

Evaluasi Kinerja *Testbed Routing Protocol* berbasis NodeMCU ESP8266 pada Perangkat IoT

HERIANSYAH, AHMAD REYNALDI NOPRIANSYAH, SWADEXI ISTIQPHARA

Institut Teknologi Sumatera
Email: heri@el.itera.ac.id

Received 1 Oktober 2020 | *Revised* 15 November 2020 | *Accepted* 28 November 2020

ABSTRAK

Jaringan Ad hoc pada perangkat Internet of Things (IoT) mempunyai sifat yang dinamis dengan node pada jaringan yang berperan sebagai router dan bergerak bebas secara random tanpa bantuan infrastruktur komunikasi sehingga topologi berubah sangat cepat seiring dengan perubahan posisi. Perubahan ini sangat mempengaruhi kualitas layanan pada perangkat IoT itu sendiri. Penelitian ini bertujuan untuk mengevaluasi protocol routing yang sudah ada dengan cara mengimplementasikan routing protocol tersebut di perangkat testbed berbasis NodeMCU ESP8266. Hal ini bertujuan untuk memilih protocol routing yang paling optimal sebelum proses implementasi dilaksanakan. Pengujian ini berlaku untuk routing protocol yang sudah ada maupun yang baru. Kinerja protocol jaringan diukur melalui nilai Quality of Service (QoS) ditempatkan pada scenario berbeda yang terdiri dari throughput, delay, jitter, dan packet delivery ratio sesuai dengan perbedaan beban jaringan, mobilitas, dan ukuran jaringan. Hasil penelitian ini menunjukkan bahwa testbed yang dibangun berhasil mensimulasikan routing protocol yang ada untuk menghasilkan QoS yang baik pada perangkat IoT.

Kata kunci: *IoT, routing protocol, testbed, QoS.*

ABSTRACT

Ad hoc networks on Internet of Things (IoT) devices have dynamic characteristics where the nodes on this network can operate as routers and move freely randomly without using any communication infrastructure so that the topology changes very quickly along with changes in position. This adjustment has a significant impact on the IoT device's service quality. This study aims to evaluate the existing routing protocols by implementing the routing protocol in a testbed based on NodeMCU ESP8266. It aims to choose the most optimal routing protocol before the implementation process is carried out. This test applies to both existing and new routing protocols. Network protocol performance is measured by the Quality of Service (QoS) value which includes throughput, delay, jitter, and packet delivery ratio in different scenarios based on network load, mobility, and different network sizes. The results show that this study was successful in simulating routing protocol in order to provide good QoS on IoT devices.

Keywords: *IoT, routing protocol, testbed, QoS.*

1. PENDAHULUAN

Jaringan *Ad hoc* merupakan jaringan yang memiliki karakteristik yang sangat dinamis (**Veni, 2010**) *node-node* yang ada pada jaringan ini dapat berkomunikasi dengan menjalankan fungsinya sebagai *router* sementara mereka bergerak bebas secara acak tanpa menggunakan infrastruktur sehingga topologi kerap kali berubah tanpa bisa diprediksi (**Wang, 2013**). Berdasarkan karakteristik tersebut, diperlukan *routing protocol* tertentu untuk mewujudkan komunikasi yang handal, kontinyu dan tanpa batas antar *node* pada jaringan tersebut sesuai dengan kebutuhan.

Pengembangan *routing protocol* terus meningkat seiring dengan banyaknya perangkat IoT yang memerlukan *routing protocol* yang optimal untuk setiap skenario yang berbeda. Dalam perjalanannya, peneliti banyak menghasilkan konsep *routing protocol* yang memiliki kinerja yang cukup baik dan menarik untuk dianalisis. Konsep *routing* tersebut biasanya hanya diimplementasikan dalam bentuk simulasi dan sangat jarang diuji secara langsung ke perangkat IoT yang sudah ada. Penelitian ini akan mengambil bagian dalam membantu peneliti dalam mengimplementasikan konsep *routing protocol* yang sudah dibuat atau mencari *routing protocol* yang paling optimal yang mereka perlukan pada jaringan IoT yang mereka miliki khususnya pada jaringan *Ad hoc*.

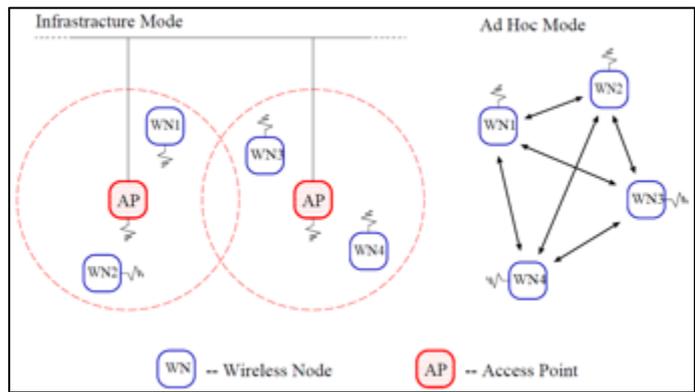
Untuk mengevaluasi kinerja dari suatu *routing protocol*, penelitian ini menyajikan suatu *testbed* berbentuk *prototype* berbasis NodeMCU ESP8266 sebagai perangkat yang dapat mengimplementasikan *routing protocol* yang sudah ada ataupun yang baru. *Testbed* yang dibangun diharapkan menghasilkan gambaran yang optimal terkait dengan kinerja *routing protocol* yang ada agar pengguna tidak hanya merasakan performansinya secara visual atau melalui simulasi aplikasi, akan tetapi dapat memberikan gambaran yang lebih nyata. Untuk mengetahui unjuk kerja dari *routing protocol* yang deprogram pada perangkat ini, peneliti membangun sebuah aplikasi monitoring berbasis web dengan cara menyambungkan salah satu ujung *node* pada jaringan *ad hoc* ke internet untuk melihat QoS yang dihasilkan dari setiap *testbed* yang dibangun. Website tersebut dapat memudahkan pengguna untuk melihat dan menganalisa secara langsung kinerja dari *routing protocol* yang sudah disiapkan sebelumnya. Adapun parameter QoS yang digunakan pada penelitian ini terdiri dari *throughput*, *delay*, *jitter*, dan *packet delivery ratio*.

2. METODE PENELITIAN

2.1. Arsitektur Jaringan *Ad hoc*

Pada jaringan *Ad hoc*, setiap unit membuat jaringan sementara dan dinamis tanpa bantuan dari infrastruktur jaringan seperti *Access Point* (AP). Semua unit independen satu sama lain dan dapat bekerja sama untuk menjaga konektivitas jaringan. Arsitektur jaringan *Ad hoc* dicirikan oleh topologi acak dan dinamis, serta komunikasi yang bersifat multi *hop*. Tidak diperlukan dukungan kabel, sehingga jaringan ini dapat terbentuk dengan sendirinya.

Gambar 1 menunjukkan perbedaan antara mode jaringan dengan menggunakan infrastruktur dan tanpa infrastruktur. Pada gambar tersebut dapat kita lihat, meskipun jangkauan transmisi mereka saling menutupi posisi geografis, *node* WN1 dan WN3 adalah bagian dari infrastruktur yang berbeda, dipisahkan oleh AP yang berbeda. Dengan demikian, mereka dapat berkomunikasi hanya melalui AP mereka masing-masing. Sementara, dalam mode *Ad hoc*, setiap *node* dapat berkomunikasi dengan setiap *node* lain yang ada di dalam jangkauan transmisinya. Ini berarti bahwa Jaringan *Ad hoc* tidak memerlukan bantuan perangkat pusat apa pun (**Barolli, Ikeda, Xhafa, & Duresi, 2010**).



Gambar 1. Arsitektur Jaringan Infrastruktur dan Jaringan Ad hoc (Barolli et al., 2010).

2.2. Parameter Kinerja Jaringan

Kinerja jaringan diukur dengan menggunakan parameter QoS (**Leon-Garcia & Widjaja**). QoS dapat mengukur konsistensi, tingkat keberhasilan pengiriman data, dan lain-lain. Ada beberapa parameter yang dapat digunakan untuk mengukur QoS yaitu:

a. *Throughput*

Throughput merupakan laju data aktual per satuan waktu yang diukur dari pengirim ke penerima (**Kompella & Ephremides, 2012**). *Throughput* juga bisa disebut sebagai *bandwidth* dalam kondisi yang sebenarnya. *Bandwidth* lebih bersifat tetap, sementara *throughput* memiliki sifat yang dinamis tergantung trafik yang sedang terjadi pada jaringan. *Throughput* mempunyai satuan bps (*bits per seconds*).

b. *Delay*

Delay dapat didefinisikan sebagai selang waktu yang dibutuhkan oleh suatu paket data untuk dapat berpindah dari suatu tempat ke tempat lain yang merupakan tujuannya. Pada *testbed* yang dirancang, sistem akan melakukan perhitungan pada *propagation delay* atau yang disebut juga sebagai *end-to-end delay*, yang merupakan jeda waktu yang hitung antara suatu waktu paket yang dikirimkan oleh *source node* dengan paket yang diterima oleh *destination node*. Menurut rekomendasi standar ITU-T G-114, nilai *end-to-end delay* yang dapat diterima dalam mendukung performansi suatu jaringan secara ideal dapat diklasifikasikan menjadi beberapa bagian menurut Tabel 1 dibawah ini.

Tabel 1. Klasifikasi Performansi Jaringan Berdasarkan *End-To-End Delay* (ITU, 2009)

Klasifikasi	<i>End-To-End Delay</i>
<i>great user experience</i>	<i>Up to 150 ms</i>
<i>good/acceptable user experience</i>	<i>150ms – 300 ms</i>
<i>degraded user experience</i>	<i>Over 300 ms</i>

c. *Packet Delivery Ratio (PDR)*

PDR adalah rasio antara banyaknya paket yang diterima oleh tujuan dengan banyaknya paket yang dikirim oleh sumber. Menurut rekomendasi standar TIPHON, nilai *PDR* yang dapat diterima dalam mendukung performansi suatu jaringan secara ideal dapat diklasifikasikan menjadi beberapa bagian menurut Tabel 2.

Tabel 2. Klasifikasi Performansi Jaringan Berdasarkan *Packet Delivery Ratio* (ETSI, 2002)

Klasifikasi	<i>End-To-End Delay</i>
<i>Very good</i>	100%
<i>Good</i>	97%
<i>Fair</i>	85%
<i>Poor</i>	75%

d. *Jitter*

Jitter merupakan variasi *delay* yang terjadi akibat adanya selisih waktu atau interval antar kedatangan paket di sisi penerima. *Jitter* dipengaruhi oleh variasi beban trafik dan besarnya tumbukan antar paket yang ada dalam jaringan. Semakin besar beban trafik di dalam jaringan akan menyebabkan semakin besar pula peluang terjadinya *congestion*. Dengan demikian, nilai *jitter* yang dihasilkan akan semakin besar. Keberadaan *jitter buffer* akan menambah nilai *end-to-end delay*.

2.2. *Routing Protocol*

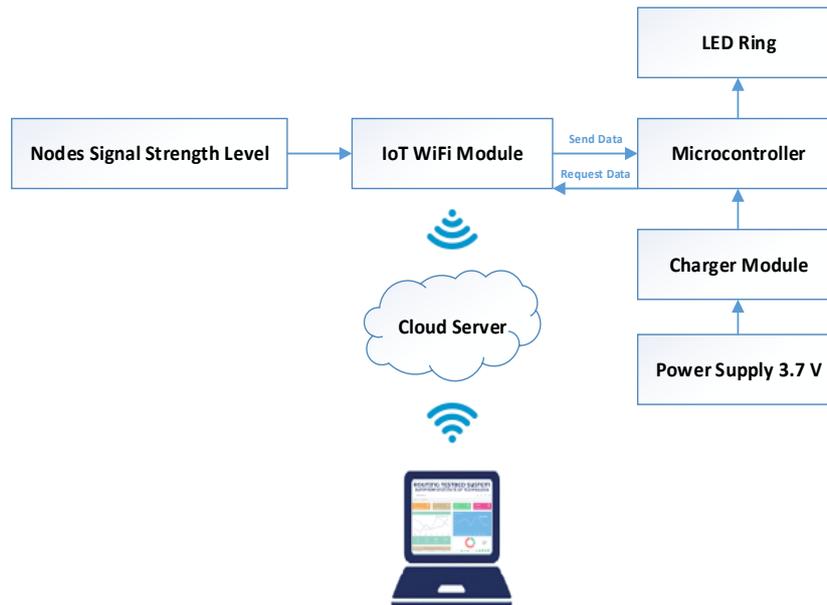
Secara umum, proses menemukan jalur atau rute antara pengirim dan target dikenal sebagai *routing*. Banyak peneliti di dunia yang menawarkan konsep baru terkait peningkatan kinerja *routing protocol* yang ada khususnya pada jaringan *Ad hoc* (Kaysina, Vasiliev, Abilov, Meitis, & Kaysin, 2018). Dalam menyesuaikan perkembangan teknologi yang terjadi dalam jaringan, protokol *routing* menggunakan berbagai algoritma, proses dan pesan (Mitra, Sarkar, & Hati, 2016). *Routing protocol* untuk jaringan *mobile ad hoc* terdiri dari *table-driven routing protocol* dan *on-demand routing protocol* serta gabungan keduanya yang disebut dengan *hybrid routing protocol* (Lei, Wang, & Li, 2016). Masing-masing dari *routing protocol* tersebut diwakili oleh masing-masing routing protocol yang populer seperti *routing protocol* DSDV untuk *table-driven*, AODV untuk *on-demand* dan ZRP untuk *hybrid*.

Syarat utama dari protokol *routing* untuk jaringan *Ad hoc* sendiri adalah mampu beradaptasi secara dinamis terhadap perubahan topologi jaringan. Hal ini diimplementasikan dengan teknik perencanaan untuk menelusuri perubahan topologi jaringan dan menemukan kembali rute baru saat rute yang lama telah rusak. Seperti yang telah disebutkan di atas bahwa jaringan *Ad hoc* tidak mempunyai infrastruktur, operasi ini dapat ditunjukkan dengan cara terdistribusi dengan sekelompok *node* yang saling bekerja sama dalam suatu jaringan. Beberapa ukuran untuk menentukan kualitas protokol *routing* yang dinamis untuk jaringan *Ad hoc* antara lain *Routing overhead*, *Timeliness* (ketepatan waktu), *Path optimality*, Bebas dari *loop*, Kompleksitas penyimpanan dan tingkat *Scalability*.

3. HASIL DAN PEMBAHASAN

3.1. Perancangan Perangkat Keras

Sistem *node* terbagi menjadi 3 fungsi, yaitu sebagai *destination node*, *connecting node*, dan *source node*. Pada sistem *node* terdapat 4 komponen utama, yaitu *power supply*, *microcontroller*, modul *charger*, dan lampu indikator. *Source node* akan mengirim pesan ke *destination node* melalui beberapa *connecting node*. Pesan yang diterima akan diolah oleh *destination node* untuk dapat dikirimkan ke *database*. Berikut merupakan rancangan *hardware* pada sistem *testbed* yang ditunjukkan pada Gambar 2 dibawah ini.



Gambar 2. Desain Hardware Sistem Testbed

3.2. Perancangan Perangkat Lunak

3.2.1 Perancangan Sistem Kerja Testbed

Pada perancangan awal, sistem akan menggunakan *mesh network firmware*. Desain *mesh network* yang akan dibuat sesuai dengan desain *mesh network* dari ESP-Mesh dengan alur sebagai berikut :

a. Pesan yang digunakan berbasis JSON

Komponen pesan yang dikirimkan terdiri dari dua informasi, yaitu waktu pengiriman oleh *source node*, serta jumlah data berdasarkan banyaknya data yang dikirimkan.

b. Metode pengiriman data

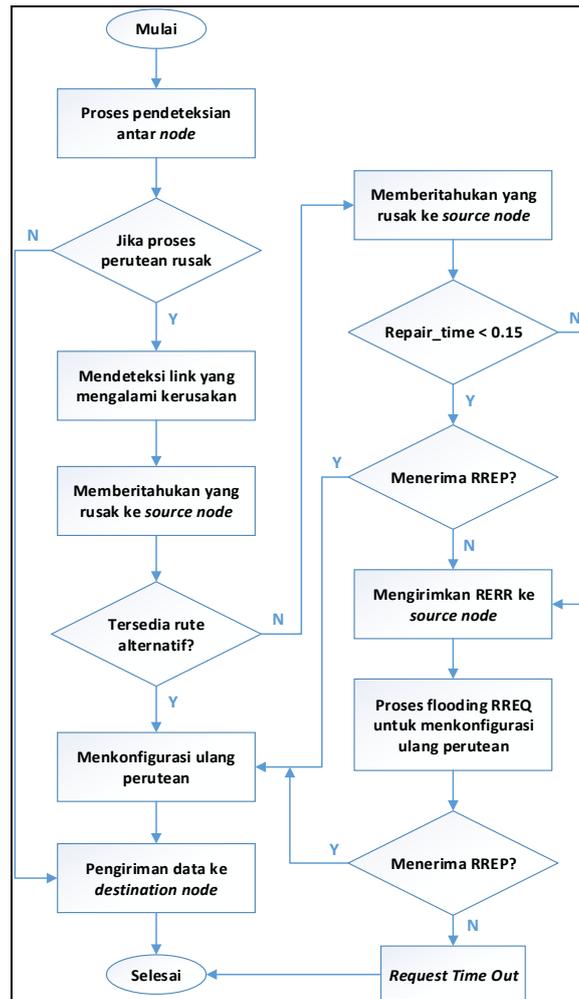
Metode pengiriman pesan menggunakan *single message*, yang mana sebuah pesan akan dikirimkan oleh *source node* menuju *destination node* dengan melewati beberapa pilihan *connecting node* melalui suatu *gateway* tertentu. Pada proses pengalamatan, tiap alamat *node* berasal dari alamat mac SOFT_AP, yang sebenarnya 4 *byte* terakhir alamat mac.

c. Fungsi *time synchronization*

Fungsi *time synchronization* akan dieksekusi pada *background task*, yang mana memiliki fungsi sebagai pengkondisian waktu dari tiap *node* pada jaringan *mesh* sehingga *task* akan berjalan hampir bersamaan di tiap *node*. Pada jaringan *mesh*, Fungsi *time synchronization* akan menggunakan algoritma dari NTP *protocol*. Klien NTP secara teratur memilih satu atau lebih server NTP.

d. *Routing message*

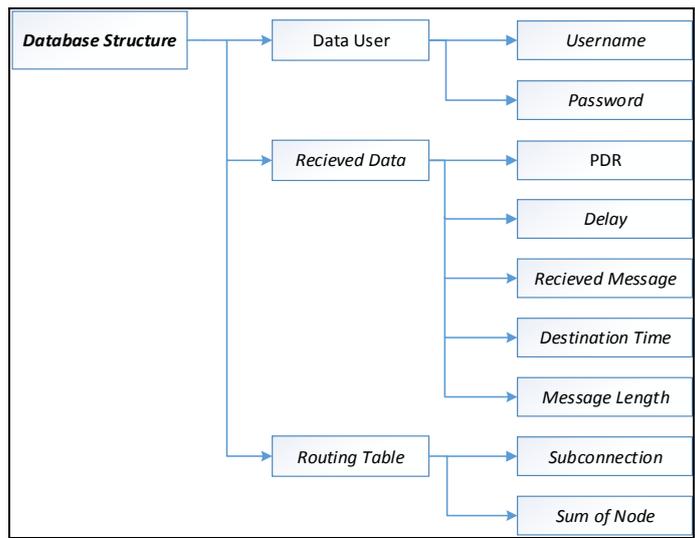
Pesan akan dikirim hingga sampai pada tujuan melalui *hop* atau loncatan dari *node* terdekatnya. Melalui teknik ini, pesan akan terkirim oleh *source node* menuju *destination node*. Secara umum, berikut merupakan alur sistem kerja *testbed* dalam melakukan proses *routing* hingga pengiriman data/paket dari *source node* menuju ke *destination node*.



Gambar 3. Diagram Alur Sistem Kerja *Testbed*

3.2.1 Perancangan *Website Monitoring Testbed*

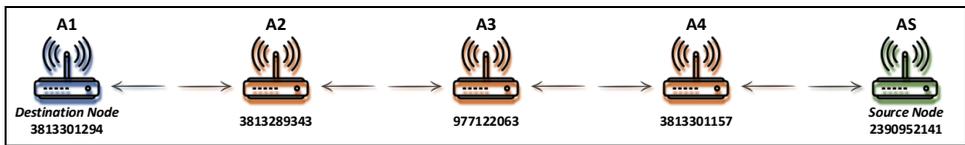
Sistem yang dirancang melalui beberapa tahap disesuaikan dengan kebutuhan sistem yang ada, yaitu: perancangan basis data, perancangan struktur menu, perancangan proses, dan perancangan *layout*. Beberapa tahapan dilakukan pada perancangan antara lain: mengumpulkan beberapa informasi dari setiap *node* secara *real time* dan mengirimkan ke *cloud*, penyimpanan data hasil pengukuran parameter QoS, dan visualisasi data, baik dalam bentuk tabel maupun kurva dan serta analisis sistem oleh pengguna. Berikut merupakan struktur *database* yang telah dirancang yang ditunjukkan pada Gambar 4 dibawah ini.



Gambar 4. Diagram Struktur Database

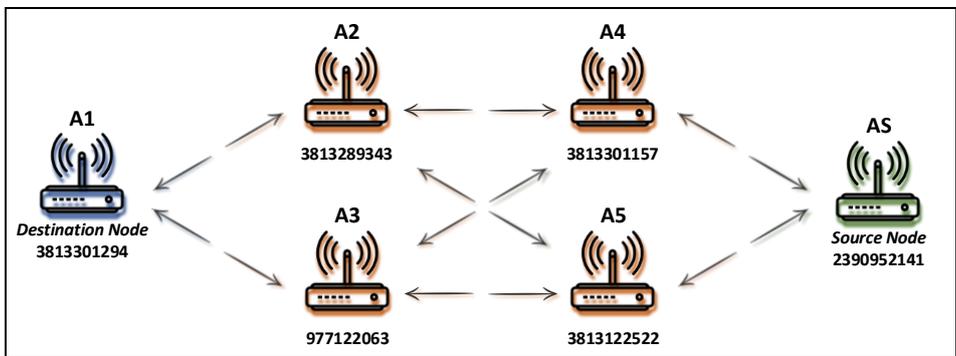
3.2. Skenario Topologi Jaringan

Penelitian ini akan diuji dalam 3 skenario. Pada skenario pertama, pengujian sistem akan dilakukan dengan menggunakan lima buah *node*. Dua buah *node* merupakan end *node* (*source node* dan *destination node*), serta tiga *node* lainnya merupakan *connecting node*. Pengujian ini dilakukan dalam keadaan *Line of Sight* (LOS) seperti yang ditunjukkan pada Gambar 5 dibawah ini.



Gambar 5. Implementasi Pengujian Skenario 1

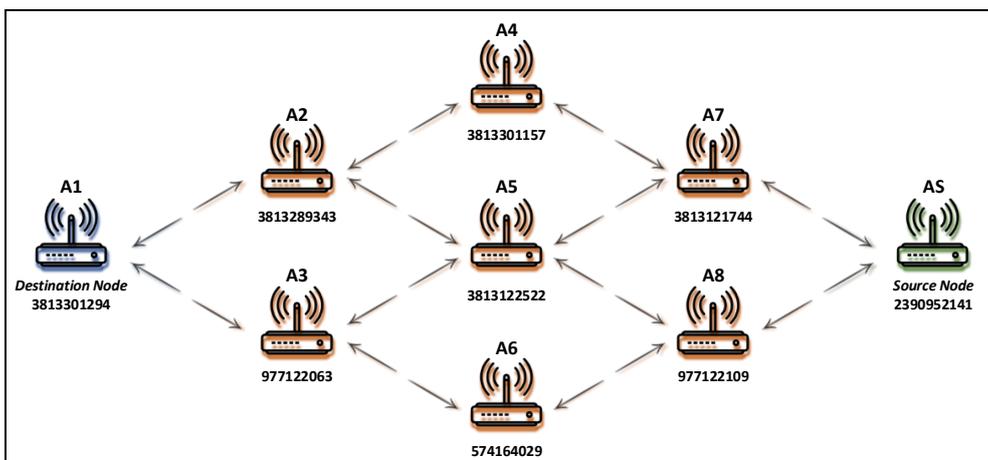
Skenario kedua, pengujian sistem akan dilakukan dengan menggunakan enam buah *node*. Dua buah *node* merupakan end *node* (*source node* dan *destination node*), serta empat *node* lainnya merupakan *connecting node* seperti yang ditunjukkan pada Gambar 6 dibawah ini.



Gambar 6. Implementasi Pengujian Skenario 2

Skenario ketiga, pengujian sistem akan dilakukan dengan menggunakan sembilan buah *node*. Dua buah *node* merupakan end *node* (*source node* dan *destination node*), serta tujuh

node lainnya merupakan *connecting node* seperti yang ditunjukkan pada Gambar 7 dibawah ini.

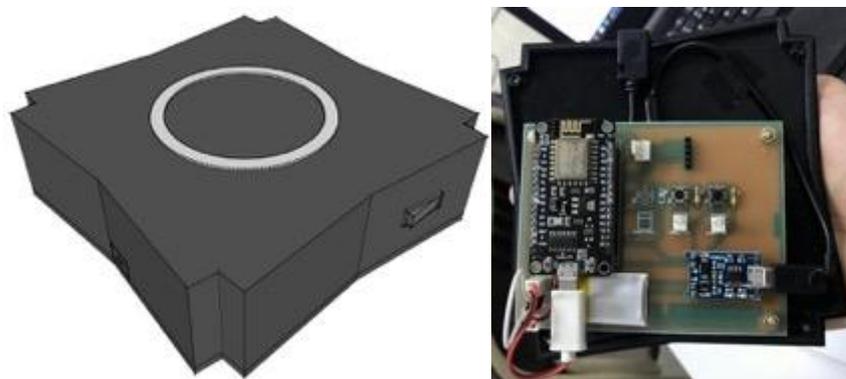


Gambar 7. Implementasi Pengujian Skenario 3

4. PENGUJIAN DAN ANALISIS

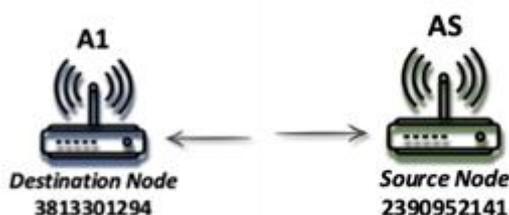
4.1 Pengujian RSSI (*Receive Signal Strength Indicator*)

Pengujian RSSI dilakukan untuk mengukur indikator kekuatan sinyal yang diterima antar *node* ESP8266. Pengujian ini dilakukan untuk mengetahui respon yang diberikan antar dua *node* yang terkoneksi satu sama lain terkait jarak maksimal dan waktu koneksi. Gambar 8 menunjukkan hasil implementasi dari rancangan alat yang akan kita uji nilai RSSI nya.



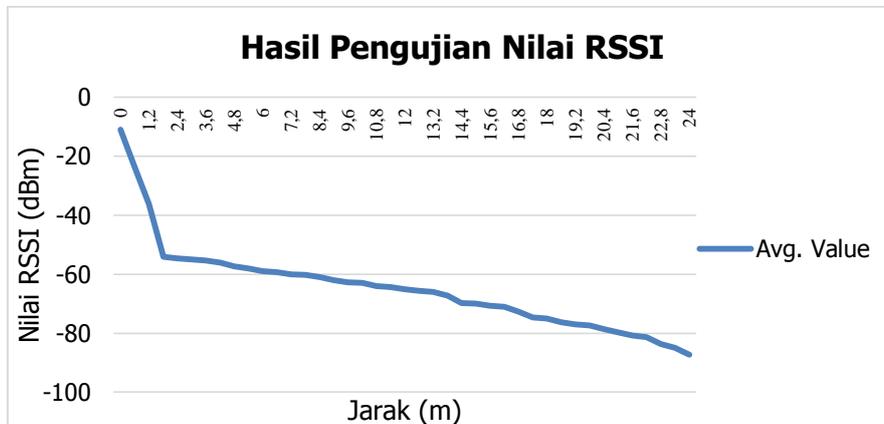
Gambar 8. Hasil Implementasi Perangkat

Pengujian ini dilakukan dalam keadaan *Line of Sight* (LOS) seperti yang ditunjukkan pada Gambar 9. Berdasarkan hasil pengukuran RSSI, diperlukan minimal 30 meter sebagai jarak maksimumnya untuk memutuskan hubungan antara dua *node* atau lebih. Dengan kata lain, pengujian sistem *testbed* yang terdiri dari 10 *node* harus dilakukan pada kawasan yang cukup luas.



Gambar 9. Skema Pengujian di Ruang Terbuka

Pengujian LOS dilakukan di luar ruangan dalam satu garis lurus sehingga sinyal pengirim dapat langsung diterima. Berikut grafik yang dihasilkan pada pengujian nilai RSSI dan waktu koneksinya yang ditunjukkan pada Gambar 10.



Gambar 10. Grafik Pembacaan Data Nilai RSSI

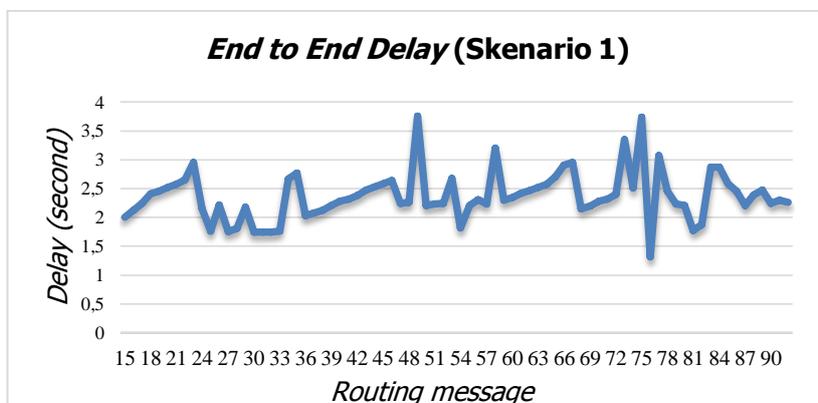
Dari grafik hasil pengukuran, dapat disimpulkan bahwa semakin jauh jarak antara dua *node*, maka kekuatan sinyal Wi-Fi akan semakin lemah dan waktu yang diperlukan untuk menghubungkan dua *node* juga akan semakin lama.

4.3.1 Pengujian Parameter Kinerja Jaringan

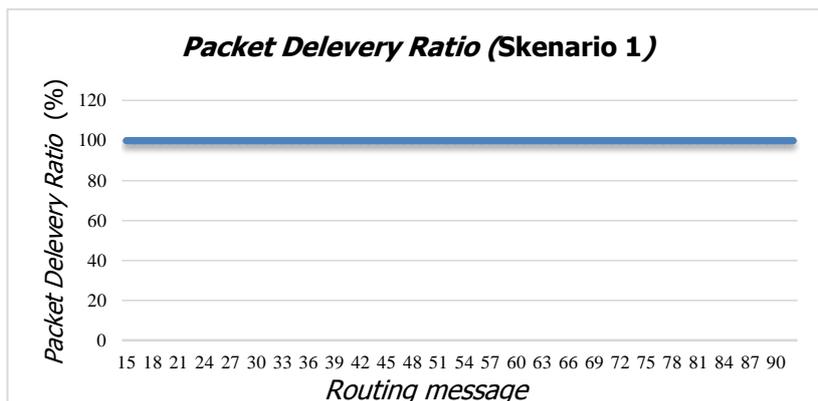
Pada penelitian ini digunakan beberapa skenario dengan jaringan yang sama untuk sistem kerja yang sama. Pengujian parameter kinerja jaringan dilakukan untuk mengetahui beberapa parameter yang mendukung performansi dari sistem *testbed*. Pengujian ini dilakukan dengan mengirimkan pesan dari *source node* (AS) menuju ke *destination node* (A1).

4.3.1.1 Hasil Pengujian Skenario 1

Dengan menggunakan skenario ini, dapat dilihat performansi sistem *testbed* yang dihasilkan dengan menambahkan banyaknya pesan yang dikirimkan.



Gambar 11. Grafik Parameter Delay Pada Skenario 1

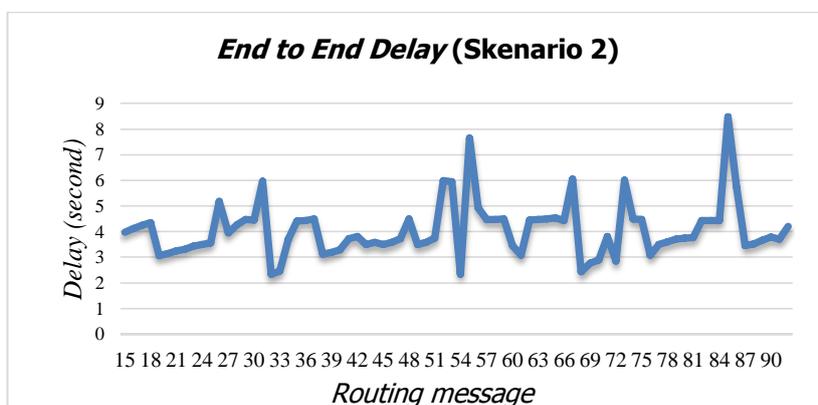


Gambar 12. Grafik Parameter *Packet Delivery Ratio* (PDR) pada Skenario 1

Gambar 11 dan 12 diatas merupakan grafik parameter *delay* dan *packet delivery ratio* (PDR) yang dihasilkan terhadap banyaknya pesan yang dikirimkan pada skenario pertama. Gambar 11 tersebut menunjukkan bahwa rata-rata *delay* pengiriman pada skenario pertama sebesar 2,375 *seconds*. Selain itu, *packet delivery ratio* (PDR) yang dihasilkan seperti yang ditunjukkan pada Gambar 12 semuanya mencapai 100% untuk semua pengiriman. Stabilitasnya nilai PDR yang dihasilkan disebabkan oleh jumlah *node* yang diuji tidak terlalu banyak, sehingga mekanisme dari sistem *testbed* masih dapat memberikan layanan yang baik. Adapun rata-rata nilai *throughput* yang dihasilkan pada pengujian skenario pertama yaitu sebesar 183,161 bps dengan nilai *jitter* minimal sebesar 1,27 *seconds*.

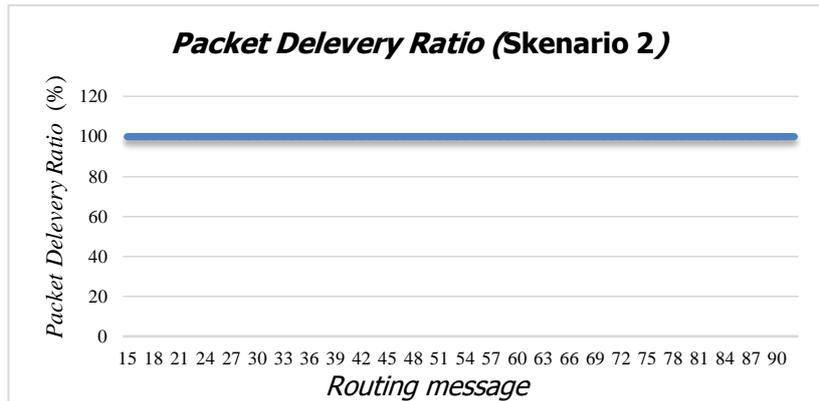
4.3.1.2 Hasil Pengujian Skenario 2

Pada skenario kedua, dapat dilihat performansi sistem *testbed* yang dihasilkan terhadap penambahan banyaknya pesan yang dikirimkan.



Gambar 13. Grafik Parameter *Delay* pada Skenario 2

Gambar 13 dan 14 diatas merupakan grafik parameter *delay* dan *packet delivery ratio* (PDR) yang dihasilkan terhadap banyaknya pesan yang dikirimkan pada skenario pertama. Gambar 13 tersebut menunjukkan bahwa rata-rata *delay* pengiriman pada skenario pertama sebesar 4,06 *seconds*. Selain itu, *packet delivery ratio* (PDR) yang dihasilkan semuanya mencapai 100% untuk semua pengiriman.

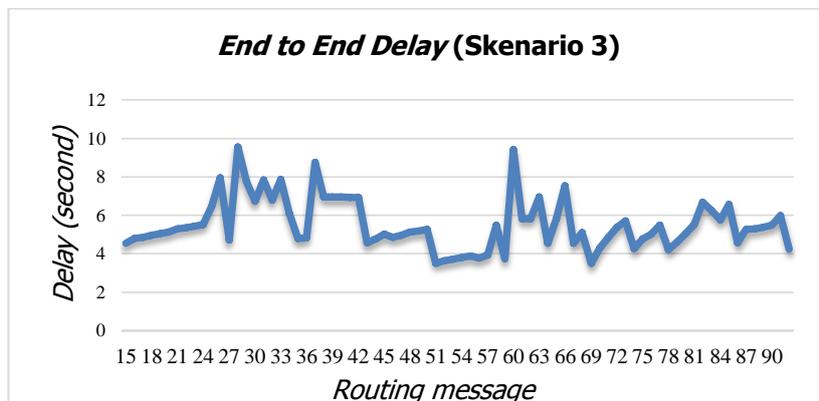


Gambar 14. Grafik Parameter *Packet Delivery Ratio* (PDR) pada Skenario 2

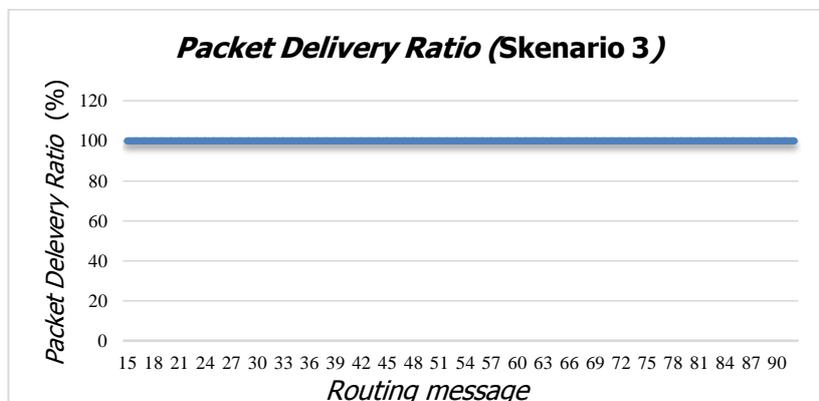
Stabilnya nilai PDR yang dihasilkan disebabkan oleh sistem melewati *hop* yang lebih sedikit, sehingga mekanisme dari sistem *testbed* masih dapat memberikan layanan yang baik. Adapun rata-rata nilai *throughput* yang dihasilkan pada pengujian skenario pertama yaitu sebesar 110,252 bps dengan nilai *jitter* minimal sebesar 2,13 *seconds*.

4.3.1.3 Hasil Pengujian Skenario 3

Pada skenario kedua, dapat dilihat performansi sistem *testbed* yang dihasilkan terhadap penambahan banyaknya pesan yang dikirimkan.



Gambar 15. Grafik Parameter Delay pada Skenario 3

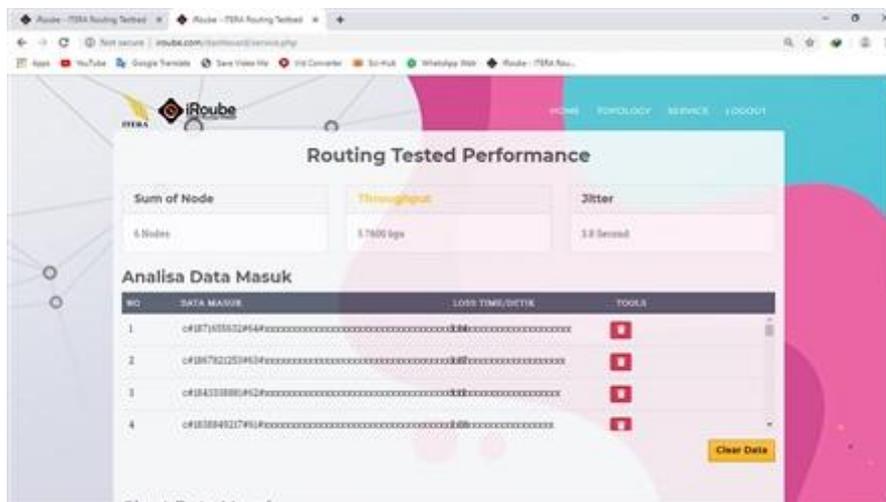


Gambar 16. Grafik Parameter *Packet Delivery Ratio* (PDR) pada Skenario 3

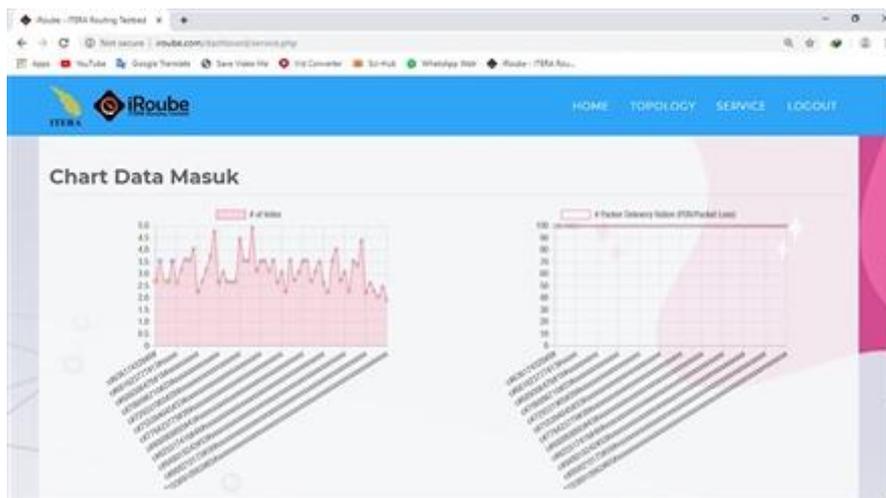
Gambar 15 dan 16 diatas merupakan grafik parameter *delay* dan *packet delivery ratio* (PDR) yang dihasilkan terhadap banyaknya pesan yang dikirimkan pada skenario pertama. Gambar 15 tersebut menunjukkan bahwa rata-rata *delay* pengiriman pada skenario pertama sebesar 5,53 *seconds*. Selain itu, *packet delivery ratio* (PDR) yang dihasilkan semuanya mencapai 100% untuk semua pengiriman, sehingga mekanisme dari sistem *testbed* masih dapat memberikan layanan yang baik. Adapun rata-rata nilai *throughput* yang dihasilkan pada pengujian skenario pertama yaitu sebesar 82,833 bps dengan nilai *jitter* minimal sebesar 2,86 *seconds*.

4.3.1.3 Visualisasi Parameter QoS

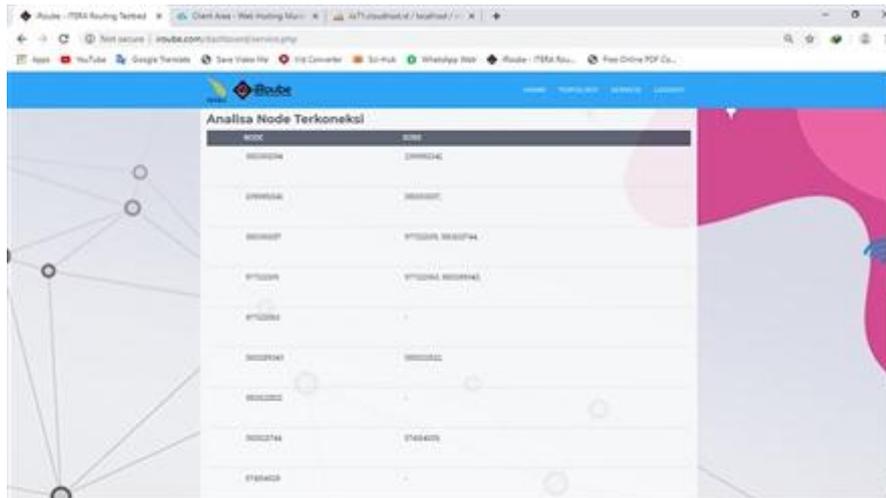
Visualisasi parameter QoS digunakan untuk memverifikasi data yang masuk pada *database* dan dapat dibentuk dari tabel ataupun kurva. Berikut merupakan hasil dari visualisasi parameter QoS pada *web server* yang ditunjukkan pada Gambar 17 dan 20 dibawah ini.



Gambar 17. Tampilan Performansi *Throughput* dan *Jitter*



Gambar 18. Tampilan Grafik *Delay* dan *Packet Delivery Ratio*



SRC	DST
192.168.1.1	192.168.1.2
192.168.1.2	192.168.1.1
192.168.1.1	192.168.1.3
192.168.1.3	192.168.1.1
192.168.1.1	192.168.1.4
192.168.1.4	192.168.1.1
192.168.1.1	192.168.1.5
192.168.1.5	192.168.1.1
192.168.1.1	192.168.1.6
192.168.1.6	192.168.1.1
192.168.1.1	192.168.1.7
192.168.1.7	192.168.1.1
192.168.1.1	192.168.1.8
192.168.1.8	192.168.1.1
192.168.1.1	192.168.1.9
192.168.1.9	192.168.1.1
192.168.1.1	192.168.1.10
192.168.1.10	192.168.1.1

Gambar 19. Tampilan *Routing Table*



Gambar 20. Tampilan Topologi Jaringan

5. KESIMPULAN

Berdasarkan proses analisis, implementasi, dan pengujian yang dilakukan didapatkan hasil bahwa *testbed* yang dibangun berhasil mensimulasikan *routing protocol* yang ada untuk melihat QoS yang dihasilkan oleh perangkat IoT. Hal ini dapat ditunjukkan dengan nilai rata-rata *throughput*, *delay*, *jitter*, *packet delivery ratio* secara *realtime* melalui sistem *monitoring* berbasis *web*.

DAFTAR RUJUKAN

- Barolli, L., Ikeda, M., Xhafa, F., & Duresi, A. (2010). A testbed for MANETs: Implementation, experiences and learned lessons. *IEEE Systems Journal*, 4(2), 243–252. <https://doi.org/10.1109/JSYST.2010.2047174>
- ETSI. (2002). Telecommunications and Internet Protocol Harmonization Over Networks

- (TIPHON) Release 3; End-to-end Quality of Service in TIPHON systems; Part 7: Design guide for elements of a TIPHON connection from an end-to-end speech transmission performance point of. In *Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON)* (Vol. 1).
- ITU. (2009). *Series G: Transmission System And Media, Digital Systems And Networks*.
- Kaysina, I. A., Vasiliev, D. S., Abilov, A., Meitis, D. S., & Kaysin, A. E. (2018). Performance evaluation testbed for emerging relaying and coding algorithms in Flying Ad Hoc Networks. *Moscow Workshop on Electronic and Networking Technologies, MWENT 2018 - Proceedings, 2018-March*(Figure 1), 1–5. <https://doi.org/10.1109/MWENT.2018.8337169>
- Kompella, S., & Ephremides, A. (2012). Stable throughput regions in wireless networks. *Foundations and Trends in Networking*, 7(4), 235–338. <https://doi.org/10.1561/13000000039>
- Lei, D., Wang, T., & Li, J. (2016). Performance analysis and comparison of routing protocols in Mobile Ad Hoc network. *Proceedings - 5th International Conference on Instrumentation and Measurement, Computer, Communication, and Control, IMCCC 2015*, (61272491), 1533–1536. <https://doi.org/10.1109/IMCCC.2015.325>
- Leon-Garcia, A., & Widjaja, I. (2004). *Communication Networks*. New York: Mc Graw Hill.
- Mitra, D., Sarkar, S., & Hati, D. (2016). A Comparative study of Routing Protocols. *International Research Journal of Advance Engineering and Science*, 2(1), 44–50. <https://doi.org/10.1109/ICTA.2015.7426884>