Journal of Informatics Management and Information Technology
ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

Kriptografi Ringan dengan Menggunakan Algoritma di *Internet Of Things (IoT)*

Rudhi Wahyudi Febrianto[1,*], Arief Zulianto[2]

[1] Fakultas Teknik Informatika, Program Studi Teknik Informatika, Sekolah Tinggi Teknologi Bandung, Bandung, Indonesia
[2] Prodi Magister Teknik Informatika, Pascasarjana, Universitas Langlangbuana
Email: rudhiwahyudifebrianto@sttbandung.ac.id[1], madzul@gmail.com[2]
Email Penulis Korespondensi: rudhiwahyudifebrianto@sttbandung.ac.id

Abstrak−Internet of Things (IoT) memungkinkan suatu objek menghasilkan data dan bertukar data. Pengaplikasian IoT menggunakan mikrokontroler seperti Arduino belum memiliki fitur untuk menjaga keamanan data di dalamnya. Selain itu, Arduino memiliki kapabilitas komputasi terbatas. Oleh karena itu, perlu diterapkan kriptografi dengan algoritma yang memiliki komputasi rendah pada Arduino untuk menjaga keamanan data. Kriptografi perlu diterapkan pada pengaplikasian IoT menggunakan mikrokontroler untuk menjaga keamanan proses transaksi data dan menjaga keaslian asal suatu data. Penerapan kriptografi pada mikrokontroler juga harus ringan dan dapat berjalan pada mikrokontroler terutama mikrokontroler Arduino. Pada Arduino, kemampuan komputasi bersifat terbatas namun kebanyakan algoritma komputasi keamanan yang ada memiliki komputasi yang tinggi. Oleh karena itu, protokol kriptografi yang diterapkan harus memiliki algoritma yang efisien dan kemampuan komputasi yang rendah. Tujuan penelitian kami adalah untuk memahami apakah pendekatan kriptografi ringan dapat digunakan untuk menumbuhkan IoT yang aman berdasarkan desain. Sebagai langkah pertama dalam proses penelitian kami, kami melakukan tinjauan literatur sistematis pada kriptografi ringan untuk mengumpulkan pengetahuan tentang penggunaan teknologi ini dan untuk mendokumentasikan tingkat saat ini. Kami menemukan 5 kasus penggunaan kriptografi ringan dalam literatur. Kami juga menemukan beberapa masalah dalam Waktu eksekusi algoritma pada Arduino Uno memiliki waktu yang lebih besar dua kali lipat dibandingkan dengan waktu eksekusi algoritma pada PC, Perubahan nilai pada panjang data dan pasangan tanda tangan digital berpengaruh terhadap hasil verifikasi keabsahan tanda tangan digital. Kami mendokumentasikan dan mengkategorikan penggunaan saat ini kriptografi ringan, dan memberikan beberapa rekomendasi untuk masa depan bekerja untuk mengatasi masalah yang disebutkan di atas.

Kata Kunci: *Internet of Things*; Kriptografi Ringan; Mikrokontroler Arduino; Teknologi; Keamanan Data

Abstract−Internet of Things (IoT) allows objects to generate data and exchange data. IoT applications using microcontrollers such as Arduino do not yet have features to maintain the security of the data in them. Additionally, Arduino has limited computing capabilities. Therefore, it is necessary to apply cryptography with algorithms that have low computation on the Arduino to maintain data security. Cryptography needs to be applied to IoT applications using microcontrollers to maintain the security of data transaction processes and maintain the authenticity of the origin of data. The application of cryptography on microcontrollers must also be light and able to run on microcontrollers, especially Arduino microcontrollers. On Arduino, computing capabilities are limited but most existing security computing algorithms have high computation. Therefore, the applied cryptographic protocol must have efficient algorithms and low computational capabilities. Our research goal is to understand whether lightweight cryptographic approaches can be used to foster secure IoT by design. As a first step in our research process, we conducted a systematic literature review on lightweight cryptography to gather knowledge about the use of this technology and to document its current level. We found 5 use cases of lightweight cryptography in the literature. We also found several problems in that the algorithm execution time on the Arduino Uno was twice as long as the algorithm execution time on a PC. Changes in the value of the data length and digital signature pair affected the results of verifying the validity of the digital signature. We document and categorize current uses of lightweight cryptography, and provide several recommendations for future work to address the issues mentioned above.

Keywords: Internet of Things; Lightweight Cryptography; Arduino microcontroller; Technology; Data Security

1. PENDAHULUAN

Internet of things (IoT) tengah menjadi perbincangan dalam dunia teknologi pada saat ini. Istilah IoT umumnya mengacu pada sebuah skenario suatu jaringan internet, kemampuan konektivitas dan komputasi berada dalam sebuah objek yang memungkinkan objek tersebut untuk menghasilkan data, bertukar data, dan mengambil data dengan sedikit campur tangan manusia [1], [2]. Pada dasarnya IoT merupakan konstruksi yang saling menghubungkan perangkat umum satu sama lain. Perangkat umum dapat berupa jam tangan, televisi, termostat, mobil, dan lampu. Selain perangkat umum yang disebutkan sebelumnya, pengaplikasian IoT juga biasa digunakan pada mikrokontroler [3], [4], [5], [6], [7], [8]. Salah satu contoh mikrokontroler yang umum digunkan dalam pengaplikasian IoT adalah Arduino. Pengaplikasian IoT pada mikrokontroler ini dapat digunakan sebagai sarana pertukaran data atau pengiriman data. Salah satu contoh pengaplikasiannya adalah sistem location based perangkat berdaya komputasi rendah dengan Arduino [9], [10], [11], [12], [13], [14], [15], [16], [17], [18]. Sistem ini bekerja dengan mengirimkan data berupa longitude, latitude, dan Internet Protocol (IP) address dari perangkat mikrokontroler ke suatu server. Pengaplikasian dengan sistem tersebut perlu memiliki fitur keamanan di dalamnya. Keamanan yang dimaksud dapat berupa keamanan saat melakukan pengiriman data antar alat elektronik. Jika pada client server keamanan proses pengiriman data dilindungi dengan Hypertext Transfer Protocol Secure (HTTPS), sebaliknya pada IoT belum terdapat keamanan saat proses transaksi data. Keamanan proses transaksi data pada IoT merupakan hal yang penting, bukan hanya pada saat data dikirimkan tetapi juga bagaimana data tidak diubah oleh seseorang atau pihak ketiga sehingga data tersebut bersifat asli dan untuk mengetahui keaslian asal data tersebut. Kriptografi perlu diterapkan pada pengaplikasian IoT menggunakan mikrokontroler untuk menjaga keamanan proses transaksi data dan menjaga keaslian asal suatu data. Penerapan kriptografi pada mikrokontroler juga harus ringan dan

ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

dapat berjalan pada mikrokontroler terutama mikrokontroler Arduino. Pada Arduino, kemampuan komputasi bersifat terbatas namun kebanyakan algoritma komputasi keamanan yang ada memiliki komputasi yang tinggi [19], [20], [21].

Terdapat berbagai penelitian dengan judul Lightweight cryptography methods, penelitian tersebut berfokus untuk metode kriptografi konvensional, seperti untuk AES (enkripsi), SHA-256 (hashing) dan RSA/Elliptic Curve (penandatanganan), bekerja dengan baik pada sistem yang memiliki daya pemrosesan dan kemampuan memori yang wajar/masuk akal. Lalu penelitian dengan judul Pengamanan Internet of Things untuk Tanda Tangan Digital Menggunakan Algoritma Elgamal Signature Scheme, penelitian tersebut berfokus Internet of Things (IoT) memungkinkan suatu objek menghasilkan data dan bertukar data. Pengaplikasian IoT menggunakan mikrokontroler seperti Arduino belum memiliki fitur untuk menjaga keamanan data di dalamnya. Selain itu, Arduino memiliki kapabilitas komputasi terbatas. Oleh karena itu, perlu diterapkan kriptografi dengan algoritma yang memiliki komputasi rendah pada Arduino untuk menjaga keamanan data. Penjagaan keamanan data terutama pada keaslian asal data, dilakukan dengan menggunakan tanda tangan digital. Penerapan tanda tangan digital dapat dilakukan salah satunya dengan algoritma Elgamal signature scheme. Lalu penelitian dengan judul Implementasi Algoritma AEGIS untuk Payload Data Protokol MQTT dan CoAP pada Raspberry Pi 3, penelitian tersebut berfokus Raspberry Pi 3 adalah perangkat yang dapat digunakan sebagai middleware. Middleware bertanggung jawab dalam memfasilitasi komunikasi dan manajemen informasi dari komponen heterogen. Dalam rancangan sistem middleware penelitian sebelumnya masih terdapat celah keamanan sistem pada pengiriman data yang membuat paket data yang dikirimkan dapat diketahui oleh pihak yang tidak bertanggung jawab. Suatu metode yang dapat menjamin confidentiality dan authentication yaitu algoritma AEGIS diajukan sebagai solusi untuk mengatasi permasalahan tersebut. Lalu penelitian dengan judul Security in Wireless Sensor Networks Using Lightweight Cryptography, penelitian tersebut berfokus untuk Wireless Sensor Networks (WSNs) memiliki banyak penggunaan dan aplikasi di tingkat individu dan organisasi. Sensor terintegrasi di banyak bidang termasuk kesehatan, lalu lintas, pertanian, industri, dan lain-lain. Kegunaan sensor adalah terkait dengan penggunaan banyak teknologi termasuk nirkabel protokol komunikasi, Internet of Things, Cloud Computing, komputasi mobile, dan teknologi berkembang lainnya. Ini interaksi yang melibatkan transmisi informasi penting dalam banyak kasus memaksakan kebutuhan untuk mengamankan data ini dari semua kemungkinan serangan. Lalu penelitian dengan judul Implementasi Algoritma SPECK dalam Sistem Monitoring Sumber Daya pada Raspberry Pi, penelitian tersebut berfokus Sistem monitoring merupakan sesuatu yang awam dilakukan saat ini. Namun perlu diperhatikan bahwa sistem keamanan juga diperlukan. Bila orang asing menguasai data sistem monitoring, maka dapat membahayakan pemilik sistem monitoring. Aspek yang perlu dijaga sistem monitoring adalah confidentiality. Algoritma SPECK dapat memenuhi keamanan confidentiality dan memiliki keunggulan pada sistem berspesifikasi rendah. Algoritma SPECK diimplementasikan pada sistem monitoring pada manager dan agent saat melakukan pengiriman dan penerimaan data [22], [23], [24], [25].

Walaupun pada penelitian tersebut telah dilakukan kriptografi ringan di IoI, namun metode kriptografi konvensional tidak baik untuk berskala dunia dengan sistem tertanam pada jaringan sensor. Dengan demikian, metode kriptografi ringan diusulkan untuk mengatasi banyak masalah kriptografi konvensional ini termasuk kendala yang berkaitan dengan ukuran fisik, persyaratan pemrosesan, keterbatasan memori dan yang menguras energi. Lalu Waktu eksekusi proses tanda tangan digital membutuhkan waktu lebih lama dibandingkan dengan waktu eksekusi proses verifikasi. Algoritma Elgamal signature scheme membutuhkan waktu dua kali lebih lama karena banyaknya perhitungan sistematis pada perangkat Arduino Uno. Lalu terdapat delay dalam pembacaan arduino dengan algoritma AEGIS. keamanan dunia maya dan serangan untuk Wireless Sensor Networks (WSNs) harus diperhatikan. Lalu hasil pengujian serangan aktif dengan metode known plaintext attack mengunakan algoritma SPECK mampu untuk dapat bertahan dari serangan tersebut selama 24 jam.

2. METODOLOGI PENELITIAN

2.1 Kerangka Dasar Penelitian

Ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat lain disebut kriptografi. Dalam kriptografi memiliki tiga tujuan utama yaitu:

a. Confidentiality: kriptografi kerahasiaan data yang bertujuan untuk memastikan informasi hanya pihak-pihak tertentu yang berhak mengakses pesan.
b. Integrity: kriptografi keutuhan data yang bertujuan pesan tidak ada perubahan dalam proses pengiriman atau penerimaan.
c. Authentication: kriptografi keaslian data yang bertujuan memastikan penerima bahwa pesan yang diterima merupakan pesan asli dari sumber yang dapat dipercaya.

Dengan 3 aspek utama tersebut kriptografi dibagi menjadi 2 pengelompokan menurut kunci yang digunakan yaitu algoritma simetris dan asimetris. Kriptografi dengan kunci simetris yang memiliki kunci sama pada proses enkripsi dan deskripsinya. Sedangkan kriptografi dengan kunci asimetris yang proses enkripsi dan deskripsi menggunakan kunci yang berbeda.

Journal of Informatics Management and Information Technology
ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

2.2 Tahapan Penelitian

Tujuan dari penelitian kami adalah untuk memahami apakah kriptografi ringan secara umum, dapat bekerja di IoT. langkah pertama dari penelitian ini, kami mengumpulkan kasus kriptografi ringan di IoT untuk mengumpulkan bukti dari literatur tentang tingkat kemampuan pembacaan IoT terhadap algoritma. Untuk mencapai tujuan itu, kami merumuskan masalah-masalah, sebagai berikut:

Pertanyaan 1 atau P1) Apakah saja persyaratan pemrosesan, keterbatasan memori yang dapat menguras energi pada krptografi ringan di IoT?

Pertanyaan 2 atau P2) Kenapa Waktu eksekusi proses tanda tangan digital membutuhkan waktu lebih lama dibandingkan dengan waktu eksekusi proses verifikasi untuk Tanda Tangan Digital Menggunakan Algoritma Elgamal Signature Scheme?

Pertanyaan 3 atau P3) Bagaimana delay dalam pembacaan arduino dengan algoritma AEGIS?

Pertanyaan 4 atau P4) Bagaimana keamanan dunia maya dan serangan untuk Wireless Sensor Networks (WSNs) yang harus diperhatikan?

Pertanyaan 5 atau P5) Bagaimana hasil pengujian serangan aktif dengan metode known plaintext attack mengunakan algoritma SPECK mampu untuk dapat bertahan dari serangan tersebut selama 24 jam?

P1, P2 dan P3 bertujuan untuk menemukan dalam literatur, penggunaan komponen apa saja yang ada pada kriptografi ringan dan IoT yang menjadi penyebab masalah.
P4 dan P5 bertujuan untuk menemukan dalam literatur, keamanan apa saja yang ada dan rentan akan serangan.

2.3 Proses pencarian

Untuk melakukan penelitian, peneliti mengumpulkan 5 penelitian, untuk memutuskan mana dari mereka yang dianalisis secara mendalam. Akhirnya, peneliti mendapat hasil, dari mana peneliti mengekstrak informasi yang diperlukan untuk menjawab pertanyaan.

3. HASIL DAN PEMBAHASAN

Pada bagian ini kami melaporkan hasil yang diambil dari makalah yang dianalisis, disusun berdasarkan pertanyaan penelitian. Diskusi tentang hasil akan mengikuti di Bagian IV.

3.1 Pertanyaan 1

P1 : Apakah saja persyaratan pemrosesan, keterbatasan memori yang dapat menguras energi pada krptografi ringan di IoT?

Dalam kasus implementasi perangkat lunak, implementasi ukuran dan konsumsi RAM melalui penempatan (*byte* per siklus) adalah metrik yang lebih disukai untuk aplikasi ringan. Semakin kecil semakin baik. Dalam kasus perangkat lunak, *framework Fair Evaluation of Lightweight Cryptographic Systems* (FELICS) diusulkan untuk mengevaluasi kinerja blok ringan atau kinerja *stream cipher* dalam ukuran implementasi, konsumsi RAM dan waktu yang dibutuhkan untuk melakukan operasi tertentu.

Table 1. *FELICS results for lightweight ciphers*

General info			AVR (8-bit)			MSP (16-bit)			ARM (32-bit)		
Name	**Block**	**Key**	**Code**	**RAM**	**Time**	**Code**	**RAM**	**Time**	**Code**	**RAM**	**Time**
Chaskey	128	128	770	84	1597	490	86	1351	178	80	614
SPECK	64	96	448	53	2829	328	48	959	256	56	1003
SPECK	64	128	452	53	2917	332	48	2013	276	60	972
Chaskey-LTS	128	128	770	84	413	492	86	2064	178	80	790
SIMON	64	96	600	57	4269	460	56	2905	416	64	1335
SIMON	64	128	608	57	4445	468	56	3015	388	64	1453
LEA	128	128	906	80	4023	722	78	2814	520	112	1171
RECTANGLE	64	128	602	56	4381	480	54	2651	452	76	2432
RECTANGLE	64	80	606	56	4433	480	54	2651	452	76	2338
SPARX	64	128	662	51	4397	580	52	2261	654	72	2338
SPARX	128	128	1184	74	5478	1036	72	3057	1468	104	2935
RC5-20	64	128	1068	63	8812	532	60	15,925	372	64	1919
AES	128	128	1246	81	3408	1170	80	4497	1348	124	4044
HIGHT	64	128	636	56	6231	636	52	7117	670	100	5532
Fantomas	128	128	1712	76	9689	1920	78	3602	2184	184	4550
Robin	128	128	2530	108	7813	1942	80	4913	2188	184	6250

Tabel 1 menunjukkan hasil FELICS dari algoritma cipher ringan populer untuk tiga implementasi yang berbeda: AVR 8-bit, MSP 6- bit dan ARM 32-bit. Salah satu yang menunjukkan untuk pengganti AES untuk kriptografi ringan

Journal of Informatics Management and Information Technology
ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

adalah *PRESENT*. Ini menggunakan ukuran blok yang lebih kecil dan potensi kunci yang lebih kecil (seperti kunci 80-bit). *PRESENT* pengguna baik 80-bit (10 karakter hex) atau kunci enkripsi 128-bit (16 karakter hex). Ini beroperasi pada blok 64-bit dan menggunakan substitusi-permutasi.

Table 2. *Lightweight hash functions*

Lightweight hash function	Digest [bits]	Code [bytes]	RAM [bytes]	RAM [bytes]	RAM stack	Cycle (8- byte msg)	Cycle (50- byte msg)	Cycle (100- byte msg)	Cycle (500- byte msg)
SPONGENT 256/256/128	256	364	16	96	5	1 542 923	3 856 916	6 170 900	25 454 100
SPONGENT 160/160/80	160	598	10	60	6	795 294	2 783 241	4 771 186	20 674 746
S-Quark	256	1106	4	60	5	708 783	1 417 611	2 339 023	9 4270 23
D-Quark	176	974	2	42	5	631 871	1 516 685	2 570 035	10 996 835
Keccak [r = 40, c = 160]	160	752	5	45	3	58 063	162 347	278 269	1 205 627
Keccak [r = 144, c = 256]	256	608	18	92	4	90 824	181 466	317 221	1 313 291
PHOTON-160 / 36/36	160	764	9	39	11	620 921	1 655 364	2 793 265	11 999 914
PHOTON-256 / 32/32	256	1244	4	68	10	254 871	486 629	787 896	3 105 396

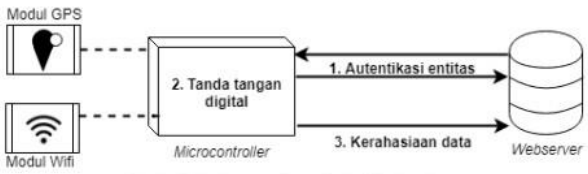
Dengan SPN, seperti halnya AES (Rijndael), kami beroperasi di blok plaintext dan menerapkan kunci dan kemudian menggunakan sejumlah putaran yang kami gunakan substitusi kotak (S-box) dan kotak permutasi (P-box). Operasi yang digunakan biasanya dicapai melalui XOR/rotasi bitwise, dan bagian dari kunci diperkenalkan melalui putaran operasi. Proses dekripsi kemudian kebalikan dari enkripsi putaran, dan S-box/P-box dibalik dalam operasinya.

CLEFIA adalah *cipher* blok ringan yang dipelajari dengan baik dan ditulis oleh Taizo Shirai, Kyoji Shibutani, Toru Akishita, Shiho Moriai dan Tetsu Iwata dan dapat diimplementasikan dengan gerbang 6K. Itu didefinisikan oleh Sony dan memiliki kunci 128, 192 dan 256 bit dan ukuran blok 128-bit. Bersamaan dengan ini, termasuk dalam Standar Internasional ISO/IEC 29192 untuk metode cipher blok ringan (ISO/IEC 29192-2:2012). RC5 ('Ron's *Cipher* 5'), dibuat pada tahun 1994 oleh Ron L. Rivest, juga menunjukkan potensi besar untuk metode kriptografi ringan. Ini adalah cipher blok yang memiliki ukuran blok variabel (32, 64 atau 128 bit), jumlah putaran variabel dan ukuran kunci variabel (0 hingga 2048 bit). Dengan demikian dapat digunakan untuk mencocokkan enkripsi dengan kemampuan perangkat. Jika itu adalah perangkat berdaya rendah dengan memori terbatas dan jejak fisik yang relatif kecil, kita dapat menggunakan ukuran blok 32-bit dan kunci 80-bit, hanya dengan beberapa putaran. Tetapi kami dapat meningkatkan keamanan jika perangkat dapat mengatasinya dan menggunakan ukuran blok 128-bit dan kunci 128-bit. Itu juga bisa fleksibel, di mana satu perubahan di kedua sisi dapat meningkatkan atau mengurangi persyaratan.

3.2 Pertanyaan 2

P2 : Kenapa Waktu eksekusi proses tanda tangan digital membutuhkan waktu lebih lama dibandingkan dengan waktu eksekusi proses verifikasi untuk Tanda Tangan Digital Menggunakan Algoritma Elgamal *Signature Scheme*?

Sistem dirancang untuk menyediakan tiga layanan keamanan. Tiga layanan keamanan tersebut adalah autentikasi entitas, tanda tangan digital, dan kerahasiaan data. Lingkungan simulasi pada penelitian ini dibangun pada sistem keamanan IoT yang memiliki tiga tahapan utama seperti pada Gambar 1. Tahapan pertama yang dilakukan adalah Arduino melakukan inisialisasi ke server untuk pertukaran kunci atau autentikasi entitas. Tahapan kedua, yaitu pembuatan tanda tangan digital pada data lokasi mikrokontroler untuk autentikasi asal data. Tahapan ketiga, yaitu Arduino melakukan enkripsi pada data sebelum dikirimkan ke server untuk menjaga kerahasian data.



Gambar 1. Lingkungan sistem simulasi keseluruhan

Proses pembangkitan kunci dilakukan oleh pengirim. Kunci yang dibangkitkan berguna untuk memberikan tanda tangan digital pada data. Kunci yang dibangkitan terdapat dua jenis, yaitu kunci private dan kunci public. Berikut pada Tabel 1 adalah hasil dari proses pembangkitan kunci.

Tabel 3. Hasil pembangkitan kunci

Parameter	Nilai
p	61091
G	60383

Journal of Informatics Management and Information Technology
ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

k	60719
Kunci private	60647
Kunci public	35865

Tabel 4. Hasil pembangkitan tanda tangan digital

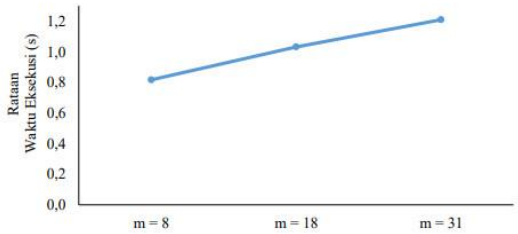
Parameter	Nilai
Data (m)	2147483647
r	32914
S	25611

Verifikasi tanda tangan digital menghasilkan hasil perbandingan dari dua proses komputasi. Berikut pada Tabel 5 adalah hasil verifikasi keabsahan tanda tangan digital. Berdasarkan pada Tabel 5, perbandingan dari hasil dua proses tersebut memiliki nilai yang sama yang artinya verifikasi keabsahan tanda tangan digital berhasil dilakukan.

Tabel 5. Hasil verifikasi tanda tangan digital

Proses Komputasi	Hasil
$G^m \bmod p$	16717
Kunci public^r $r^s \bmod p$	16717

Hasil analisis kinerja yang dilakukan pada Arduino Uno dan pada PC. Analisis kinerja yang dilakukan berupa analisis kinerja terhadap waktu eksekusi. Pengukuran kinerja algoritma diukur bedasarkan panjang bilangan acak prima (p) yang digunakan, panjang data (m) yang digunakan, dan lama waktu eksekusi algoritma.



Gambar 2. Panjang Message (m)

Analisis keamanan algoritma dilihat dari hasil verifikasi *signature* saat ada perubahan pada m dan pada pasangan tanda tangan digital (r,s). Pada Tabel 4 dan 5, perubahan pada m ataupun data pasangan tanda tangan digital (r,s) menyebabkan perbedaan nilai pada proses komputasi 1 dan proses komputasi 2. Perbedaan nilai yang dihasilkan menujukkan bahwa proses verifikasi telah gagal. Panjang kunci private dihasilkan pada penelitian ini sebesar 18 bit. Jika dilakukan metode brute force, maka akan menghasilkan 2 18 = 262 144 kemungkinan bilangan untuk menebak nilai dari kunci private yang dihasilkan. Jika brute force dilakukan pada PC penelitian ini membutuhkan waktu sekitar 262 detik atau sekitar empat menit. Idealnya untuk panjang kunci asimetris, minimal 2048 bit seperti pada RSA.

Tabel 6. Data awal dan data sesudah diubah

	Nilai		
Parameter	**Data Awal**	**Data Pesan**	**Data(r,s) diubah**
Pesan	2011684	2011876	2011684
r	20965	20965	20432
s	481	481	555

Tabel 7. Hasil verifikasi dengan data awal dan data setelah diubah

	Nilai		
Parameter	**Data Awal**	**Data Pesan diubah**	**Data(r,s) diubah**
Proses Komputasi 1	29154	27233	27223
Proses Komputasi 1	29541	29154	39187

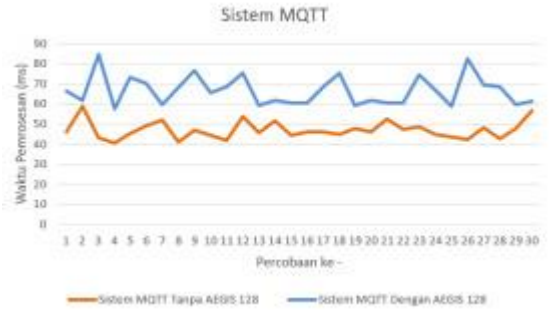
3.3 Pertanyaan 3

P3 : Bagaimana *delay* dalam pembacaan arduino dengan algoritma AEGIS?

Pengujian kinerja waktu sistem bertujuan untuk mengetahui perbedaan signifikan karena implementasi AEGIS 128. Pada sistem *middleware* yang menggunakan protokol MQTT atau menggunakan protokol CoAP akan diuji data sampel waktu pemrosesan saat AEGIS 128 diimplementasikan dan tanpa AEGIS 128. Dari hasil data sampel waktu pemrosesan pada masing-masing sistem akan dibandingkan. Grafik pada Gambar 3 menampilkan perbedaan waktu sistem dengan menggunakan protokol MQTT saat proses pengiriman dilakukan dari perangkat nodeMCU ke *client* tanpa AEGIS

Journal of Informatics Management and Information Technology
ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

128 dan pada sistem menggunakan AEGIS 128. Dari hasil rata-rata selisih sistem MQTT menggunakan Algoritma AEGIS 128 dan tanpa Algoritma AEGIS 128 adalah 19,592 milisecond.



Gambar 3. Grafik hasil perbandingan waktu proses sistem MQTT

Gambar 4. Grafik hasil perbandingan waktu proses sistem CoAP

Grafik pada Gambar 4 menampilkan perbedaan waktu sistem dengan menggunakan protokol CoAP saat proses pengiriman dilakukan dari perangkat nodeMCU ke *middleware* tanpa AEGIS 128 dan pada sistem menggunakan AEGIS 128. Hasil rata-rata selisih sistem CoAP menggunakan Algoritma AEGIS 128 dan tanpa Algoritma AEGIS 128 adalah 28,141 milisecond.

Pengujian *Quality of Service* (QoS) merupakan pengujian untuk mengukur kualitas layanan yang diperoleh dari suatu jaringan komunikasi. Dasar pengujian QoS yaitu menganalisis parameter dari *packet loss*, *delay* dan *jitter*. *Packet loss* merupakan indikasi bahwa terdapat paket yang hilang atau gagal terkirim ke tujuan. Nilai packet loss didapatkan dari nilai *expected* dan nilai actual. Nilai *expected* yaitu nilai harapan dari banyak paket yang terkirim ke tujuan, sedangkan nilai actual adalah nilai dari paket yang berhasil terkirim ke tujuan. *Delay* adalah waktu jeda paket terkirim dari sumber sampai dengan waktu pengiriman ACK dari tujuan. Jitter adalah variasi nilai dari *delay* pengiriman paket. Berdasarkan Tabel 1 hasil pengujian pengiriman data dengan protokol MQTT dari perangkat nodeMCU ESP8266 ke *middleware* tanpa mekanisme keamanan. Menghasilkan *success rate* 100 % dan packet loss 0%. Dengan nilai rata-rata *delay* adalah 25,7 detik dan jitter adalah 1,41 detik.

Tabel 8. Hasil pengujian skenario MQTT

Expected	Actual	Success Rate	Packet Loss rate	Delay	Jitter
20	20	100%	0%	24.8	1.36
20	20	100%	0%	26.0	1.43
20	20	100%	0%	2.8	1.43
20	20	100%	0%	26.0	1.43
20	20	100%	0%	26.0	1.43
	Rata-Rata (detik)			**25.7**	**1.41**

Tabel 9. Hasil pengujian scenario CoAP

Expected	Actual	Success Rate	Packet Loss rate	Delay	Jitter
20	20	100%	0%	14.3	0.0013
20	20	100%	0%	14.3	0.0013
20	20	100%	0%	14.3	0.0013

Journal of Informatics Management and Information Technology
ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

20	20	100%	0%	14.3	0.0013
20	20	100%	0%	14.3	0.0013
	Rata-Rata (detik)			14.3	0.0013

Berdasarkan Tabel 8 hasil pengujian pengiriman data dengan protokol CoAP dari sensor nodeMCU ESP8266 ke *middleware* tanpa mekanisme keamanan. Menghasilkan *success rate* 100 % dan packet loss 0 %. Dengan nilai rata-rata delay adalah 14,3 detik dan jitter adalah 0,0013 detik.

Berdasarkan Tabel 9 hasil pengujian pengiriman data pada protokol MQTT dari sensor nodeMCU ESP8266 ke middleware dengan Algoritma AEGIS 128. Menghasilkan *success rate* 99 % dan *packet loss* 1%. Dengan nilai rata-rata delay adalah 26,3 detik dan jitter adalah 1,46 detik.

Tabel 10. Hasil pengujian skenario MQTT AEGIS 128

Expected	Actual	Success Rate	Packet Loss rate	Delay	Jitter
20	19	95%	5%	26.18	1.36
20	20	100%	0%	26.37	1.45
20	20	100%	0%	26.37	1.45
20	20	100%	0%	26.37	1.45
20	20	100%	0%	26.37	1.45
	Rata-Rata (detik)			26.37	1.46

Tabel 11. Hasil pengujian skenario MQTT AEGIS 128

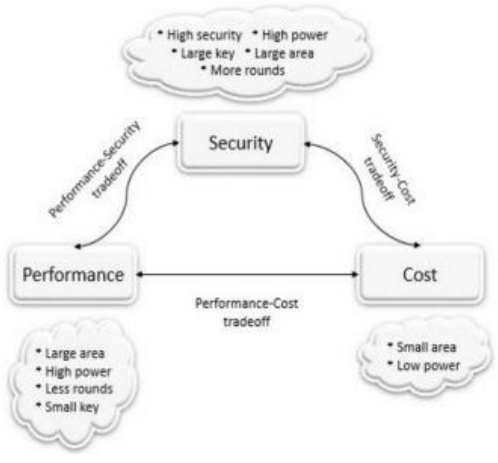
Expected	Actual	Success Rate	Packet Loss rate	Delay	Jitter
20	19	95%	5%	14.6	0.0021
20	20	100%	0%	14.6	0.0018
20	20	100%	0%	14.6	0.0018
20	18	90%	10%	14.6	0.0023
20	16	100%	20%	14.4	0.0025
	Rata-Rata (detik)			14.5	0.0021

Berdasarkan Tabel 11 hasil pengujian pengiriman data dengan protokol CoAP dari sensor nodeMCU ESP8266 ke middleware dengan Algoritma AEGIS 128. Menghasilkan *success rate* 93 % dan *packet loss* 7 %. Nilai rata-rata *delay* adalah 14,5 detik dan *jitter* adalah 0,0021 detik.

3.4 Pertanyaan 4

P4 : Bagaimana keamanan dunia maya dan serangan untuk *Wireless Sensor Networks* (WSNs) yang harus diperhatikan?

Kriptografi Ringan mengacu pada satu set algoritma kriptografi simetris dan asimetris yang dirancang untuk memastikan keamanan yang memadai dalam sistem yang memiliki kendala khusus untuk menjalankan lingkungan dan kemampuan daya seperti WNS. Ringan dalam kriptografi dapat dicapai pada tingkat perangkat keras dan perangkat lunak, di mana ukuran chip, energi di tingkat perangkat keras, ukuran kode, dan kompleksitas RAM di tingkat perangkat lunak digunakan untuk mengukur tingkat optimasi yang dipenuhi dengan menerapkan LWC. Pandangan kriptografer adalah bahwa implementasi perangkat keras dan perangkat lunak dioptimalkan, dengan mengingat bahwa apa yang ringan untuk implementasi perangkat keras mungkin tidak ringan untuk implementasi perangkat lunak dan sebaliknya



Gambar 5. Desain Algoritma Kriptografi Ringan

Journal of Informatics Management and Information Technology
ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

a. Pendekatan Kriptografi Ringan Simetris untuk persyaratan minimal dan kinerja optimal, Sebagai berikut:
   1. *Advanced Encryption Standard* (AES): Ini adalah cipher blok standar NIST yang beroperasi pada blok 128-bit dan menggunakan ukuran kunci 256 bit, 192 bit, atau 128 bit. Enkripsi dapat dilakukan dengan sepuluh, 12, atau 14 putaran tergantung pada ukuran input. Setiap putaran AES melewati empat langkah dasar: sub byte, shift rows, MixedColumns, dan AddRoundKey. *Array byte* diperlakukan sebagai matriks 4 x 4. Implementasi AES 128-bit yang ringan membutuhkan 3100 GE (gate setara) dan menawarkan *throughput* 80 Kb/s pada daya 100 KHz. Namun, serangan Man-in-the-Middle (MITM) masih mengancam AES Ringan.
   2. DESLX: varian DES (*Data Encryption Standard*) ringan yang beroperasi pada blok 64-bit menggunakan kunci 184-bit. Enkripsi menggunakan DESLX terjadi dengan 16 putaran. Perubahan signifikan diterapkan pada DES standar untuk mencapai tampilan DESLX yang ringan, seperti hanya menggunakan satu S-box, bukan delapan Sbox dan menerapkan pendekatan pemutihan esensial yang memerlukan dua gerbang XOR untuk ditambahkan ke desain: satu untuk pra -*whitening* pada *plaintext* dan satu untuk *post-whitening* pada *ciphertext* . Implementasi DES yang ringan ini membutuhkan 2168 GE (*gate* setara) dan menawarkan *throughput* 44,4 Kb/s pada daya 100 KHz [28]. Serangan yang mengancam DES tidak serta merta mengancam DESLX karena properti dari satu S-box yang digunakan dalam DESLX berbeda dengan properti dari DES S-box.
   3. Keamanan Tinggi dan Ringan *HIGHT*: cipher blok ringan beroperasi pada blok 64-bit menggunakan kunci 128-bit yang dihasilkan pada enkripsi dan dekripsi menggunakan struktur GFS. Enkripsi menggunakan *HIGHT* terjadi dengan 32 putaran. Implementasi ringan ini membutuhkan 3048 GE (setara gerbang) dan menawarkan throughput 188,2 Kb/s pada daya 100 KHz. Block cipher ini rentan terhadap serangan saturasi.
   4. *PRESENT*: Ini dianggap sebagai algoritma kriptografi ultra-ringan yang beroperasi pada blok 64-bit menggunakan ukuran kunci 80-bit atau 128-bit. Enkripsi menggunakan PRESENT terjadi dalam 32 putaran. Lapisan substitusi SPN saat ini menggunakan S-box 4-bit untuk input dan output. Block cipher ini dikenal dengan tingkat keamanan yang tinggi dan kesederhanaannya. *PRESENT* membutuhkan hingga 1570 GE (setara gerbang) saat menggunakan kunci 80-bit dan hingga 1884 GE saat menggunakan kunci 128-bit, dan menawarkan throughput hingga 200 Kb/s pada daya 100 KHz. *PRESENT* rentan terhadap serangan diferensial.
   5. KATAN dan KTANTAN: Ini adalah stream cipher yang beroperasi pada blok ukuran 32-bit, 48-bit atau 64-bit dan keduanya menggunakan kunci 80-bit. Enkripsi oleh KATAN dan KTANTAN terjadi dalam 254 putaran. Perbedaan utama antara desain adalah bahwa KTANTAN membutuhkan sekitar setengah dari ekuivalen gerbang yang dibutuhkan KATAN untuk implementasi perangkat keras dengan semua ukuran blok. Untuk blok berukuran 32-bit, KTANTAN membutuhkan 462 EG, sedangkan KATAN membutuhkan 802 EG. Untuk blok berukuran 48-bit, KTANTAN membutuhkan 588 EG, sedangkan KATAN membutuhkan 927 EG. Akhirnya, untuk blok 64-ukuran bit, KTANTAN membutuhkan 688 EG, sedangkan KATAN membutuhkan 1054 EG. KATAN dan KTANTAN menawarkan throughput 12,5, 18,8 dan 25,1 untuk Kb/s pada daya 100 KHz saat mengenkripsi blok masing-masing berukuran 32-bit, 48-bit, dan 64-bit. Kedua cipher rentan terhadap serangan diferensial.
   6. *PRINCE*: sandi ringan yang dikenal menggunakan properti refleksi: enkripsi dan dekripsinya sama, tetapi masing-masing menggunakan kunci yang berbeda untuk mengurangi persyaratan desain. *Cipher* ini beroperasi pada blok berukuran 64-bit menggunakan kunci 128-bit, dan menggunakan struktur SPN. RSA: pendekatan boros sumber daya yang tidak terlalu fungsional untuk pengoptimalan ringan. RSA bergantung pada pemilihan dua bilangan prima besar untuk menemukan kunci publik dan kunci privatnya, yang biasanya antara 1024 dan 4096 bit. 121 Enkripsi oleh *PRINCE* terjadi dalam 12 putaran *PRESENT* membutuhkan hingga 3491 GE (setara gerbang) dan menawarkan *throughput* hingga 533,3 Kb/s pada daya 100 KHz. *PRINCE* rentan terhadap serangan refleksi.

Tabel 12. Pendekatan LWC Simetris

Cipher Name	Block Size	Key Size	# of Rounds	Structure	# of GEs	Throughput Kb/s at 100 Khz	Vulnerable Attacks
AES	128-bit	128-bit 192-bit 256-bit	12 14 16	SPN	3100	80	MITM Attack
DESLX	64-bit	184-bit	64	Feistel	2168	44,4	-
HIGHT	64-bit	128-bit	32	GFS	3048	188,2	Saturation Attack
PRESENT	64 bit	80-bit 128-bit	31	SPN	Up to 1570 Up to 1884	Up to 200	Differential Attack
PRINCE	64-bit	128-bit	12	SPN	Up to 3491	Up to 533,3	Reflection Attack
KATAN	32-bit 48-bit 64-bit	80-bit	254	Stream-Cipher	802 927 1054	12,5 18,8 25,1	Differential Attack
KTANTAN	32-bit 48-bit	80-bit	254	Stream-Cipher	462 588	12,5 18,8	Differential Attack

ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

	64-bit				688	25,1	
TWIN	64-bit	80-bit 128-bit	36	GFN	1799 2285	178	Biclique Attack

b. Pendekatan Kriptografi Ringan Asimetris, Sebagai berikut:
   1. Kriptografi Kurva Elliptik ECC: ECC ringan dianggap sebagai metode kunci publik yang paling efisien karena memerlukan konsumsi daya yang lebih sedikit, area yang lebih kecil, dan siklus clock yang lebih sedikit. Ini didasarkan pada sistem aljabar dan menggunakan kunci kecil yang dihasilkan menggunakan algoritma diskrit. Beberapa implementasi yang dioptimalkan diusulkan untuk ECC ringan. Sebagian besar desain membutuhkan tidak kurang dari 10.000 GE. Lima operasi disediakan dalam aritmatika medan utama: Perkalian, Penambahan, Pengurangan, Bagi-bagi, dan Pengurangan. Mengurangi kompleksitas operasi Perkalian adalah yang paling layak dari operasi ini karena dapat dilakukan untuk implementasi hingga 30 k GE. Optimalisasi terjadi dengan menggunakan bit shifting dan penambahan untuk perkalian daripada menggunakan mikroprosesor
   2. RSA: pendekatan boros sumber daya yang tidak terlalu fungsional untuk pengoptimalan ringan. RSA bergantung pada pemilihan dua bilangan prima besar untuk menemukan kunci publik dan kunci privatnya, yang biasanya antara 1024 dan 4096 bit. Pendekatan konvensional kriptografi asimetris masih belum menjanjikan untuk bobot ringan karena tidak ada implementasi yang diusulkan memiliki waktu eksekusi yang wajar.

3.5 Pertanyaan 5

P5 : Bagaimana hasil pengujian serangan aktif dengan metode *known plaintext attack* mengunakan Algoritma SPECK mampu untuk dapat bertahan dari serangan tersebut selama 24 jam?

Pengujian serangan yang dilakukan pada penelitian oleh Yohanes Heryka Febriarso , Ari Kusyanti dan Primantara Hari Trisnawan (2021), dengan judul Implementasi Algoritma SPECK dalam Sistem Monitoring Sumber Daya pada Raspberry Pi. Hasil dari pengujian serangan pasif menggunakan alat bantu yaitu aplikasi *Wireshark*. *Wireshark* memiliki skenario pihak ketiga melakukan serangan *sniffing*, dengan aplikasi *Wireshark* dengan kondisi sistem monitoring belum dilakukan implementasi *Algoritma SPECK*. Bahwa komunikasi antar manager dan agent dapat dilihat dengan jelas oleh aplikasi *Wireshark*. Sedangkan kondisi sistem monitoring sudah diimplementasikan *Algoritma SPECK*. Aplikasi *Wireshark* hanya melihat angka biner acak saja. *Command line* pada Windows 10 menunjukkan pengujian serangan aktif dengan cara *known plaintext attack* untuk mendapatkan pasangan key dan *initialization vector* (IV) dengan kondisi penyerang sudah memiliki plaintext. Serangan dilakukan selama 24 jam, akan tetapi penyerang masih tidak mendapatkan pasangan key dan IV yang benar.

3.6 Pembahasan

Penggunaan komponen apa saja yang ada pada kriptografi ringan dan IoT yang menjadi penyebab masalah. Dalam kasus implementasi perangkat lunak, implementasi ukuran dan konsumsi RAM melalui penempatan (byte per siklus) adalah metrik yang lebih disukai untuk aplikasi ringan. Semakin kecil semakin baik. Dalam kasus perangkat lunak, framework Fair Evaluation of Lightweight Cryptographic Systems (FELICS) diusulkan untuk mengevaluasi kinerja blok ringan atau kinerja stream cipher dalam ukuran implementasi, konsumsi RAM dan waktu yang dibutuhkan untuk melakukan operasi tertentu. CLEFIA adalah cipher blok ringan yang dipelajari dengan baik dan ditulis oleh Taizo Shirai, Kyoji Shibutani, Toru Akishita, Shiho Moriai dan Tetsu Iwata dan dapat diimplementasikan dengan gerbang 6K. Dengan demikian dapat digunakan untuk mencocokkan enkripsi dengan kemampuan perangkat. Jika itu adalah perangkat berdaya rendah dengan memori terbatas dan jejak fisik yang relatif kecil, kita dapat menggunakan ukuran blok 32-bit dan kunci 80-bit, hanya dengan beberapa putaran. Tetapi kami dapat meningkatkan keamanan jika perangkat dapat mengatasinya dan menggunakan ukuran blok 128-bit dan kunci 128-bit. Itu juga bisa fleksibel, di mana satu perubahan di kedua sisi dapat meningkatkan atau mengurangi persyaratan.

Waktu eksekusi proses tanda tangan digital membutuhkan waktu lebih lama dibandingkan dengan waktu eksekusi proses verifikasi untuk Tanda Tangan Digital Menggunakan Algoritma Elgamal Signature Scheme. Proses pembangkitan kunci dilakukan oleh pengirim. Kunci yang dibangkitkan berguna untuk memberikan tanda tangan digital pada data. Verifikasi tanda tangan digital menghasilkan hasil perbandingan dari dua proses komputasi. Pada Tabel 5 adalah hasil verifikasi keabsahan tanda tangan digital. Berdasarkan pada Tabel 5, perbandingan dari hasil dua proses tersebut memiliki nilai yang sama yang artinya verifikasi keabsahan tanda tangan digital berhasil dilakukan. Analisis keamanan Algoritma dilihat dari hasil verifikasi signature saat ada perubahan pada m dan pada pasangan tanda tangan digital (r,s).Namun pada Tabel 4 dan 5, perubahan pada m ataupun data pasangan tanda tangan digital (r,s) menyebabkan perbedaan nilai pada proses komputasi 1 dan proses komputasi 2. Perbedaan nilai yang dihasilkan menujukkan bahwa proses verifikasi telah gagal. Panjang kunci private dihasilkan pada penelitian ini sebesar 18 bit. Jika dilakukan metode brute force, maka akan menghasilkan 2 18 = 262 144 kemungkinan bilangan untuk menebak nilai dari kunci private yang dihasilkan. Jika brute force dilakukan pada PC penelitian ini membutuhkan waktu sekitar 262 detik atau sekitar empat menit. Idealnya untuk panjang kunci asimetris, minimal 2048 bit seperti pada RSA.

Delay dalam pembacaan arduino dengan Algoritma AEGIS. Pengujian kinerja waktu sistem bertujuan untuk mengetahui perbedaan signifikan karena implementasi AEGIS 128. Pada sistem middleware yang menggunakan protokol MQTT atau menggunakan protokol CoAP akan diuji data sampel waktu pemrosesan saat AEGIS 128 diimplementasikan dan tanpa AEGIS 128. Pengujian Quality of Service (QoS) merupakan pengujian untuk mengukur kualitas layanan yang

ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

diperoleh dari suatu jaringan komunikasi. Dasar pengujian QoS yaitu menganalisis parameter dari packet loss, delay dan jitter. Packet loss merupakan indikasi bahwa terdapat paket yang hilang atau gagal terkirim ke tujuan. Nilai packet loss didapatkan dari nilai expected dan nilai actual. Nilai expected yaitu nilai harapan dari banyak paket yang terkirim ke tujuan, sedangkan nilai actual adalah nilai dari paket yang berhasil terkirim ke tujuan. Delay adalah waktu jeda paket terkirim dari sumber sampai dengan waktu pengiriman ACK dari tujuan. Jitter adalah variasi nilai dari delay pengiriman paket. Namun Berdasarkan Tabel 8 hasil pengujian pengiriman data dengan protokol CoAP dari sensor nodeMCU ESP8266 ke middleware tanpa mekanisme keamanan. Menghasilkan success rate 100 % dan packet loss 0 %. Dengan nilai rata-rata delay adalah 14,3 detik dan jitter adalah 0,0013 detik. Berdasarkan Tabel 9 hasil pengujian pengiriman data pada protokol MQTT dari sensor nodeMCU ESP8266 ke middleware dengan Algoritma AEGIS 128. Menghasilkan success rate 99 % dan packet loss 1%. Dengan nilai rata-rata delay adalah 26,3 detik dan jitter adalah 1,46 detik.

Keamanan dunia maya dan serangan untuk Wireless Sensor Networks (WSNs). Kriptografi Ringan mengacu pada satu set algoritma kriptografi simetris dan asimetris yang dirancang untuk memastikan keamanan yang memadai dalam sistem yang memiliki kendala khusus untuk menjalankan lingkungan dan kemampuan daya seperti WNS. Ringan dalam kriptografi dapat dicapai pada tingkat perangkat keras dan perangkat lunak, di mana ukuran chip, energi di tingkat perangkat keras, ukuran kode, dan kompleksitas RAM di tingkat perangkat lunak digunakan untuk mengukur tingkat optimasi yang dipenuhi dengan menerapkan LWC. Namun, serangan Man-in-the-Middle (MITM) masih mengancam AES Ringan. Serangan yang mengancam DES tidak serta merta mengancam DESLX karena properti dari satu S-box yang digunakan dalam DESLX berbeda dengan properti dari DES S-box. HIGHT Block cipher ini rentan terhadap serangan saturasi. PRESENT, KATAN dan KTANTAN rentan terhadap serangan diferensial. PRINCE rentan terhadap serangan refleksi.

Hasil pengujian serangan aktif dengan metode known plaintext attack mengunakan Algoritma SPECK mampu untuk dapat bertahan dari serangan tersebut selama 24 jam. hasil dari pengujian serangan pasif menggunakan alat bantu yaitu aplikasi Wireshark. Pada Gambar 6 memiliki skenario pihak ketiga melakukan serangan sniffing dengan aplikasi Wireshark dengan kondisi sistem monitoring belum dilakukan implementasi Algoritma SPECK. Dapat dilihat bahwa pada Gambar 7 memperlihatkan komunikasi antar manager dan agent dapat dilihat dengan jelas oleh aplikasi Wireshark. Sedangkan Gambar 6 memiliki kondisi sistem monitoring sudah diimplementasikan Algoritma SPECK. Dapat dilihat bahwa pada Gambar 8 aplikasi Wireshark hanya melihat angka biner acak saja. Pada Gambar 8 menunjukkan pengujian serangan aktif dengan cara known plaintext attack untuk mendapatkan pasangan key dan initialization vector (IV) dengan kondisi penyerang sudah memiliki plaintext. Serangan dilakukan selama 24 jam, akan tetapi penyerang masih tidak mendapatkan pasangan key dan IV yang benar. Namun apabila adanya serangan aktif selama 24 jam perlu adanya uji coba kembali.

4. KESIMPULAN

Peneliti melakukan Tinjauan Literatur Sistematik untuk menyelidiki kasus penggunaan kriptografi ringan dalam literatur dan faktor mana yang memengaruhi metode kriptografi konvensional tidak baik untuk berskala dunia dengan sistem tertanam pada jaringan sensor. Dengan demikian, metode kriptografi ringan diusulkan untuk mengatasi banyak masalah kriptografi konvensional ini termasuk kendala yang berkaitan dengan ukuran fisik, persyaratan pemrosesan, keterbatasan memori dan yang menguras energi. Lalu Waktu eksekusi proses tanda tangan digital membutuhkan waktu lebih lama dibandingkan dengan waktu eksekusi proses verifikasi. Algoritma Elgamal signature scheme membutuhkan waktu dua kali lebih lama karena banyaknya perhitungan sistematis pada perangkat Arduino Uno. Lalu terdapat delay dalam pembacaan arduino dengan Algoritma AEGIS. keamanan dunia maya dan serangan untuk Wireless Sensor Networks (WSNs) harus diperhatikan. Lalu hasil pengujian serangan aktif dengan metode known plaintext attack mengunakan Algoritma SPECK mampu untuk dapat bertahan dari serangan tersebut selama 24 jam. Dalam penelitian ini dilakukan untuk mengatasi masalah di masa depan akan terdiri dari pengujian kriptografi yang aman dan paling cocok dalam merancang arsitektur berlapis untuk aplikasi IoT.

REFERENCES

[1] F. Rozi, H. Amnur, F. Fitriani, and P. Primawati, "Home Security Menggunakan Arduino Berbasis Internet Of Things," *INVOTEK: Jurnal Inovasi Vokasional dan Teknologi*, vol. 18, no. 2, pp. 17–24, 2018, doi: 10.24036/invotek.v18i2.287.

[2] S. Bresciani, A. Ferraris, and M. Del Giudice, "The management of organizational ambidexterity through alliances in a new context of analysis: Internet of Things (IoT) smart city projects," *Technol Forecast Soc Change*, vol. 136, pp. 331–338, 2018, doi: 10.1016/j.techfore.2017.03.002.

[3] M. U. Ali, S. Hur, and Y. Park, "Wi-Fi-based effortless indoor positioning system using IoT sensors," *Sensors (Switzerland)*, vol. 19, no. 7, 2019, doi: 10.3390/s19071496.

[4] M. Mohammadi, A. Al-Fuqaha, M. Guizani, and J. S. Oh, "Semisupervised Deep Reinforcement Learning in Support of IoT and Smart City Services," *IEEE Internet Things J*, vol. 5, no. 2, pp. 624–635, 2018, doi: 10.1109/JIOT.2017.2712560.

[5] M. S. M. Saleh, A. A. B. Sajak, R. Mohamad, and M. A. M. Zaaba, "IoT Real-Time Soil Monitoring Based on LoRa for Palm Oil Plantation," *J Phys Conf Ser*, vol. 1874, no. 1, 2021, doi: 10.1088/1742-6596/1874/1/012047.

[6] V. A. Memos, K. E. Psannis, Y. Ishibashi, B. G. Kim, and B. B. Gupta, "An Efficient Algorithm for Media-based Surveillance System (EAMSuS) in IoT Smart City Framework," *Future Generation Computer Systems*, vol. 83, pp. 619–628, 2018, doi: 10.1016/j.future.2017.04.039.

ISSN 2774-4744 (Media Online)
Vol 4, No 3, July 2024 | Hal 81-91
DOI: 10.47065/ jimat.v4i3.403
https://hostjournals.com/jimat

[7] F. Ullah, M. R. Naeem, L. Mostarda, and S. A. Shah, "Clone detection in 5G-enabled social IoT system using graph semantics and deep learning model," *International Journal of Machine Learning and Cybernetics*, 2021, doi: 10.1007/s13042-020-01246-9.
[8] B. El Boudani *et al.*, "Implementing deep learning techniques in 5g iot networks for 3d indoor positioning: Delta (deep learning-based co-operative architecture)," *Sensors (Switzerland)*, vol. 20, no. 19, pp. 1–20, 2020, doi: 10.3390/s20195495.
[9] D. K. Sawlani, *Keputusan Pembelian Online: Kualitas Website, Keamanan dan Kepercayaan*. Surabaya: Scopindo Media Pustaka, 2021.
[10] Fachruddin, Saparudin, E. Rasywir, and Y. Pratama, "Network and layer experiment using convolutional neural network for content based image retrieval work," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 20, no. 1, pp. 118–128, 2022, doi: 10.12928/TELKOMNIKA.v20i1.19759.
[11] X. Li, L. Bing, W. Lam, and B. Shi, "Transformation Networks for Target-Oriented Sentiment Classification," *ArXiv*, 2018.
[12] Z. Jianqiang, G. Xiaolin, and Z. Xuejun, "Deep Convolution Neural Networks for Twitter Sentiment Analysis," *IEEE Access*, vol. 6, no. c, pp. 23253–23260, 2018, doi: 10.1109/ACCESS.2018.2776930.
[13] H. T. Nguyen and M. Le Nguyen, "Effective Attention Networks for Aspect-level Sentiment Classification," *Proceedings of 2018 10th International Conference on Knowledge and Systems Engineering, KSE 2018*, pp. 25–30, 2018, doi: 10.1109/KSE.2018.8573324.
[14] Y. Pratama and E. Rasywir, "Automatic Cost Estimation Analysis on Datawarehouse Project with Modified Analogy Based Method," in *Proceedings of 2018 International Conference on Electrical Engineering and Computer Science, ICECOS 2018*, IEEE, 2019, pp. 171–176. doi: 10.1109/ICECOS.2018.8605195.
[15] D. Zaenal Abidin and E. Rasywir, "Penerapan Data Mining Klasifikasi Untuk Memprediksi Potensi Mahasiswa Berprestasi Di Stikom Dinamika Bangsa Jambi Dengan Metode Naive Bayes," *Jurnal Ilmiah Mahasiswa Teknik Informatika*, vol. 3, no. 2, 2021.
[16] S. Assegaff, E. Rasywir, and Y. Pratama, "Experimental of vectorizer and classifier for scrapped social media data," vol. 21, no. 4, pp. 815–824, 2023, doi: 10.12928/TELKOMNIKA.v21i4.24180.
[17] H. Prastiwi, J. Pricilia, and E. Raswir, "Implementasi Data Mining Untuk Menentuksn Persediaan Stok Barang Di Mini Market Menggunakan Metode K-Means Clustering Jurnal Informatika Dan Rekayasa Komputer (JAKAKOM)," *Jurnal Informatika Dan Rekayasa Komputer (JAKAKOM)*, vol. 1, no. April, pp. 141–148, 2022.
[18] E. Rasywir, R. Sinaga, and Y. Pratama, "Evaluasi Pembangunan Sistem Pakar Penyakit Tanaman Sawit dengan Metode Deep Neural Network (DNN)," vol. 4, pp. 1206–1215, 2020, doi: 10.30865/mib.v4i4.2518.
[19] R. Munir, *Kriptografi*, Kedua. Bandung: Informatika Bandung, 2019.
[20] R. F. Malik *et al.*, "The Indoor Positioning System Using Fingerprint Method Based Deep Neural Network," *IOP Conf Ser Earth Environ Sci*, vol. 248, no. 1, 2019, doi: 10.1088/1755-1315/248/1/012077.
[21] X. Ding and R. Yang, "Vehicle and Parking Space Detection Based on Improved YOLO Network Model," *J Phys Conf Ser*, vol. 1325, no. 1, 2019, doi: 10.1088/1742-6596/1325/1/012084.
[22] F. Ubaidillah and I. M. Suartana, "Analisis Peforma Multimedia Streaming Menggunakan Clustering Controller Pada Software Defined Network," *Journal of Informatics and …*, vol. 03, pp. 207–215, 2021.
[23] M. Syani, "Implementasi Intrusion Detection System (Ids) Menggunakan Suricata Pada Linux Debian 9 Berbasis Cloud Virtual Private Servers (Vps)," *Jurnal Inkofar*, vol. 1, no. 1, pp. 13–20, 2020, doi: 10.46846/jurnalinkofar.v1i1.155.
[24] J. Wei, C. H. Chiu, F. Huang, J. Zhang, and C. Cai, "A cost-effective decentralized vehicle remote positioning and tracking system using BeiDou Navigation Satellite System and Mobile Network," *EURASIP J Wirel Commun Netw*, vol. 2019, no. 1, pp. 0–8, 2019, doi: 10.1186/s13638-019-1436-y.
[25] S. Subedi and J. Y. Pyun, "A survey of smartphone-based indoor positioning system using RF-based wireless technologies," *Sensors (Switzerland)*, vol. 20, no. 24, pp. 1–32, 2020, doi: 10.3390/s20247230.