Analisis Perbandingan Performansi MQTT dan HTTP pada Platform IoT Node-Red

Elza Fitria Dwi Permatasari[1], Aji Gautama Putrada, S.T., M.T.[2], Dr. Maman Abdurohman, S.T., M.T.[3]

[1,2,3]Fakultas Informatika, Universitas Telkom, Bandung
[1]elzafitria@students.telkomuniversity.ac.id, [2]ajigps@telkomuniversity.ac.id,
[3]abdurohman@telkomuniversity.ac.id

Abstrak

Aplikasi web saat ini berkembang pesat mulai dari *front-end* hingga *back-end*, dan teknologi *Web* ini mendukung perkembangan teknologi komputasi terdistribusi dimana teknologi ini memungkinkan melakukan proses di banyak mesin, dan hasilnya dimanfaatkan oleh banyak mesin. Salah satu dukungan teknologi *Web* pada komputasi terdistribusi adalah pada pembangunan *Web Service* yang berfungsi sebagai aplikasi transaksi data antar mesin yang terlibat di dalamnya. Konsep *Web Service* muncul untuk menjembatani sistem-sistem informasi yang digunakan oleh masing-masing sumber. Pada prinsip dasarnya, suatu komunikasi data merupakan proses mengirimkan data dari satu komputer ke komputer yang lain untuk terselenggaranya proses pengiriman paket data tersebut. Arsitektur yang digunakan pada tugas akhir ini adalah arsitektur *publish/subscribe* dan *request/reply*. Dalam tugas akhir ini telah diimplementasikan sistem monitoring suhu dan kelembaban udara yang berbasis MQTT dan HTTP yang terkoneksi dengan NodeMCU, dan sensor suhu DHT22 yang berfokus pada perbandingan performansi kedua protokol yang digunakan. Hasil dari analisis pada tugas akhir ini, yaitu rangakain alat NodeMCU dan Sensor DHT22 bekerja dengan baik dan mendapatkan data suhu ruangan. Parameter performansi *throughput* terbesar didapatkan pada protokol MQTT dengan nilai 110000 bits/s. Parameter performansi *delay* dan *packet loss* terkecil didapatkan pada protokol MQTT dengan nilai 0,00042 % *packet loss* dan 0,0086902 untuk *delay*. Kinerja sistem monitoring udara lebih baik menggunakan protokol MQTT dibandingkan dengan HTTP dilihat dari nilai performansi seperti *throughput*, *delay*, dan *packet loss*.

Kata kunci: MQTT, HTTP, NodeMCU, DHT22, *Web Service*

Abstract

Web applications are currently developing rapidly from the front-end to the back-end, and this Web technology supports the development of distributed computing technology where this technology allows processes in many machines, and the results are used by many machines. One of the support of Web technology in distributed computing is the development of a Web Service that functions as a data transaction application between machines involved in it. The concept of Web Service appears to bridge the information systems used by each source. Basically, a data communication is a process of sending data from one computer to another for the delivery of the data packet. The architecture used in this thesis is the publish / subscribe and request / reply architecture. In this final project a MQTT and HTTP based temperature and humidity monitoring system has been implemented that is connected to NodeMCU, and a DHT22 temperature sensor that focuses on comparing the performance of the two protocols used. The results of the analysis in this thesis, namely the range of NodeMCU tools and DHT22 sensors work well and get room temperature data. The biggest throughput performance parameter is found in the MQTT protocol with a value of 110000 bits / s. The smallest delay and packet loss performance parameters are obtained in the MQTT protocol with a value of 0,00042% packet loss and 0.0086902 for delay. The performance of air monitoring systems is better using the MQTT protocol compared to HTTP in terms of performance values such as throughput, delay, and packet loss.

Keywords: MQTT, HTTP, NodeMCU, DHT22, Web Service

1. Pendahuluan

Latar Belakang

Aplikasi web saat ini berkembang pesat mulai dari *front-end* hingga *back-end*, dan teknologi *Web* ini mendukung perkembangan teknologi komputasi terdistribusi dimana teknologi ini memungkinkan melakukan proses di banyak mesin, dan hasilnya dimanfaatkan oleh banyak mesin. Salah satu dukungan teknologi *Web* pada komputasi terdistribusi adalah pada pembangunan *Web Service* yang berfungsi sebagai aplikasi transaksi data antar mesin yang terlibat di dalamnya. Konsep *Web Service* muncul untuk menjembatani sistem-sistem informasi yang digunakan oleh masing-masing sumber.

Pada prinsip dasarnya, suatu komunikasi data merupakan proses mengirimkan data dari satu komputer ke komputer yang lain untuk terselenggaranya proses pengiriman paket data tersebut. Untuk komunikasi antar protokol IoT dapat menggunakan arsitektur *Publish/Subscribe* atau *request/reply*.

Arsitektur *publish/subscribe* adalah salah satu arsitektur yang digunakan untuk komunikasi antar perangkat IoT. Contoh dari arsitektur *software* yang menggunakan arsitektur *publish/subscribe* adalah MQTT dan HTTP. Arsitektur *publish/subscribe* yang menggunakan broker adalah MQTT, sedangkan arsitektur *publish/subscribe* tidak menggunakan broker adalah HTTP.

Dalam tugas akhir ini telah dilakukan perbandingan dua arsitektur komunikasi perangkat IoT yang diimplementasikan dengan perangkat IoT menggunakan MQTT dan HTTP yang terkoneksi dengan NodeMCU, dan sensor suhu DHT22 yang berfokus pada perbandingan performansi kedua arsitektur komunikasi yang digunakan.

Batasan Masalah

Dalam pengerjaan Tugas Akhir ini, penulis membatasi hal-hal yang berhubungan dengan tugas akhir yang penulis buat. Berikut Batasan masalah dalam tugas akhir ini, yaitu:

1. Node yang digunakan berupa 7 NodeMCU dan 7 Sensor DHT22.
2. Parameter performansi yang diukur yaitu *throughput*, *delay*, dan *packet loss*.
3. Analisis parameter performansi dilakukan dengan Wireshark.
4. Waktu yang digunakan untuk pengujian selama 90 Menit.

Tujuan

Tujuan dari pengerjaan Tugas Akhir ini adalah untuk membandingkan MQTT dengan HTTP dan menganalisis performansi parameter *throughput*, *delay*, dan *packet loss* dari kedua protokol.

Tabel 1. Keterkaitan antara tujuan, pengujian, dan kesimpulan

Tujuan	Pengujian	Kesimpulan
Mengimplementasikan node untuk MQTT dan HTTP.	Mengambil data suhu ruangan menggunakan NodeMCU dan Sensor DHT22. Untuk menampilkan data pada arsitektur komunikasi MQTT dan HTTP menggunakan platform *Node-red*.	Alat dapat terintegrasi dengan sistem. Sistem dapat mengeluarkan data suhu udara dan kelembaban udara secara *wireless*.
Melakukan analisis performansi *throughput*, *delay*, dan *packet loss* pada arsitektur komunikasi MQTT dan HTTP.	Mengambil data suhu ruangan yang didapat dari hasil *running* implementasi menggunakan arsitektur komunikasi MQTT dan HTTP dan dianalisis parameter performansi *throughput*, *delay*, dan *packet loss* menggunakan *Wireshark*.	Parameter performansi telah didapat dari hasil *running* implementasi alat dengan menggunakan masing-masing protokol.
Melakukan perbandingan arsitektur komunikasi MQTT dengan HTTP pada monitoring udara dan mendapatkan kesimpulan dari analisis tersebut.	Melakukan perbandingan dari hasil parameter performansi yang didapat, dan menganalisis hasil tersebut.	Kinerja MQTT lebih baik dibandingan dengan HTTP.

Organisasi Tulisan

Pada bagian 2 akan dijelaskan penelitian sebelumnya serta landasan teori yang terkait dengan penelitian. Pada bagian 3 dijelaskan proses sistem yang dibangun. Lalu, pada bab 4 dijelaskan mengenai pengujian performansi dari data suhu ruangan yang didapat dari hasil *running* pada *Node-red* dan dianalisis parameter performansi *throughput*, *delay*, dan *packet loss* menggunakan *Wireshark*. Terakhir pada bab 5 terdapat kesimpulan dan saran pada penelitian ini.

2. Studi Terkait

Dengan pesatnya penggunaan sensor berbasis *Internet of Things* (IoT), *Wireless Sensor Network* (WSN) dan studi tentang *Internet of Things* (IoT) dan *Wireless Sensor Network* (WSN) dibutuhkan juga protokol *IoT*

untuk komukasi. Tujuan dilakukannya penelitian "*Messaging Technologies for the Industrial Internet and the Internet of Things"* adalah membandingkan antara beberapa protokol *IoT* sehingga didapatkan protokol yang mendukung skenario yang dibutuhkan seperti misalnya komunikasi *Inter* dan *Intra Device*, komunikasi *Device to Cloud*, dan komunikasi *Inter Data Center*, khususnya untuk performansi dan toleransi kesalahan pada setiap pengiriman data [2].

Pada jurnal [6] [7] dilakukan skenario eksperimental yang telah dikembangkan menggunakan perangkat sensor, Arduino Uno, Raspberry Pi 3, layanan web Amazon, broker MQTT dan platform cloud pribadi. Analisis teoritis menunjukkan bahwa terjadinya penundaan karena meningkatnya jumlah pakey dan muatan pesan telah berkurang dibandingkan dengan pendekatan yang ada.

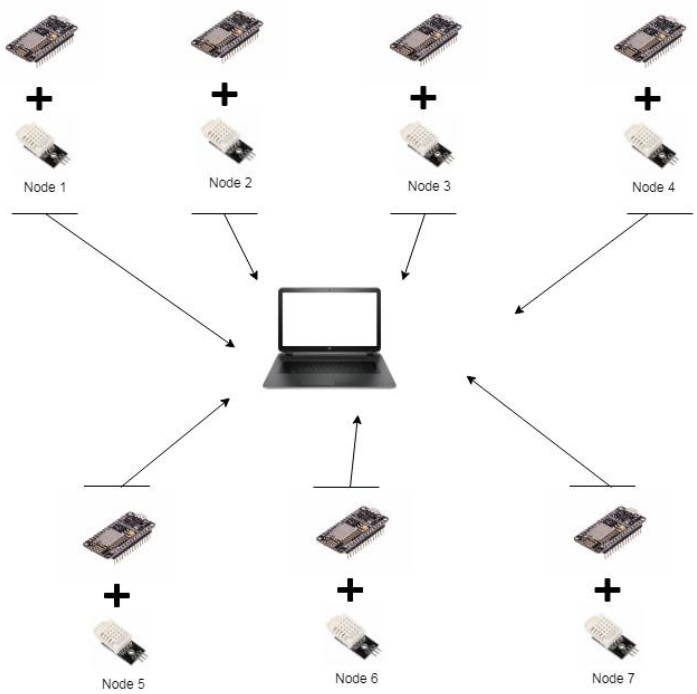
Penulis juga mengutip referensi dari jurnal yang disusun oleh Konstantinos Karamitsios dan Theofanis Orphanoudakis dalam tulisannya yang berjudul "*Efficient IoT data aggregation for connected health applications"* pada tahun 2017 [10]. Pada paper tersebut berfokus pada eksploitasi platform IoT untuk pemantauan status perawatan kesehatan jarak jauh. Dan juga pada paper tersebut menaksir tumpukan protokol IoT yang berbeda untuk agregasi data sensor. Dilakukan evaluasi IEEE 11073 transportasi data media melalui koneksi berbasis TCP dan UDP yang memgeksploitasi protokol CoAP dan MQTT yang diusulkan untuk enkapsulasi data IoT dan interkoneksi *server client*.

Dari penelitian tersebut, penulis melakukan implementasi perbandingan antara dua protokol IoT yaitu MQTT dan HTTP. Kemudian dengan menggunakan *Wireshark* didapatkan hasil perbandingan antara keduanya.

3. Sistem yang Dibangun

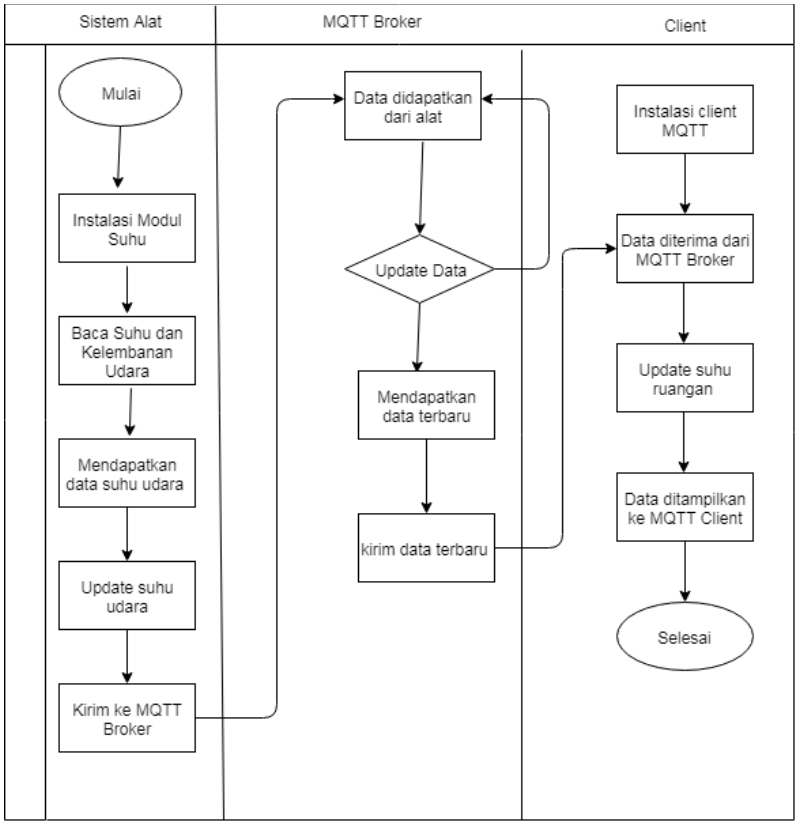
Gambaran Umum Sistem

Pada tugas akhir ini dirancang dan dibangun suatu sistem monitoring udara untuk mendapatkan hasil perbandingan dari dua protokol IoT. Gambaran sistem monitoring udara yaitu terdapat 7 node yang berisikan NodeMCU dan Sensor DHT22 disebar dalam beberapa ruang. Diperoleh data suhu dan kelembaban ruangan dari 7 node tersebut. Data ditampilkan dalam satu platform yang sama namun sesuai dengan kebutuhan protokol yang digunakan. Kedua arsitektur komunikasi menggunakan platform *Node-red*. Ketika masing-masing protokol dijalankan, dilakukan juga pengolahan data menggunakan Wireshark untuk mengetahui hasil *throughput*, *delay*, dan *packetloss*. Gambar 3.1 berikut ini adalah gambaran dari alat yang dibangun.

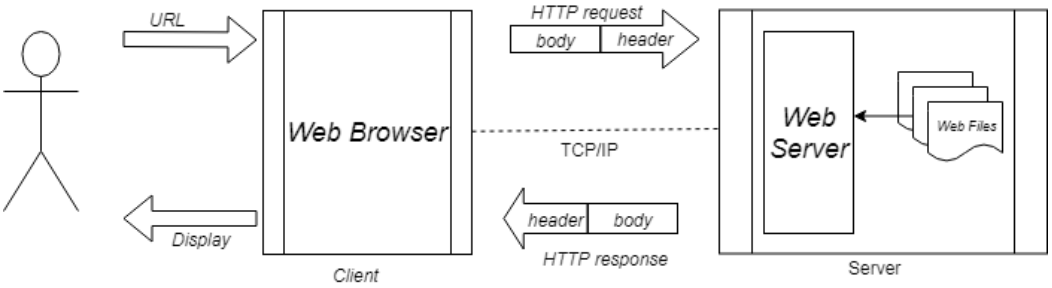


Gambar 3. 1 Gambaran Umum Sistem

Pada gambar 3.1 dijelaskan bahwa ketujuh node yang sudah tersambung dengan program, akan ditampilkan pada platform sesuai dengan protokolnya. Gambar 3.2 dan Gambar 3.3 berikut menunjukkan gambaran alur kerja sistem untuk menampilkan data pada masing-masing arsitektur komunikasi.



Gambar 3. 2 Alur Kerja MQTT



Gambar 3. 3 Alur Kerja HTTP

Protokol MQTT (***Message Queue Telemetry Transport)***

MQTT dirancang sebagai protokol transport yang ringan, terbuka dan sederhana, agar mudah diimplementasikan [3]. Karakteristik ini membuat MQTT ideal digunakan dalam lingkungan jaringan yang terbatas, seperti *bandwidthit* yang rendah dan memori yang terbatas pada suatu *embedded*. Protokol transport yang digunakan MQTT adalah TCP/IP. Sedangkan koneksi komunikasi data untuk layanan TCP/IP dapat digunakan protokol seperti TLS dan *WebSocket*.

Hypertext Transfer Protocol **(HTTP)**

HTTP sebagian besar merupakan protokol pengiriman pesan web, yang pada awalnya dikembangkan oleh Tim Berners-Lee. HTTP mendukung permintaan/respons RESTful arsitektur Web. HTTP adalah protokol berbasis teks dan memang demikian tidak menentukan ukuran header dan payload pesan, melainkan tergantung pada server web atau teknologi pemrograman.

Skenario Pengujian

Pengujian *Scalability*

Secara umum scalability adalah kemampuan suatu sistem, jaringan, atau proses untuk menangani jumlah penambahan beban yang diberikan atau potensinya untuk ditingkatkan guna menangani pemanbahan beban tersebut. Adapun spesifikasi dari perangkat keras yang digunakan adalah sebagai berikut:

a. Processor : Intel® Core™ i3
b. vCPU : 2.30 GHz (4 CPUs), ~ 2.3 GHz
c. RAM : 4096 MB
d. Storage : 500 GB
e. OS : Windows 7 Ultimate 64-bit

Pengujian *Throughput*

Throughput [7] merupakan kecepatan rata-rata data yang diterima oleh suatu-suatu node. Berikut rumus untuk menghitung throughput menggunakan Wireshark.

$$Throughput = \frac{Jumlah\ data\ yang\ dikirim}{Waktu\ pengiriman\ data}$$

Pengujian *Delay*

Delay merupakan waktu tunda saat paket yang diakibatkan oleh proses transmisi dari satu titik (node) menuju titik (node) lain yang menjadi tujuannya. *Delay* diperoleh dari selisih waktu pengiriman antara satu paket dengan paket yang lainnya dalam satuan detik. Berikut rumus untuk menghitung *delay* menggunakan Wireshark
.

$$Delay = \frac{Timespan}{Packet}$$

Pengujian *Packet Loss*

Packet loss merupakan banyaknya paket yang hilang pada suatu jaringan paket yang disebabkan oleh tabrakan (*collision*), penuhnya kapasitas jaringan, dan penurunan paket yang disebabkan oleh habisnya TTL (*Time To Live*) paket. Berikut rumus untuk menghitung *packet loss* menggunakan Wireshark.

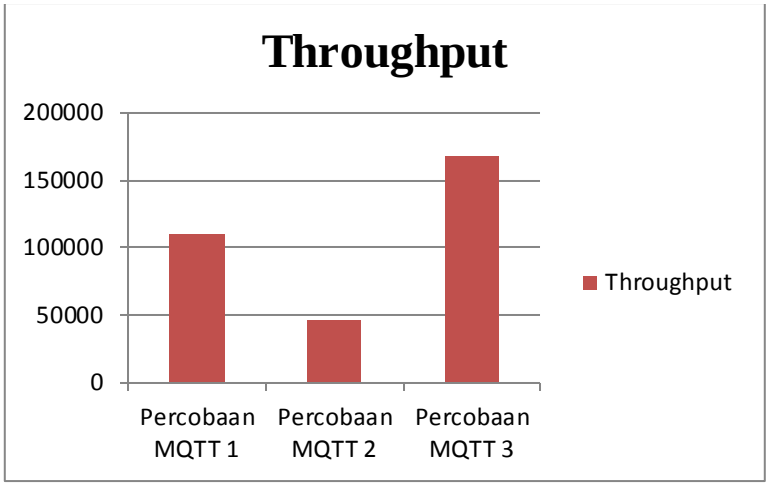
$$Packet\ Loss = (\frac{\text{Paket data yang dikirim} - \text{paket data yang diterima}}{Paket\ data\ yang\ dikirim})\ x\ \ 100\%$$

Lalu, setelah dilakukan running node dan pengujian, dilakukan analisis parameter performansi dan dilakukan perbandingan antara kedua protokol dari hasil analisis parameter performansi. Dari analisis tersebut dapat diketahui manakah protokol yang lebih efektif untuk sistem monitoring udara.

4. Evaluasi

Pengujian *Throughput* MQTT

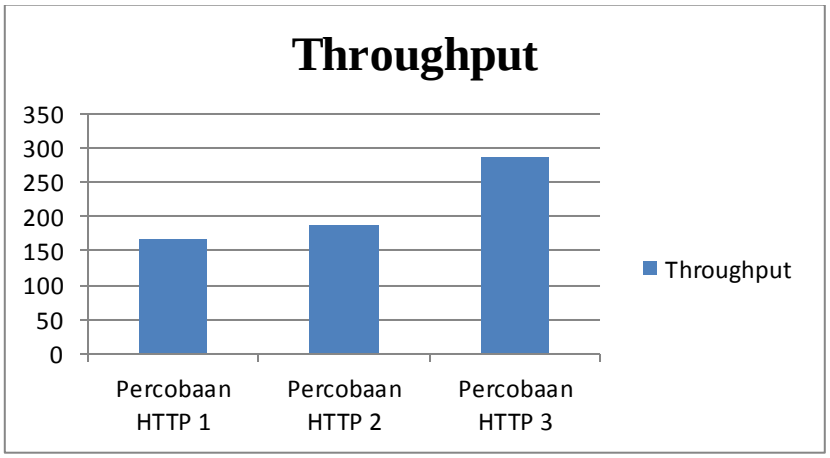
Pengujian dan analisis parameter *throughput* MQTTQ dilakukan untuk mengetahui seberapa besar bandwidth yang dibutuhkan oleh MQTT. Ditunjukkan pada gambar 4.1.



Gambar 4. 1 Throughput MQTT

Pengujian *Throughput* HTTP

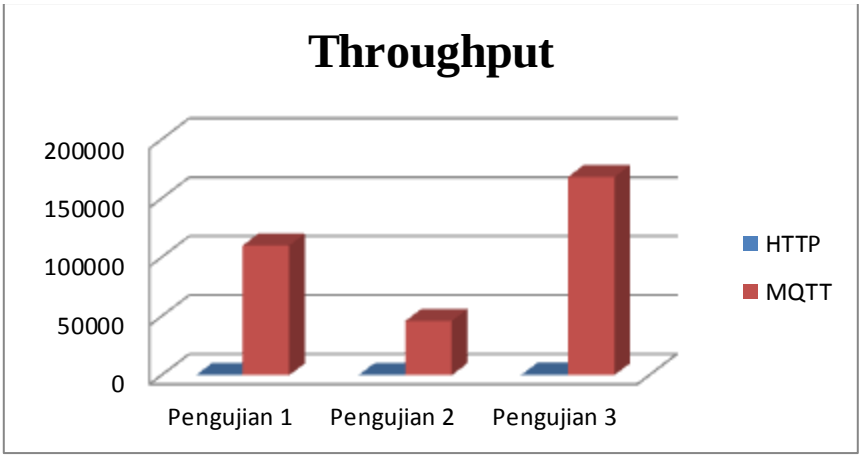
Pengujian dan analisis parameter *throughput* HTTP dilakukan untuk mengetahui seberapa besar kecepatan rata-rata yang diterima oleh suatu node dalam selang waktu pengamatan tertentu. *Throughput* merupakan *bandwidth* actual saat ini juga dimana kita sedang melakukan koneksi. Satuan yang dimilikinya sama dengan bandwidth yaitu bps. Ditunjukkan pada gambar 4.2.



Gambar 4. 2 *Throughput* HTTP

Perbandingan Pengujian dan Analisis *Throughput* MQTT dan HTTP

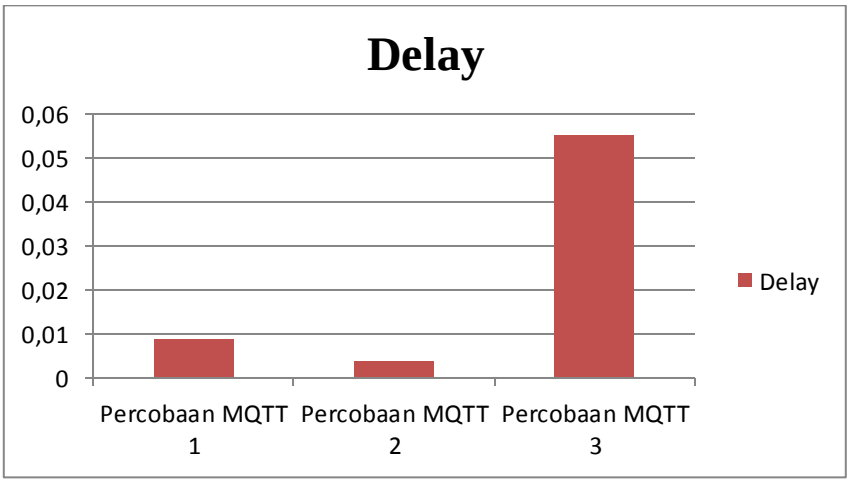
Berdasarkan gambar 4.3 dapat ditarik kesimpulan bahwa *throughput* yang dihasilkan oleh MQTT lebih baik jika dibandingkan dengan *throughput* yang dihasilkan oleh HTTP berdasarkan pengaruh waktu penanganan sistem dimana *throughput* pengujian 1 MQTT pada sistem monitoring udara adalah sebesar 110000 bits/s, pengujian 2 sebesar 46000 bits/s, dan pengujian 3 sebesar 168000 bits/s. Hal ini menggambarkan bahwa kecepatan pengiriman data MQTT lebih cepat dibandingkan dengan HTTP.



Gambar 4. 3 Perbandingan *Throughput* MQTT dan HTTP

Pengujian *Delay* MQTT

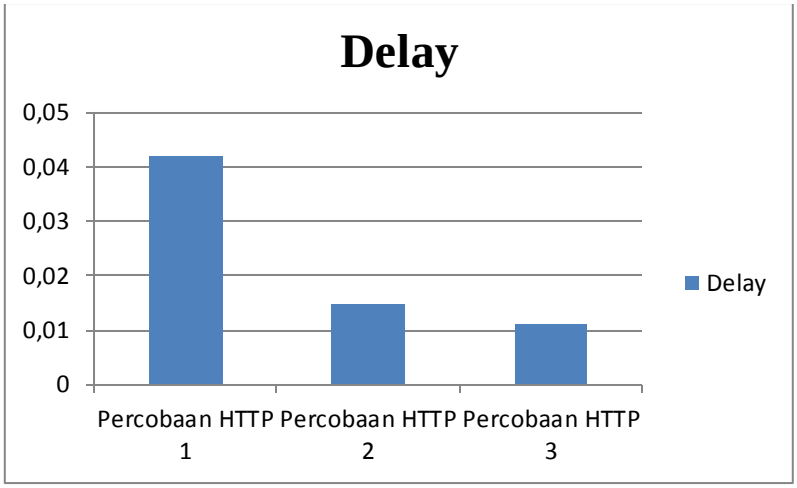
Menentukan waktu yang dibutuhkan *packet* yang dikirim oleh pengirim dan sampai di penerima dikirimkan kembali *packet* baru oleh si pengirim. Pada gambar 4.4 ditunjukkan nilai *delay* pada MQTT.



Gambar 4. 4 *Delay* MQTT

Pengujian *Delay* HTTP

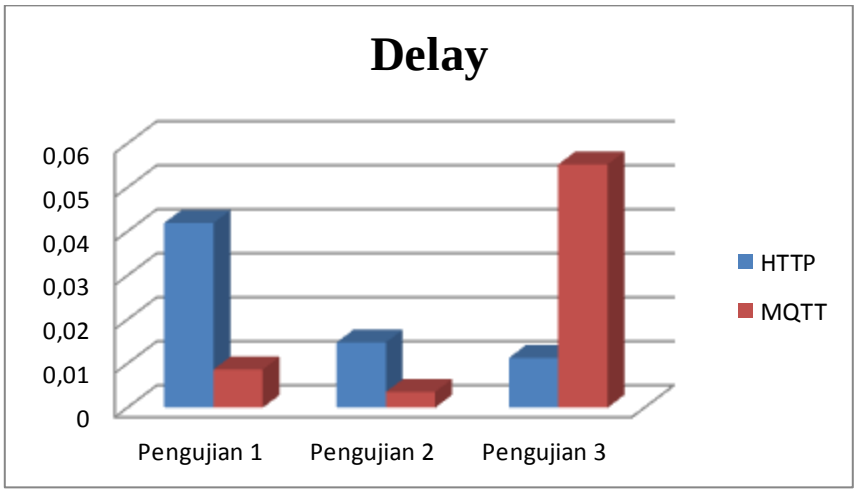
Pengujian *delay* dalam penelitian tugas akhir ini adalah melakukan perhitungan terhadap rata-rata jeda antara pengiriman satu paket dengan paket selanjutnya yang dilakukan oleh sensor DHT22. Ditunjukkan pada gambar 4.5 *delay* yang didapat pada HTTP.



Gambar 4. 5 *Delay* HTTP

Perbandingan Pengujian dan Analisis *Delay* MQTT dan HTTP

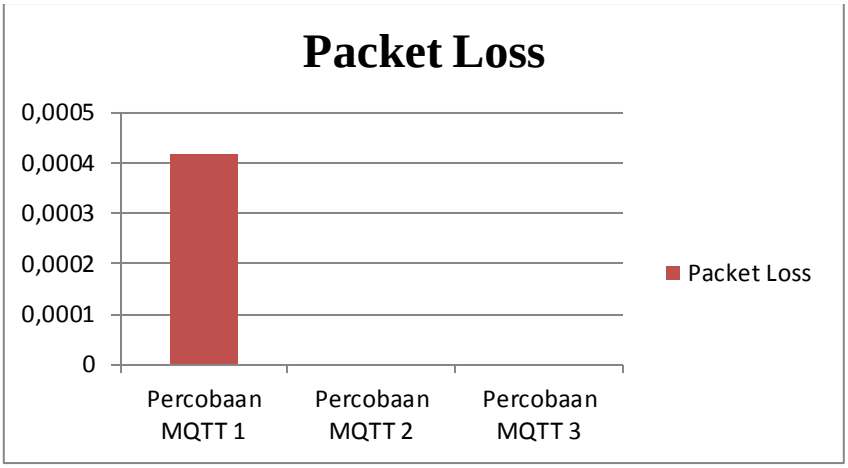
Pengujian delay untuk masing-masing protokol yang terkecil pada saat penggunaan protokol MQTT. Dengan waktu *delay* pada MQTT pengujian pertama sebesar 0,0086902 second (s) dengan banyak 714220 packet, sementara perngujian kedua sebesar 0,003576 s banyak packet yang diproses 1950272 packet, dan pada pengujian ketiga sebesar 0,055249709 s dengan banyak packet yang diproses 12322816 packet.



Gambar 4. 6 Perbandingan *Delay* MQTT dan HTTP

Pengujian *Packet Loss* MQTT

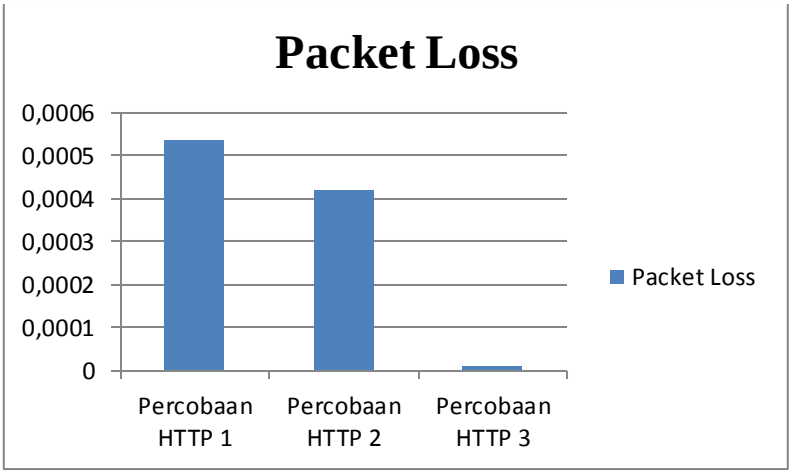
Packet loss merupakan jumlah data yang hilang pada lalu lintas jaringan diakibatkan oleh tabrakan (*collision*), kapasitas jaringan, dan penurunan *packet* dikarenakan kehabisan waktu tunggu *packet*. Dalam menentukan *packet loss* dengan aplikasi *wireshark* menggunakan filter *"tcp.analysis.lost_segment&&mqtt"* untuk menentukan jumlah packet yang hilang dengan protokol MQTT. Ditunjukkan pada gambar 4.7 *packet loss* yang didapat pada protokol MQTT.



Gambar 4. 7 *Packet Loss* MQTT

Pengujian *Packet Loss* HTTP

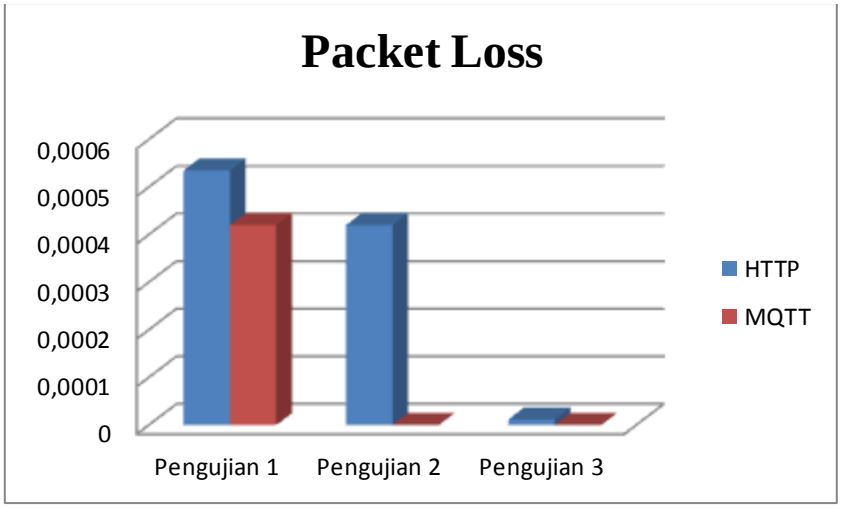
Untuk menentukan *packet loss* dengan aplikasi *wireshark* menggunakan filter *"tcp.analysis.lost_segment&&http"* untuk menentukan jumlah *packet* yang hilang pada HTTP. Ditunjukkan pada gambar 4.8 *packet loss* yang didapat pada HTTP.



Gambar 4. 8 *Packet Loss* HTTP

Perbandingan Pengujian dan Analisis *Packet Loss* MQTT dan HTTP

Berdasarkan gambar 4.9 dapat ditarik kesimpulan bahwa *packet loss* terkecil yang dialami sistem monitoring udara dengan menggunakan MQTT sebagai media komunikasi data. *Packet loss* yang didapat pada MQTT pengujian 1 sebesar 0,00042 % sedangkan pada HTTP pengujian 1 sebesar 0,0001 %. *Packet loss* dapat terjadi dikarenakan kepadatan dalam jaringan, dari hasil implementasi tersebut *packet* yang berada di jaringan saat penerapan MQTT adalah jumlah *packet* terkecil yang ditangani sistem selama implementasi, dari gambar di bawah dapat dilihat *packet loss* MQTT terkecil dari *packet loss* HTTP.



Gambar 4. 9 Perbandingan *Packet Loss* MQTT dan HTTP

5. Kesimpulan

Berdasarkan hasil analisis pengujian perbandingan dua protokol pada sistem monitoring udara, dapat ditarik kesimpulan, yaitu:

1. Rangakain alat NodeMCU dan Sensor DHT22 bekerja dengan baik dan mendapatkan data suhu ruangan.
2. Parameter performansi *throughput* terbesar didapatkan pada MQTT. Parameter performansi *delay* dan *packet loss* terkecil didapatkan pada MQTT.
3. Kinerja sistem monitoring udara lebih baik menggunakan MQTT dibandingkan dengan HTTP dilihat dari nilai performansi seperti *throughput*, *delay*, dan *packet loss*.

6. Saran

Berdasarkan hasil penelitian yang telah dilakukan dalam tugas akhir ini, terdapat beberapa saran untuk perbandingan dua protokol ini pada penelitian selanjutnya, yaitu:

1. Diharapkan adanya sensor yang lebih banyak agar dihasilkan tingkat akurasi yang lebih tepat.
2. Diharapkan adanya nilai parameter selain *throughput*, *delay*, dan *packet loss*.
3. Menambahkan *packaging* pada tiap node-node tersebut.

Daftar Pustaka

[1] M. Firza Pahlevi, Julio Risky Sybiro Coubat Alifian Odi Mahendra, "Sistem Monitoring Suhu dan Kelembaban Ruangan Berbasis Web ," Sekolah Tinggi Teknik PLN , Jakarta , Course Report 2018.

[2] Andrew (PrismTech) Foster, *Messaging Technologies for the Industrial Internet and the Internet of Things Whitepaper*., 2015.

[3] S. Bocchino, P. Pagano, G. Pellerano, M. Petracca A. Azzara, "Middleware Solutions in WSN: The IoT oriented approach in the ICSI Project," in *21st Int. Conf. Software, Telecommun, Comput. Netowork, SoftCOM 2013*, 2013.

[4] Oktoverano Lengkong Jacquline Waworundeng, "Sistem Monitoring dan Notifikasi Kualitas Udara dalam Ruangan dengan Platform IoT," *Cogito Smart Journal*, pp. 94-103, 2018.

[5] Lutfi Aldi Nugroho, "ANALISIS PERBANDINGAN ANTARA PROTOKOL MESSAGE QUEUE TELEMTRY TRANSPORT (MQTT) DAN DATA DISTRIBUTION SERVICE (DDS) ," Telkom University, Bandung, Bachelor Thesis 2017.

[6] Konstantinos Karamitsios dan Theofanis Orphanoudakis, "Efficient IoT data aggregation for connected health applications," *IEEE Symposium on Computers and Communications (ISCC)*, 2017.

[7] Debashis De, Rajkumar Buyya Deepsubhra Guha Roy Bipasha Mahato, "Application-aware end-to-end delay and message loss estimation in Internet of Things (IoT) - MQTT-SN protocol," *Future Generation Computer Systems* , pp. 300-316, 2018.

[8] Feridi. (2019, January) Mengenal RESTful Web Services. [Online].

https://www.codepolitan.com/mengenal-restful-web-services

[9] Angga Nugraha, "Analisis Kualitas Jaringan VSAT Ditinjau dari Delay, Throughput, dan Packet Loss," Institut Pertanian Bogor, Bogor, Bachelor Thesis 2015.

[10] (2019, July) Wikipedia. [Online]. https://en.wikipedia.org/wiki/Node-RED