

Simulasi Hybrid MP-QUIC dalam Internet-of-Things Pertanian Pintar

Muhammad Hafidh Firmansyah¹, Wahyu Kurnia Dewanto²,

hafidh@polije.ac.id¹, wahyu@polije.ac.id²

Politeknik Negeri Jember

Teknologi Informasi

Jl. Mastrip, Krajan Timur, Sumbersari, Kec. Sumbersari, Kabupaten Jember, Jawa Timur 68121

ARTICLE INFO

Article history:

Received January 12, 2024

Received in form 10 March 2024

Accepted 2 April 2024

Available online 1 Juli 2024

ABSTRACT

At this moment, IoT technology was implemented in a broader area. The development of IoT technology can help human life become more manageable. IoT systems can help human activity through the natural time control system provided before. But there are some drawbacks when implementing the IoT system. One of them is the transmission process in the networking area. It's happened because of weather, and other natural problems. Sometimes, IoT transmission can't reach on time. It can make errors reading data on the user side. We proposed a transmission system, namely a Hybrid MP-QUIC transmission protocol, for helping in sending the data. Using this transmission protocol can reduce the response time between client and server.

Keywords: IoT, QUIC, MP-QUIC, Transmission Data.

1. Pendahuluan

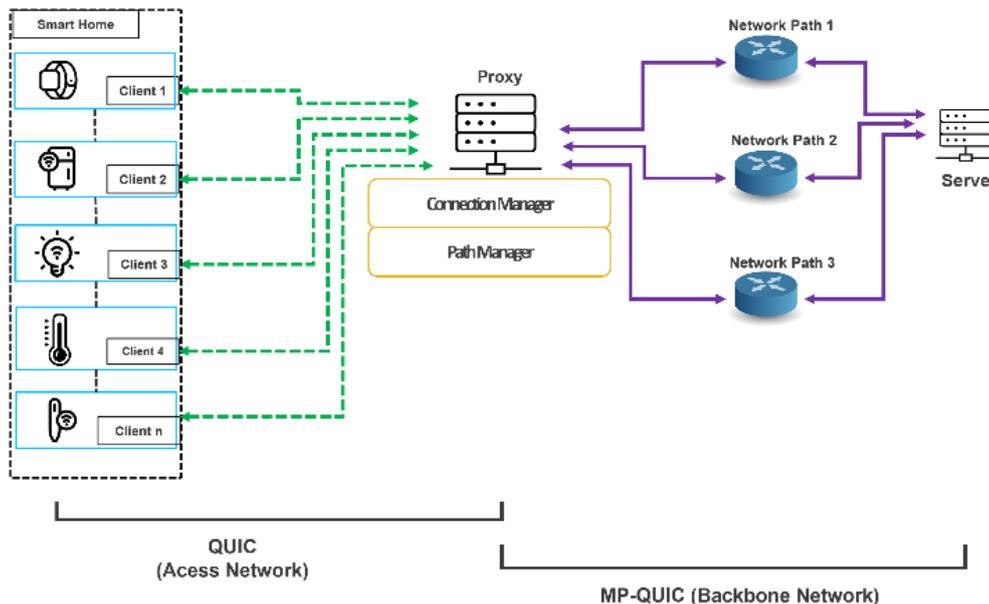
IoT (Prof.Ms.P.V.Dudhe et al., 2017) (Hejazi et al., 2018; Jin & Kim, 2018) banyak digunakan dalam berbagai bidang. Salah satunya adalah bidang pertanian, dengan sistem ekosistem IoT yang besar, menjadi tantangan besar untuk menerapkan teknologi IoT dalam mengirimkan data dalam skenario waktu nyata. Biasanya, IoT menggunakan protokol seperti MQTT atau CoAP(Nikolov, 2020). Namun, kita dapat menggunakan protokol lain seperti generasi baru HTTP/3. Teknologi QUIC (Iyengar & Thomson, 2021; Luglio et al., 2020; Sy et al., 2019) ini memberikan koneksi yang jauh lebih baik dan lebih aman untuk menerapkan sistem baru untuk pengiriman data antara klien dan server. Pengembangan QUIC lebih berfokus pada level userspace, membuatnya lebih mudah diimplementasikan atau dimodifikasi dalam transmisi. Komunitas pengembangan yang aktif dapat membantu dalam implementasi di berbagai lingkungan. Kami menggabungkan Multipath QUIC dengan skema hibrida, yang berarti QUIC bagus dalam kondisi jaringan buruk dan menggunakannya. Pada saat yang sama, kami mendeteksi jaringan dalam kondisi buruk dan menggunakan protokol lain jika kami tidak melihat adanya lingkungan yang buruk. Kami mengukur korupsi, tingkat kesalahan, keterlambatan, bandwidth, dan antrian pada sensor IoT ke server cloud. Kami membuat lingkungan simulasi yang terjadi di rumah kaca, seperti mengirimkan data kondisi aktual lingkungan. Kami memilih NS3(Ivey et al., 2017) sebagai simulator dan teknologi kontainer sebagai simulasi perangkat sensor.

Berdasarkan deskripsi di atas, para peneliti tertarik untuk mengetahui bagaimana sistem QUIC dapat menangani sistem rumah kaca dengan kondisi lingkungan internet yang sebenarnya.

2. Metode penelitian

2.1 Deskripsi Sistem

Dalam penelitian ini, kami menggunakan docker untuk mensimulasikan lingkungan jaringan, dan setiap kontainer dapat muncul sebagai perangkat sensor yang akan mengirimkan data ke broker. Pada skema yang diusulkan ini, kami menggabungkan penelitian terakhir yang sudah ada yaitu MP-QUIC dengan koneksi hibrida. Koneksi hibrida mengacu pada penggunaan MP-QUIC pada waktu tertentu dan protokol lain pada waktu lain karena QUIC dapat berjalan pada kondisi jaringan yang sangat buruk. Skema penelitian akan dijelaskan oleh gambar di bawah ini.



Gambar 1 Skema Jaringan Antara Client - Server

Dalam Gambar 1, dijelaskan bagaimana kondisi jaringan yang akan digunakan, di mana proxy akan berperan sebagai broker lokal dan juga untuk mempertahankan serta mengelola pengiriman paket dan penyaringan paket. Jaringan yang digunakan pada jaringan akses adalah hibrida, yaitu QUIC dan MQTT, MP-QUIC akan digunakan di jaringan tulang punggung, jumlah sensor yang dapat ditangani oleh proxy adalah jumlah yang tidak terbatas, namun jalur jaringan antara proxy dan server hanya ada tiga jalur, tiga jalur tersebut mewakili jaringan seluler, satelit, dan kabel. Dan juga, kami menggunakan tipe paket prioritas dan non-prioritas untuk membuat jenis pengiriman paket yang berbeda. Paket prioritas akan digunakan untuk mengirimkan paket yang membutuhkan respons segera pada sistem kontrol rumah kaca. Sebagai contoh, ketika sensor mendeteksi kebocoran air, sensor harus segera mengirimkan informasi ini, sehingga informasi tersebut dapat tiba dengan cepat. Paket non-prioritas akan mengirimkan informasi reguler, seperti kelembaban dan suhu lingkungan.

2.2 Metode Waterfall

Dalam penelitian ini menggunakan metode waterfall (Ji & Sedano, 2011), dimana metode ini dibagi menjadi kedalam beberapa bagian, diantaranya adalah :

- a. Analisis Kebutuhan Software

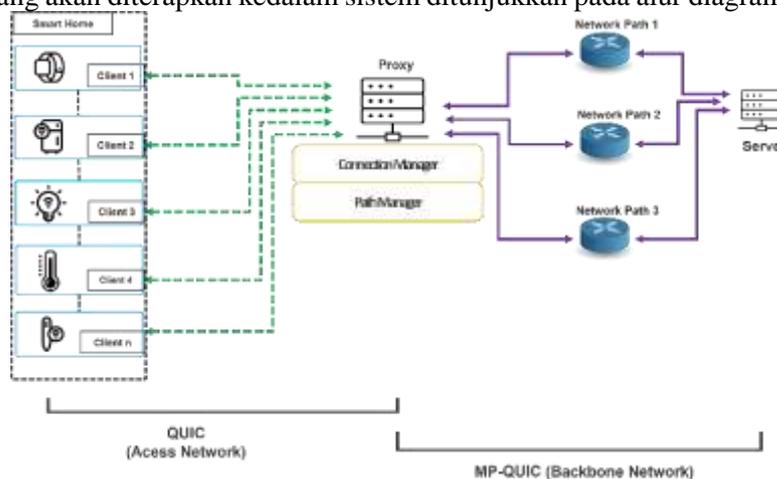
Dalam kegiatan ini dilakukan perancangan terhadap sistem yang akan dilakukan implementasi, dimana dalam hal ini adalah penerapan sistem jaringan berbasis IoT dalam lingkup pertanian cerdas.

b. Kode Program

Pada bagian ini dilakukan proses pembuatan kode program yang akan digunakan dalam sistem. Dimana akan dibagi menjadi dua bagian jenis,

Condition	QUIC	MQTT
Good Connection	No	Yes
Bad Connection	Yes	No

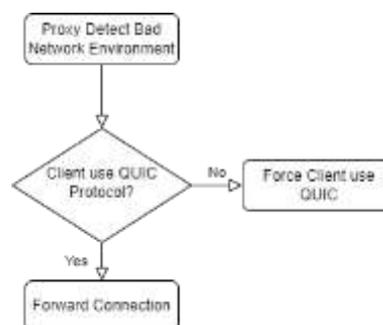
Sistem yang akan diterapkan kedalam sistem ditunjukkan pada alur diagram berikut ini.



Gambar 2 Desain Sistem Jaringan

c. Pengujian

Dalam proses pengujian ini, beberapa komponen sistem akan diuji, terutama fungsionalitas jaringan dalam kondisi baik atau buruk, sebagaimana diperlihatkan dalam diagram berikut. Pengujian ini bertujuan untuk mengevaluasi respons sistem terhadap variasi kondisi jaringan yang mungkin terjadi dalam penggunaan nyata. Dengan memperhatikan berbagai skenario yang mungkin terjadi, kita dapat memastikan bahwa sistem dapat berkinerja secara optimal dalam berbagai situasi, baik saat jaringan beroperasi dengan lancar maupun saat mengalami gangguan. Hasil pengujian ini akan menjadi panduan berharga dalam memperbaiki dan meningkatkan kinerja sistem secara keseluruhan.



Gambar 3 Uji Coba Jaringan

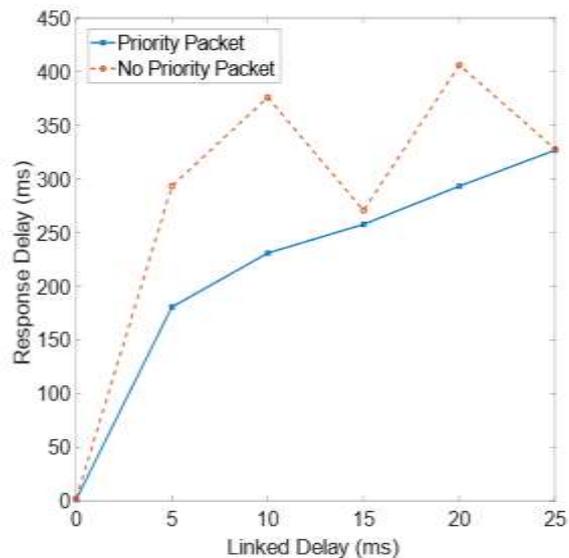
d. Pemeliharaan

Dalam tahapan pemeliharaan, sistem yang telah diimplementasikan akan diperiksa secara berkala untuk memastikan tingkat akurasi yang optimal. Langkah-langkah ini diperlukan untuk memastikan bahwa sistem tetap beroperasi dengan baik dan memberikan hasil yang konsisten sesuai dengan kebutuhan pengguna. Pemeliharaan berkala juga memungkinkan untuk mengidentifikasi dan menangani masalah potensial secara proaktif sebelum mereka menjadi lebih serius, sehingga dapat meminimalkan dampaknya terhadap kinerja keseluruhan sistem. Dengan demikian, pemeliharaan rutin menjadi bagian integral dari proses pengelolaan sistem yang efektif dan efisien.

3. Hasil dan Pembahasan

3.1. Response Delay

Dalam proses yang kami terapkan, penting untuk menetapkan kriteria-kriteria dan bobot yang sesuai untuk memastikan pencapaian hasil yang optimal. Oleh karena itu, kami telah mengidentifikasi kriteria-kriteria yang relevan dan menetapkan bobotnya agar dapat memberikan nilai yang terbaik. Salah satu langkah penting dalam proses ini adalah menetapkan jumlah antrian sebanyak lima antrian, dengan tingkat kesalahan sebesar 5%. Selain itu, untuk mengukur keterlambatan terkait, kami merencanakan sejumlah nilai yang berbeda, mulai dari 0 hingga 25 milidetik, sesuai dengan standar internasional yang berlaku. Seluruh proses pengukuran ini kemudian direkam dan hasilnya ditampilkan dalam Gambar 4.

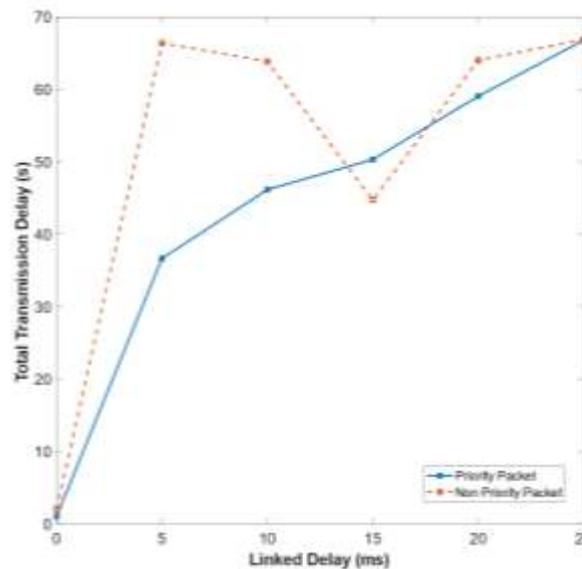


Gambar 4 Hasil Linked Delay

Gambar 4 memberikan gambaran mengenai hasil keterlambatan respons antara paket prioritas dan non-prioritas. Kami mencatat bahwa paket prioritas yang menggunakan QUIC sebagai protokol transmisi data dalam kondisi jaringan buruk menunjukkan hasil yang memuaskan, dimana QUIC mampu menjaga keterlambatan respons seminimal mungkin. Namun, terdapat perbedaan yang cukup signifikan ketika kami mengamati paket non-prioritas. Saat keterlambatan terkait meningkat, keterlambatan respons paket non-prioritas menjadi lebih tinggi dan tidak stabil. Dari sini, dapat disimpulkan bahwa hasil pengujian menunjukkan konsistensi nilai yang lebih tinggi pada paket prioritas, sementara paket non-prioritas cenderung mengalami fluktuasi nilai yang lebih besar. Hal ini menegaskan bahwa sistem kami mampu memberikan nilai hasil yang stabil dan dapat diandalkan dalam berbagai kondisi jaringan.

3.1. Transmission Delay

Dalam tahap analisis ini, kami fokus untuk memahami keterlambatan transmisi, yang merupakan waktu yang dibutuhkan oleh server atau klien untuk mengirimkan data dalam berbagai kondisi jaringan. Kami mengacu pada nilai-nilai keterlambatan yang sama seperti yang digunakan dalam analisis keterlambatan respons. Selama pengujian, kami membagi pengiriman paket menjadi dua jenis prioritas: pengiriman paket prioritas dan pengiriman paket non-prioritas, yang ditampilkan dalam Gambar 5.



Gambar 5 Hasil Transmission Delay

Gambar 5 memperlihatkan hasil akhir dari eksperimen kami. Hasil total keterlambatan transmisi paket prioritas menunjukkan kinerja yang baik, hal ini disebabkan oleh kemampuan proxy untuk mengelola jumlah paket prioritas seefisien mungkin. Namun, terdapat perbedaan yang mencolok saat kami melihat pengiriman paket non-prioritas. Grafik menunjukkan tingkat ketidakstabilan yang lebih tinggi ketika data dikirimkan, meskipun pada satu titik waktu, paket non-prioritas memperoleh nilai terendah dibandingkan dengan paket prioritas. Namun, nilai tersebut kemudian kembali meningkat, menunjukkan adanya fluktuasi yang signifikan. Dengan demikian, hasil ini memberikan gambaran tentang keberhasilan proxy dalam membuat paket prioritas pada nilai terendah, serta menyoroti tantangan dalam menjaga konsistensi nilai pada paket non-prioritas. Evaluasi ini penting untuk memastikan sistem kami mampu memberikan kinerja yang optimal dalam mengelola transmisi data dalam kondisi jaringan yang beragam.

4. Kesimpulan

Dari hasil eksperimen jaringan yang kami lakukan, kami menyimpulkan bahwa keterlambatan respons pada paket prioritas selalu mendapatkan nilai terbaik, demikian pula dengan hasil total keterlambatan transmisi. Dengan menerapkan teknik hibrida yang kami kembangkan, kami mampu mencapai nilai yang optimal dibandingkan dengan paket non-prioritas. Sistem proxy yang kami desain dapat mendeteksi kondisi jaringan dengan baik, sehingga mampu beralih ke protokol yang sesuai saat kondisi jaringan memburuk. Berdasarkan temuan ini, kami bermaksud untuk menerapkan sistem kami pada lingkungan IoT yang nyata, serta melakukan peningkatan pada algoritma yang kami gunakan di sisi proxy.

Dengan skema yang kami usulkan, kami berharap dapat mempercepat pengembangan jaringan IoT serta membuatnya lebih mudah diimplementasikan. Kami percaya bahwa dengan pendekatan ini, kondisi jaringan tidak akan lagi ditentukan semata oleh faktor-faktor alam, melainkan dapat diatur dan dikendalikan secara lebih efektif melalui logika dan teknologi yang kami terapkan. Namun, kami menyadari bahwa perlu adanya pengembangan lebih lanjut terhadap skema proxy dan skema perangkat sensor guna meningkatkan kinerja dan keandalan sistem secara keseluruhan. Dengan demikian, kami berkomitmen untuk terus mengembangkan solusi ini guna mendukung perkembangan teknologi IoT yang lebih baik di masa depan.

References

- [1] Hejazi, H., Rajab, H., Cinkler, T., & Lengyel, L. (2018). Survey of platforms for massive IoT. *2018 IEEE International Conference on Future IoT Technologies, Future IoT 2018, 2018-Janua*, 1–8. <https://doi.org/10.1109/FIOT.2018.8325598>
- [2] Ivey, J. S., Swenson, B. P., & Riley, G. F. (2017). Simulating networks with NS-3 and enhancing realism with DCE. *Proceedings - Winter Simulation Conference, 2(Riley 2008)*, 690–704. <https://doi.org/10.1109/WSC.2017.8247825>
- [3] Iyengar, J., & Thomson, M. (2021). *QUIC: A UDP-Based Multiplexed and Secure Transport* (Issue 9000). RFC Editor. <https://doi.org/10.17487/RFC9000>
- [4] Ji, F., & Sedano, T. (2011). Comparing extreme programming and Waterfall project results. *2011 24th IEEE-CS Conference on Software Engineering Education and Training, CSEE and T 2011 - Proceedings*, 482–486. <https://doi.org/10.1109/CSEET.2011.5876129>
- [5] Jin, W., & Kim, D. H. (2018). IoT device management architecture based on proxy. *Proceedings of 2017 6th International Conference on Computer Science and Network Technology, ICCSNT 2017, 2018-Janua*, 84–87. <https://doi.org/10.1109/ICCSNT.2017.8343663>
- [6] Luglio, M., Quadrini, M., Roseti, C., Zampognaro, F., & Romano, S. P. (2020). A QUIC-based proxy architecture for an efficient hybrid backhaul transport. *2020 23rd Conference on Innovation in Clouds, Internet and Networks and Workshops, ICIN 2020, Icin*, 144–146. <https://doi.org/10.1109/ICIN48450.2020.9059508>
- [7] Nikolov, N. (2020). Research of MQTT, CoAP, HTTP and XMPP IoT Communication protocols for Embedded Systems. *2020 29th International Scientific Conference Electronics, ET 2020 - Proceedings*, 18–21. <https://doi.org/10.1109/ET50336.2020.9238208>
- [8] Prof.Ms.P.V.Dudhe, Prof.Ms.N.V.Kadam, & Prof. R. M. Hushangabade. (2017). *Internet of Things (IOT): An Overview and its Applications*.
- [9] Sy, E., Mueller, T., Moennich, M., & Federrath, H. (2019). Accelerating QUIC's connection establishment on high-latency access networks. *IEEE International Conference on Performance, Computing and Communications (IPCCC)*.