Perbandingan Protokol Keamanan *Wireless Sensor Network* (WSN) LEAP dan RKP

Taufik Akbar[1)], Fazmah Arif Yulianto[2)], Sidik Prabowo[3)]

Prodi S1 Teknik Informatika, Telkom School of Computing, Telkom University
Jalan Telekomunikasi no. 1, Dayeuhkolot Bandung 42057 Indonesia

[1]t.akbar34@yahoo.co.id, [2]fazmaharif@telkomuniversity.ac.id, [3]pakwowo@telkomuniversity.ac.id

Abstrak

Protokol keamanan pada jaringan merupakan hal yang sangat penting dan mempengaruhi keberlangsungan jaringan. Jaringan sensor nirkabel merupakan jaringan yang rentan dalam hal keamanan, kerentanan ini dapat menyebabkan kebocoran informasi, pemalsuan informasi, bahkan penyusupan kedalam jaringan yang dapat merusak jalur informasi pada jaringan. Terbatasnya kapasitas memori dan energi pada sensor menyebabkan tidak sembarang protokol kemanan dapat digunakan. Penelitian ini dilakukan untuk menganalisis performa protokol keamanan Localized Encryption and Authentication Protocol (LEAP) dan Random Key Pre-distribution (RKP) yang dapat digunakan pada jaringan sensor nirkabel dan membandingkan manakah yang lebih baik dengan menggunakan aplikasi NS3.

Kata Kunci : *Wireless Sensor Network, Jaringan Sensor Nirkabel, LEAP, RKP, NS3*

1. PENDAHULUAN

Jaringan sensor nirkabel merupakan kumpulan sensor yang tersusun menjadi sebuah jaringan yang dapat merasakan dan mengontrol lingkungannya sehingga dapat dilakukannya interaksi antara manusia atau komputer dengan lingkungan sekitarnya. Pada awalnya jaringan sensor nirkabel digunakan hanya untuk kepentingan militer, sekarang jaringan sensor nirkabel sudah mulai dikembangkan untuk berbagai bidang lainnya, seperti pertanian, kesehatan, dan lain-lain.

Node sensor tidak dapat selalu diawasi satu persatu oleh manusia. Oleh karena itu, keamanan node sensor pada jaringan sensor nirkabel sangat dibutuhkan, terutama saat pengiriman data dan informasi. Data dan informasi yang dikirimkan node sensor bersifat rahasia, karena isi data dan informasi yang dikirimkan node sensor adalah hal penting yang merupakan keadaan sebenarnya dari lingkungan. Apabila kerahasiaan data dan informasi tidak ada atau rusak, sehingga data dan informasi dapat dilihat dan diubah oleh pihak ketiga yang dapat mengakibatkan penyusupan kedalam jaringan, kebocoran informasi, bahkan pemalsuan informasi yang dapat merusak jalur informasi pada jaringan.

Terbatasnya memori dan energi yang dapat disimpan oleh node sensor mengakibatkan komputasi yang dilakukan node sensor haruslah lebih sederhana dibandingkan komputasi yang dilakukan oleh komputer yang biasa digunakan sehari-hari. Begitu pula dengan protokol keamanan yang digunakan untuk jaringan sensor nirkabel, tidak hanya harus memenuhi kebutuhan keamanan tapi juga harus sesuai dengan sumber daya dan kemampuan yang dimiliki oleh node sensor, sehingga tidak sembarang protokol keamanan dapat diaplikasikan pada jaringan sensor nirkabel.

Dalam penelitian ini, akan dilakukan analisis terhadap perbandingan performa antara dua protokol keamanan, yaitu LEAP dan RKP dimana kedua protokol keamanan tersebut bertipe manajemen kunci. Walaupun keduanya bertipe manajemen kunci, tapi kedua protokol keamanan tersebut memiliki cara manajemen kunci yang berbeda. Protokol keamanan LEAP menggunakan empat jenis kunci yang digunakan berdasarkan node yang dituju untuk berkomunikasi. RKP menggunakan *key ring* yang berisi kunci-kunci yang nantinya digunakan untuk membangun jalur komunikasi. Selain itu, LEAP dan RKP mendukung aspek confidentiality dan authentication. Perbandingan protokol keamanan LEAP dan RKP dilakukan menggunakan NS3.

2. TINJAUAN PUSTAKA

2.1 *Karya Terkait*

Cukup banyak paper-paper yang membahas dan mengulas mengenai protokol keamanan pada jaringan sensor nirkabel dan juga membandingkannya dengan protokol keamanan lainnya. Diantaranya adalah, Sangwan dan kawan-kawan mengulas beberapa protokol keamanan pada jaringan sensor nirkabel, sepeti SPINS, LEAP, TINYSEC, ZigBee, dan SM pada paper berjudul "*A Review of various security protocols in Wireless Sensor Network*" [18]. Setelah melakukan review, mereka menyimpulkan bahwa protokol keamanan LEAP memiliki karekteristik untuk mendukung aspek *confidentiality* dan *authentication*, tapi tidak mendukung aspek *freshness*, *integrity*, dan *availability*.

Hong dan kawan-kawan, selain melakukan review terhadap protokol keamanan juga menggambarkan perbandingan protokol keamanan pada paper berjudul "*Comparison of Security Protocols for Wireless Sensor Networks*" [8]. Perbandingan yang dilakukan berfokus pada aspek confidentiality dan authentication. Pada paper ini, protokol keamanan LEAP dan RKP dimasukan kedalam kategori manajemen kunci. Pada perbandingan performa berdasarkan serangan spoofing, protokol keamanan LEAP dan RKP menyediakan otentikasi untuk menangani serangan dan menjaga data dan informasi..

2.2 *Jaringan Sensor Nirkabel*

Jaringan sensor nirkabel merupakan jaringan nirkabel yang memanfaatkan sistem *embedded* dan kumpulan node sensor yang dapat merasakan, memonitoring, mengirim data, dan mengontrol lingkungan sehingga memungkinkan terjadinya interaksi manusia atau komputer dengan lingkungan sekitar. Node sensor akan diletakan secara random disekitar lingkungan yang akan dipantau. Node sensor yang satu dan node sensor lainnya akan mengorganisasi dengan sendirinya membentuk sebuah jaringan.[16]

2.3 *Localized Encryption and Authentication Protocol* (LEAP)

Localized Encryption and Authentication Protocol (LEAP) merupakan salah satu protokol keamanan yang digunakan pada jaringan sensor nirkabel yang menggunakan manajemen kunci. LEAP menggunakan algoritma kunci simetrik dimana node sensor pengirim dan node sensor penerima menggunakan kunci yang sama untuk melakukan enskripsi dan dekripsi data. LEAP mengasumsikan bahwa jaringan sensor nirkabel tidak akan sepenuhnya aman hanya dengan mekanisme kunci tunggal [2], sehingga LEAP menggunakan empat jenis kunci yang akan digunakan oleh setiap node sensor, yaitu *individual key, pairwise key, cluster key, dan group key*.

- ***Individual key*.** *Individual key* merupakan kunci yang digunakan node sensor untuk berhubungan dengan node sink.
- ***Pairwise key*.** *Pairwise key* digunakan untuk membangun jalur komunikasi yang aman antar node sensor yang berdekatan.
- ***Cluster key*.** *Cluster key* merupakan kunci yang dibagikan oleh node sensor kepada semua node sensor tetangganya. *Cluster key* digunakan untuk mengamankan jaringan komunikasi pesan broadcast lokal.
- ***Group key*.** *Group key* merupakan kunci yang dibagikan oleh node sink kepada seluruh node sensor di dalam jaringan. *Group key* digunakan untuk mengenkripsi pesan yang akan di-broadcast oleh node sink keseluruh jaringan.

2.4 *Random Key Pre-distribution* (RKP)

Setiap node akan memilih kunci secara acak dan membentuknya menjadi *key ring* sebelum disebarkan kedalam jaringan, node sensor dapat membuat sambungan komunikasi yang aman dengan node sensor lainnya jika mereka berbagi setidaknya satu kunci yang sama. Skema dari RKP dibagi menjadi tiga fase, yaitu:

1. ***Key setup*.** Sebelum disebarkan, setiap node menyiapkan *key ring* yang berisi sejumlah *key* yang dipilih secara random dari *key pool* dan disimpan kedalam memori.
2. ***Shared key discovery*.** Setelah node sensor disebar maka akan masuk kedalam fase *shared key discovery*, dimana setiap node akan membroadcast daftar *key identifier*, sensor node yang menerima broadcast daftar *key identifier* akan membandingkannya dengan kunci yang ada pada *key ring*. Apabila terdapat key yang sama, maka node sensor akan mengotentikasi untuk

memverifikasi dan kemudian *key* tersebut akan dimasukan kedalam *key chain* dan dibuat jalur komunikasi yang aman menggunakan *key* tersebut.

3. ***Path key establishment.*** Node sensor akan mencoba membentuk *graph* yang seluruhnya tersambung untuk berkomunikasi dengan node sensor lainnya yang belum terhubung setelah fase sebelumnya. Node sensor pertama akan melakukan *multi-hop* melalui node sensor penengah yang dimana node sensor tersebut sudah menjalin komunikasi yang aman baik antara node sensor pertama dan dengan node sensor yang tidak terhubung dengan node sensor pertama.

2.5 RC5

Algoritma enkripsi RC5 berbasis *symmetric block cipher* yang dimana menggunakan *secret key* yang sama untuk melakukan enkripsi dan dekripsi, *ciphertext* yang dihasilkan oleh RC5 adalah *block cipher*, yaitu *ciphertext* yang memiliki panjang bit yang sudah ditentukan dan selalu tetap, diantaranya 32 bit, 64 bit, dan 128 bit. RC5 berorientasi word yang dimana semua operasi komputasi dasarnya memiliki w-bit words sebagai input dan output dan hanya menggunakan operasi komputasi primitif yang umumnya ditemukan pada mikroprosesor sehingga membuat RC5 menjadi algoritma enkripsi yang simpel dan cepat.

2.6 *Cipher Block Chaining Message Authentication Code* (CBC-MAC)

CBC-MAC memiliki beberapa versi berbeda yang bervariasi dengan pembeda, seperti *padding, length variability*, dan *key search strengthening*. CBC-MAC pada umumnya melakukan *padding* dengan mempertimbangkan blok masukan akhir sebagai blok parsial data dan menambahkan nol untuk memenuhi blok.

Diasumsikan pesan berbentuk string biner dinotasikan dengan M dengan panjang beberapa positif l, sehingga M dapat dipecah menjadi beberapa blok seperti [5]

$$M = \text{[illegible]}, \text{[illegible]}, \ldots, \text{[illegible]} \quad |\text{[illegible]}| = \text{[illegible]} \tag{2,1}$$

Kemudian masing-masing blok dilewatkan melalui enkripsi E dengan kunci K dan hasilnya kemudian di-XOR dengan blok berikutnya. Jika EK merupakan enkripsi dengan menggunakan kunci K, kemudian cipher block chaining ditentukan oleh [5]

$$\text{[illegible]}(\text{[illegible]}) = \sum_{\text{[illegible]}=1}^{\text{[illegible]}} \text{[illegible]}[\text{[illegible]} \oplus \text{[illegible]}_{-1}]$$
$$\text{untuk } i = 1 \ldots \text{[illegible]} \text{ dan } \text{[illegible]}_0 = 0\text{[illegible]} \tag{2,2}$$

2.7 *Hash Message Authentication Code (HMAC)*

Tujuan utama dari HMAC adalah penggunaan tanpa modifikasi, fungsi hash yang tersedia, memiliki performa baik, dan kode perangkat lunak yang tersedia secara bebas. HMAC dapat dibuktikan aman jika fungsi hash *embedded* memiliki kekuatan kriptografi yang layak. Dimana H menjadi simbol dari fungsi hash yang diinisialisasi dengan Initial Value yang tetap (IV), yang menghasilkan nilai hash n-bit. HMAC bekerja pada input M dengan panjang yang arbitrary, yang merupakan kelipatan dari b bit. Hmac menggunakan string random tunggal yang dinotasikan dengan K sebagai kunci. Jika panjang kunci lebih besar dari b, maka akan menjadi input ke fungsi hash untuk menghasilkan kunci n-bit. Panjang kunci yang direkomendasikan untuk digunakan oleh HMAC adalah ≥ n. Dimana HMAC dapat dinyatakan sebagai berikut [5]

$$\text{[illegible]}(\text{[illegible]}) = \text{[illegible]}(K^{+} \oplus \text{[illegible]}(\text{[illegible]}(K^{+} \oplus \text{[illegible]}))) \tag{2,3}$$

K^{+} merupakan K yang diisi nol sehingga menghasilkan panjang b bits, ipad adalah pad bagian dalam yang merupakan 36 *byte hex* diulang sebanyak b/8 kali dan opad adalah pad bagian luar yang merupakan 5C *byte hex* diulang sebanyak b/8 kali. Operasi XOR dinotasikan dengan simbol ⊕.

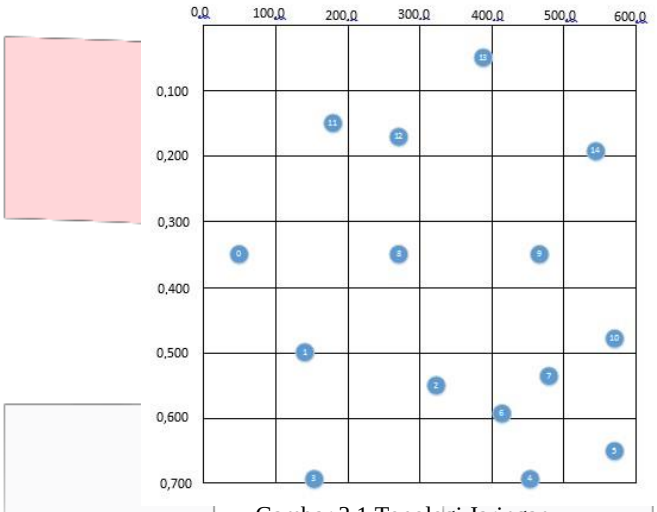
3 PERANCANGAN DAN SKENARIO SIMULASI

3.1 Deskripsi Sistem

Sistem ini bertujuan untuk melakukan simulasi dari protokol keamanan LEAP dan RKP. NS3 digunakan sebagai sarana simulasi dari protokol jaringan yang dijalankan pada sistem operasi Linux Ubuntu 15.04. Eksternal library Cryptopp digunakan untuk pemrosesan kriptografi pada simulasi di NS3.

3.2 Skenario Simulasi

Simulasi pengujian pada protokol LEAP dan RKP dibagi kedalam 3 skenario. Skenario yang dilakukan berdasarkan aspek yang diujikan, yaitu *confidentiality*, *authentication*, dan *energy consumption*. Topologi jaringan yang dibuat untuk simulasi pengujian secara umum dapat dilihat pada Gambar 3.1.



Gambar 3.1 Topologi Jaringan

3.2.1 Skenario 1: *Confidentiality*

Pada skenario ini setiap node sensor pada masing-masing protokol keamanan LEAP dan RKP akan mengirimkan *ciphertext* yang dibuat dan *ciphertext* yang diterima oleh node sink akan didekripsi untuk menghasilkan *plaintext*.

3.1.2 Skenario 2: *Authentication*

Pada skenario ini, pengujian aspek *authentication* dilakukan dengan menambahkan dua node palsu pada jaringan yang nantinya akan berperan sebagai penyerang didalam jaringan dan akan ikut serta mengirimkan *ciphertext* seperti node anggota pada jaringan, jenis serangan yang digunakan adalah *fabrication attack*. Pengujian ini akan menghasilkan jumlah pesan palsu yang terdeteksi dan persentase pesan yang didrop hasil dari pemeriksaan *authentication* yang dikirim oleh node penyerang.

3.1.3 Skenario 3: *Energy Consumption*

Pengujian aspek *energy consumption* dilakukan pada protokol keamanan LEAP, protokol keamanan RKP, dan topologi tanpa protokol keamanan. Hasil rata-rata pengukuran *energy consumption* akan disajikan kedalam tabel yang kemudian dibandingkan dan disajikan kedalam bentuk grafik batang. Dikarenakan NS3 tidak support keamanan dan menggunakan library cryptopp untuk mengimplementasikan kriptografi, sehingga energi komputasi dari kriptografi tidak dapat dihitung. Energi yang dihasilkan merupakan energi dari transmisi data.

4 HASIL PENGUJIAN DAN ANALISIS

4.1 *Confidentiality*

Protokol keamanan LEAP dan RKP menghasilkan masing-masing 16 byte *ciphertext* hasil dari enkripsi *plaintext*. Kedua protokol keamanan menggunakan algoritma enkripsi yang sama, yaitu RC5 yang berfungsi sebagai pelaku enkripsi sehingga ciphertext yang dihasilkan hampir serupa.

Tabel 4.1 Hasil Pengujian *Confidentiality*

	Jumlah pesan yang dikirim seluruh sensor	Jumlah pesan yang diterima sink	Jumlah pesan yang berhasil didekripsi
Topologi menggunakan protokol keamanan LEAP	10.309	10.309	10.309
Topologi menggunakan protokol keamanan RKP	10.309	10.309	10.309

Sebanyak 10.309 *ciphertext* dikirimkan secara kumulatif oleh seluruh node sensor pada simulasi pengujian *confidentiality*. Dapat dilihat pada Tabel 4.1, masing-masing protokol keamanan mengirimkan 10.309 pesan berisi *ciphertext* dan semua pesan berhasil diterima oleh node sink. *Ciphertext* yang diterima node sink kemudian didekripsi untuk melihat isi *plaintext* yang dikirimkan.

4.2 *Authentication*

Pengujian aspek authentication bertujuan agar pihak luar yang tidak memiliki kewenangan dan bukan anggota tidak berpartisipasi dalam jaringan. Pengujian dilakukan dengan cara melakukan verifikasi pesan melalui perhitungan MAC yang membuktikan apakah benar pesan dikirimkan oleh anggota jaringan ataukah pihak luar.

Tabel 4.2 Hasil Pengujian *Authentication*

	Pesan Palsu yang Dikirim	Keseluruhan Pesan yang Dikirim	Pesan Palsu yang Terdeteksi	Rasio Pesan Palsu yang Didrop
Protokol Keamanan LEAP	1.893	18.440	1.893	100 %
Protokol Keamanan RKP	1.998	17.037	1.998	100 %

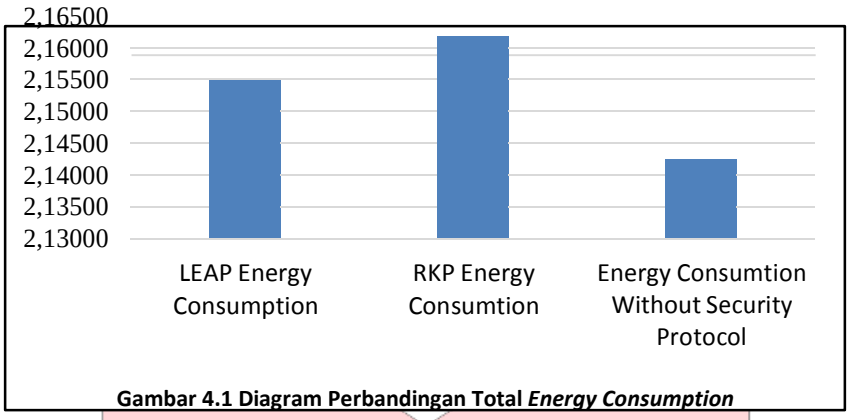
Pada jalannya simulasi semua pesan akan diterima node sink yang kemudian akan dilakukan proses verifikasi, apabila ada pesan yang tidak terverifikasi sebagai pesan yang dikirim oleh node anggota jaringan maka pesan akan didrop. Berdasarkan Tabel 4.2, node penyerang pada protokol keamanan LEAP mengirimkan 1.893 pesan dan pada protokol keamanan RKP mengirimkan 1.998 pesan. Terlihat bahwa rasio pesan yang didrop baik pada protokol keamanan LEAP dan RKP adalah 100%, dimana pesan palsu yang dikirim node penyerang berhasil dideteksi dan didrop.

4.3 *Energy Consumption*

Pengujian energy consumption dilakukan pada setiap node sensor yang ada didalam jaringan pada masing-masing protokol keamanan LEAP dan RKP dengan waktu simulasi 1500 detik untuk setiap node sensor. Simulasi juga dilakukan pada topologi jaringan tanpa protokol keamanan yang digunakan sebagai pembantu pembanding pengujian.

Tabel 4.3 Rata-Rata Total *Energy Consumption*

Rata-rata Konsumsi Energi Seluruh Node LEAP	Rata-rata Konsumsi Energi Seluruh Node RKP	Rata-rata Konsumsi Energi Seluruh Node Tanpa Protokol Keamanan
2,15482 J	2,16158 J	2,14220 J



Gambar 4.1 Diagram Perbandingan Total *Energy Consumption*

Rata-rata energi yang dikonsumsi oleh seluruh node sensor pada masing-masing protokol keamanan setelah 1500 detik dapat dilihat pada Tabel 4.3. Berdasarkan Tabel 4.3, protokol keamanan LEAP mengkonsumsi total energi sebesar 2,15482 J dan protokol keamanan RKP mengkonsumsi total energi sebesar 2,16158 J . Bila dibandingkan konsumsi energi dari protokol keamanan LEAP lebih sedikit dibandingkan dengan protokol keamanan RKP dengan perbedaan yang tipis, yaitu 0,00676 J.

NS3 yang tidak support keamanan dan menggunakan eksternal library cryptopp menyebabkan energi dari komputasi yang terjadi pada protokol keamanan LEAP dan RKP tidak bisa dihitung. Perbedaan konsumsi energi terjadi karena panjang paket yang dihasilkan protokol keamanan LEAP dan RKP berbeda. Hal ini disebabkan karena perbedaan MAC yang dihasilkan oleh protokol keamanan LEAP dan RKP, sehingga besar ukuran paket keseluruhan berbeda.

Berdasarkan data pada Tabel 4.3 dapat dihitung daya yang diperlukan oleh rata-rata node sensor pada masing-masing protokol keamanan dan topologi tanpa protokol keamanan. Hasil perhitungan dapat dilihat pada Tabel 4.4 dimana daya yang dibutuhkan oleh protokol keamanan LEAP adalah 0,001436 Watt dan protokol keamanan RKP adalah 0,001441 Watt.

Tabel 4.2 Rata-rata Daya yang Dibutuhkan Seluruh Node Sensor

Rata-rata Daya yang Dibutuhkan Seluruh Node LEAP	Rata-rata Daya yang Dibutuhkan Seluruh Node RKP	Rata-rata Daya yang Dibutuhkan Seluruh Node Tanpa Protokol Keamanan
0,001436 W	0,001441 W	0,001428 W

Berdasarkan data pada Tabel 4.3 dapat dihitung waktu hidup rata-rata node sensor dari tiap protokol keamanan. Kapasitas energi yang digunakan pada perhitungan adalah 1500 J. Hasil perhitungan dapat dilihat pada Tabel 4.4 dimana rata-rata waktu hidup seluruh node sensor dari protokol keamanan LEAP dan RKP berdasarkan kapasitas energi yang diberikan hanya memiliki selisih 2 jam 16 menit 3 detik atau 8.163 detik.

Tabel 4.5 Rata-rata Waktu Hidup Seluruh Node Sensor

Rata-rata Waktu Hidup Seluruh Node LEAP	Rata-rata Waktu Hidup Seluruh Node RKP	Rata-rata Waktu Hidup Seluruh Node Tanpa Protokol Keamanan
12hari 5jam 7menit 5detik	12hari 2jam 51menit 2detik	12hari 9jam 23menit 24detik

Kesimpulan

1. Simulasi protokol keamanan LEAP dan RKP berhasil dilakukan pada NS3, dengan bantuan library eksternal cryptopp sebagai pemroses kriptografi dalam simulasi protokol keamanan LEAP dan RKP.
2. Hasil analisis memperlihatkan kemampuan LEAP dan RKP dalam aspek confidentiality sama dimana keduanya menggunakan RC5 sebagai algoritma enkripsinya. Pada aspek authentication kedua protokol dapat memverifikasi pesan yang masuk dengan baik walaupun menggunakan algoritma yang berbeda, yaitu LEAP menggunakan CBC-MAC dan RKP menggunakan HMAC. Pada aspek energy consumtion RKP mengkonsumsi energi lebih banyak dibandingkan LEAP walaupun hanya berbeda tipis, yaitu 0,00676 Joule.

Daftar Pustaka

[1] Bellare, M., Kilian, J. & Rogaway, P., 2001. The Security of the Cipher Block Chaining Message Authentication Code.

[2] Chaba, Y., Sharma, R. & Singh, Y., 2010. Analysis of Security Protocols in Wireless Sensor Network. Int. J. Advanced Networking and Applications, 2(3), pp. 707-713.

[3] Chan, H., Perrig, A. & Song, D., 2003. Random Key Predistribution Schemes for Sensor Networks. 2003 IEEE Symposium on Security and Privacy, p. 197.

[4] Chen, S. et al., 2014. Internet of Things: Wireless Sensor Networks. International Electrotechnical Commission.

[5] Deepakumara, J., Heys, H. M. & Venkatesan, R., 2003. Performance Comparison of Message Authentication Code (MAC) Algorithms for the Internet Protocol Security (IPSEC). NECEC.

[6] Fu, H., Kawamura, S., Zhang, L. & Zhang, M., 2005. Replication Attack on Random Key Pre-distribution Schemes for Wireless Sensor Networks. Proceedings of the 2005 IEEE Workshop on Information Assurance and Security.

[7] Hichem, E., Jemai, A. & Mastouri, A., 2011. Study of key pre-distribution schemes in wireless sensor networks: case of BROSK (use of WSNet). Applied Mathematics & Information Sciences – An International Journal, 5(3), pp. 655-667.

[8] Hong, C. P., Kim, T. H., Kim, C. H. & Kim, H., t.thn. Comparison of Security Protocols for Wireless Sensor Networks.

[9] Hwang, J. & Kim, Y., 2004. Revisiting Random Key Pre-distribution Schemes for Wireless Sensor Networks. Proceedings of the 2nd ACM workshop on Security of Ad hoc and Sensor Network.

[10] Jajodia, S., Setia, S. & Zhu, S., 2004. LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks.

[11] Jang, J., Kwon, T. & Song, J., 2007. A Time-Based Key Management Protocol for Wireless Sensor Network.

[12] Kaji, Y., Matsumoto, R. & Mohri, H., 2008. Key Predistribution Schemes for Sensor Networks Using Finite Plane Geometry. IEICE TRANS. INF. & SYSt., Volume E91-D.

[13] Kaliski Jr., B. S. & Yin, Y. L., 1998. On the Security of the RC5 Encryption Algorithm, s.l.: RSA Laboratories Technical Report TR-602.

[14] Kim, S. G., 2015. Reliable Random Key Pre-Distribution Schemes for Wireless Sensor Networks. International Journal of Information and Education Technology, Volume 5.

[15] Lopez, J. & Zhou, J., 2008. Wireless Sensor Network Security. 1 penyunt. Amsterdam: IOS Press.

[16] Pratama, I. P. A. E. & Suakanto, S., 2015. Wireless Sensor Network. Bandung: Informatika.

[17] Rivest, R. L., 1997. The RC5 Encryption Algorithm, s.l.: MIT Laboratory for Computer Science.

[18] Sangwan, A., Sindhu, D. & Singh, K., 2011. A Review of various security protocols in Wireless Sensor Network. IJCTA, Volume 2, pp. 790-797.