Analisis Performansi dan Simulasi *Security Protocol* TinySec dan LLSP pada *Wireless Sensor Network*

Simulation of Analytical Performance between TinySec and LLSP in Wireless Sensor Network

Ryandito Danuansa[1], Fazmah Arif Yulianto[2], Sidik Prabowo[3]
[123]Prodi S1 Teknik Informatika, Fakultas Informatika, Universitas Telkom
[1]rdnuansa@telkomuniversity.ac.id [2]fazmaharif@telkomuniversity.ac.id [3]pakwowo@telkomuniversity.ac.id

Abstrak

Di zaman sekarang, jaringan sensor nirkabel atau *Wireless Sensor Network* (WSN) sudah mulai banyak diterapkan di berbagai bidang kehidupan. WSN terbentuk dari beberapa perangkat *node* sensor yang saling terhubung melalui jaringan nirkabel (*wireless*) dan dapat saling bertukar data secara real-time. Ukurannya yang kecil mengakibatkan perangkat sensor memiliki keterbatasan sumber daya, terutama dalam menjamin aspek keamanan seperti *confidentiality, integrity* dan *authenticity*. Pada penelitian ini digunakan dua buah protokol keamanan, yaitu TinySec dan Link-Layer Security Protocol (LLSP), dan dilakukan uji untuk dibandingkan tingkat performansi antara kedua protokol tersebut. Parameter performansi yang menjadi tolak ukur adalah konsumsi energi, *confidentiality*, *integrity* dan *authentication*. Dari hasil pengujian yang dilakukan, protokol LLSP dapat menghemat konsumsi energi hingga 15% dari protokol TinySec, karena adanya perbedaaan panjang byte untuk melakukan operasi keamanan yang dibutuhkan

Kata kunci: ***Wireless sensor network (WSN), Security protocol*, TinySec, *Link-Layer Security Protocol (LLSP)*.**

Abstract

In the current era, Wireless Sensor Network (WSN) has started being implemented in various areas of life. WSN is formed from multiple sensor nodes that are connected via a wireless network (wireless) and can exchange data in real-time. The sensor node is small sized, so it has a limited resources to do the computation process, especially in ensuring the security aspects such as confidentiality, integrity and authenticity. This research will use two security protocols, TinySec and Link-Layer Security Protocol (LLSP), and will be tested against each other's to compare the performance level of each protocol. The performance parameters that will be used are energy consumption, confidentiality, integrity and authentication. Based on system testing, LLSP can reduce energy consumption up to 15% compared to TinySec. It's caused by a differences in byte length for overhead communication.

Keywords: Wireless sensor network (WSN), Security protocol, TinySec, Link-Layer Security Protocol (LLSP).

1. Pendahuluan

Akhir-akhir ini jaringan sensor nirkabel atau Wireless Sensor Network (WSN) sudah mulai banyak diterapkan di berbagai bidang kehidupan. Hal itu dikarenakan kemampuan sensor yang saat ini terus dikembangkan. Sejauh ini sensor yang digunakan dalam WSN mampu mendeteksi kondisi fisik (keberadaan, suara, getaran, dsb) dan lingkungan (suhu, kelembaban, tekanan, dsb). WSN merupakan jaringan wireless (nirkabel) yang terdiri dari sejumlah node-node/device yang dapat saling memproses dan bertukar data secara real time. Umumnya node tersebut berupa sensor kecil yang dapat mendeteksi dan memonitor kondisi di lingkungan sekitarnya. Karena itulah karakteristik utama yang dibutuhkan dari WSN antara lain adalah konsumsi energi dan biaya yang rendah pada node-nya.

Selain itu, yang menjadi isu lainnya pada WSN adalah tingkat keamanan. Perangkat yang tidak terawat atau lingkungan yang kurang pengawasan dapat menjadi celah bagi attacker untuk melakukan serangan. WSN juga tidak terbatas oleh fisik karena transmisinya menggunakan broadcast medium, sehingga membuatnya lebih rentan lagi terhadap berbagai serangan [1]. Dari celah tersebut attacker dapat menyadap dan mengubah-ubah data yang sedang ditransmisikan.

Karena itulah perlu adanya security requirements untuk menangani isu-isu keamanan tersebut. Requirements yang perlu diberikan oleh WSN antara lain adalah data integrity, authentication, dan data confidentiality [2] [3]. Layanan tersebut disediakan oleh berbagai protokol keamanan yang dirancang untuk WSN. Salah satunya yaitu TinySec. TinySec adalah data link layer security protocol pertama yang diimplementasikan dalam WSN [4].

Sedangkan protokol LLSP merupakan pengembangan dari TinySec. Kedua protokol tersebut didesain sebagai protokol yang ringan dan hemat energi, namun tetap dapat memberikan level keamanan yang baik.

Pada penelitian ini, akan dilakukan perbandingan dua buah *security protocol*, yaitu TinySec dan LLSP, ke dalam beberapa skenario kasus umum yang terdapat pada WSN. *Security protocol* LLSP dapat memberikan keamanan yang setara atau lebih baik dari TinySec namun dengan konsumsi energi yang lebih rendah.

2. Dasar Teori

2.1. *Wireless Sensor Network* (WSN)

Wireless Sensor Network (WSN) adalah sebuah jaringan yang terdiri dari sekumpulan *node-node* sensor yang memiliki kemampuan untuk mendeteksi suatu keadaan di sekitarnya, memproses data yang diperoleh, dan saling berkomunikasi satu sama lain untuk mengirimkan data ke node pusat [5]. Pada umumnya WSN digunakan untuk memantau keadaan suatu lingkungan, seperti suhu, suara, tekanan, dll. WSN mulai banyak diterapkan karena beberapa keunggulannya, antara lain yaitu rendah daya atau energi, skalabilitas yang baik, mudah diimplementasikan, ukuran yang kecil, dan sebagainya. Tetapi WSN juga memiliki masalah pada keamanan dan rentan terhadap serangan yang dilakukan pada jaringan, seperti sniffing attack atau man-in-the-middle. Untuk menangani masalah tersebut dapat dilakukan dengan mengimplementasikan protokol keamanan atau security protocol. Namun karena ukurannya yg kecil WSN memiliki keterbatasan resource atau sumber daya (seperti memori, energi, processor), sehingga untuk melakukan proses komputasi yang kompleks akan membutuhkan energi yang besar pula. Hal ini lah yang menjadi permasalah utama dalam WSN. Untuk itu *security protocol* dituntut untuk dapat memberikan kemanan yang baik namun dengan konsumsi energi yang rendah.

2.2. Protokol Keaman TinySec

TinySec merupakan protokol keamanan yang rendah energi dan ditujukan untuk diimplementasikan pada perangkat sensor yang memiliki keterbatasan resource. Fokus dari TinySec adalah menjamin keamanan pada authenticity, integrity dan confidentiality. Untuk menangani authenticity dan integrity TinySec menggunakan metode Cipher Block Chaining dengan Message Authentication Code (CBC-MAC), sedangkan untuk menangani confidentiality digunakan metode enkripsi CBC-RC5/Skipjack [6] [7].

Untuk menangani *overhead* maka MAC yang biasanya memiliki panjang 8 atau 16 byte dimodelkan ke dalam panjang 4 byte. Untuk model jaringan konvensional, panjang ini memang tidak cukup dalam menangani security, namun untuk jaringan sensor hal ini sudah cukup memadai [6]. TinySec memiliki *initialization vector* (IV) sepanjang 8-byte, yang terdiri dari *Dest,* yaitu alamat tujuan, AM, yaitu *Active Message, Len,* yaitu panjang data, *Src,* yaitu alamat sumber/pengirim dan *Ctr,* yaitu sebuah *counter value*..

2.3. *Link-Layer Security Protocol* (LLSP)

Protocol keamanan LLSP merupakan pengembangan dari protokol TinySec. LLSP dirancang dengan tujuan untuk menjadi protokol yang dapat mengonsumsi energi lebih hemat dari TinySec. LLSP dapat mengurangi konsumsi energi dengan cara meminimalkan *security overhead* pada setiap paket. Sama seperti TinySec, LLSP juga memastikan keamanan pada *authentication, confidentiality* dan *integrity*. Namun ada satu keunggulan pada LLSP yang tidak dimiliki TinySec yaitu adalah *replay protection. Replay protection* dapat mencegah pesan-pesan lama terkirim kembali, sehingga pesan yang diterima adalah pesan yang benar-benar baru dan dapat mempertahankan *data freshness*. Untuk menangani permasalahan pada aspek *confidentiality* digunakan metode enkripsi *Advance Encryption Standard with chipper blok chaining* (AES-CBC). Sedangkan untuk *authentication* digunakan metode MAC, dan untuk aspek *integrity* adalah CBC-MAC. Pada *replay protection*, metode yang digunakan adalah dengan mempertahankan *counter* 4-*byte* antara *sender* dan *receiver. Feedback shift register* (FSR) digunakan untuk mengupdate 4-*byte counter* tersebut [1].

Pada struktur format paket data LLSP, seperti yang ditunjukkan oleh **Error! Reference source not found.**, tidak terdapat 2-byte *counter* seperti yang ada pada TinySec, namun *Ctr* akan ditambahkan dalam perhitungan MAC. Struktur IV dari LLSP, yaitu *Dest,* AM, Len, Src. Nilai *Ctr* termasuk ke dalam IV, kegunaannya agar menambah variasi pada enkripsi, sehingga mengurangi kemungkinan IV yang berulang [1]. Pada paket LLSP data *payload* juga dienkripsi, namun *header* paket (*Dest, AM, Len, Src*) tidak dienkripsi.

Untuk melakukan perhitungan MAC pada LLSP digunakan persamaan 1 sebagai berikut [1].

$$MAC = H(K, Dest||AM||Len||Src||Ctr||Data) \quad (1)$$

Dest adalah alamat identitas tujuan, AM adalah tipe *message handler*, Len adalah panjang data, Src adalah alamat sumber atau pengirim, Ctr adalah nilai *counter*, dan Data informasi yang dimiliki sensor.

2.4. RC5

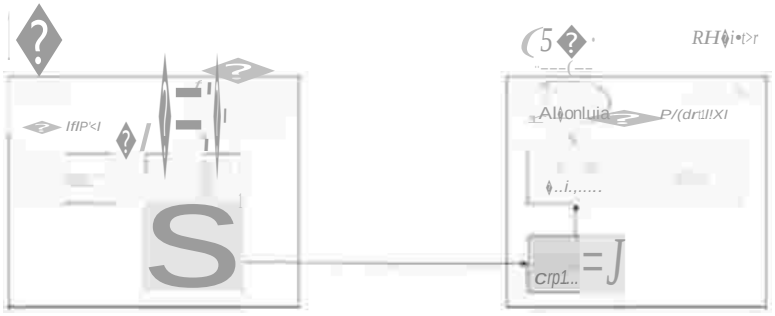
RC5 adalah algoritma *block cipher* yang diciptakan oleh Ronald L. Rivest. RC5 merupakan *symmetric block cipher* [8], yang berarti pada saat melakukan proses enkripsi dan dekripsinya menggunakan *secret cryptographic key* yang sama. *Plaintext* dan *ciphertext* pada RC5 memiliki ukuran panjang yang tetap dalam bentuk *bit sequence* (*block*). RC5 memiliki ukuran *block* 32, 64, atau 128 bit, dan ukuran *key* antara 0 sampai 2040 bit, serta jumlah *round*-nya antara 0-255. Algoritma RC5 terbagi menjadi tiga bagian, yaitu *key expansion*, enkripsi, dan dekripsi [8].

2.5. *Advanced Encryption System* (AES)

Sebelumnya algoritma AES diberi nama Rijndael yang merupakan kombinasi dari kedua nama penciptanya asal Belgia, yaitu Vincent Rijmen dan Joan Daemen. Setelah ditetapkan sebagai standar algoritma enkripsi oleh *Federal Information Processing Standards Publication (FIPS)*, algoritma ini baru diberi nama *Advanced Encryption Standard*. AES adalah sebuah *symmetric block cipher* yang dapat memproses blok data 128 bit, menggunakan *cipher keys* dengan panjang 128, 192, dan 256 bit. Karena dapat menggunakan tiga *key* yang berbeda maka algoritma ini dikenal juga dengan "AES-128", "AES-192", dan "AES-256" [9]. AES memiliki 9, 11, atau 13 *round* dalam satu kali prosesnya, baik enkripsi maupun dekripsi. Pada setiap *round*-nya terdiri dari beberapa tahap, yaitu *substitution byte, shift rows, mix column,* dan *add round key*.

3. Perancangan Sistem

Kedua buah protokol, yaitu TinySec dan LLSP, akan diimplementasikan dan diuji ke dalam beberapa skenario pengujian. Pada skenario-skenario tersebut digunakan beberapa parameter yang akan dijadikan nilai ukur dalam menentukan tingkat performansi. Pengujian sistem dilakukan pada dua aspek, yaitu dari segi aspek keamanan dan energi. Cara kerja kedua protokol secara garis besar dapat diilustrasikan pada Gambar 3-1. Output yang didapatkan dari setiap skenario untuk masing-masing protokol akan dibandingkan untuk menentukan protokol keamanan mana yang memikili performansi terbaik.

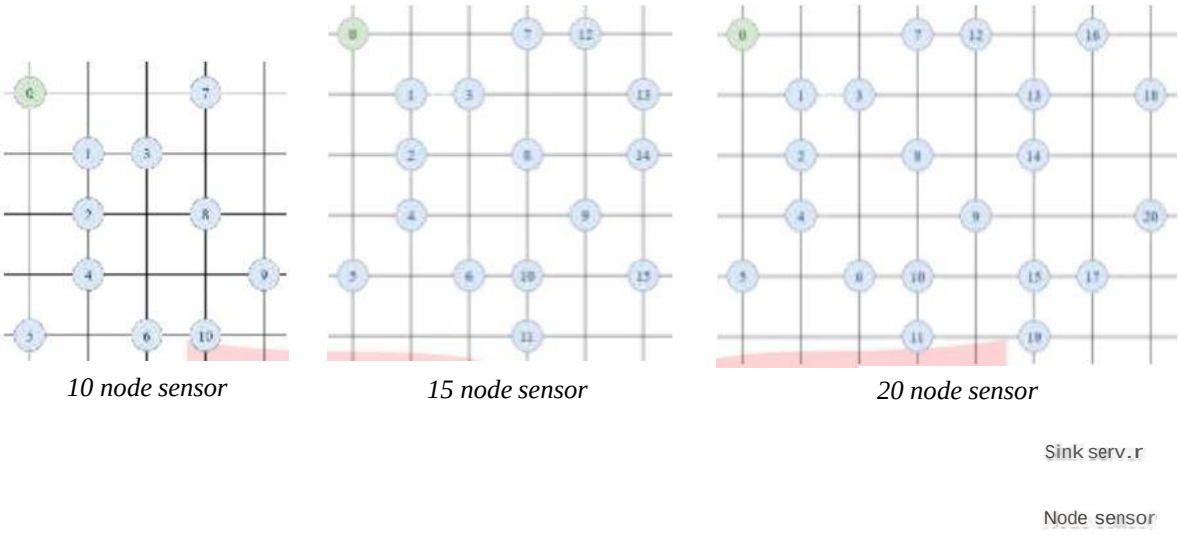


Gambar 3-1. Skema kerja protokol keamanan

Simulasi pengujian akan dilakukan menggunakan aplikasi NS-3, dengan menggunakan topologi jaringan seperti pada Gambar 3-2, untuk skenario pengujian aspek energi. Sedangkan untuk pengujian aspek keamanan *confidentiality*, *integrity* dan *authenticity* digunakan topologi dengan 15 *node* sensor namun ditambahkan 3 buah *node* penyerang. Untuk implementasi modul-modul keamanan pada protokol digunakan *library* eksternal yang dapat dijalankan pada NS3 yaitu Crypto++. Implementasi menggunakan Crypto++ juga sudah banyak diterapkan pada beberapa penelitian, seperti yang telah dilakukan oleh John Kelsey [10] yang membahas *cryptanalytic attack* pada *pseudorandom number generator*. Qiong Huang [11] dan Lihao Xu [12] juga menggunakan Crypto++ dalam penelitiannya.

Untuk menjamin keamanan pada aspek *confidentiality* perlu diimplementasikan modul enkripsi. Protokol TinySec menggunakan algoritma enkripsi RC5 sedangkan protokol LLSP menggunakan algoritma enkripsi AES. Untuk algoritma *cipher block*-nya, kedua protokol menggunakan metode CBC. Agar perbandingan performansi antara kedua protokol seimbang maka digunakan panjang key dan *block size* yang sama. Konfigurasi key default dari kedua algoritma enkripsi adalah sama yaitu 16 *byte* atau 128 bit. Jenis serangan yang diimplementasikan adalah *sniffing*.

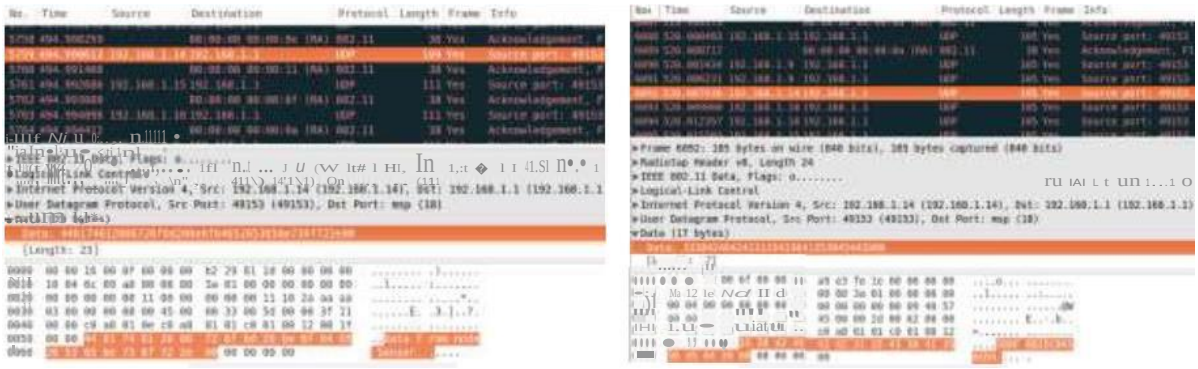
Untuk menjamin keamanan pada kedua aspek *integrity* dan *authenticity* dapat menggunakan *message authentication code* (MAC) [6] [13]. Pada protokol TinySec dan LLSP digunakan MAC dengan mode *cipher block* CBC. Untuk implementasinya digunakan *key* yang di-*generate* secara *random* di awal, kemudian *key* tersebut akan diimplementasikan pada tiap-tiap *node*-nya. Jenis serangan yang diimplementasikan pada simulasi yaitu fabrikasi.



Gambar 3-2. Skenario topologi jaringan untuk pengujian energi.

4. Pengujian dan Hasil

4.1. *Confidentiality*

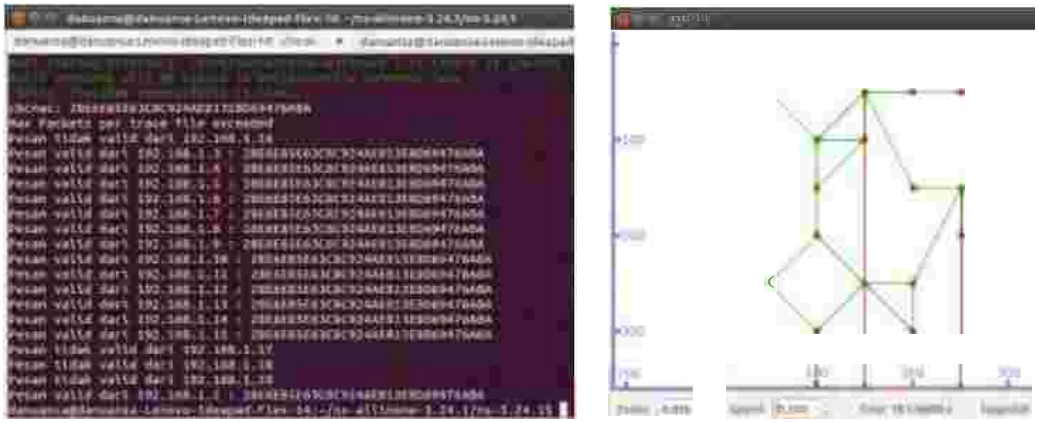


Gambar 4-1. Hasil proses *sniffing* tanpa enkripsi (kiri) dan dengan protokol TinySec (kanan).

Berdasarkan hasil pengujian, kedua protokol mampu menyembunyikan *plaintext* dan dapat menjamin kerahasiaan data dalam jaringan. Namun algoritma enkripsi AES memiliki performansi yang lebih baik dari RC5. Berdasarkan analisis yang dilakukan oleh Razvi [14], RC5 memang mempunyai kompleksitas yang lebih tinggi daripada AES dari sudut pandang *code*. Namun proses *key schedule* RC5 mengonsumsi banyak waktu, sedangkan pada WSN keterbatasan *resource* dan biaya komputasi merupakan faktor utama yang harus diperhitungkan dalam menentukan mekanisme keamanan yang akan digunakan dalam WSN. Komputasi yang dilakukan AES memakan energi lebih rendah daripada RC5 dan akan membuat *node* dapat bertahan lebih lama. Sehingga dapat dikatakan bahwa algoritma AES lebih cocok untuk diimplementasikan ke dalam WSN karena memenuhi dalam semua aspek, yaitu level keamanan yang tinggi dan keterbatasan *resource*.

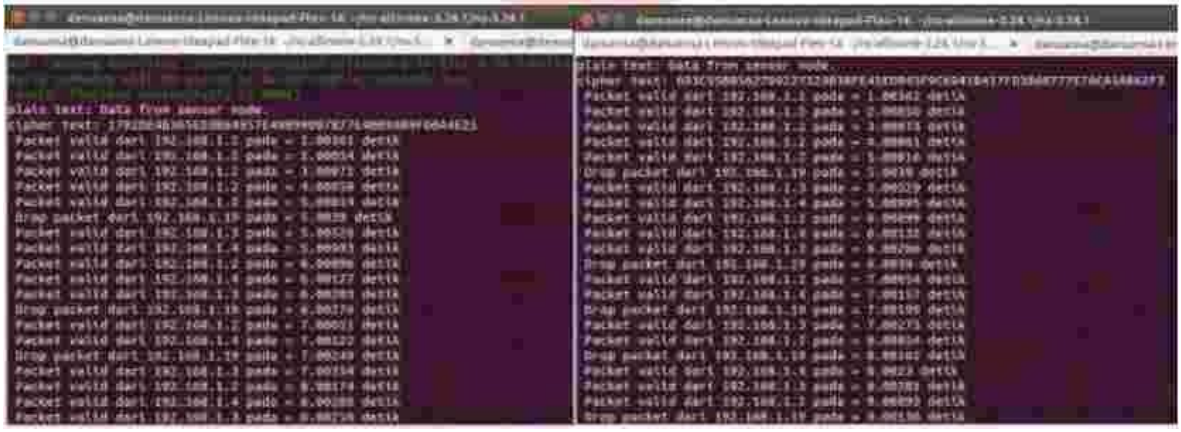
4.2. *Integrity* dan *authenticity*

Untuk menjamin keamanan pada aspek *integrity* dan *authenticity*, protokol TinySec dan LLSP menggunakan metode CBC-MAC. Perbedannya terletak pada proses perhitungan *hash* dan MAC-nya. Berikut ini adalah hasil yang didapatkan dari pengujian. Gambar 4-2 menunjukkan hasil yang diperoleh dari pengujian untuk protokol LLSP. Dapat dilihat bahwa kedua protokol dapat memilah antara paket-paket yang berasal dari *node* sensor asli dengan *node* sensor asing yang akan merusak pesan asli dari *node* sensor. Gambar 4-2 juga dapat dilihat hasil visualisasi dari pengujian proses otentikasi menggunakan aplikasi PyViz untuk kedua protokol. Dari gambar tersebut dapat dilihat bahwa terdapat panah-panah merah yang melambangkan paket yang dikirim dari *node* tersebut di-*drop*. Berarti dari proses pengujian telah ditemukan paket-paket yang di-*drop* karena tidak sesuai dengan kriteria paket yang valid.



Gambar 4-2. Paket *check* CBC-MAC LLSP (kiri) dan paket *drop* dari *node* penyerang.

Selain dari visualisasi, pembuktian paket yang di-*drop* juga ditunjukkan pada Gambar 4-3. Dari hasil tersebut terdapat paket-paket yang di-*drop* oleh protokol TinySec dan LLSP seperti yang berasal dari alamat IP 192.168.1.19 pada detik ke 5. Hal itu dikarenakan *node-node* asing tersebut tidak memiliki *cipher* yang valid dalam konten paket yang dikirimnya. Isi pesan yang valid mengandung *cipher* yang dihasilkan dari proses *hashing*



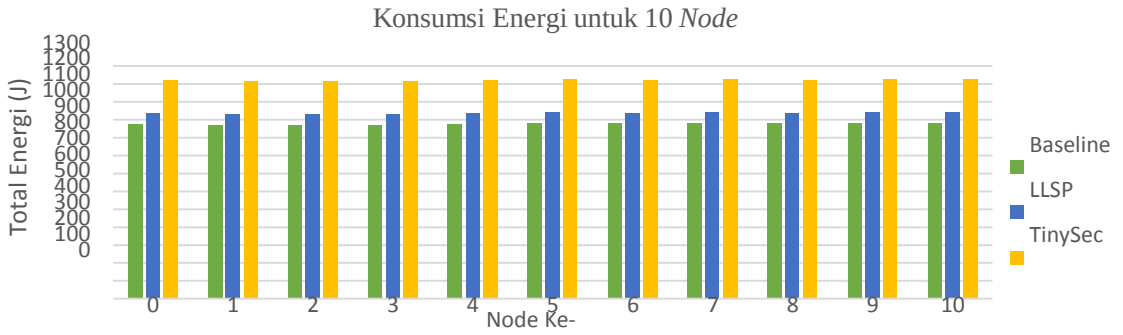
Gambar 4-3. Paket *drop* TinySec (kiri) dan paket *drop* LLSP (kanan).

Dari hasil simulasi didapatkan bahwa kedua metode yang digunakan oleh masing-masing protokol dapat melakukan perhitungan MAC dan melakukan *drop* pada paket-paket yang berasal dari *node* penyerang. Keduanya menggunakan 4 *byte* untuk panjang MAC untuk menjamin keamanan *authenticity* dan *integrity*, sehingga akan terdapat 2^{32} kemungkinan bagi penyerang untuk menemukan kombinasi MAC yang valid [4] [6]. Namun protokol TinySec hanya menyediakan 2 *byte counter* yang digunakan untuk kombinasi IV, sedangkan pada protokol LLSP menyediakan 4 *byte counter* yang sinkronis pada *node* pengirim dan penerima yang terus di-*update* dengan *feedback shift register* (FSR) [15] [13]. Dengan cara tersebut LLSP mampu mendeteksi jika terdapat paket-paket lama/duplikat yang dikirim ulang oleh *node* penyerang setelah melakukan *sniffing*. Sehingga LLSP dapat menyediakan perlindungan terhadap *replay attack* sedangkan TinySec tidak menyediakan perlindungan pada level keamanan tersebut.

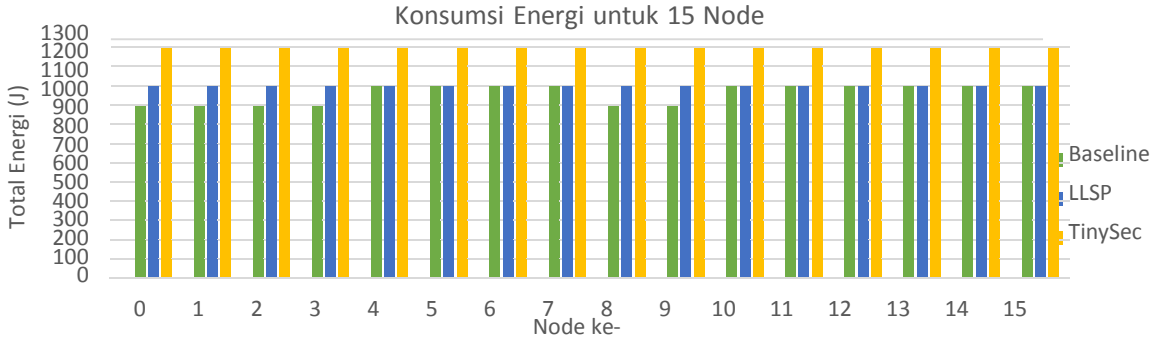
4.3. Energi

Pengujian untuk aspek energi dilakukan pada tiga skenario, yaitu pada topologi dengan jumlah *node* sensor 10, 15, dan 20 buah. Berdasarkan hasil pengujian yang dilakukan selama 1500 detik, berikut ini adalah total konsumsi energi yang didapat dari masing-masing skenario pengujian. Untuk skenario pertama, yaitu pengujian dengan menggunakan 10 *node* sensor, disajikan melalui Gambar 4-4. Skenario kedua, yaitu pengujian dengan menggunakan 15 *node*, disajikan dalam Gambar 4-5. Sedangkan untuk skenario ketiga, disajikan pada Gambar 4-6. Berdasarkan hasil pengujian yang disajikan pada grafik tersebut, diperoleh data bahwa total energi konsumsi yang dihabiskan oleh *node* sensor yang mengimplementasikan protokol keamanan LLSP lebih rendah dari *node* yang mengimplementasikan protokol TinySec. Hal ini disebabkan karena protokol TinySec menambahkan panjang paket sebesar 5 *byte* per paket untuk *communication overhead*, sedangkan pada protokol LLSP hanya menambahkan sebanyak 3 *byte* per paket [6]. Dengan bertambahnya panjang paket maka akan bertambah pula

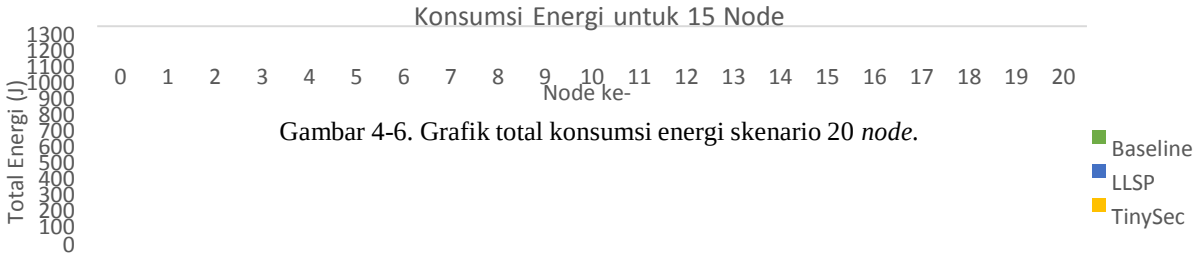
bandwidth yang digunakan, *latency*, serta konsumsi energi untuk mengirimkan paket [7]. Selain itu faktor lainnya adalah proses komputasi untuk melakukan enkripsi data. Algoritma RC5, yang dipakai oleh TinySec, memiliki siklus *key expansion* yang lebih besar dibandingkan algoritma AES, yang ada pada LLSP [14].



Gambar 4-4. Grafik total konsumsi energi skenario 10 *node*.



Gambar 4-5. Grafik total konsumsi energi skenario 15 *node*.



Gambar 4-6. Grafik total konsumsi energi skenario 20 *node*.

5. Kesimpulan

Berdasarkan hasil simulasi pengujian dan studi literature yang telah dilakukan, maka kesimpulan yang dapat ditarik yaitu, untuk aspek konsumsi energi protokol keamanan LLSP lebih unggul dibandingkan dengan protokol TinySec karena dapat menghemat konsumsi energi hingga 15%. Hal itu dikarenakan adanya penambahan byte overhead yang lebih besar pada protokol TinySec. Kedua protokol dapat menjamin keamanan atau kerahasiaan data yang dikirimkan antar node. Namun LLSP memiliki panjang IV sebesar 10 *byte*, sedangkan TinySec hanya mengalokasikan panjang IV sebesar 8 *byte*. Algoritma enkripsi AES, yang diterapkan oleh LLSP, lebih cocok diimplementasikan dalam WSN daripada RC5 karena dapat memenuhi tantangan utama dalam WSN yaitu keterbatasan resource. Untuk *integrity* dan *authenticity*, kedua protokol berhasil memilah paket-paket yang mengandung paket yang tidak valid dan melakukan *drop* terhadap paket tersebut. Keduanya memiliki panjang MAC yang sama yaitu 4 *byte* sehingga dalam aspek *integrity* kedua memberikan perfomansi yang tidak jauh berbeda. Namun pada LLSP terdapat 4 *byte counter* dan *feedback shift register* (FSR) yang dapat mendeteksi paket duplikat yang dikirim ulang, sehingga LLSP dapat memberikan pertahanan yang lebih baik pada aspek *authenticity*.

Daftar Pustaka

[1] L. E. Lighfoot, J. Ren and T. Li, "An Energy Efficient Link-Layer Security Protocol," in *Electro/Information Technology*, Chicago, 2007.

[2] R. W. Anwar, M. Bakhtiari, A. Zainal, A. H. Abdullah and K. N. Qureshi, "Security Issues and Attacks in Wireless Sensor Network," *World Applied Sciences Journal* , vol. 30, no. 10, pp. 1224-1227, 2014.

[3] J. P. Walters, Z. Lian, W. Shi and V. Chaudhary, "Wireless Sensor Network Security: A Survey," Detroit, 2006.

[4] S. M. AlMheiri and H. S. AlQamzi, "Data Link Layer Security Protocols in Wireless Sensor Networks: A Survey," *IEEE*, pp. 312-317, 2013.

[5] J. Sen, "A Survey on Wireless Sensor Network Security," *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 1, no. 2, pp. 55-78, 2009.

[6] C. Karlof, N. Sastry and D. Wagner, "TinySec: A Link Layer Security Architecture for Wireless Sensor Networks," in *SenSys Embedded Network Sensor Systems*, New York, 2004.

[7] A. S. Ahmed, "An Evaluation of Security Protocols on Wireless Sensor Network," in *Seminar on Internetworking*, 2009.

[8] R. L. Rivest, "The RC5 Encryption Algorithm," *Proceedings of the Second International Workshop on Fast Software Encryption (FSE)*, pp. 86-96, 1994.

[9] FIPS, "Advanced Encryption Standard (AES)," FIPS, 2001.

[10] J. Kelsey, B. Schneier, D. Wagner and C. Hall, "Cryptanalytic Attacks on Pseudorandom Number Generators," *Fast Software Encryption*, vol. 1372, pp. 168-188, 1998.

[11] Q. Huang, D. S. Wong, J. Li and Y.-M. Zhao, "Generic Transformation from Weakly to Strongly Unforgeable Signatures," *Journal of Computer Science and Technology*, vol. 23, no. 2, pp. 240-252, 2008.

[12] L. Xu, "Computation-Efficient Multicast Key Distribution," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, no. 5, pp. 577-587, 2008.

[13] J. Ren, T. Li and D. Aslam, "A Power Efficient Link-Layer Security Protocol (LLSP)," in *Military Communications Conference, 2005*, Atlantic City, 2005.

[14] K. S. M. Razvi Doomun, "Analytical Comparison of Cryptographic," *International Journal of Network Security*, vol. 9, no. 1, p. 82–94, 2009.

[15] A. A. Andhale and P. B. Jagdale, "Light Weight Security Protocol for Wireless Sensor Network's (WSN)," *International Journal of Engineering Research & Technology (IJERT)*, vol. 3, pp. 2885-2890, 2014.

[16] X. Chen, K. Makki, K. Yen and N. Pissinou, "Sensor Network Security: A Survey," *Communications Surveys & Tutorials, IEEE*, vol. 11, no. 2, pp. 52-73, 2009.

[17] Y. Bazband, K. and F. , "Performance Comparison Wireless Sensor Network Security Protocols : LLSP and Tiny," *JESRT*, pp. 650-652, 2014.

[18] M. Krishnan, "Intrusion Detection in Wireless Sensor Networks," 2010.

[19] W. Stalling, Cryptography and Network Security Fifth Edition, Prentice Hall, 2011.

[20] W. Dai, "Crypto++ Library," [Online]. Available: https://www.cryptopp.com/.

[21] Stanford, "CC2420 chip," [Online]. Available: http://tinyos.stanford.edu/tinyos-wiki/index.php/CC2420#Where_used.

[22] D. Selent, "Advanced Encryption Standard," *InSight: RIVIER ACADEMIC JOURNAL*, vol. 6, no. 2, pp. 1-14, 2010.

[23] R. H. Rajdeep Bhanot, "A Review and Comparative Analysis of Various Encryption," *International Journal of Security and Its Applications*, vol. 9, no. 4, pp. 289-306, 2015.