

# Simple LoRa Protocol: Protokol Komunikasi LoRa Untuk Sistem Pemantauan Multisensor

## Simple LoRa Protocol: LoRa Communication Protocol for Multisensor Monitoring Systems

Eko Didik Widiyanto<sup>1\*</sup>, Al Arthur Faizal<sup>2</sup>, Dania Eridani<sup>3</sup>, Richard Dwi Olympus Augustinus<sup>4</sup>, Michael SM Pakpahan<sup>5</sup>

Departemen Teknik Komputer Fakultas Teknik Universitas Diponegoro

Jalan Prof. Sudharto, Tembalang, Semarang, Indonesia

e-mail: didik@live.undip.ac.id<sup>1\*</sup>, aafaizal@student.ce.undip.ac.id<sup>2</sup>, dania@ce.undip.ac.id<sup>3</sup>, rdoaugustinus@ce.undip.ac.id<sup>4</sup>, msmpakpahan@ce.undip.ac.id<sup>5</sup>

**Abstrak** – Standar terbuka pada sistem komunikasi LoRa (Long Range) menyebabkan perbedaan cara implementasi dalam setiap penggunaannya. LoRaWAN (Long Range Wide Area Network) sebagai protokol bawaan LoRa masih mempunyai beberapa kekurangan yang menyebabkannya kurang efektif untuk diimplementasikan pada sistem tertentu. Penelitian ini mengembangkan protokol SLP (Simple LoRa Protocol) sebagai alternatif protokol komunikasi dan arsitektur jaringan yang didesain berdasarkan kekurangan yang dimiliki oleh LoRaWAN. Protokol ini mendefinisikan format data dan proses komunikasi antara client dan gateway dalam mode setup untuk pendaftaran node secara mandiri dan polling untuk transaksi data. Proses setup mampu mengenali node dengan konfigurasi sensor beragam dan dapat mengirimkan data dari node ke gateway dengan baik. Hasil pengujian menunjukkan bahwa SLP menghasilkan peningkatan performa QoS dalam throughput dan packet loss dari LoRaWAN menggunakan modulasi yang sama.

**Kata kunci:** LoRa; LoRaWAN; Sistem Komunikasi; Protokol; Arsitektur Jaringan

**Abstract** – Due to its open standard, LoRa (Long Range) communication system has a different approach on each of its implementation. LoRaWAN (Long Range Wide Area Network) as a default protocol for LoRa still has several shortcomings that made it less effective while implemented in a certain system. This study develops the SLP (Simple LoRa Protocol) protocol as an alternative communication protocol and network architecture designed based on the shortcomings of LoRaWAN. This protocol defines the data format and communication process between the client and gateway in setup mode for independent node registration and polling for data transactions. The setup process is able to recognize nodes with various sensor configurations and can send data from the node to the gateway properly. The tests show that SLP results in improved QoS performance in throughput and packet loss from LoRaWAN using the same modulation.

**Keywords:** LoRa, LoRaWAN, Communication System, Protocol, Network Architecture

## 1. Pendahuluan

LoRa merupakan sebuah sistem komunikasi *Low Power Wide Area Network* (LPWAN) yang memiliki kemampuan transmisi jarak jauh, didukung pengembangannya oleh IBM, Semtech, Actility, dll, yang tergabung dalam LoRa Alliance [1]. LoRa Alliance telah mendesain sebuah protokol bawaan untuk sistem komunikasi LoRa, disebut dengan LoRaWAN [2]. Protokol LoRaWAN telah memenuhi tujuan desain utama LPWAN yang diantaranya adalah jarak jauh, hemat biaya, rendah energi, skalabilitas tinggi, dan QoS (*Quality of service*) [3]. Namun, desain protokol LoRaWAN masih mempunyai beberapa kekurangan yang bisa dioptimalkan. Berdasarkan keterangan dari [1], *frame* data LoRa terdiri dari *preamble*, *header*, dan *payload* data yang masih kosong sebagai tempat bagi data yang akan dikirimkan. Belum terdapat format *payload* data spesifik yang implementasinya khusus untuk sistem tertentu. Pada [2] dipaparkan bahwa arsitektur jaringan LoRaWAN terdiri dari *end-node*, gateway, dan server, dimana koneksi gateway dan server menggunakan koneksi internet. Hal ini menjadikan LoRaWAN kurang efektif untuk dipasang di tempat terpencil yang tidak terdapat jaringan internet. Sementara [4] menyatakan bahwa LoRaWAN kurang sesuai untuk dipasangkan pada sistem real-time karena *air-time* dan kemungkinan tabrakan data yang tinggi.

Adelantado, dkk[4] telah meneliti batasan kemampuan LoRaWAN dan memberi kondisi ideal untuk beberapa kasus penggunaan. Lukas, Aditya dan Gunawan[5] membuat sebuah sistem monitoring bak penampungan air berbasis LoRa dengan Raspberry Pi. D. H. Kim, Lim and J. D. Kim[6] membuat sistem monitoring berbasis LoRa dan WiFi. Pakpahan, Widiyanto, dan Septiana[7] membuat sistem monitoring limbah batik berbasis LoRa, dan membuat database berbasis HTTP request. LoRa Alliance Inc [8] menjabarkan spesifikasi LoRa secara menyeluruh.

Penelitian sebelumnya, seperti pada [5], [6], [7], dan [9] berhasil membuat sistem monitoring berbasis LoRa, namun masih membutuhkan jaringan komunikasi lain untuk dapat mengolah dan menyajikan data. Sementara [4], [8], [10], [12] melakukan penelitian tentang LoRa dan menghasilkan beberapa informasi penting yang bisa diambil dan hasil pengujiannya dapat digunakan sebagai perbandingan.

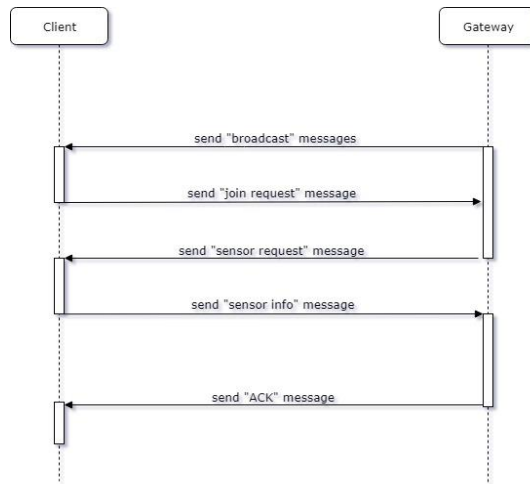
Penelitian ini bertujuan untuk mengembangkan sebuah protokol komunikasi LoRa sebagai alternatif dari LoRaWAN. Protokol ini dirancang spesifik untuk sistem monitoring multisensor tanpa menggunakan internet di dalam arsitekturnya, yang diberi nama SLP (*Simple LoRa Protocol*). Protokol ini mendefinisikan format data dan proses komunikasi antara client (node) dan gateway dalam jaringan LoRa.

## 2. Metode Penelitian

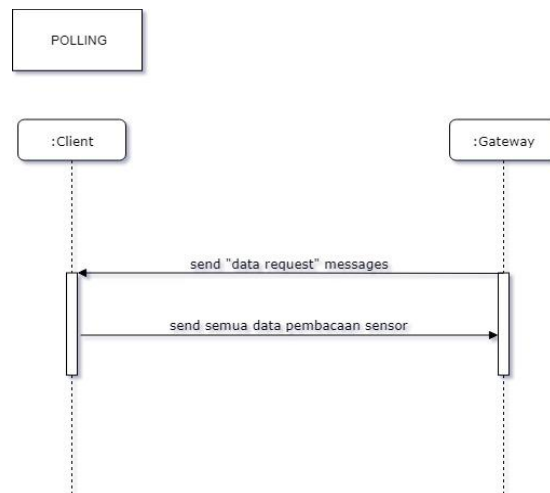
Tujuan desain utama dari protokol SLP adalah mampu melakukan komunikasi antara gateway dan client yang dipasangkan dengan beberapa sensor melalui komunikasi LoRa, tanpa koneksi internet. Maka dari itu, beberapa hal utama yang harus didesain adalah sekuen komunikasi yang memungkinkan sistem untuk melakukan komunikasi dengan lancar, format *payload* data yang dilengkapi dengan format id untuk setiap pesan yang terdapat di sekuen, serta implementasi dan pengujian sistem.

### 2.1 Sekuen Komunikasi

Sekuen komunikasi SLP mengatur urutan pesan yang dikirim antara client dan gateway. Proses komunikasi dibagi menjadi 2 mode, yaitu *setup* dan *polling*. Mode *setup* merupakan proses pengenalan client yang akan masuk ke jaringan gateway, sementara mode *polling* merupakan proses pengambilan data dari client oleh gateway. Di dalam kedua mode tersebut, gateway dan client saling bertukar pesan untuk berkomunikasi. Gambar 1 menunjukkan diagram sekuen dari mode *setup*, sementara Gambar 2 menunjukkan diagram sekuen untuk mode *polling*.



Gambar 1. Diagram sekuen mode *setup*.



Gambar 2. Diagram sekuen mode *polling*.

Dalam proses komunikasi, terdapat kemungkinan terjadinya tabrakan data saat pengiriman. Maka dari itu dibuat sebuah desain sistem *state* untuk *client* yang disebut dengan *4-state-handshake*. Sistem ini dibuat untuk mengatasi perbedaan kemampuan antar client dalam menerima perintah *gateway* yang kadang menyebabkan komunikasi *client* terhenti pada langkah tertentu, yang pada akhirnya mengakibatkan tabrakan data. Tabel 1 menunjukkan detail dari sistem *4-state-handshake*.

Tabel 1. State client dalam sistem *4-state-handshake*.

State Client	Kondisi	
	Client	Gateway
1	Menerima pesan <i>broadcast</i> , mengirim pesan <i>join request</i>	Mengirim pesan <i>broadcast</i>
2	Menerima pesan <i>sensor request</i> , Mengirim pesan <i>sensor information</i>	Menerima pesan <i>join request</i> , mengirim pesan <i>sensor request</i>
3	Menerima pesan <i>join ack</i>	Menerima pesan <i>sensor information</i> , mengirim pesan <i>join ack</i>
4	Memasuki mode <i>polling</i>	Memasuki mode <i>polling</i>

2.2 Format ID dan Pesan

Format id dibuat untuk memberi identitas terhadap alat yang terpasang pada sistem, serta dijadikan sebagai alamat pengiriman dalam pesan. Terdapat 3 format id yang dibuat, yaitu id *gateway*, *client*, dan sensor. Setiap id terdiri dari 4 bytes, mengadaptasi format id *internet protocol* (IP) yang juga terdiri dari 4 bytes. Gambar 3, 4, dan 5 menunjukkan format id yang terdapat di SLP.

MSB			LSB
0-255	0-255	0-255	0-255

Gambar 3. Format ID *client*.

MSB			LSB
0-255	0-255	0-255	0-255

Gambar 4. Format ID *gateway*.

MSB			LSB
ID MEREK		ID SENSOR	
0-255	0-255	0-255	0-255

Gambar 5. Format ID sensor.

Format pesan dibuat untuk tiap-tiap pesan yang terdapat di sekuen komunikasi. Format pesan ini merupakan modifikasi dari *payload data* dalam frame pesan LoRa yang masih kosong. Format pesan akan mengatur kode yang digunakan untuk setiap jenis pesan, isi pesan, dan panjang byte yang digunakan untuk setiap pesannya. Isi dari tiap byte format pesan SLP menggunakan format kode ASCII (bernilai 0-255). Total terdapat 7 pesan dalam sekuen mode *setup* dan *polling*. Jenis pesan tersebut adalah *broadcast*, *join request*, *sensor request*, *sensor information*, *ACK*, dan *data request*. Setiap pesan terdiri dari 2 byte kode perintah yang menunjukkan jenis pesan, 4 byte id *gateway* dan 4 byte id *client* sebagai alamat pengiriman pesan, dan bit sisanya merupakan isi pesan tersebut. Gambar 6 dan Gambar 7 menunjukkan format pesan yang terdapat di *gateway* dan *client*.

B	C	:	Gateway id (4 bytes)	
S	R	:	Client id (4 bytes)	Gateway id (4 bytes)
J	A	:	Client id (4 bytes)	Gateway id (4 bytes)
D	R	:	Client id (4 bytes)	Gateway id (4 bytes)

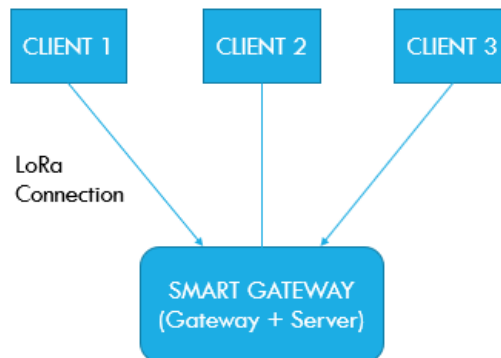
Gambar 6. Format pesan pada *gateway*.

D	S	:	Gateway id (4 bytes)	Client id (4 bytes)	*	Sensor Data#1 (4 bytes)	*	Sensor Data#... (4 bytes)
J	R	:	Gateway id (4 bytes)	Client id (4 bytes)				
S	I	:	Gateway id (4 bytes)	Client id (4 bytes)	*	Sensor id #1 (4 bytes)	*	Sensor id #... (4 bytes)

Gambar 7. Format pesan pada *client*.

### 2.3 Arsitektur Jaringan

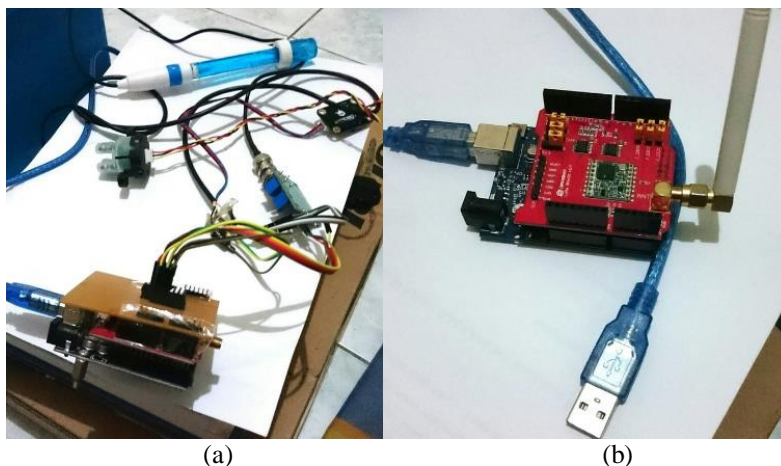
Arsitektur jaringan SLP dibuat berdasarkan dari kekurangan yang masih dimiliki oleh desain arsitektur LoRaWAN, yaitu menggunakan koneksi internet. Arsitektur jaringan SLP terdiri dari *smart gateway* dan beberapa *client* yang telah dipasangi dengan sensor. Satu *client* dapat melayani maksimal 3 sensor. Gambar 8 menunjukkan bagan arsitektur jaringan SLP.



Gambar 8. Arsitektur jaringan SLP.

### 2.4 Implementasi

Implementasi perangkat lunak sistem menggunakan Arduino IDE 1.8.8. Sementara untuk implementasi perangkat keras, papan rangkaian yang digunakan sebagai *gateway* adalah Arduino Uno, dan untuk *client* menggunakan dua jenis papan berupa Arduino Uno dan Dragino Yun. Baik *gateway* maupun *client* dilengkapi dengan LoRa Shield v1.4 untuk melakukan komunikasi LoRa. Gambar 9 menunjukkan bentuk fisik *client* dan *gateway*.



Gambar 9. Implementasi sistem: (a) Bentuk fisik *client*, (b) Bentuk fisik *gateway*.

### 3. Hasil dan Pembahasan

Untuk memastikan sistem dapat bekerja dengan baik dan mengetahui batasan kemampuan sistem, dilakukan 3 jenis pengujian, yaitu pengujian perangkat lunak (*white box*), pengujian fungsionalitas sistem, dan pengujian QoS. Perangkat yang digunakan untuk pengujian terdiri dari 1 *gateway* dan 4 *client* yang terdiri dari 2 Arduino UNO yang masing-masing dipasang 3 sensor, dan 2 Dragino Yun yang tidak dipasangkan dengan sensor. Penggunaan jaringan komunikasi LoRa menggunakan LoRa Shield v1.4 yang dipasangkan pada setiap *client* dan *gateway*. Modulasi LoRa yang digunakan dapat dilihat pada Tabel 2.

Tabel 2. Modulasi LoRa pada pengujian.

Parameter	Value
<i>ISM Band Frequency</i>	868.0 Mhz
<i>Transmitting Power</i>	14dbm
<i>Spreading Factor</i>	7
<i>Bandwidth</i>	125 kHz
<i>Code Rate</i>	4/5
<i>Jarak Komunikasi</i>	1m

### 3.1 Hasil Pengujian White Box

Pengujian *white box* dibagi menjadi dua, yaitu pengujian percabangan kode program dimana setiap kemungkinan yang bisa menjadi luaran dari fungsi akan dicoba satu persatu. Pengujian *test case* dimana sistem diuji untuk berjalan dalam beberapa kondisi nyata yang mungkin terjadi pada saat implementasi sebenarnya.

Pengujian percabangan kode program dilakukan pada 11 fungsi utama SLP. Sementara pengujian *test case* dilakukan dengan 4 contoh kasus spesial yang mungkin terjadi pada saat implementasi. Tabel 3 menunjukkan hasil pengujian kasus (*test case*).

Tabel 3. Hasil pengujian kasus.

No	Kasus	Keluaran
1	Id_gateway dan/atau id_client yang terdapat di pesan tidak sesuai	<i>Gateway/client</i> yang menerima tidak akan memproses pesan tersebut, lalu melanjutkan ke langkah berikutnya
2	Ketika proses setup dilaksanakan, <i>client</i> keluar dari jaringan ( <i>offline</i> )	Proses setup berhenti untuk <i>client</i> , lalu <i>gateway</i> meneruskan proses <i>setup</i> untuk <i>client</i> yang lain. Setelah kembali <i>online</i> , <i>client</i> akan mengulang proses <i>setup</i> dari awal.
3	Ketika proses <i>setup</i> dilaksanakan, <i>gateway</i> <i>offline</i>	<i>Client</i> akan menunggu sampai timeout (30 detik) terhitung dari komunikasi terakhir yang diterima dari <i>gateway</i> , jika melebihi dari itu, <i>client</i> akan kembali ke kondisi 0.
4	<i>Client</i> menerima <i>broadcast</i> dari <i>gateway</i> yang RSSI nya lebih bagus dari <i>gateway</i> tempat <i>client</i> sekarang.	<i>Client</i> akan menerima pesan <i>broadcast</i> tersebut dan berpindah ke <i>gateway</i> yang baru

Dari hasil pengujian *white box*, protokol SLP dapat menjalankan semua fungsi dan kode pemrograman yang ada dengan baik. Tidak ditemukan masalah ketika dilakukan pengujian percabangan program maupun *test case*, semua kondisi dan masalah yang mungkin terjadi dalam proses komunikasi dapat diselesaikan oleh SLP dengan baik.

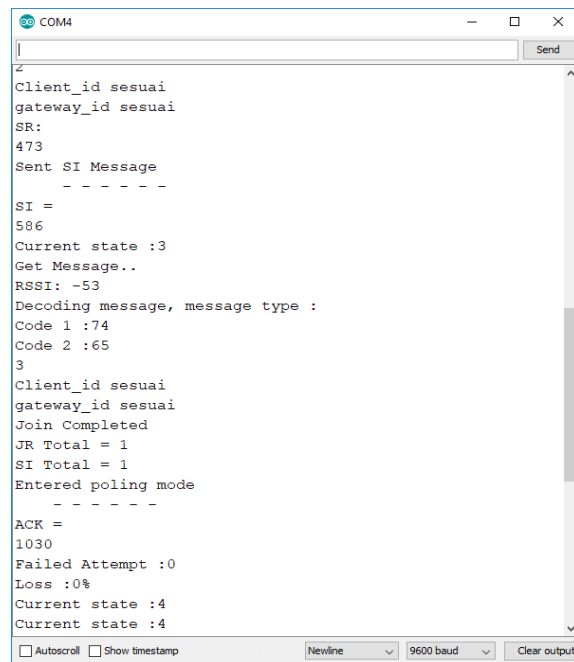
### 3.2 Hasil Pengujian Fungsionalitas Protokol

Pengujian fungsionalitas protokol dibagi menjadi 2, pertama adalah pengujian sekuen komunikasi yang dilakukan untuk menguji apakah sistem mengerjakan sekuen denganurut. Pengujian yang kedua adalah pengujian pengiriman data yang dilakukan untuk menguji keberhasilan sistem dalam mengirim data. Keluaran yang diharapkan dari pengujian ini adalah sistem dapat melakukan proses komunikasi dengan baik. Gambar 10 dan 11 menunjukkan hasil pengujian sekuen komunikasi SLP.



```
COM4
Current state :0
Get Message..
RSSI: -55
Decoding message, message type :
Code 1 :66
Code 2 :67
1
BC
-55
Gateway ID :fz1s
Sent a Join Request Message
-----
JR = 182
Current state :2
Get Message..
RSSI: -53
Decoding message, message type :
Code 1 :83
Code 2 :82
2
Client_id sesuai
gateway_id sesuai
SR:
473
Sent SI Message
-----
SI =
586
Current state :3
Autoscroll Show timestamp Newline 9600 baud Clear output
```

Gambar 10. Hasil percobaan sekuen komunikasi (1).



```
COM4
Client_id sesuai
gateway_id sesuai
SR:
473
Sent SI Message
-----
SI =
586
Current state :3
Get Message..
RSSI: -53
Decoding message, message type :
Code 1 :74
Code 2 :65
3
Client_id sesuai
gateway_id sesuai
Join Completed
JR Total = 1
SI Total = 1
Entered poling mode
-----
ACK =
1030
Failed Attempt :0
Loss :0%
Current state :4
Current state :4
Autoscroll Show timestamp Newline 9600 baud Clear output
```

Gambar 11. Hasil percobaan sekuen komunikasi (2).

Pengujian sekuen komunikasi dilaksanakan menggunakan 1 *gateway* dan 1 *client* pada mode *setup* saat *client* berada di tahap *4-state-handshake*. Urutan sekuen pesan yang terjadi adalah : BC (*broadcast*) – JR (*join request, state client = 2*) - SR (*sensor request*) – SI (*sensor information – state client = 3*) – ACK (*acknowledgement – state client = 4*). Hasil tersebut menunjukkan kesesuaian dengan *sequence diagram* yang sudah dibuat (Gambar 6). Sistem dapat mengeksekusi sekuensi komunikasi dengan baik.

```

CLIENT start:
Frequency : 868.00
Client ID : abcd{□s□
Sensor connected: 4
slot_memori: 0
data_sensor: 121
- - - - -
slot_memori: 1
data_sensor: 155
- - - - -
slot_memori: 2
data_sensor: 1187
- - - - -
slot_memori: 3
data_sensor: -127
- - - - -

```

Gambar 12. Hasil percobaan ketepatan data (1).

```

COM9
DS :1187
DS :-127
Request Data from client ID: 979899100
Get incoming message
DS :121
DS :155
DS :1187
DS :-127

```

Gambar 13. Hasil percobaan ketepatan data (2).

Pengujian pengiriman data dilaksanakan menggunakan 1 *gateway* dan 1 *client* pada mode *polling*. Gambar 12 merupakan potongan tangkapan layar dari *serial monitor client* yang menunjukkan informasi tentang data hasil pembacaan sensor yang akan dikirim ke *gateway*. Data yang tercantum adalah 121, 155, 1187, dan -127. Gambar 13 merupakan *screenshot* dari *serial monitor gateway* menunjukkan data yang dikirim dari *client*. Data sensor yang ditampilkan oleh *gateway* adalah 121, 155, 1187, dan -127, sama seperti yang tercantum pada *client*. Pengujian dilanjutkan sampai *client* berhasil mengirimkan 20 paket terkirim dengan tingkat kesesuaian 100%. Kesesuaian data yang terkirim ke *gateway* membuktikan bahwa sistem dapat melakukan pengiriman data dengan benar.

### 3.3 Hasil Pengujian Quality of Service

Pengujian ini dilaksanakan untuk mengukur kemampuan SLP dalam melakukan komunikasi data. Parameter *QoS* yang diujikan adalah *throughput* dan *packet loss* yang termasuk dalam prioritas utama *QoS* dalam sistem yang mempunyai kegunaan untuk transfer data [11]. Persamaan yang digunakan untuk menghitung *throughput* dan *packet loss* dapat dilihat pada Persamaan 1 dan 2.

$$T = \frac{Dt}{w} \quad (1)$$

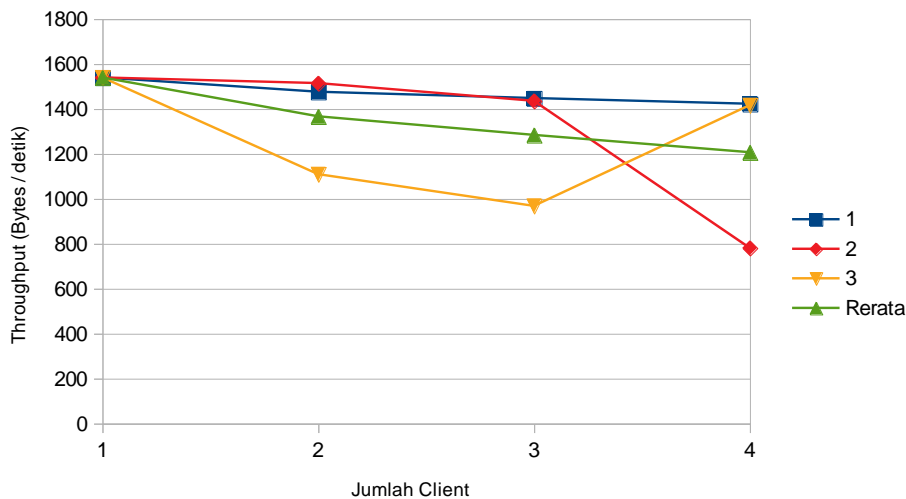
T : *Throughput*  
 Dr : Data terkirim  
 w : Waktu *Throughput*

$$PL = \frac{(Pt - Pd)}{Pt} \times 100\% \quad (2)$$

PL : *Packet Loss*  
 Pt : Paket terkirim  
 Pd : Paket diterima

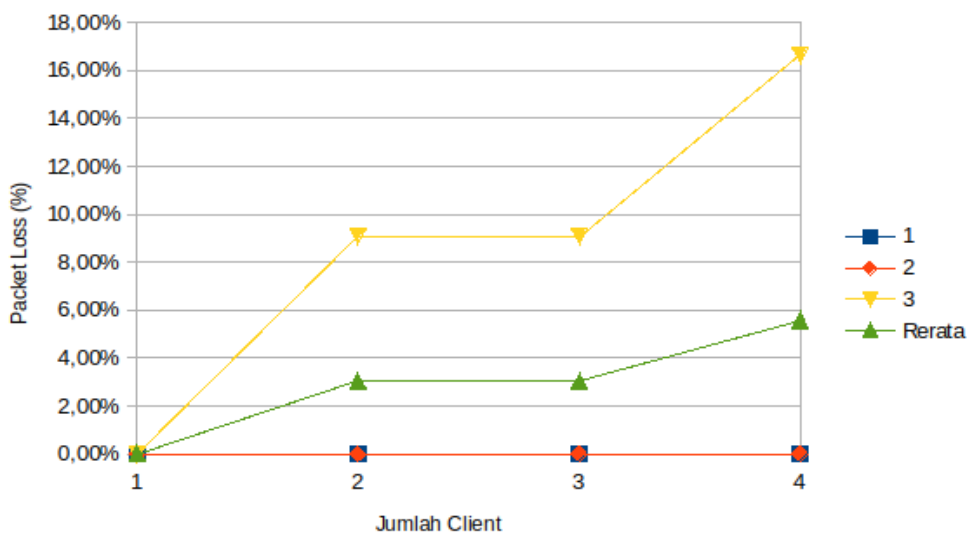


Penghitungan *throughput* dilakukan sebanyak 3 kali pada mode *polling*, dengan mengirimkan 10 paket data yang masing-masing berukuran 62 Byte (4960 bit). Rerata hasil pengujian disajikan dalam bentuk grafik pada Gambar 14. Dari 3 kali pengujian dan 4 konfigurasi jumlah *client*, *throughput* rata-ratanya adalah 1348,87 Bps. *Throughput* cenderung menurun sebesar 6% dengan bertambahnya jumlah *client*. Hasil ini menunjukkan peningkatan dari *throughput* protokol LoRaWAN yang dihitung oleh [12] dengan modulasi yang sama dengan hasil 484.47 Bps. Hal ini disebabkan karena komunikasi SLP bersifat sinkron dan dilakukan terus-menerus, sedangkan LoRaWAN menggunakan komunikasi asinkronus dan menerapkan batasan maksimal pengiriman data *client* yang disebut *duty-cycle*.



Gambar 14. Grafik *throughput* dari 3 pengujian.

Penghitungan *packet loss* juga dilakukan pada waktu yang bersamaan dengan pengujian *throughput*. Hasil pengukuran *packet loss* SLP disajikan dalam bentuk grafik pada Gambar 15. Dari 3 percobaan dengan 4 variasi jumlah *client*, hasil pengukuran tersebut menunjukkan menghasilkan rerata sebesar 2,9% yang juga meningkat dibandingkan dengan implementasi LoRa yang diterapkan pada [10] dengan hasil 4%. *Packet loss* cenderung meningkat dengan bertambahnya jumlah *client*, walaupun di dua percobaan pertama tidak ada *packet loss*.



Gambar 15. Grafik *packet loss* dari 3 pengujian.

#### 4. Kesimpulan

Protokol SLP yang dikembangkan telah mampu melakukan setup koneksi dari node dengan beragam sensor dan melakukan transaksi data dengan gateway dengan baik. Dengan menggunakan modulasi yang sama, SLP yang bersifat sinkron dapat menghasilkan throughput dan packet loss yang lebih baik daripada sistem pembandingan dalam referensi.

#### Ucapan Terima Kasih

Penelitian ini didanai oleh Kementerian Riset, Teknologi, dan Pendidikan Tinggi melalui skema Penelitian Dasar Unggulan Perguruan Tinggi dengan nomor kontrak 101-37/UN7.P4.3/PP/2019.

#### Referensi

- [1] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A Study of LoRa: Long Range & Low Power Networks for the Internet of Things," *Sensors*, vol. 16, no. 9, 2016.
- [2] LoRa Alliance Technical Marketing Workgroup, "A technical overview of LoRa and LoRaWAN," November, 2015.
- [3] U. Raza, P. Kulkarni and M. Sooriyabandara, "Low Power Wide Area Networks: An Overview," in *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 855-873, Secondquarter 2017.
- [4] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui and T. Watteyne, "Understanding the Limits of LoRaWAN," in *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34-40, Sept. 2017. doi: 10.1109/MCOM.2017.1600613XLukas, Lukas & Aditya Tanumihardja, Wisena & Gunawan, Edy. (2015). On the Application of IoT: Monitoring of Troughs Water Level Using WSN. 10.1109/ICWISE.2015.7380354.
- [5] D. H. Kim, J. Y. Lim and J. D. Kim, "Low-Power, Long-Range, High-Data Transmission Using Wi-Fi and LoRa," 2016 6th International Conference on IT Convergence and Security (ICITCS), Prague, 2016, pp. 1-3.
- [6] S.M. Pakpahan, Michael & Widiyanto, Eko Didik & Septiana, Risma. (2018). Analysis on Batik Water Waste Monitoring System based on LoRa Communication. 171-175. 10.1109/ICITACEE.2018.8576954.
- [7] LoRa Alliance Inc, "LoRa Specification", 2015
- [8] D.Bahrin, "Sistem Kontrol Penerangan Menggunakan Arduino Uno Pada Universitas Ichsan Gorontalo", *Ilkom*, no.3 vol.9 (2017).
- [9] Widiyanto, Eko Didik & S.M. Pakpahan, Michael & Arthur Faizal, Al & Septiana, Risma. (2018). LoRa QoS Performance Analysis on Various Spreading Factor in Indonesia. 1-5. 10.1109/ISESD.2018.8605471.
- [10] M. Z. Chowdhury et al., "Characterizing QoS Parameters and Application of Soft-QoS Scheme for 3G Wireless Networks," 2008 10th International Conference on Advanced Communication Technology, Gangwon-Do, 2008, pp.760-764. doi: 10.1109/ICACT.2008.4493867
- [11] Petajarvi, Juha & Mikhaylov, Konstantin & Pettissalo, Marko & Janhunen, Janne & Inatti, Jari. (2017). Performance of a low-power wide-area network based on LoRa technology: Doppler robustness, scalability, and coverage. *International Journal of Distributed Sensor Networks*. Vol. 13. 1-16. 10.1177/1550147717699412.