

ANALISA DAN IMPLEMENTASI SMART HOME MENGGUNAKAN PROTOKOL MQTT DAN XMPP BERBASIS INTERNET OF THINGS

ANALYSIS AND IMPLEMENTATION SMART HOME USING MQTT AND XMPP PROTOCOLS BASED ON INTERNET OF THING

Feri Teja Kusuma¹, Rendy Munady², Arif Indra Irawan³

^{1,2,3}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

¹feritejak@student.telkomuniversity.ac.id, ²rendymunadi@telkomuniversity.ac.id,

³arifindrairawan@telkomuniversity.ac.id

Abstrak

Penggunaan protokol yang tepat akan sangat membantu pada kinerja jaringan yang sedang dibangun. Dengan penggunaan protokol MQTT yang memiliki kelebihan yaitu hemat bandwidth yang menggunakan sedikit daya baterai disebabkan protokol ini hanya menggunakan sedikit layer, sehingga protokol ini cocok untuk perangkat yang memiliki batasan memory. Lalu terdapat Protokol XMPP yang merupakan sebuah standar IoT yang dapat berbasis text, suara maupun video dengan teknologi open XML. Protokol XMPP memiliki banyak kelebihan yang ditawarkan didalam mengembangkan layanan pesan instan terutama dari segi server.

Pada hasil penelitian didapat penerapan protokol MQTT menjadi pilihan yang tepat untuk diterapkan pada jaringan dengan bandwidth rendah dan perangkat yang memiliki daya pemrosesan memori yang terbatas dengan nilai *throughput* sebesar 316 Bps, rata-rata *delay* sebesar 490,51 ms, *packet received* 1139,6 Byte. Berbanding dengan XMPP yang lebih cocok untuk diterapkan pada bagian *client* ke *server* karena fiturnya. Penerapan dengan mode protokol ini menghasilkan probabilitas *availability* sebesar 97,647% dan *reliability* sebesar 99,4%.

Kata kunci : *internet of thinks, smart home, raspberry pi, MQTT, XMPP.*

Abstract

The use of appropriate protocols will be very helpful in the performance of the network being built. With the use of the MQTT protocol which has the advantage of saving bandwidth that uses less battery power because this protocol only uses a few layers, this protocol is suitable for devices that have memory limitations. Then there is the XMPP Protocol which is an IoT standard that can be based on text, sound or video with open XML technology. The XMPP protocol has many advantages offered in developing instant messaging services, especially in terms of servers.

The results of the research show that the application of the MQTT protocol is the right choice to be applied to low bandwidth networks and devices that have limited memory processing with a throughput of 316 Bps, an average delay of 490.51 ms, packet received 1139,6 Bytes. Compared with XMPP which is more suitable to be applied to the client to the server because of its features. The application with this protocol mode results in a *availability probability* of 97,647% and *reliability* of 99,4%.

Keywords: *internet of thinks, smart home, raspberry pi, MQTT, XMPP.*

1. Pendahuluan

Internet of Things (IoT) dapat digambarkan sebagai penghubung benda-benda seperti telepon pintar, televisi Internet, dan sensor ke Internet dimana perangkat tersebut digabungkan menjadi bentuk baru yang memungkinkan adanya komunikasi antara seseorang dan benda tersebut. IoT juga dapat diterapkan untuk menciptakan sebuah konsep baru yaitu rumah cerdas yang menyediakan beberapa perangkat pintar untuk meningkatkan kualitas hidup.

Dalam era sekarang dimana konsep IoT diterapkan pada ranah yang lebih umum, ukuran sensor dan spesifikasinya mulai disesuaikan. Untuk membangun jaringan IoT yang dapat dibuat dengan berbagai sensor dan tidak membebani jaringan, maka perlu didesain dimulai dari perangkat hingga protokol yang digunakan untuk mendapatkan jaringan dengan performa yang baik.

Kesesuaian penggunaan protokol dapat memberikan kualitas dan performa yang baik untuk jaringan yang akan dibangun. Oleh karena itu pada penelitian ini yang berjudul "Analisa dan Implementasi Protokol MQTT dan XMPP pada *Smart Home* berbasis Internet of Things" akan membandingkan penggunaan dua protokol TCP/IP di bagian application layer yang digunakan dalam jaringan IoT yaitu protokol MQTT yang memiliki kelebihan pada ukuran yang kecil untuk pesan yang dikirimkan dan XMPP yang memiliki kelebihan dalam fitur

yang tidak ditawarkan oleh MQTT seperti identitas, federasi, dan berbagai ekstensi untuk IoT yang akan diterapkan pada *smart home*.

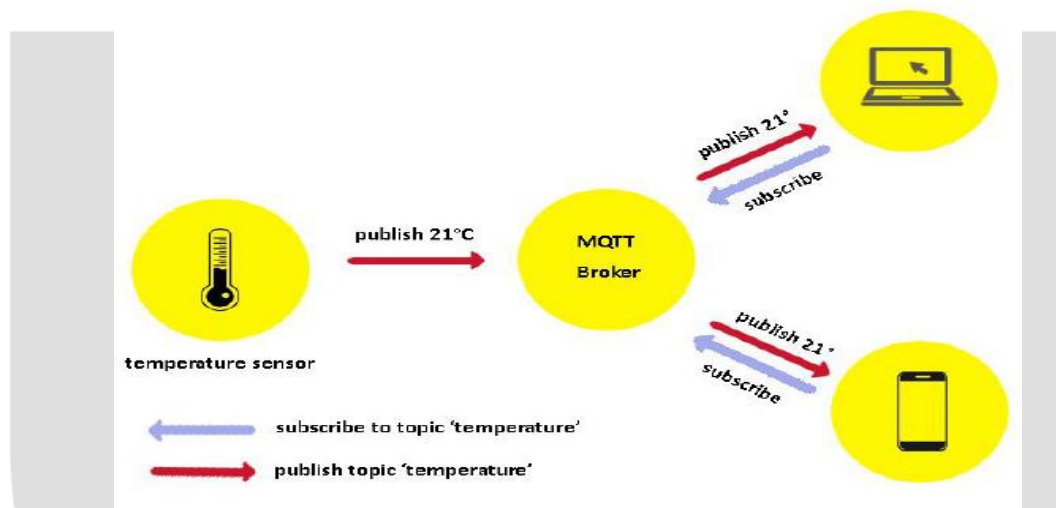
2. Dasar Teori /Material dan Metodologi/perancangan

2.1 Internet of Things

Internet of Things atau IoT merupakan sebuah topik hangat dalam industri teknologi dan ruang lingkup keinsinyuran. IoT sendiri merupakan ketika semua benda dapat terkoneksi satu dengan lainnya dengan menggunakan jaringan internet[10]. IoT memiliki potensi untuk memberikan solusi efisiensi energi, keamanan, kesehatan, pendidikan dan banyak aspek lain di kehidupan sehari-hari dengan meminimalkan campur tangan dari manusia secara langsung. Pengaplikasiannya pada pembuatan smart city, aplikasi kesehatan, aplikasi pendidikan, aplikasi produktivitas dan lainnya dimana semuanya berbasis IoT [1].

2.2 Message Queuing Telemetry Transport (MQTT)

MQTT adalah protokol konektivitas mesin ke mesin yang dikembangkan pada tahun 1999 oleh Dr. Andy Stanford dan Arlen Nipper, protokol ini berbasis open source yang memiliki kelebihan yaitu hemat bandwidth yang menggunakan sedikit daya baterai disebabkan protokol ini hanya menggunakan sedikit layer antara lain control header, packet length, variable length header dan payload sehingga minimal paket yang dikirim oleh MQTT sebesar 1 byte. Protokol ini juga menggunakan arsitektur publish/subscribe yang berbeda dengan Hypertext Transfer Protocol (HTTP) yang menggunakan arsitektur request/response. Fungsi dari arsitektur MQTT ini ialah meneruskan setiap pesan yang dikirim oleh klien kepada klien lain yang sah sehingga pesan akan terkirim secara real-time. Titik dari komunikasi MQTT disebut sebagai MQTT Broker yang bertanggung jawab untuk mengirim pesan antara pengirim kepada penerima yang sah[2].



Gambar 1. Ilustrasi Publish/Subscribe pada protokol MQTT[2]

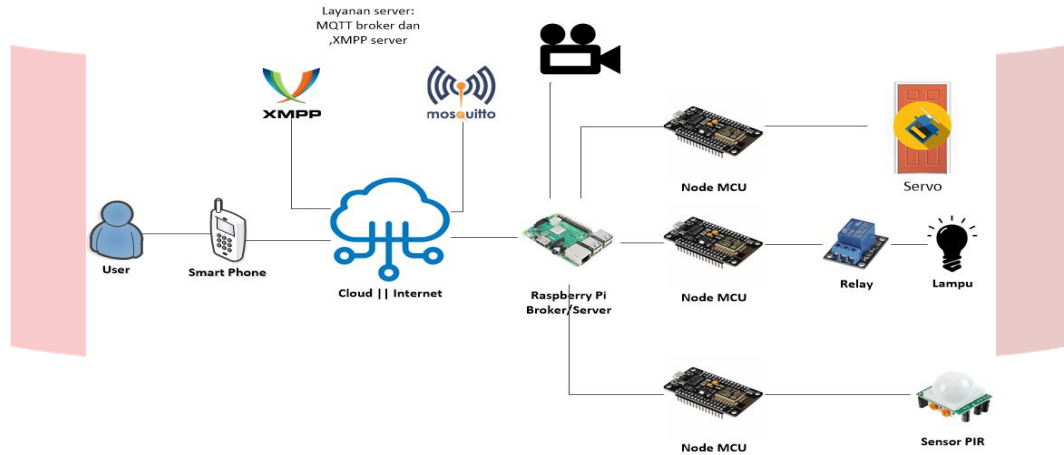
2.3 Extensible Messaging and Presence Protocol(XMPP)

Extensible Messaging and Presence Protocol yang dikenal sebagai XMPP adalah teknologi untuk komunikasi real-time, menggunakan XML yang dikenal sebagai Extensible Markup Language. Melalui XMPP, potongan-potongan kecil XML dapat dikirim dari satu entitas ke entitas lain secara real time. XMPP menyediakan layanan seperti Enkripsi Saluran, Otentikasi, Kehadiran, pesan Peer-Peer, perpesanan multi-pihak, daftar Kontak, Pesan dan Kehadiran Online. XMPP termasuk aplikasi seperti Instant Messaging, Group-Chat, Game, Geolocation, dan Voice over IP[3].

XMPP telah distandarisasi di dalam Internet Engineering Task Force (IETF). Protokol ini mendukung beberapa pola komunikasi, termasuk Asynchronous Messaging, Publish / Subscribe dan Request / Response. XMPP dikembangkan oleh komunitas open source Jabber untuk mendukung protokol pesan terbuka, aman, bebas spam dan terdesentralisasi. XMPP memungkinkan pengguna untuk berkomunikasi satu sama lain dengan mengirim pesan instan di Internet tidak peduli sistem operasi apa yang mereka gunakan. XMPP memungkinkan aplikasi IM untuk mencapai otentikasi, kontrol akses, pengukuran privasi, enkripsi hop-by-hop dan end-to-end, dan kompatibilitas dengan protokol lain[6].

Selain menyediakan kehadiran dan kemampuan pengiriman pesan, XMPP juga telah digunakan dalam aplikasi VoIP, gaming dan - yang paling baru - aplikasi internet of things (IoT). XMPP Standards Foundation (dahulu bernama Jabber Software Foundation) membantu mengembangkan rangkaian ekstensi baru yang akan membantu sensor dan aktuator menggunakan XMPP untuk bertukar informasi secara real-time[7].

3. Perancangan Sistem
3.1 Arsitektur Sistem

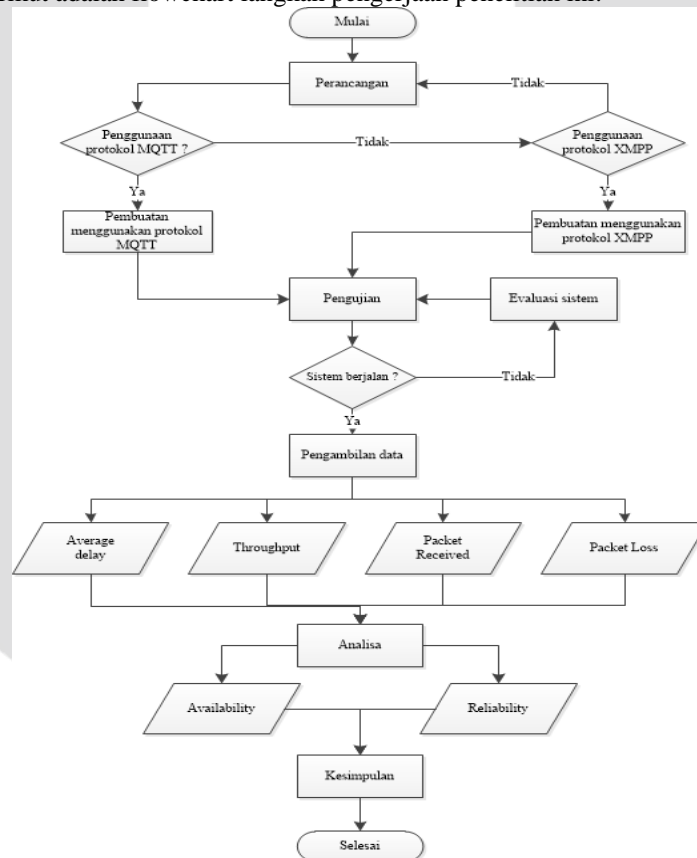


Gambar 2. Arsitektur Pembangunan IoT pada smart home security

Pada gambar 1 arsitektur ini menjelaskan peran perangkat yang digunakan. Dalam gambar tersebut pengguna dapat terhubung dengan perangkat di rumahnya melalui smartphone yang terhubung melalui jaringan internet. Ketika adanya pesan dari sensor yang akan masuk ke smartphone maka raspberry pi akan bertindak sebagai server/broker protokol sistem yang telah diatur dan meneruskannya ke pemilik rumah melalui jaringan internet. Protokol yang akan digunakan pada sistem Raspberry pi adalah MQTT dan XMPP.

3.2 Flowchart Pengerjaan

Dalam pengerjaan penelitian ini terdapat beberapa tahapan yang akan dikerjakan. Pada penelitian ini yang menjadi objek penelitiannya adalah pengujian performa dari protokol MQTT dan XMPP dalam pengaplikasian di sistem *smart home*. Berikut adalah flowchart langkah pengerjaan penelitian ini:

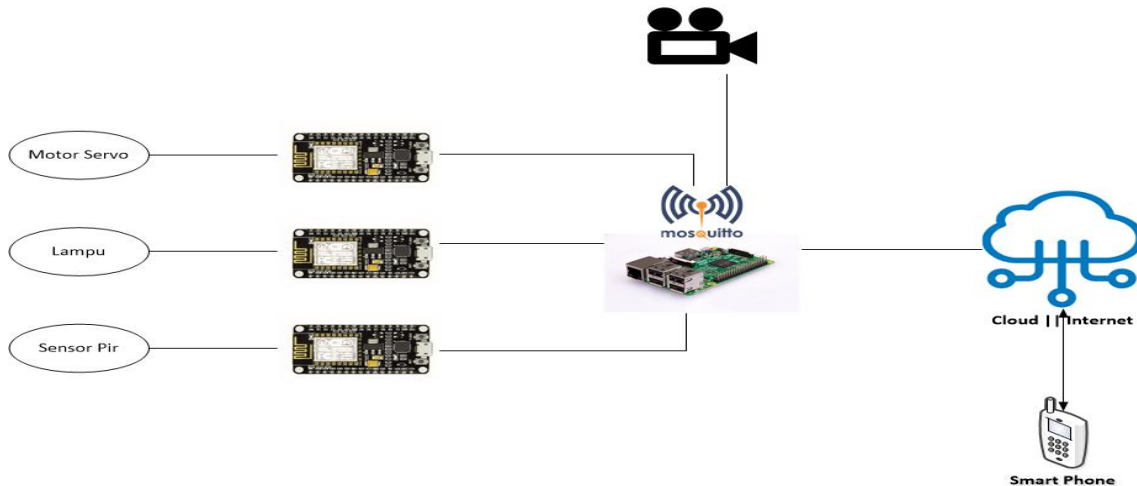


Gambar 3. Flowchart Pengerjaan

pengujian akan dilakukan 2 kali menggunakan protokol yang berbeda pada pengaplikasian sistem, kemudian nilai hasil keluaran dari pengujian akan diolah dan disimpulkan manakah protokol yang lebih baik digunakan pada sistem.

3.3 Skema Pengujian jaringan pada MQTT

Pada jaringan dengan protokol MQTT ini akan digunakan tiga buah NodeMCU dan sebuah kamera yang terhubung pada Raspberry PI, lalu dihubungkan pada client Smart Phone melewati internet.



Gambar 4. Perancangan Sistem dengan Protokol MQTT

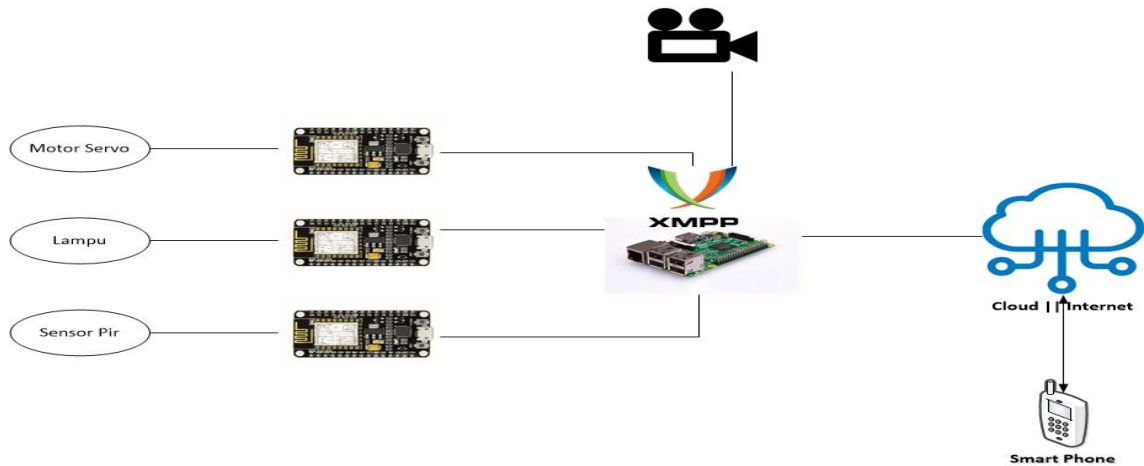
Untuk pengujian jaringan ini, NodeMCU akan digunakan sebagai *wireless sensor* yang melakukan *publish* maupun *subscribe* sesuai pengujian yang akan dilakukan ke Raspberry PI yang terpasang MQTT broker dengan opsi QOS yang digunakan adalah tipe QOS 0. Setiap pengujian akan direkam selama 60 menit melalui Wireshark dengan skenario *wireless sensor* bertambah, dimulai dari 1, 2, 3 dan juga dilakukan pengujian ke web penyedia layanan protokol *server* dalam melakukan pengiriman informasi, lalu diolah pada Microsoft Excell untuk mengetahui nilai QOS pada model jaringan ini.

Tabel 1. Skenario pengujian MQTT

Protokol	Jumlah Wireless Sensor	Skenario
MQTT	1	Kirim
	2	Kirim
	3	Kirim

3.4 Skema Pengujian jaringan pada XMPP

Model jaringan yang dipakai pada pengujian ini tidak berbeda dengan yang dilakukan pada MQTT, hanya saja pada XMPP setiap NodeMCU dibedakan dengan Jabber ID sebagai identitas.



Gambar 5. Perancangan Sistem dengan Protokol XMPP

Pada pengujian ini setiap NodeMCU akan digunakan sebagai *wireless sensor* yang akan meminta maupun mengirimkan suatu informasi nilai kepada Raspberry PI yang terpasang XMPP *server* Setiap pengujian akan direkam selama 60 menit melalui Wireshark dengan skenario *wireless sensor* bertambah, dimulai dari 1, 2, 3 dan juga dilakukan pengujian ke web penyedia layanan protokol *server* dalam melakukan pengiriman informasi, lalu diolah pada Microsoft Excell untuk mengetahui nilai QOS pada model jaringan ini.

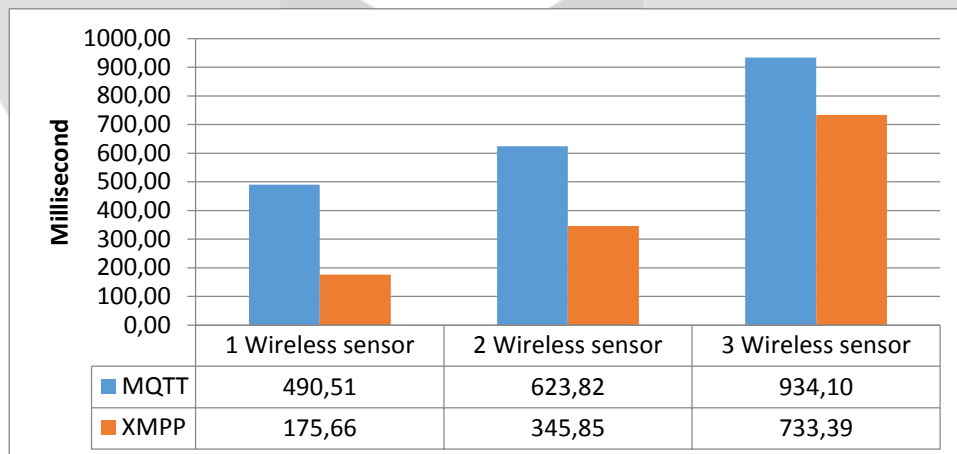
Tabel 2. Skenario pengujian XMPP

Protokol	Jumlah Wireless Sensor	Skenario
XMPP	1	Kirim
	2	Kirim
	3	Kirim

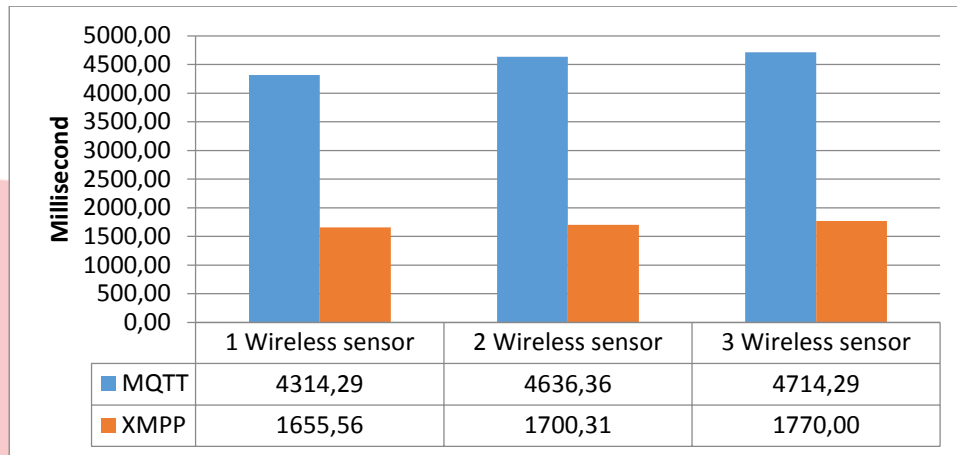
4. Analisis Hasil Simulasi

4.1 Pengujian Average delay

Pengujian average delay dalam penelitian ini adalah melakukan perhitungan terhadap jeda antara pengiriman suatu paket dengan paket selanjutnya, yang dilakukan oleh broker/server sebagai publisher. Pengujian akan dilakukan dengan cara mengirim paket dengan interval waktu rata rata 0,5.



Gambar 6. Hasil pengujian delay pengiriman wireless sensor ke *end point device* MQTT dan XMPP interval rata-rata 0,5 detik



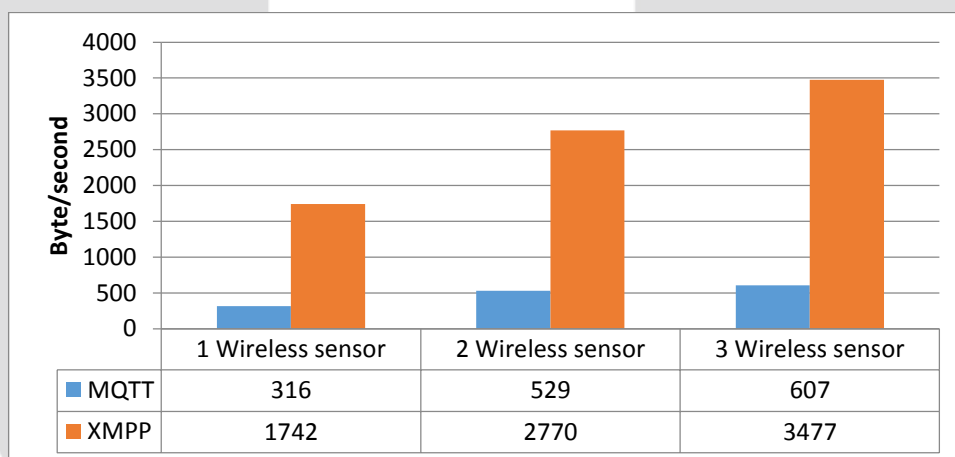
Gambar 7. Hasil pengujian delay pengiriman wireless sensor ke *end point device* MQTT dan XMPP interval rata-rata 5 detik

Hasil pengujian pengiriman packet dari broker/server ke wireless sensor yang ditunjukkan gambar 6 dan 7 didapat average delay protokol XMPP lebih unggul dibandingkan dengan MQTT pada pengiriman data *end to end*. Pada pengujian ini didapat nilai terbaik antara delay kedua protokol pada saat jumlah wireless sensor berjumlah satu buah, yaitu 490,51 milidetik dan 4314,29 milidetik untuk MQTT dan jaringan XMPP pada 175,66 milidetik dan 1655,56 milidetik

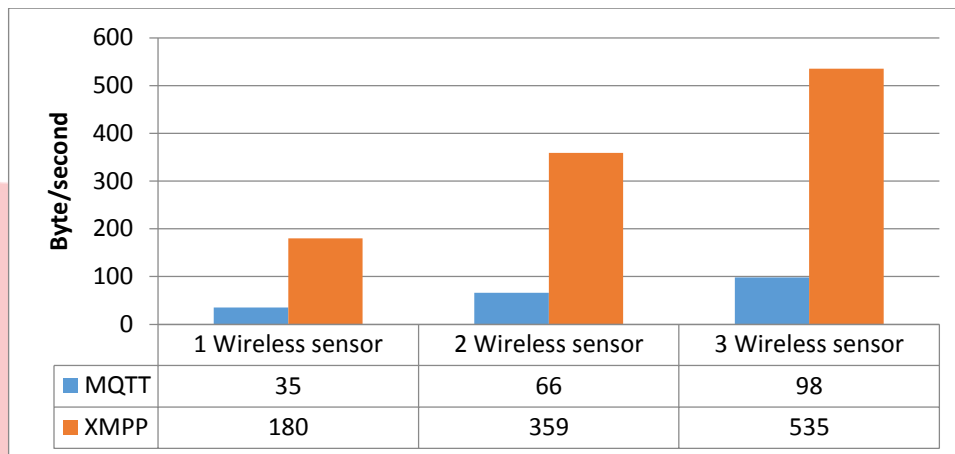
Dari pengujian yang berlangsung, protokol XMPP memiliki nilai average delay yang lebih rendah dibanding protokol MQTT. Dalam hal ini protokol XMPP akan cocok diimplementasikan untuk kondisi jaringan yang membutuhkan akses *realtime*.

4.2 Pengujian Throughput

Pengujian Throughput merupakan pengujian untuk mengetahui seberapa besar bandwidth yang dibutuhkan oleh jaringan. Besar dari throughput ini dapat dijadikan acuan dalam merancang kebutuhan bandwidth agar tidak terjadi *bottleneck* dalam suatu jaringan yang dibangun. Pengujian akan dilakukan dengan cara mengirim paket dengan interval waktu rata rata 0,5



Gambar 8. Hasil pengujian throughput pengiriman wireless sensor ke *end point device* MQTT dan XMPP interval rata-rata 0,5 detik



Gambar 9. Hasil pengujian throughput pengiriman wireless sensor ke *end point device* MQTT dan XMPP interval rata-rata 5 detik

Pada hasil pengujian yang ditunjukkan oleh gambar 8 dan 9, *throughput* tertinggi terdapat pada protokol ketika mengirim ke tiga wireless sensor yaitu sebesar 607 Byte/second dan 98 Byte/second untuk MQTT dan XMPP mendapatkan 3477 Byte/second dan 535 Byte/second.

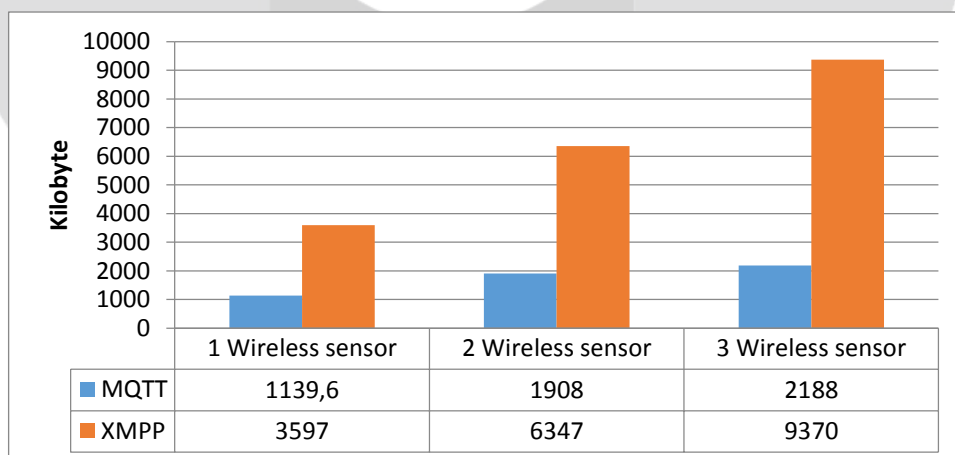
Dari pengujian Throughput untuk kedua protokol yang diteliti, protokol XMPP membutuhkan *bandwidth* yang lebih besar dibanding protokol MQTT, hal ini disebabkan oleh kecepatan pengiriman paket dengan perbedaan besar ukuran paket yang di kirimkan dari masing masing protokol yang diteliti.

4.3 Received Packet dan Packet Loss

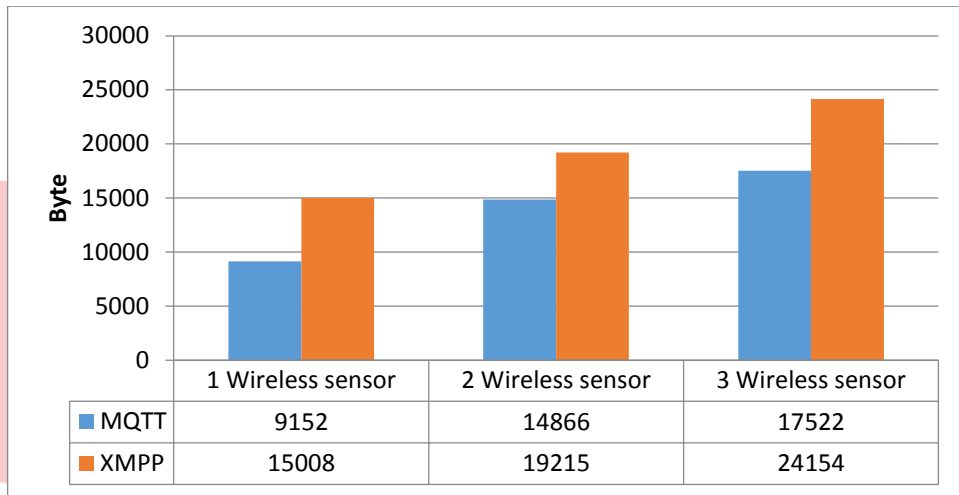
Pengujian received dan loss packet ini bertujuan untuk mengetahui beban dan kualitas yang akan broker/server proses pada saat menggunakan protokol tertentu. Pengujian received packet akan menghitung besaran paket yang diterima dalam proses pengiriman dari wireless sensor menuju *end point device*. Sedangkan pengujian packet loss merekam pengiriman paket yang gagal diterima disini yang lain pada jaringan dalam suatu protokol yang digunakan.

Tabel 3. Hasil pengujian *received* dan *loss packet* saat pengiriman dari wireless sensor ke *end point device* interval rata-rata 0,5 detik

Jumlah WSN	Received Packet(Kilo Byte)		Packet Loss (Byte)	
	MQTT	XMPP	MQTT	XMPP
1 Wireless sensor	1139,6	3597	9152	15008
2 Wireless sensor	1908	6347	14866	19215
3 Wireless sensor	2188	9370	17522	24154



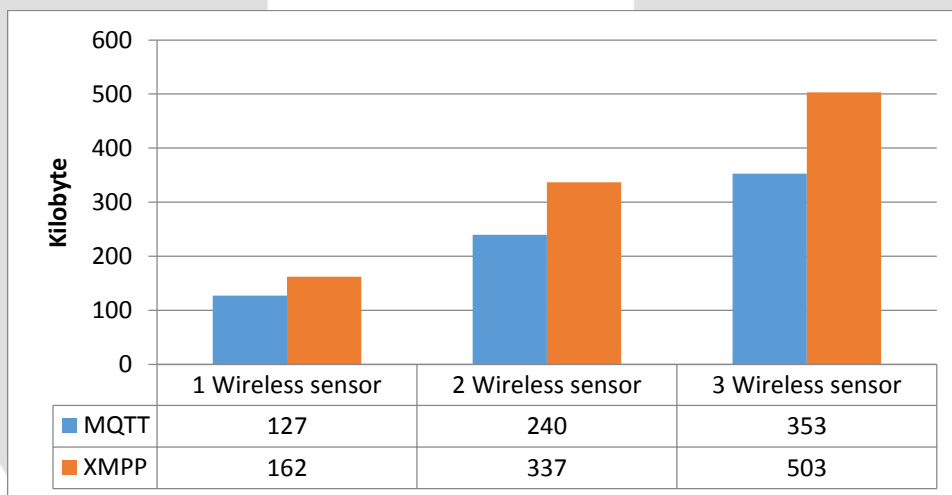
Gambar 10. Hasil pengujian received packet wireless sensor ke *end point device* MQTT dan XMPP interval rata-rata 0,5 detik



Gambar 11. Hasil pengujian *packet loss wireless sensor ke end point device* MQTT dan XMPP interval rata-rata 0,5 detik

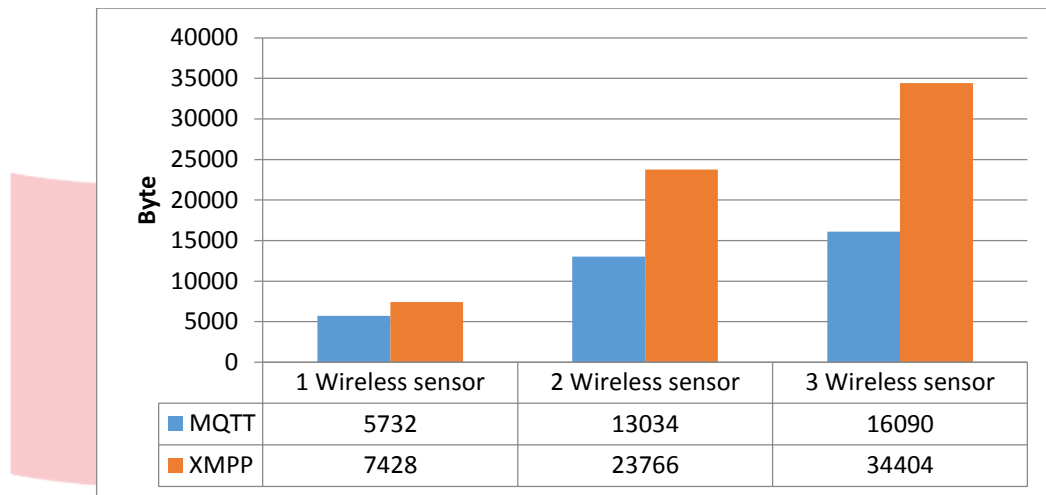
Tabel 4. Hasil pengujian *received dan loss packet* saat pengiriman dari wireless sensor ke *end point device* interval rata-rata 5 detik

Jumlah WSN	Received Packet(Kilo Byte)		Packet Loss (Byte)	
	MQTT	XMPP	MQTT	XMPP
1 Wireless sensor	127	162	5732	7428
2 Wireless sensor	240	337	13034	23766
3 Wireless sensor	353	503	16090	34404



Gambar 12. Hasil pengujian *received packet wireless sensor ke end point device* MQTT dan XMPP interval rata-rata 5 detik

Pada gambar 12. menunjukkan paket yang diterima dari pengiriman oleh XMPP ke server lebih besar dari pada MQTT, pada 3 wireless sensor yang menggunakan XMPP didapat 503 KB besar paket dalam waktu satu jam, sedangkan 3 wireless sensor yang menggunakan MQTT diterima sebesar 353 KB.



Gambar 13. Hasil pengujian *packet loss* wireless sensor ke *end point device* MQTT dan XMPP interval rata-rata 0,5 detik

Pada hasil pengujian yang ditujukan oleh gambar 11 dan 13 ketika pengujian paket yang dikirim dari wireless sensor ke *end point device*, *packet loss* yang terjadi pada MQTT lebih rendah daripada XMPP.

Besarnya *received packet* dari MQTT adalah yang paling kecil akibat besar pesan pada *header* paket yang dimiliki hanya berukuran 2 Byte serta ditambah 2 Byte tambahan untuk *topic length* dan sisanya secara opsional untuk isi pesan. Sedangkan XMPP menggunakan pengiriman berupa XML yang berisi *message*, *presence* dan *IQ* sehingga memiliki ukuran yang lebih besar dan juga XMPP mengirimkan paket ACK saat pengiriman yang membuatnya paket yang diterima lebih banyak.

Dari hasil keseluruhan pengujian *packet loss*, protokol MQTT memiliki hasil yang lebih reliabel untuk dipasangkan kedalam jaringan.

4.4 Pengujian Reliability dan Availability

Berdasarkan hasil pengujian *average delay*, *throughput*, *received and loss packet* sistem, pada jaringan ini akan digunakan protokol MQTT disisi wireless sensor ke broker untuk mengatasi beban paket serta bandwidth dan protokol XMPP pada server ke client untuk mengatasi hal kecepatan akses dan keamanan di jaringan publik.

a. Reliability

Pengujian *Reliability* digunakan untuk menentukan kemungkinan sistem tetap beroperasi secara terus menerus dalam menjalankan fungsinya. Pengujian direkam dalam rentang waktu setiap satu jam.

Tabel 4. Pengujian Reliability

Jam ke	Jumlah Kirim	Gagal	Persentase
1	849	0	100%
2	1746	0	100%
3	2569	3	99,6%
4	3388	5	99,4%
5	4208	6	99,8%

b. Availability

Pengujian *Availability* digunakan untuk mengetahui probabilitas suatu sistem untuk melakukan operasionalnya. Pengujian ini dilakukan selama tujuh hari pengamatan.

Tabel 5. Pengujian Availability

waktu(jam)	status	Availability
0-52	up-time	97,647%
52-53	down-time	
53-130	up-time	
130-131	down-time	
131-168	up-time	

Berdasarkan pengujian *reliability* dan *availability* pada sistem yang dilakukan, kemungkinan sistem melakukan fungsi atau operasi yang diperlukan pada saat tertentu sebesar 99,4% dengan *reliability* paling rendah dalam operasi sistem sebesar 97,3% yang dapat dikatakan sistem yang dibuat dapat diandalkan.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis data dari penelitian ini, maka dapat diperoleh kesimpulan sebagai berikut

1. Berdasarkan pengujian performa protokol yang dilakukan pada saat koneksi antara sensor sampai ke *end point device*, protokol MQTT memiliki hasil yang lebih unggul pada parameter packet receive, throughput dan packet loss.
2. protokol MQTT akan lebih baik di implementasikan ketika jaringan menyediakan *bandwidth* yang kecil agar tidak terjadi *bottleneck*. Ditinjau dari pengujian throughput, protokol MQTT menghasilkan 607 Byte/second dibanding protokol XMPP yang menghasilkan sebesar 3477 Byte/second pada jaringan yang sama.
3. Protokol XMPP cocok diterapkan pada komunikasi *endpoint device*, dari Raspberry pi ke *client* secara dua arah berdasarkan hasil *test performance* pada *average delay* untuk mendapatkan kecepatan akses. Selain itu XMPP menyediakan layanan keamanan sehingga aman dilewatkan melalui *cloud platform*.
4. *Availability* sistem yang diperoleh dari pengujian memiliki persentase 97,647% dengan *reliability* sistem yang diuji memiliki persentase terendah sebesar 99,4 %, maka sistem yang digunakan dapat dikatakan reliabel.

5.2 Saran

Dengan hasil pengujian dan penelitian diatas. Maka penulis menyarankan kedua protokol digunakan secara bersamaan, menggunakan MQTT untuk komunikasi M2M dan XMPP untuk komunikasi melewati *cloud platform*. Untuk penelitian selanjutnya penulis menyarankan hal sebagai berikut:

1. Pengujian dilakukan bersama fitur keamanan pada tiap protokol.
2. Melakukan pengujian pengiriman gambar/video.
3. Menambah jumlah node pada pengujian.
4. Penggunaan jenis node dan topologi yang berbeda

Daftar Pustaka:

- [1] K. Rose, S. Eldridge, and C. Lyman, "The internet of things: an overview," *Internet Soc.*, no. October, p. 53, 2015.
- [2] J. E. Luzuriaga, M. Perez, P. Boronat, J. C. Cano, C. Calafate, and P. Manzoni. Improving MQTT Data Delivery in Mobile Scenarios: Results from a Realistic Testbed. *Mobile Information Systems*, 2016:1–11, August 2016..
- [3] Krishnapur. Pooja M, M. Shruthi, Y, Chetan Adhikari. "Fast Realtime Data Transfer using XMPP", 2016
- [4] M. T. Jones, "Meet the Extensible Messaging and Presence Protocol (XMPP)," IBM developerWorks, Markham, ON, Canada, 2009.
- [5] Heng. Wang, Xiong. Daijin, Wang. Ping, Liu. Yuqiang, " A Lightweight XMPP Publish/Subscribe Scheme for Resource-Constrained IoT Devices", Chongqing University of Posts and Telecommunications,, Chongqing, China, 2017.
- [6] M. T. Jones, "Meet the Extensible Messaging and Presence Protocol (XMPP)," IBM developerWorks, Markham, ON, Canada, 2009
- [7] Heng. Wang, Xiong. Daijin, Wang. Ping, Liu. Yuqiang, " A Lightweight XMPP Publish/Subscribe Scheme for Resource-Constrained IoT Devices", Chongqing University of Posts and Telecommunications,, Chongqing, China, 2017.