] K. Ali Abdurahman, I. M. Rendy Munadi, and A. S. Indra Irawan, "Perancangan Dan Implementasi Hidroponik Berbasis Internet of Things (IoT) Menggunakan Protokol HTTP," *e-Proceeding of Engineering*, vol. 7, no. 2, pp. 3862–3868, 2020.
- [11] Yuswandari and H. Yuana, "Rancang Bangun Sistem Kendali Jarak Jauh Lampu Menggunakan Thingsboard Berbasis IoT," *JIP (Jurnal Informatika Polinema)*, vol. 7, no. 1, pp. 29–36, 2020, doi: 10.33795/jip.v7i1.437.
- [12] A. F. Pratama, S. Baqaruzi, and A. Muhtar, "Quality of Service Packet Loss Pada Sistem Home Monitoring Water Flow Berbasis Internet of Things," *Jurnal ELECTRON*, vol. 2, no. 1, pp. 10–17, 2021, doi: <https://doi.org/10.33019/electron.v2i1.2369>.
- [13] A. R. Maulana, H. Walidainy, M. Irhamsyah, Fathurrahman, and Akhyar, "Analisis Quality of Service (QoS) Jaringan Internet Pada Website E-Learning Universitas Syiah Kuala Berbasis Wireshark," *KITEKTRO: Jurnal Komputer, Teknologi Informasi, dan Elektro*, vol. 6, no. 2, pp. 27–30, 2021, doi: 10.24815/kitektro.v6i2.22284.
- [14] N. Naik, "Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP," 2017 IEEE International Systems Engineering Symposium (ISSE), Vienna, Austria, 2017, pp. 1–7, doi: 10.1109/SysEng.2017.8088251.
- [15] Hairatunnisa, H. A. Nugroho, and R. Margiono, "Analisis Kinerja Protokol MQTT Dan HTTP Pada Akuisisi Data Magnet Berbasis Internet of Things," *Jurnal Ilmiah Informatika*, vol. 6, no. 2, pp. 71–80, Dec. 2021, doi: 10.35316/jimi.v6i2.1351.