Implementasi Known-Plaintext Attack Algoritme Pada Grain-128a Berbasis LoRa

Olivia Very Noorlinda[1], Ari Kusyanti[2], Kasyful Amron[3]

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: [1]oliviavery28@gmail.com, [2]ari.kusyanti@ub.ac.id, [3]kasyful@ub.ac.id

Abstrak

Long-Range (LoRa) merupakan teknologi Low Power Wide Area Network (LPWN) yang menggunakan modulasi Chirp Spread Spectrum (CSS) yang berasal dari skema modulasi spread spectrum yang mampu mempertahankan daya rendah serta meningkatkan jangkauan komunikasi yang dibutuhkan pada Internet of things (IoT). Perangkat LoRa melakukan komunikasi melalui radio gelombang dengan frekuensi yang sama. Pada penelitian ini LoRa berperan sebagai node dan gateway yang melakukan pengiriman data pada LoRa melalui broadcast dengan frekuensi yang sama. LoRa memiliki kelemahan pada keamanan datanya, di mana data yang dikirimkan ke node gateway masih rentan terhadap serangan pihak ketiga. Keamanan pada LoRa dapat ditingkatkan dengan menambahkan algoritme Grain-128a pada node dan gateway yang bertujuan untuk memberikan keamanan pada data yang dikirimkan oleh node ke gateway. Grain-128a merupakan algoritme enkripsi yang menawarkan autentikasi opsional, yaitu melakukan enkripsi dengan autentikasi dan tanpa autentikasi pada data. Pada penelitian ini Grain-128a berhasil melakukan pengamanan pada data suhu dan pada modul komunikasi LoRa berdasarkan dari pengujian yang telah dilakukan. Pada pengujian known-plaintext attack yang dilakukan, penyerang tidak berhasil mengetahui plaintext dari ciphertext yang dihasilkan algoritme Grain-128a. Hal ini membuktikan bahwa penyerang tidak berhasil melakukan penyerangan, dan algoritme Grain-128a berhasil melakukan pengamanan pada data.

Kata kunci: LoRa, IoT, Algoritme Grain 128-a, Enkripsi, *Known-Plaintext-Attack,* Keamanan

Abstract

Long-Range (LoRa) is a Low Power Wide area Network (LPWN) technology that uses Chirp Spread Spectrum (CSS) modulation derived from a spread spectrum modulation scheme that is able to maintain low power and increase the range of communication needed on the Internet of things (IoT). LoRa devices communicate via radio waves with the same frequency. In this study, LoRa acts as a node and gateway that transmits data on LoRa via broadcast with the same frequency. LoRa has a weakness in its data security, where the data sent by the node to the gateway is still vulnerable to third party attacks. LoRa security can be improved by adding the Grain-128a algorithm to nodes and gateways which aim to provide security for the data sent by the node to the gateway. Grain-128a is an encryption algorithm that offers optional authentication, which is to do encryption with authentication and without authentication on data. In this research, Grain-128a succeeded in securing the temperature and humidity data on the LoRa communication module based on the testing that has been done. In the known-plaintext attack test, the attacker was unable to find out the plaintext from the ciphertext generated by the Grain-128a algorithm. This proves that the attacker was unsuccessful in carrying out the attack, and the Grain-128a algorithm was successful in securing the data.

Keywords: *LoRa, IoT, Algoritme Grain 128-a, Encryption, Known-Plaintext-Attack,, Security*

1. PENDAHULUAN

Internet of things (IoT) merupakan salah satu teknologi internet yang memungkinkan pengguna dapat mengelola banyak perangkat sekaligus dari jarak jauh melalui Internet. Untuk mengelola perangkat-perangkat yang terhubung dalam IoT, IoT membutuhkan konsumsi daya yang rendah dengan jangkauan jarak jauh. Oleh karena itu dibutuhkan sebuah teknologi dengan daya yang rendah dan jangkauan jarak jauh.

Long-Range (LoRa) merupakan teknologi *Low Power Wide Area Network (LPWN)* yang menawarkan daya yang rendah dan jangkauan jarak jauh. LoRa merepresentasikan *physical*

layer atau *wireless modulation* yang digunakan untuk membangun komunikasi jarak jauh. LoRa menggunakan modulasi *Chirp Spread Spectrum* (CSS) milik sendiri yang berasal dari skema modulasi lokasi tersebar yang mampu mempertahankan daya rendah serta dapat meningatkan jangkauan komunikasinya. Pada penelitian ini LoRa berperan sebagai *node* dan *gateway*. *Node* akan mengirimkan data berupa suhu dan kelembapan menuju *gateway*. Pada penelitian sebeumnya yang dilakukan Arijjudin (2019) terdapat celah keamanan pada saat komunikasi antar LoRa berlangsung, yaitu pihak yang tidak memiliki wewenang dapat melihat data yang dikirimkan oleh *node* menuju *gateway* yang memiliki wewenang. Oleh karena itu dibutuhkan upaya untuk melakukan pengamanan data saat komunikasi antar LoRa berlangsung.

Keamanan pada LoRa dapat ditingkatkan dengan menambahkan kriptografi pada data yang dikirimkan pada saat komunikasi berlangsung. Salah satu konsep kriptografi yaitu *confidentiality*, yang merupakan konsep keamanan dimana pihak lain tidak dapat membaca pesan yang dirahasiakan kecuali pihak yang memiliki wewenang. Pada penelitian sebelumnya NXP Semiconductor (2018) telah melakukan pengamanan pada LoRa dengan menggunakan algoritme AES. Namun algoritme AES merupakan algoritme standar enkripsi yang sudah lama. Sehingga dibutuhkan algoritme baru untuk melakukan keamanan pada data. Algoritme Grain-128a dipilih karena merupakan versi terbaru dari algoritme Grain dengan konsumsi daya yang rendah dan pada algoritme Grain-128a terdapat autentikasi pada pesan yang membuat tingkat keamanan Grain-128a lebih tinggi dari algoritme AES. Algoritme Grain-128a adalah algoritme dengan konsep kerahasiaan yang dilengkapi dengan konsep integritas pada kriptografi. Algoritme Grain-128a menawarkan autentikasi opsional, yaitu enkripsi dengan autentikasi dan tanpa autentikasi yang berperilaku mirip dengan Grain-128 (Ågren et al., 2011). Pada penelitian yang dilakukan oleh Watanabe dkk (2018), Grain-128a membutuhkan 164 byte RAM dan 385 *times*(µs) untuk 16-byte *input data* pada ARM Cortex-M3, yang secara signifikan lebih baik dari AES dan lebih cepat dibandingkan dengan *block cipher* SKINNY-128-128. Algoritme Grain-128a digunakan untuk memberikan keamanan pada data yang akan dikirimkan oleh *node* ke *gateway*.

Dengan mengimplementasikan algoritme Grain-128a pada data yang akan dikirimkan oleh *node* ke *gateway* melalui modul komunikasi LoRa, maka data yang dikirimkan dapat terjamin keamanannya. Pada penelitian akan dilakukan pengujian untuk memastikan data yang dikirimkan dari node menuju gateway tidak dapat dibaca oleh pihak lain yang tidak memiliki wewenang. Pengujian yang dilakukan yaitu pengujian pasif dengan melakukan serangan *sniffing* dan pengujian aktif dengan melakukan *known-plaintext attack* (KPA).

2. LANDASAN KEPUSTAKAAN

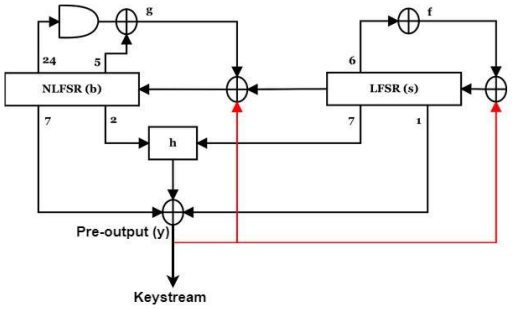
Pada penelitian yang dilakukan Arijuddin dengan judul "Pengembangan Sistem Perantara Pengiriman Data Menggunakan Modul Komunikasi LoRa dan Protokol MQTT Pada *Wireless Sensor Network*". Pada penelitian tersebut mendapatkan hasil yaitu data yang diambil dari sensor berhasil dikirimkan *node* ke *gateway* dengan menggunakan modul komunikasi LoRa. Pada penelitian tersebut komunikasi antar *node* dan *gateway* dengan modul komunikasi LoRa masih rentan terhadap serangan pihak ketiga. Sehingga pada penelitian ini, peneliti akan menambahkan pengamanan data pada *node* dan *gateway*.

Pada penelitian yang dilakukan oleh NXP Semiconductors dengan judul "*IoT Device Secure Connection with LoRa*". Pada penelitian tersebut melakukan penerapan keamanan pada LoRa menggunakan algoritme AES128. Namun algoritme AES merupakan algoritme standar enkripsi yang sudah lama. Pada penelitian ini algoritme yang digunakan untuk mengamankan data menggantikan algoritme AES-128 adalah algoritme Grain-128a.

Pada penelitian yang dilakukan oleh Ågren dkk dengan judul "*Grain-128a: a new version of Grain-128 with optional authentication*". Penelitian tersebut merupakan penjelasan dan detail dari algoritme Grain128a. Algoritme Grain-128a dengan *optional authentication* didesain berdasarkan serangan dan pengamatan yang diketahui pada algoritme Grain-128 untuk mengatasi kelemahan yang ada pada Grain-128. Pada algoritme Grain-128a terdapat autentikasi pada pesan yang membuat tingkat keamanan Grain-128a lebih tinggi dari algoritme AES. Pada penelitian ini algoritme Grain-128a digunakan untuk mengamankan data saat komunikasi antar *node* dan *gateway* pada LoRa berjalan.

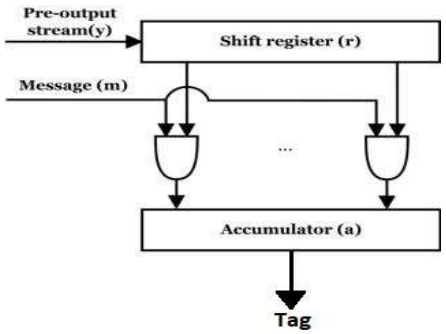
2.1. Algoritme Grain-128a

Algoritme Grain-128a merupakan versi baru dari algoritme Grain128. Algoritme Grain-128a dirancang untuk mengatasi kelemahan yang ada pada Grain-128 dan dapat memberikan autentikasi jika diperlukan, sehingga Grain128a menawarkan autentikasi opsional, yaitu enkripsi dengan autentikasi dan tanpa autentikasi (Ågren et al., 2011). Autentikasi yang tersedia pada Grain-128a sendiri adalah autentikasi pada pesan guna untuk mengetahui pesan tidak diubah pada saat dekripsi.



Gambar 1. Alur dari Grain-128a

Pada gambar 1 merupakan alur dari Grain-128a. Pada Grain-128a terdapat LFSR dan NFSR. Perhitungan yang dilakukan untuk menghasilkan *keystream* yang digunakan untuk enkripsi yaitu melakukan perhitungan fx, gx, *boolean* (hx), dan *pre-output* (y). Pada proses *generate keystream* akan melakukan perhitungan dengan melakukan *clock* 256 kali tanpa menghasilkan *keystream* dan nilai *pre-output* di-XOR-kan kembali ke LFSR dan NFSR. Kemudian dilakukan *clock* sebanyak 320 kali untuk menghasilkan *keystream*, nilai *pre-output* yang dihasilkan tidak di-XOR-kan kembali ke LFSR dan NFSR.



Gambar 2. Mekanisme autentikasi pada Grain-128a

Gambar 2 merupakan mekanisme proses autentikasi pada Grain-128a. Proses autentikasi menggunakan nilai *pre-output* dan pesan. Perhitungan yang dilakukan pada autentikasi pesan yang melakukan perhitungan *update* pada akumulator dan *shift register* yang dilakukan perulangan sebanyak panjang pesan. Hasil dari autentikasi adalah menemukan *tag* yang diperoleh dari *final update* dari akumulator.

2.2. LoRa

Long Range (LoRa) merupakan perangkat IoT yang menawarkan daya yang rendah dan jangkauan jarak jauh. LoRa merepresentasikan *physical layer* atau *wireless modulation* dan menggunakan modulasi *Chirp Spread Spectrum* (CSS) milik sendiri. Modulasi CSS pada LoRa yang berasal dari skema modulasi spektrum tersebar yang mampu mempertahankan daya rendah serta dapat meningatkan jangkauan komunikasinya. Modulasi CSS ini dapat beroperasi di tingkat bawah yang membuat lebih tahan terhadap gangguan dan gangguan (Zourmand, 2019).

2.3. Known-Plaintext Attack

Known-plaintext attack (KPA) merupakan serangan yang bertujuan untuk mencari kunci yang digunakan pada *ciphertext* untuk mengetahui *plaintext* dari *ciphertext* tersebut.

3. PERANCANGAN

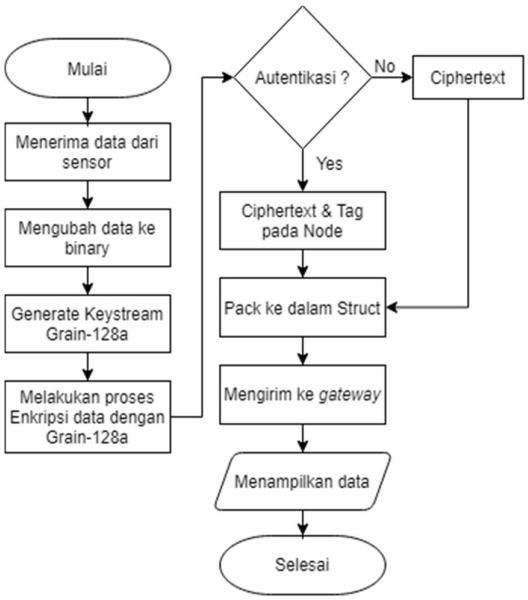
Perancangan dilakukan untuk memberikan gambaran secara umum sistem yang dibuat dan hasil dari perancangan digunakan untuk melakukan proses implementasi. Perancangan yang dilakukan pada penelitian ini terdiri dari perancangan pengamanan dan perancangan pengujian. Data yang digunakan dalan penelitian ini ialah data yang diperoleh dari sensor yang terpasang pada *node*.

Gambar 2. Gambaran sistem secara umum

Pada gambar 3 merupakan gambaran sistem dasar secara umum. Data diperoleh dari sensor akan dikirimkan ke *node*. Pada *node*, data yang diperoleh dari sensor akan ditampilkan terlebih dahulu lalu data dikirimkan ke *gateway* melalui komunikasi antar LoRa. Kemudian *gateway* menerima data yang dikirim *node* melalui komunikasi antar LoRa dan ditampilkan.

3.1. Perancangan Pengamanan

Perancangan pengamanan akan melakukan proses pengamanan data yang diperoleh dari sensor. Data yang digunakan dalam penelitian ini ialah data suhu dan kelembapan. Pada penelitian ini menggunakan algoritme Grain-128a untuk melakukan pengaman pada data. Proses pengamanan yang dilakukan ialah enkripsi dan dekripsi yang akan diletakkan pada *node* dan *gateway*.



Gambar 3. Perancangan alur enkripsi pada *node*

Pada gambar 4 merupakan perancangan alur enkripsi pada *node*. *Node* menerima data suhu dan kelembapan dari sensor yang berada dilapangan. Data suhu dan kelembapan akan diubah menjadi bentuk *binary*. Lalu algoritme Grain-128a melakukan proses *generate keystream* untuk menghasilkan *keystream* yang digunakan pada proses enkripsi. Kemudian data akan diamankan dengan cara melakukan proses enkripsi dan menghasilkan *ciphertext*. Pada algoritme Grain-128a terdapat proses autentikasi yang menghasilkan *tag*. Jika pada pengamanan yang dilakukan terdapat autentikasi pada data, maka hasil yang diperoleh dari proses pengamanan pada data ialah *ciphertext* dan *tag*. Sedangkan jika tidak terdapat autentikasi pada data maka hasil dari proses proses pengamanan pada data ialah *ciphertext* saja. Kemudian hasil dari proses pengamanan data di-*pack* ke dalam *struct* dan dikirimkan ke *gateway* melalui komunikasi antar LoRa. Node menampilkan data asli dan data setelah dilakukan pengamanan dengan algoritme Grain-128a.

Gateway akan melakukan proses dekripsi dengan menggunakan algoritme Grain-128a. Alur pertama pada *gateway* ialah melakukan *generate keystream* algoritme Grain-128a. Jika terdapat autentikasi maka *gateway* melakukan unpack data struct dan menerima data *ciphertext* dan *tag,* sedangkan jika tidak terdapat autentikasi maka *gateway* melakukan *unpack* data struct dan menerima *ciphertext* saja. Jika tidak terdapat proses autentikasi maka *ciphertext* yang diterima dilakukan proses dekripsi untuk menghasilkan *plaintext* dengan menggunakan *keystream* algoritme Grain-128a yang telah dihasilkan pada proses sebelumnya. Kemudian *gateway* menampilkan *plaintext* yang berhasil didekripsikan. Sedangkan jika terdapat proses autentikasi maka *ciphertext* dan *tag* yang diterima dari *node* dilakukan proses dekripsi untuk menghasilkan *plaintext, plaintext* yang dihasilkan akan dilakukan proses autentikasi pada *gateway* untuk menghasilkan *tag* pada *gateway*. Kemudian *tag* yang diperoleh dari *node* akan dibandingkan dengan *tag* yang dihasilkan pada *gateway* untuk mengetahui *plaintext* atau pesan yang di dekripsi benar atau tidaknya. Jika *tag* dari *node* sama dengan *tag* dari *gateway* maka pesan yang di dekripsi benar dan pesan akan ditampilkan. Sedangkan jika *tag* dari *node* dan *tag* dari *gateway* tidak sama maka pesan yang di dekripsi tidak benar atau salah dan pesan akan ditampilkan.

3.2. Perancangan Pengujian

Perancangan pengujian digunakan pada tahap pengujian sistem yang dibangun. Perancangan pengujian diperlukan guna untuk mengetahui keberhasilan penerapan pengamanan data menggunakan algoritme Grain-128a pada modul komunikasi LoRa. Pada lingkungan uji untuk pengujian akan terdiri dari beberapa perangkat yang akan berperan sebagai *node*, *gateway*, dan penyerang. Pengujian yang dilakukan pada penelitian ini diantaranya ialah pengujian pengujian pasif, dan pengujian aktif.

Pengujian pertama yang dilakukan ialah pengujian pasif dengan cara melakukan *sniffing*. Penyerangan *sniffing* merupakan serangan dimana penyerang dapat memantau semua informasi yang melewati jaringan. Pada saat pengujian ini penyerang akan berperan sebagai *gateway* baru. Pada skenario penyerangan ini, serangan dilakukan dengan cara penyerang melakukan *sniffing* dengan menggunakan *brute force* untuk mendapatkan id *node* agar penyerang dapat mengetahui data yang

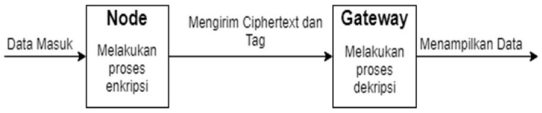
dikirimkan oleh *node* menuju *gateway*. Data yang didapatkan dari serangan ini ialah data yang telah diamankan yaitu data *ciphertext*.

Pengujian yang dilakukan selanjutnya ialah pengujian aktif. Pengujian aktif akan melakukan serangan aktif yang memiliki tujuan untuk modifikasi data. Pada pengujian aktif akan dilakukan penyerangan dengan menggunakan *known-plaintext attack*. *Known-plaintext attack* (KPA) ialah serangan yang bertujuan untuk mendapatkan *key* yang digunakan pada *ciphertext* yang didapatkan tanpa mengetahui algoritme yang digunakan pada *node* untuk mengamankan data. Dimana *ciphertext* yang digunakan dalam pengujian ini ialah *ciphertext* yang didapatkan dari pengujian pasif yaitu melalui *sniffing*. Setelah *key* berhasil didapatkan dari *ciphertext* maka penyerang juga dapat mengetahui data *plaintext* dari *ciphertext* tersebut. Hal ini dapat digunakan untuk melakukan perubahan pada data dan kemudian mengirimkannya ke *gateway* sah tanpa sepengetahuan *node* dan *gateway*. Hasil *key* yang diperoleh oleh penyerang akan dibandingkan dengan *key* yang digunakan oleh sistem. Sehingga jika *key* yang diperoleh penyerang berbeda maka penyerang tidak dapat mengetahui informasi apapun melaui *ciphertext* yang digunakan.

4. IMPLEMENTASI DAN PENGUJIAN

4.1. Implementasi Pengembangan *Node* dan *Gateway*

Pada subbab ini akan melakukan implementasi sistem dengan penambahan pengamanan pada data. Penelitian ini menggunakan LoRa untuk melakukan pengiriman data. Perangkat LoRa yang digunakan ialah LoRa dengan frekuensi 915 mhz. Ada dua perangkat LoRa yang digunakan yang memiliki peran sebagai *node* dan *gateway*.



Gambar 5. Alur Pengamanan data pada LoRa

Gambar 5 merupakan alur pengaman data pada LoRa. Pengamanan data diterapkan pada *node* dan *gateway*. Pada *node* pengaman yang dilakukan ialah enkripsi dengan *optional* autentikasi. Data suhu dan kelembapan dari sensor di amankan dengan algoritme Grain-128a pada *node*. Jika terdapat autentikasi dalam algoritme Grain-128a maka *node* mengirimkan data *ciphertext* dan *tag* yang di-*pack* ke dalam *struct* menuju *gateway*. Sedangkan jika tidak terdapat autentikasi dalam algoritme Grain-128a maka *node* mengirimkan data *ciphertext* dan di-*pack* ke dalam *struct* menuju *gateway*.

Gateway menerima data *struct* yang dikirimkan *node*. Kemudian data *struct* yang diterima di-*unpack.* Jika terdapat autentikasi dalam algoritme Grain-128a maka *gateway* menerima data *ciphertext* dan *tag*. Sedangkan jika tidak terdapat autentikasi maka *gateway* menerima *ciphertext* saja. Jika tidak terdapat autentikasi, *ciphertext* yang diterima dilakukan proses dekripsi dan menghasilkan *plaintext*. Sedangkan jika terdapat autentikasi maka *plaintext* yang dihasilkan dari proses dekripsi *ciphertext* di autentikasi dan menghasilkan *tag* pada *gateway*. Kemudian *tag* pada *node* dibandingkan dengan *tag* pada *gateway*. Jika *tag* pada *node* sama dengan *tag* pada *gateway* maka pesan yang di dekripsi benar dan pesan akan ditampilkan. Sedangkan jika *tag* dari *node* dan *tag* dari *gateway* tidak sama maka pesan yang di dekripsi tidak benar dan pesan akan ditampilkan.

4.2. Pengujian

4.2.1. Pengujian Pasif

Pengujian serangan pasif pada penelitian ini yaitu dengan melakukan serangan *sniffing*. Pada saat serangan *sniffing* terjadi, penyerang dapat mengetahui data yang melewati jaringan komunikasi antar LoRa. Hal ini dikarenakan modul komunikasi LoRa melakukan *broadcast* pada data yang dikirimkan oleh *node* sebagai pengirim. Sehingga LoRa yang memiliki frekuensi sama dengan pengirim dapat menerima data yang dikirimkan. Pada pengujian ini *gateway* baru ditambahkan yang memiliki peran sebagai *gateway* penyerang. *Gateway* penyerang memiliki frekuensi yang sama dengan *node* dan *gateway* sah. Sehingga pada saat *node* mengirimkan data ke *gateway* sah melalui *broadcast* LoRa, *gateway* penyerang juga mendapatkan data yang dikirimkan oleh *node*.

4.2.2. Pengujian Aktif

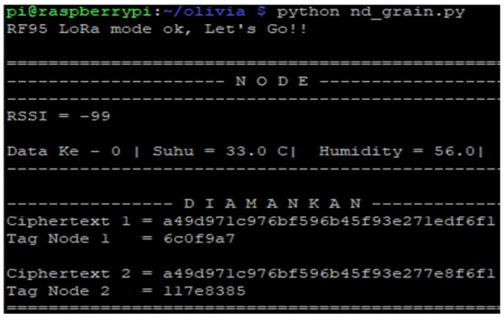
Pengujian aktif dilakukan setelah pengujian pasif dilakukan. Pada pengujian ini dilakukan serangan aktif yang bertujuan untuk modifikasi data. Pada serangan aktif ini, penyerang tidak mengetahui algoritme yang digunakan oleh sistem. Pada pengujian aktif akan dilakukan

serangan dengan *known-plaintext attack* (KPA). Serangan KPA memiliki tujuan untuk mengetahui *key* yang digunakan *ciphertext*. *Key* yang diperoleh dari serangan aktif KPA digunakan untuk mengetahui *plaintext* dari *ciphertext*. Sehingga penyerang akan melakukan modifikasi dari *plaintext* yang diperoleh dari *ciphertext* tersebut dan membuat *plaintext* baru. Untuk melakukan serangan *known-plaintext attack,* penyerang membutuhkan beberapa hal diantaranya *ciphertext* dan contoh *plaintext*. Pada serangan ini, *ciphertext* yang digunakan diperoleh dari pengujian pasif.

5. HASIL DAN ANALISIS

5.1. Hasil Implementasi

Pada subbab ini melakukan analisis pada sistem setelah pengamanan pada data dengan algoritme Grain-128a digunakan. Pengamanan pada data sangatlah penting dikarenakan data yang dikirimkan oleh LoRa yang berperan sebagai pengirim dapat diketahui oleh berbagai pihak yang memiliki frekuensi sama dengan LoRa pengirim. Oleh karena itu diperlukan pengamanan tambahan pada data sehingga pihak yang tidak memiliki wewenang tidak dapat mengetahui data yang dikirimkan oleh *node* ke *gateway* sah. Algoritme Grain-128a digunakan agar data yang dikirimkan oleh *node* tidak dapat diketahui oleh pihak yang tidak memiliki wewenang. Pihak yang memiliki wewenang pada sistemnya juga akan menggunakan algoritme Grain-128a untuk dapat melakukan dekripsi dari data yang dikirimkan oleh *node* untuk mendapatkan data *plaintext*.

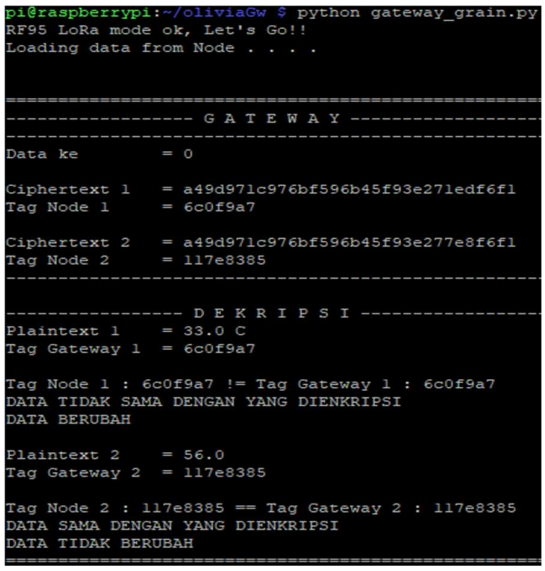


Gambar 4. Pengamanan pada *node*

Pada gambar 6 merupakan hasil keluaran ketika sistem dengan pengamanan dijalankan pada node. Pengamanan yang dilakukan ialah enkripsi dan autentikasi pada data. Proses enkripsi menghasilkan *ciphertext* dan autentikasi menghasilkan *tag*. Dapat dilihat dari gambar diatas pada data ke - 0 yang diamankan ialah suhu dengan nilai "33.0" dan *humidity* dengan nilai "56.0". Hasil dengan pengamanan pada data suhu dapat dilihat pada *Ciphertext* 1 dan *tag* 1 dalam bentuk *hexadecimal*. Hasil dengan pengamanan pada data *humidity* pada *Ciphertext* 2 dan *tag* 2. Hasil dari pengamanan data dikirimkan ke *gateway* melalui komunikasi LoRa dalam bentuk data *struct*.

Gambar 5. Pengamanan pada *gateway*

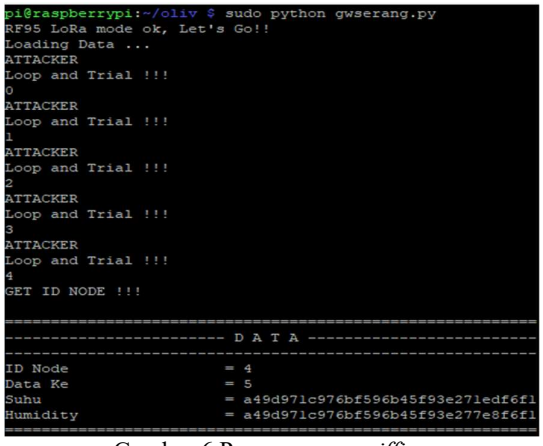
Pada gambar 7 merupakan tampilan



gateway menerima data yang telah diamankan dan melakukan proses dekripsi. Pengamanan yang dilakukan pada *gateway* ialah proses dekripsi dan autentikasi pada *plaintext* hasil dekripsi. Dapat dilihat dari gambar diatas data yang diterima dari *node* ialah data yang telah diamankan yaitu *ciphertext* 1, *tag node* 1, *ciphertext* 2 dan *tag node* 2. *Ciphertext* 1 dilakukan proses dekripsi terlebih dahulu dan menghasilkan *plaintext* 1. Kemudian *plaintext* 1 dilakukan proses autentikasi untuk memastikan bahwa pesan yang dikirimkan tidak berubah dengan membandingkan *tag* yang dihasilkan *gateway* sama dengan *tag* pada *node*. *Plaintext* 1 merupakan hasil dekripsi data suhu yang dikirimkan *node*. *Ciphertext* 2 dan *tag node* 2 diproses sama dengan *ciphertext* 1 dan *tag node* 1. Dapat dilihat dari hasil dekripsi pada *plaintext* 1 dan *plaintext* 2 bahwa data hasil dekripsi sama dengan data yang dienkripsi. Hal ini membuktikan bahwa pengamanan data menggunakan Grain-128a pada modul komunikasi LoRa berhasil diimplementasi.

5.2. Pengujian Pasif

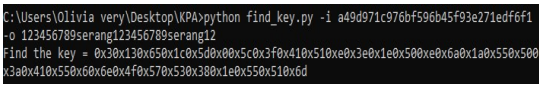
Pada pengujian ini *gateway* baru sebagai penyerang ditambahkan. Melalui *broadcast* pada LoRa dengan frekuensi sama dengan *node* pada saat mengirimkan data menuju *gateway* sah, *gateway* penyerang yang tidak memiliki wewenang melakukan *brute force* untuk mendapatkan id *node*. Setelah penyerang mendapatkan id *node* maka penyerang dapat mengetahui data yang dikirimkan oleh *node*.



Gambar 6 Penyerangan *sniffing*

Gambar 8 merupakan hasil dari penyerangan *sniffing* yang dilakukan oleh *gateway* penyerang. Dapat dilihat pada serangan terjadi, penyerang melakukan *brute force* id *node* terlebih dahulu. Dapat dilihat dari gambar diatas bahwa data yang didapatkan ialah *ciphertext* yang tidak bisa dibaca oleh penyerang. Sehingga penyerang tidak dapat mengetahui nilai sebenarnya dari data *ciphertext* yang dikirimkan oleh *node* ke *gateway* sah. Hal ini membuktikan bahwa pengamanan pada data yang dilakukan oleh *node* berhasil dilakukan.

5.3. Pengujian Aktif



Gambar 9 Pengujian aktif

Pada gambar 9 merupakan pengujian aktif yang dilakukan. Dari serangan yang dilakukan berhasil mendapatkan key. Sedangkan *key* yang digunakan algoritme pada sistem ialah "0123456789abcdef123456789abcdef0". Dapat dilihat bahwa hasil key yang didapatkan dari serangan KPA diatas tidak sama dengan *key* yang digunakan oleh algoritme keamanan pada sistem. Sehingga penyerang tidak dapat mengetahui *plaintext* dari *ciphertext* tersebut melalui *key* yang diperoleh dan tidak dapat melakukan modifikasi pada *plaintext*. Hal ini membuktikan bahwa penyerang tidak berhasil melakukan penyerangan dan algoritme Grain-128a berhasil melakukan pengamanan pada data.

6. PENUTUP

Berdasarkan dari beberapa pengujian yang telah dilakukan pada penelitian ini maka dapat diambil kesimpulan bahwa algoritme Grain-128a yang diimplementasikan pada *node* dan *gateway* berhasil melakukan pengamanan data pada modul komunikasi LoRa. Hal ini dibuktikan dari pengujian pasif dan aktif yang dilakukan dengan cara melakukan serangan tidak berhasil mengetahui data asli yang dikirimkan oleh *node* ke *gateway*.

7. DAFTAR PUSTAKA

Ågren, M., Hell, M., Johansson, T. and Meier, W., 2011. *Grain-128a: a new version of Grain-128 with optional authentication*. *International Journal of Wireless and Mobile Computing*, [online] 5(1), p.48. *Available* at: <http://www.inderscience.com/link.php?id=44106>.

Arijuddin, H., Bhawiyuga, A. and Amron, K., 2019. Pengembangan Sistem Perantara Pengiriman Data Menggunakan Modul Komunikasi LoRa dan Protokol MQTT Pada *Wireless Sensor Network*. 3(2), pp.1655–1659.

Semiconductors, N.X.P., 2018. *IoT Device Secure Connection with LoRa*.

Watanabe, Y., Yamamoto, H. and Yoshida, H., 2018. E*xtending FELICS for Automotive PKES System. 2019 FSE Rump session proceedings,* pp.152-158.

Zourmand, A., 2019. *Internet of Things (IoT) using LoRa technology*. *2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, (June), pp.324–330.