**Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)** p-ISSN: 2355-7699
Vol. 12, No. 3, Juni 2025, hlm. 517-524 e-ISSN: 2528-6579

# IMPLEMENTASI ALGORITMA SPARKLE SCHWAEMM128-128 UNTUK PROSES ENKRIPSI PENGIRIMAN DATA PADA SENSOR GPS BERBASIS IOT DENGAN PROTOKOL MODUL NRF24L01

**Avis Sugianto[1], Ari Kusyanti[2], Primantara Hari Trisnawan[3]**

[1,2,3]Universitas Brawijaya, Malang
Email: [1]avis7sugianto@student.ub.ac.id, [2]ari.kusyanti@ub.ac.id, [3]prima@ub.ac.id
*Penulis Korespondensi

(Naskah masuk: 9 April 2025, diterima untuk diterbitkan: 3 Juni 2025)

**Abstrak**

Penelitian ini bertujuan untuk menerapkan algoritma enkripsi Sparkle Schwaemm128-128, yang merupakan algoritma enkripsi ringan yang berdasarkan pada permutasi Sparkle, untuk proses enkripsi pengiriman data pada sensor GPS berbasis IoT dengan protokol modul NRF24L01. Metode penelitian yang digunakan adalah eksperimen dengan menggunakan perangkat keras NodeMCU ESP8266, sensor GPS NEO-6Mv2, modul komunikasi NRF24L01, dan *platform* Arduino IDE. Hasil penelitian menunjukkan bahwa algoritma Sparkle Schwaemm128-128 dapat mengenkripsi dan mendekripsi data sensor GPS dengan kecepatan 8012 mikrodetik untuk waktu enkripsi dan 13108 untuk waktu dekripsi. Sisa memori penggunaan pada mikrokontroler NodeMCU ESP8266 berkisar 51765 *byte* yaitu hanya 36,75% memori yang terpakai pada node sensor dan 52174 *byte* yaitu hanya 36,36% memeri yang terpakai pada node pusat. Penelitian ini diharapkan dapat memberikan solusi keamanan yang efisien dan efektif untuk sistem IoT yang menggunakan sensor GPS dan modul komunikasi NRF24L016.

**Kata kunci**: *IoT, sensor GPS, enkripsi, Sparkle Schwaemm128-128, Modul NRF24L01.*

# *IMPLEMENTATION OF THE SPARKLE SCHWAEMM128-128 ALGORITHM FOR DATA ENCRYPTION IN IOT-BASED GPS SENSOR DATA TRANSMISSION USING NRF24L01 MODULE PROTOCOL*

***Abstract***

*This research aims to implement the Sparkle Schwaemm128-128 encryption algorithm, which is a lightweight encryption algorithm based on Sparkle permutation, for the data encryption process in IoT-based GPS sensors with the NRF24L01 module protocol. The research method used is an experiment using NodeMCU ESP8266 hardware, NEO-6Mv2 GPS sensor, NRF24L01 communication module, and Arduino IDE platform. The results showed that the Sparkle Schwaemm128-128 algorithm can encrypt and decrypt GPS sensor data at 8012 microseconds for encryption time and 13108 for decryption. The remaining memory usage on the NodeMCU ESP8266 microcontroller ranges from 51765 bytes which is only 36.75% of the memory used on the sensor node and 52174 bytes which is only 36.36% of the memory used on the central node. This research is expected to provide an efficient and effective security solution for IoT systems that use GPS sensors and NRF24L01 communication modules.*

**Keywords**: *IoT, GPS sensor, encryption, Sparkle Schwaemm128-128, NRF24L01 modul*

## 1. PENDAHULUAN

*Internet of Things* (IoT) adalah konsep yang memungkinkan perangkat-perangkat yang terhubung melalui jaringan internet untuk saling berkomunikasi dan bertukar data. Menurut definisi dari Rose et al. (2015), IoT secara umum merujuk pada suatu skenario di mana objek dapat terhubung dalam suatu jaringan, memiliki kemampuan konektivitas dan komputasi, serta mampu menghasilkan, berbagi, dan mengakses data secara otomatis atau minim adanya keikutsertaan user secara langsung. Salah satu aplikasi IoT adalah sistem pemantauan lokasi menggunakan sensor *Global Positioning System* (GPS). Namun, sistem ini memiliki tantangan dalam hal keamanan data, karena data sensor GPS dapat disadap, dimodifikasi, atau dipalsukan oleh pihak yang tidak bertanggung jawab. Sehingga diperlukan adanya mekanisme enkripsi yang dapat menjaga keaslian dan kerahasiaan data sensor GPS.

Enkripsi adalah proses mengubah informasi yang dapat diketahui artinya secara langsung menjadi informasi yang telah disandikan yang tidak dapat diketahui isinya oleh pihak yang tidak memiliki kunci

rahasia. Enkripsi dapat dilakukan dengan berbagai algoritma, salah satunya menurut menurut Beierle et al.,(2020) adalah algoritma Sparkle Schwaemm128-128, yang merupakan algoritma enkripsi ringan yang berdasarkan pada permutasi Sparkle. Algoritma ini memiliki keunggulan dalam hal kecepatan, keamanan, dan efisiensi sumber daya, sehingga cocok untuk diterapkan pada perangkat IoT dengan keterbatasan memori dan daya.

Penelitian ini bertujuan untuk menerapkan algoritma enkripsi Sparkle Schwaemm128-128 untuk proses enkripsi pengiriman data pada sensor GPS berbasis IoT dengan protokol modul nRF24L01. Modul nRF24L01 adalah perangkat komunikasi nirkabel yang beroperasi pada frekuensi radio 2.4 GHz dan memiliki jangkauan hingga 100 meter. Modul ini memiliki kelebihan dalam hal harga, ukuran, dan konsumsi daya yang rendah, sehingga ideal diterapkan dalam IoT bersamaan dengan sensor GPS.

Dengan demikian, penelitian ini diharapkan dapat menjadi solusi keamanan yang efisien dan efektif untuk sistem IoT yang menggunakan sensor GPS dan modul komunikasi NRF24L01. Diharapkan implementasi ini dapat dijadikan referensi bagi pengembangan IoT dan memastikan keamanan proses enkripsi dan dekripsi menggunakan algoritma Sparkle Schwaemm128-128.

## 2. LANDASAN KEPUSTAKAAN

Penelitian yang digunakan sebagai landasan kepustakaan yang juga merupakan paper resmi dari algoritma Sparkle Schwaemm128-128 ialah penelitian dari Beierle et al.,(2020) berjudul "*Lightweight aead and hashing using the sparkle permutation family*" yang membahas tentang enkripsi menggunakan algoritma enkripsi Schwamm dan Esch menggunakan Sparkle Permutation yang diimplementasikan ke dalam *chipset* mikrokontroler. Menghasilkan bahwa performa yang didapatkan ketika melakukan sparkle permutation yang menggunakan kode C dengan assembler memiliki selisih lebih sedikit pada ARM, yaitu faktor sekitar 2.5 ketika dijalankan pada Cortex-m3.

Penelitian kedua adalah penelitian yang dilakukan oleh Nur Aziz, Maulana dan Ichsan (2019) yang berjudul "Implementasi Algoritma Speck pada Sistem Monitoring Detak Jantung dan Suhu" membahas penggunaan algoritma Speck untuk mengamankan data pasien di rumah sakit, dengan fokus pada data denyut jantung dan suhu tubuh. Sensor denyut jantung dan sensor suhu DHT11 diuji, menunjukkan akurasi yang sangat baik dengan kesalahan minimal. Dalam proses enkripsi membutuhkan wakatu rata2 80,5 milidetik dan waktu dekripsi 50,3 milidetik.

Penelitian ketiga adalah penelitian yang dilakukan oleh Sardi (2020) yang berjudul "Implementasi Algoritme AES 128 bit Pada Mikrokontroler NodeMCU Menggunakan Arsitektur *WEB SERVICE REST* Untuk Keamanan Pengiriman Data" yang menerangkan tentang cara kerja sistem algoritme tersebut dengan mengubah data berbentuk *plaintext* menjadi ciphertext menggunakan algoritma AES 128 bit sebelum dikirim ke server basis data. Setelah sampai di server, data akan didekripsi menjadi bentuk semula dan disimpan ke dalam basis data. Proses pengiriman data menggunakan arsitektur web service REST dan hasil menunjukkan bahwa proses enkripsi *plaintext* sebesar 128 dan 256 bit pada mikrokontroler NodeMCU dibutuhkan durasi 266,31 dan 274,31 mikrodetik, memori yang dibutuhkan sebesar 53928.12 dan 54114.32 *byte*.

### 2.1 Algoritma Sparkle

Menurut Bierle et al.(2019) algoritma ini adalah sebuah keluarga permutasi kriptografi yang didasarkan pada desain ARX (*Addition, Rotation, XOR*), menggabungkan operasi yang sederhana namun aman, yang membuatnya cocok untuk perangkat yang memiliki keterbatasan seperti yang digunakan dalam penelitian ini. Algoritma ini merupakan salah satu finalis dari kompetisi *Lightweight Cryptography* yang diselenggarakan oleh NIST. Algoritma ini memiliki beberapa varian yang berbeda dalam ukuran blok, yaitu 256, 384, atau 512 bit. Struktur ARX (*Addition, Rotation, XOR*) yang digunakan oleh Sparkle Schwaemm128-128, yang lebih efisien dibandingkan algoritma berbasis substitusi dan permutasi lain (seperti AES).

Algoritma ini terdiri dari beberapa langkah (*step*) yang masing-masing melibatkan dua lapisan, yaitu ARXBOX *layer* dan *Linear layer*. ARXBOX *layer* adalah bagian kunci yang menggabungkan beberapa operasi kriptografi dasar untuk meningkatkan keamanan dan kompleksitas algoritma. Linear layer adalah bagian yang menggabungkan dan mengacak *output* dari ARXBOX layer menggunakan struktur Feistel dengan fungsi *Feistel linear*. Jumlah langkah yang digunakan bervariasi tergantung pada ukuran blok dan tingkat keamanan yang diinginkan.

Selain itu, algoritma ini dirancang untuk memberikan keamanan yang kuat dengan efisiensi yang tinggi, terutama pada perangkat dengan sumber daya terbatas seperti sensor IoT. Menurut penelitian yang dilakukan oleh Bierle et al., struktur ARX memberikan keseimbangan antara keamanan dan kinerja, memungkinkan algoritma ini untuk diterapkan pada berbagai perangkat dengan kebutuhan kriptografi yang berbeda. Algoritma ini juga memiliki fleksibilitas dalam hal parameter, yang memungkinkan penyesuaian berdasarkan kebutuhan spesifik dari aplikasi atau perangkat yang menggunakannya. Hal ini menjadikan algoritma ini sangat cocok untuk berbagai lingkungan dengan batasan sumber daya.

## 2.2 Algoritma enkripsi Schwaemm128-128

Algoritma Schwaemm128-128 merupakan enkripsi terotentikasi dengan data terkait (AEAD) yang menggunakan permutasi Sparkle dalam *mode sponge*.

Kode Program 1 Notasi Enkripsi Algoritma Schwaemm128-128

Schwaemm128-128-Encryption
Input: $(K, N, A, M)$ where $K \in \mathbb{F}_2^{128}$ is a key, $N \in \mathbb{F}_2^{128}$ is a nonce and $A, M \in \mathbb{F}_2^*$
Output: $(C, T)$, where $C \in \mathbb{F}_2^*$ is the ciphertext and $T \in \mathbb{F}_2^{128}$ is the authentication tag

◁ Padding the associated data and message
if $A \neq \epsilon$ then
  $A_0 \| A_1 \| \ldots \| A_{\ell A-1} \leftarrow A$ with $\forall i \in \{0, \ldots, \ell_A - 2\} : |A_i| = 128$ and $1 \leq |A_{\ell A-1}| \leq 128$
  if $|A_{\ell A-1}| < 128$ then
    $A_{\ell A-1} \leftarrow \text{pad}_{128}(A_{\ell A-1})$
    $\text{Const}_A \leftarrow 0 \oplus (1 \ll 2)$
  *else*
    $\text{Const}_A \leftarrow 1 \oplus (1 \ll 2)$
  end if
end if
if $M \neq \epsilon$ then
  $M_0 \| M_1 \| \ldots \| M_{\ell M-1} \leftarrow M$ with $\forall i \in \{0, \ldots, \ell_M - 2\} : |M_i| = 128$ and $1 \leq |M_{\ell M-1}| \leq 128$
  $t \leftarrow |M_{\ell M-1}|$
  if $|M_{\ell M-1}| < 128$ then
    $M_{\ell M-1} \leftarrow \text{pad}_{128}(M_{\ell M-1})$
    $\text{Const}_M \leftarrow 2 \oplus (1 \ll 2)$
  *else*
    $\text{Const}_M \leftarrow 3 \oplus (1 \ll 2)$
  end if
end if
◁ State initialization
$S_L \| S_R \leftarrow \text{Sparkle256}_{10}(N \| K)$ with $|S_L| = 128$ and $|S_R| = 128$
◁ Processing of associated data
if $A \neq \epsilon$ then
  for all $j = 0, \ldots, \ell_A - 2$ do
    $S_L \| S_R \leftarrow \text{Sparkle256}_7((\rho_1(S_L, A_j) \oplus S_R) \| S_R)$
end for
◁ Finalization if message is empty
$S_L \| S_R \leftarrow \text{Sparkle256}_{10}(\rho_1(S_L, A_{\ell A-1}) \oplus S_R \oplus \text{Const}_A) \| (S_R \oplus \text{Const}_t)$
end if
◁ Encrypting
if $M \neq \epsilon$ then
  for all $j = 0, \ldots, \ell_M - 2$ do
    $C_j \leftarrow \rho_2(S_L, M_j)$
    $S_L \| S_R \leftarrow \text{Sparkle256}_7((\rho_1(S_L, M_j) \oplus S_R) \| S_R)$
  end for
  $C_{\ell M-1} \leftarrow \text{trunc}_t(\rho_2(S_L, M_{\ell M-1}))$
◁ Finalization
  $S_L \| S_R \leftarrow \text{Sparkle256}_{10}((\rho_1(S_L, M_{\ell M-1}) \oplus S_R \oplus \text{Const}_M) \| (S_R \oplus \text{Const}_M))$
end if
return $(C_0 \| C_1 \| \ldots \| C_{\ell M-1}, S_R \oplus K)$

*Mode sponge* berbeda dari algoritma konvensional, *mode* ini mampu memproses enkripsi dan autentikasi secara *simultan*, yang membuatnya lebih efisien untuk pengiriman data yang membutuhkan integritas dan kecepatan. *Mode sponge* beroperasi dengan membagi blok data menjadi dua bagian, yaitu *rate* dan *capacity*, dan melakukan operasi *XOR*, permutasi, dan ekstraksi secara berulang. Pada Schwaemm128-128, *rate* dan *capacity* memiliki panjang 128 bit, sehingga total panjang blok adalah 256 bit.

Dari proses enkripsi sebelumnya diketahui melibatkan langkah-langkah yang menyertakan pengacakan *bit* dan fungsi ekstraksi yang iteratif, yang memastikan keamanan data dan autentikasi integritasnya selama proses pengiriman melalui protokol NRF24L01 pada sensor GPS berbasis IoT. Sparkle Schwaemm128-128 sudah diuji dalam berbagai situasi dan mendapatkan reputasi sebagai algoritma yang mampu menangani serangan umum seperti *ciphertext-only attack* serta menawarkan keamanan yang cukup untuk perangkat IoT.

Kode Program 1 menunjukkan persamaan untuk proses enkripsi data menggunakan algoritma Schwaemm128-128.

Kode Program 2 Notasi Dekripsi Algoritma Schwaemm128-128

Schwaemm128-128-Decryption
Input: $(K, N, A, M)$ where $K \in \mathbb{F}_2^{128}$ is a key, $N \in \mathbb{F}_2^{128}$ is a nonce , $A, C \in \mathbb{F}_2^*$ and $T \in \mathbb{F}_2^{128}$
Output: Decryption $M$ of $C$ if the tag $T$ is valid, $\perp$ otherwise

if $A \neq \epsilon$ then
  $A_0 \| A_1 \| \ldots \| A_{\ell A-1} \leftarrow A$ with $\forall i \in \{0, \ldots, \ell_A - 2\} : |A_i| = 128$ and $1 \leq |A_{\ell A-1}| \leq 128$
  if $|A_{\ell A-1}| < 128$ then
    $A_{\ell A-1} \leftarrow \text{pad}_{128}(A_{\ell A-1})$
    $\text{Const}_A \leftarrow 0 \oplus (1 \ll 2)$
  *else*
    $\text{Const}_A \leftarrow 1 \oplus (1 \ll 2)$
  *end if*
*end if*
if $C \neq \epsilon$ then
  $C_0 \| C_1 \| \ldots \| C_{\ell M-1} \leftarrow C$ with $\forall i \in \{0, \ldots, \ell_M - 2\} : |C_i| = 128$ and $1 \leq |C_{\ell M-1}| \leq 128$
  $t \leftarrow |C_{\ell M-1}|$
  if $|C_{\ell M-1}| < 128$ then
    $C_{\ell M-1} \leftarrow \text{pad}_{128}(C_{\ell M-1})$
    $\text{Const}_M \leftarrow 2 \oplus (1 \ll 2)$
  *else*
    $\text{Const}_M \leftarrow 3 \oplus (1 \ll 2)$
  end if
end if
◁ State initialization
$S_L \| S_R \leftarrow \text{Sparkle256}_{10}(N \| K)$ with $|S_L| = 128$ and $|S_R| = 128$
◁ Processing of associated data
if $A \neq \epsilon$ then
  for all $j = 0, \ldots, \ell_A - 2$ do
    $S_L \| S_R \leftarrow \text{Sparkle256}_7((\rho_1(S_L, A_j) \oplus S_R) \| S_R)$
  end for
◁ Finalization if ciphertext is empty
  $S_L \| S_R \leftarrow \text{Sparkle256}_{10}((\rho_1(S_L, A_{\ell A-1}) \oplus S_R \oplus \text{Const}_A) \| (S_R \oplus \text{Const}_t))$
end if
◁ Decrypting
if $C \neq \epsilon$ then
  for all $j = 0, \ldots, \ell_M - 2$ do
    $M_j \leftarrow \rho'(S_L, C_j)$
    $S_L \| S_R \leftarrow \text{Sparkle256}_7((\rho'_1(S_L, C_j) \oplus S_R) \| S_R)$
  end for
  $M_{\ell M-1} \leftarrow \text{trunc}_t(\rho'_2(S_L, C_{\ell M-1}))$
◁ Finalization and tag verification
  if $t < 128$ then
    $S_L \| S_R \leftarrow \text{Sparkle256}_{10}((\rho_1(S_L, \text{pad}128(M_{\ell M-1})) \oplus S_R \oplus \text{Const}_M) \| (S_R \oplus \text{Const}_M))$
  else
    $S_L \| S_R \leftarrow \text{Sparkle256}_{10}((\rho'_1(S_L, C_{\ell M-1}) \oplus S_R \oplus \text{Const}_M) \| (S_R \oplus \text{Const}_M))$
  *end if*
*end if*
if $S_R \oplus K = T$ then
  return $(M_0 \| M_1 \| \ldots \| M_{\ell M-1})$
*else*
  *return* $\perp$
end if

Kode Program 2 dijelaskan mengenai aturan untuk dekripsi data menggunakan algoritma Schwaemm128-128.

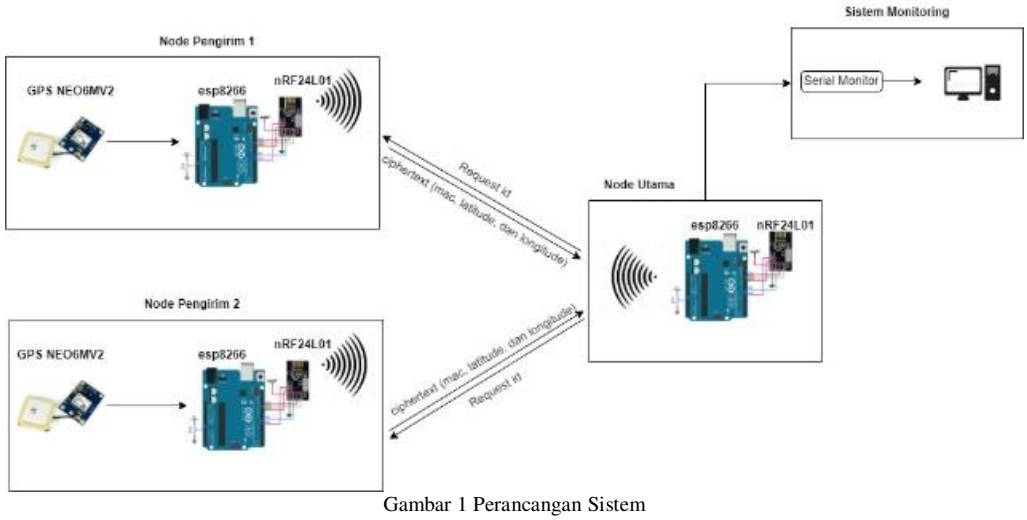
## 2.3 NodeMCU ESP8266

NodeMCU ESP8266 adalah mikrokontroler berbasis *WiFi* yang dapat diprogram menggunakan platform Arduino IDE. NodeMCU ESP8266 memiliki fitur-fitur seperti 10 port GPIO, PWM, I2C, SPI, 1 *Wire*, ADC, dan WSN (*Wireless Sensor Network*). NodeMCU ESP8266 cocok digunakan untuk aplikasi Internet of Things (IoT) yang membutuhkan komunikasi nirkabel dan pengolahan data dari berbagai sensor.

## 2.4 Modul nRF24L01

Menurut Ruhyat et al.,(2022) modul nRF24L01 adalah sebuah perangkat transceiver nirkabel yang sering digunakan dalam proyek-proyek komunikasi tanpa kabel, terutama di bidang

elektronika dan IoT.




Gambar 1 Perancangan Sistem

Modul ini dirancang oleh *Nordic Semiconductor* dan berjalan di frekuensi 2,4 GHz menggunakan modulasi GFSK (*Gaussian Frequency Shift Keying*). Modul ini dapat digunakan untuk berbagai aplikasi komunikasi tanpa kabel, terutama di bidang elektronika dan IoT. Modul ini memiliki keunggulan dalam hal keandalan, efisiensi, dan keamanan komunikasi. Modul ini juga mendukung mode *point-to-point* atau mode jaringan nirkabel, yang memungkinkan pembentukan jaringan yang fleksibel dan toleran terhadap kegagalan. Modul ini dapat dihubungkan dengan mikrokontroler melalui protokol SPI (*Serial Peripheral Interface*) dan membutuhkan lima pin utama, yaitu VCC, GND, CE, CSN, dan IRQ. Modul ini juga dapat dikombinasikan dengan protokol jaringan seperti RF24Network atau RF24Mesh untuk membangun jaringan nirkabel yang terstruktur dan dapat diandalkan.

### 2.5 Sensor GPS

Sensor GPS NEO-6Mv2 adalah modul *Global Positioning System* yang dapat memberikan informasi lokasi dan waktu yang akurat melalui sinyal satelit. Modul ini memiliki sensitivitas tinggi, start-up time cepat, dan fitur A-GPS yang mempercepat akuisisi sinyal dan meningkatkan ketepatan lokasi. Modul ini juga mendukung antena aktif dan protokol komunikasi serial seperti UART. Modul ini cocok untuk aplikasi yang memerlukan pemantauan atau navigasi lokasi yang akurat, seperti sistem pemantauan lokasi berbasis IoT.

### 2.6 WSN(*Wireless Sensor Network*)

Menurut Mainwaring et al,.(2002) *Wireless Sensor Network* (WSN) merupakan kumpulan dari beberapa atau banyak sensor *wireless* yang dibangun dari satu jenis atau berbagai jenis, tersebar di titik-titik tertentu dalam wilayah luas untuk mengumpulkan data dan mengirimkannya ke suatu tempat tertentu. WSN dapat digunakan untuk memantau kondisi lingkungan seperti suhu, kelembaban, dan polusi udara. WSN juga dapat digunakan dalam berbagai aplikasi, seperti pemantauan kesehatan, pemantauan industri, dan pemantauan lingkungan. Komponen-komponen yang umum terdapat pada WSN antara lain sensor, *external memory, power supply,* mikrokontroler, dan *transceiver* atau *communication device*.

## 3. METODE PENELITIAN DAN IMPLEMENTASI

Penelitian ini mengimplementasikan algoritma Sparkle Schwaemm128-128, yang dirancang khusus untuk perangkat dengan sumber daya terbatas seperti NodeMCU ESP8266. Analisis kebutuhan sistem dibutuhkan sebagai gambaran mengenai sistem yang akan di rancang.

Berikut merupakan penjelasan yang terperinci mengenai 1, yaitu gambaran umum sistem:

Bagian ini menjelaskan tentang pengiriman data sensor GPS dari dua sensor yang berbeda ke node pusat. Sensor primer dan sensor sekunder mengakuisisi data sensor dari GPS GY-NEO6MV2 ke mikrokontroler NodeMCU ESP8266 berupa data latitude dan longitude. Kemudian NodeMCU ESP8266 mengenkripsi data lokasi hasil sensor GPS menggunakan algoritma Sparkle Schwaemm128-128, selanjutnya mengirimkan data lokasi hasil sensor GPS yang sudah dienkripsi ke node pusat melalui modul nRF24L01.

Node pusat berfungsi mengirimkan request id ke node sensor yang dipilih, selanjutnya node pusat akan menerima data sensor GPS sesuai node yang dipilih menggunakan modul nRF24L01 sebagai media komunikasi. Selanjutnya mikrokontroler NodeMCU ESP8266 menerjemahkan data sensor dan melakukan dekripsi data sensor menggunakan algoritma Sparkle Schwaemm128-128 yang sebelumnya berupa *ciphertext* ke dalam bentuk plaintext. Data hasil

dekripsi akan ditampilkan di serial monitor untuk dianalisa.

## 4. HASIL DAN ANALISIS

Pengujian penting dilakukan untuk mengetahui kinerja dari algoritma Sparkle Scwaemm128-128. Pengujian mencakup pengujian kebutuhan sistem dilakukan untuk memastikan bahwa sistem yang sudah diimplementasikan telah sesuai.

### 4.1 Pengujian tes validitas algoritma sparkle Scwaemm128-128.

Pengujian validitas algoritma sparkle merupakan pengujian test vector algoritma Sparkle dilakukan untuk memastikan bahwa implementasi algoritma Sparkle Scwaemm128-128. pada sistem sesuai dengan skenario komputasi algoritma Sparkle pada paper resminya. Tabel 1 berisikan hasil dari pengujian validitas *ciphertext* dari algoritma sparkle Scwaemm128-128. dengan mengikuti paper resminya nilai *Nonce* dan *Key* yang sama yaitu 0123456789ABCDEF.

Tabel 1 Hasil pengujian validitas algoritma Sparkle Scwaemm128-128

| Plaintext | Ad | Ciphertext | Hasil |
|---|---|---|---|
| | | DDCE77CDB748E6D053CAB7E9190A8349 | Valid |
| 0001 | 0001 | BB027D4510D729EEC0DCE6FDFDE3027939D3 | Valid |

Pengujian validitas algoritma Sparkle Schwaemm128-128 dilakukan sebanyak 31 kali mengikuti paper resminya dan di peroleh hasil semua tes menghasilkan *Ciphertext* sesuai dan tes dinyatakan valid.

### 4.2 Pengujian memori dan waktu

Pengujian ini menjelaskan kinerja validitas test vector algoritma Sparkle Schwaemm128-128 dilakukan dengan mempertimbangkan dua parameter, yakni memori dan waktu. Terdapat 31 skenario pengujian, di mana setiap skenario terdiri dari key dan nonce, masing-masing memiliki panjang 128 bit dengan pola bit yang sama. Tabel 2 berisikan hasil dari pengujian memori dan waktu algoritma Sparkle Schwaemm128-128.

Tabel 2 Hasil pengujian memori dan waktu

| Pengujian | Waktu Enkripsi (Mikrodetik) | Waktu Dekripsi (Mikrodetik) | Memory (byte) |
|---|---|---|---|
| Rata - Rata | 443,90 | 45083 | 52176.77 |

#### 5.2.1 Pengujian fungsi waktu enkripsi dan dekripsi

Pengujian fungsi waktu enkripsi dan dekripsi bertujuan untuk mengukur waktu yang dibutuhkan oleh sistem dalam menjalankan proses enkripsi dan dekripsi. Tabel 3 merupakan hasil rata-rata pengujian waktu enkripsi dan dekripsi dilakuakan sebanyak 31 kali dengan variabel yang berbeda pada setiap skenario pengujian.

Tabel 3 Hasil pengujian perbandingan waktu enkripsi dan dekripsi

| Parameter | Dengan Algoritma | |
|---|---|---|
| | Node Sensor | Node Pusat |
| Waktu (Mikrodetik). | 8012.096 | 13108.12 |
| Parameter | Tanpa Algoritma | |
| | Node Sensor | Node Pusat |
| Waktu (Mikrodetik). | 12.48 | 363.58 |

#### 5.2.2 Pengujian pengiriman dan penerimaan data

Pengujian ini penting bertujuan memverifikasi keberhasilan *chipset* dan perangkat nRF dalam mengirimkan *ciphertext* dari sensor dan menerima *ciphertext* pada node pusat serta menerapkan fungsi *request*. Gambar 2 menunjukkan hasil pengiriman *ciphertext* oleh sensor. Sedangkan Gambar 3 merupakan node pusat yang berhasil menerima *ciphertext* kemudian melakukan proses dekripsi.

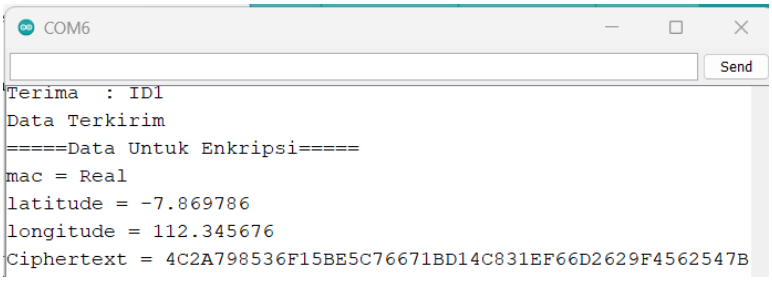
## 5. PENUTUP

### 6.1 Kesimpulan

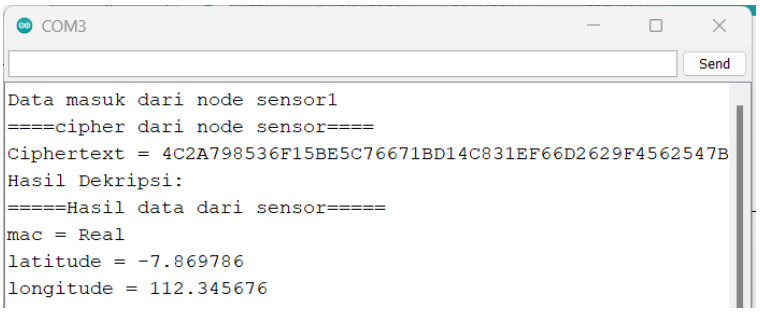
Kesimpulan yang dapat didapat dari penelitian ini adalah Sistem yang dibangun berhasil menerapkan Algoritma Sparkle Schwaemm 128-128 sesuai dengan dokumentasi asli, yang dibuktikan dengan indikator berhasil pada tes validitas algoritma (*test vector*). Algoritma ini juga berhasil diimplementasikan ke dalam NodeMCU ESP8266 dengan sisa memori penggunaan sebesar 51765 *byte* di node sensor dan 52174 *byte* di node pusat. Selain itu, Algoritma Sparkle Schwaemm 128-128 mampu melakukan enkripsi selama 8012 mikrodetik dan dapat diterima serta didekripsi selama 13108 mikrodetik.

### 6.2 Saran

Penelitian ini masih memiliki beberapa kelemahan dan keterbatasan.




Gambar 2 Hasil Pengiriman Data di Node Sensor




Gambar 3 Hasil Penerimaan Data di Node pusat

Diharapkan saran-saran berikut dapat dijadikan acuan dan memberikan kontribusi bagi penelitian lebih lanjut di masa mendatang, antara lain: pemilihan modul komunikasi yang lebih baik dan berkualitas diharapkan dapat melakukan proses pengiriman data dengan jarak jangkau yang lebih jauh dan lebih maksimal dari berbagai segi, serta mengimplementasikan jenis lain dari kombinasi antara permutasi Sparkle dan algoritma enkripsi Schwaemm agar dapat mengetahui ketahanan dan efektivitas algoritma enkripsi dengan variasi *bit* yang lebih besar pada *mikrokontroler*.

## DAFTAR PUSTAKA

AL-FUQAHA, A., ET AL., 2015. Internet of things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys & Tutorials.

ANDERSON, R., & KUHN, M., 1996. Tamper resistance: a cautionary note. Proceedings of the 2nd USENIX Workshop on Electronic Commerce.

BEIERLE, C., ET AL., 2020. Lightweight aead and hashing using the sparkle permutation family. IACR Transactions on Symmetric Cryptology.

DAUTOV, R., ET AL., 2020. Lightweight and secure data encryption for IoT: A survey. Journal of Network and Computer Applications.

FERRAG, M. A., ET AL., 2018. A survey on privacy-preserving schemes for IoT: Current status and future directions. Journal of Network and Computer Applications.

GILL, S. S., ET AL., 2019. Transformation towards fog computing from cloud computing: A survey on security issues and solutions. Journal of Network and Computer Applications.

HAMEED, S., ET AL., 2019. Security and privacy issues in IoT: A survey. IEEE Internet of Things Journal.

HENNEBERT, C., & GLIMARE, J., 2015. Data security in the IoT: Current status and open issues. International Conference on Internet of Things.

HU, Y., & FEI, T., 2019. Efficient and lightweight encryption algorithm for IoT applications. Journal of Information Security and Applications.

NUR AZIZ, M. M., MAULANA, R., & ICHSAN, M. H. H., 2019. Implementasi Algoritme Speck pada Sistem Monitoring Detak Jantung dan Suhu. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 3(9), 8680-8685. https://doi.org/10.25126/jptiik.201931964

JINDAL, A., ET AL., 2019. An evaluation of lightweight cryptographic algorithms for IoT devices. Journal of Network and Computer Applications.

JIN, X., ET AL., 2018. Lightweight encryption schemes for wireless sensor networks. Ad Hoc Networks.

MAINWARING, A., ET AL., 2002. Wireless sensor networks for habitat monitoring. Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications.

NOURA, M., ET AL., 2019. Lightweight encryption algorithms for IoT: Comparative study and performance analysis. Wireless Communications and Mobile Computing.

RAZA, S., ET AL., 2017. Lightweight security solutions for the Internet of Things: A survey. IEEE Sensors Journal.

ROMDHANI, I., ET AL., 2018. Lightweight cryptography for IoT: A comprehensive survey. Ad Hoc Networks.

ROSE, K., ET AL., 2015. The internet of things: An overview. Internet Society.

RUHYAT, M. N., RAHMADEWI, R., & SARAGIH, Y., 2022. Implementasi modul transreceiver NRF24L01 sebagai pengirim dan penerima data nirkabel pada alat sistem monitoring peringatan dini banjir. Jurnal MEDIA ELEKTRIK, 19(3), 134-142.

SARDI, R.I., 2020. Implementasi Algoritme AES 128 bit Pada Mikrokontroler NodeMCU Menggunakan Arsitektur WEB SERVICE REST Untuk Keamanan Pengiriman Data. Publikasi Tugas Akhir S-1 PSTI FT-UNRAM. [online] Available at: <http://begawe.unram.ac.id/index.php/ta/article/view/150%0Ahttp://begawe.unram.ac.id/index.php/ta/article/download/150/39>...

SHIN, S., ET AL., 2020. A study on encryption algorithms for IoT applications. Journal of Ambient Intelligence and Humanized Computing.

SICARI, S., ET AL., 2015. Security, privacy and trust in Internet of Things: The road ahead. Computer Networks.

XU, L. D., ET AL., 2014. Internet of Things in industries: A survey. IEEE Transactions on Industrial Informatics.