

## Peningkatan Performa Pembangkitan Kunci Rahasia Berbasis RSS menggunakan Metode Sindrom Bit Minimal Ringan untuk IoT

### *Performance Enhancement of RSS-Based Secret Key Generation using Lightweight Minimal Bit Syndrome Reconciliation Method for IoT Devices*

Choirun Nisa<sup>1\*</sup>, M. Cahyo Kriswantoro<sup>2</sup>

<sup>1</sup>Teknologi Komputer, Politeknik NSC Surabaya

<sup>2</sup>Teknik Informatika Medis, Universitas Muhammadiyah Lamongan

Email: <sup>1</sup>choyunnisa@gmail.com, <sup>2</sup>cahyo.krizt@gmail.com

**Abstrak** - Komunikasi yang aman pada perangkat *Internet of Things* (IoT) dapat ditingkatkan melalui metode *secret key generation* pada lapisan fisik (*physical-layer*). Pada metode *secret key generation* terdapat proses informasi rekonsiliasi untuk mencocokkan bit pada kedua perangkat yang ingin berkomunikasi. Pada proses ini seringkali menggunakan metode statistika dengan perhitungan yang panjang yang dapat meningkatkan *overhead* sistem dan juga disertai proses pengiriman bit pada jaringan publik yang tidak aman. Sehingga penelitian ini mengusulkan sebuah algoritma rekonsiliasi yang ringan yang berbasis pembentukan sindrom minimal untuk mengurangi *overhead* dan meningkatkan keamanan kunci. Metode yang diusulkan memproses nilai *Received Signal Strength* (RSS) yang telah dikuantisasi ke dalam mekanisme sindrom blok 8-bit. Sehingga, hanya 8-bit sindrom dari blok yang memiliki ketidakcocokan yang akan dikirimkan melalui jaringan publik. Pengujian sistem dilakukan pada skenario statis dan dinamis pada lingkungan *indoor* menggunakan node *Raspberry Pi* dan dihasilkan peningkatan *Key Generation Rate* (KGR) sebesar 2.2% dan pengurangan waktu pemrosesan hingga 11.6% pada kondisi statis dan 8.3% pada kondisi statis dibandingkan pada penelitian sebelumnya serta menurunkan waktu pemrosesan hingga 50% pada proses kuantisasi dan rekonsiliasi informasi. Selain itu, hasil penelitian juga menunjukkan bahwa metode yang diusulkan dapat mempertahankan *Key Disagreement Rate* (KGR) sebesar 0% pada semua ukuran blok serta lolos uji kompleksitas dan keacakan kunci rahasia dengan *NIST test* dengan hasil pada semua parameter uji memiliki nilai *p-values* diatas 0.01. Hal ini menunjukkan bahwa skema yang diusulkan mampu meningkatkan performa dari skema penelitian sebelumnya dan dapat diimplementasikan pada *device IoT*.

**Kata kunci** : Rekonsiliasi Informasi yang Aman, Pembangkitan Kunci Rahasia, Komunikasi Wireless, IoT.

**Abstract** - *Secure communication in Internet of Things (IoT) devices can be improved through physical-layer secret key generation. In secret key generation, an information reconciliation process is required to match the bit sequences between the two communicating devices. However, this process often involves complex statistical computations that increase system overhead and require the transmission of bits over an insecure public channel. Therefore, this research presents a lightweight reconciliation algorithm based on minimal bit syndrome to reduce overhead and improve key security. The proposed method processes quantized Received Signal Strength (RSS) values into 8-bit block syndrome-based reconciliation mechanism and only the 8-bit syndromes from mismatched block are transmitted over the public channel. System testing was conducted under both static and dynamic scenarios in an indoor area using Raspberry Pi nodes. The results show that the proposed scheme demonstrates an improvement in the Key Generation Rate (KGR) by 2.2% and reduces the processing time by 11.6% in dynamic scenario and 8.3% in the static scenario. Also, the proposed scheme reduce the quantization and reconciliation time by 50% compared to previous research. Furthermore, the proposed method maintained a Key Disagreement Rate (KDR) of 0% across all block size and successfully passed the NIST randomness and complexity test, with all test parameters achieving p-values above 0.01. These result indicate that the proposed scheme improve the performance of existing secret key generation methods and it is suitable for implementation in IoT devices.*

**Keywords** : *Secure information reconciliation, Secret key generation, Wireless communication, IoT*

## I. INTRODUCTION

The Internet of Things (IoT) has changed modern communication systems by allowing billions of connected systems to share data easily. However, this growth of IoT has brought serious security issues, especially in developing a secure communication channel between devices with limited resources [1]. Traditional methods for distributing cryptographic keys, which depend on pre-shared keys or public key infrastructure, often do not work well in IoT devices. This is due to factors of limited computing power, memory restrictions, and energy needs [2]. As a result, researchers are putting more effort into creating a lightweight secret key generation technique that can work well within the limits of IoT devices.

Physical layer security has emerged as a promising approach to generate cryptographic keys in IoT networks without relying on complex computational processes [3]. This method takes advantage of the natural randomness and uniqueness of wireless channel characteristics to create shared secret keys between the communicating parties [4]. However, because of noise, interference, and changes over time in the wireless channel, the information obtained by two communicating parties is not the same. This discrepancy requires advanced reconciliation techniques to achieve the same key information.

The reconciliation technique is an essential part of generating secret keys. It connects similar but not identical random data to perfectly synchronized cryptographic keys [5]. The goal of reconciliation is to remove mismatched bits of key gathered by various devices while reducing the risk of information being leaked to potential eavesdroppers [6]. Traditional reconciliation methods, like cascade reconciliation and low-density parity-check (LDPC) approaches, have been widely studied in quantum key distribution systems [7,8]. However, these techniques often demand a lot of computing power and several rounds of communication, which makes them impractical for resource-limited IoT environments. Other research [9,10], using a filter that is produced from the statistical computation to define the error position on the sequence bit. This technique produces a sequence bit filter with a high number of bits that resulting in many computational processes and high data overload.

So, this paper proposes a reconciliation method that improves bit-correction efficiency while minimizing information leakage and processing time, that suitable for IoT devices. We use a syndrome-based approach that works on quantized

Received Signal Strength (RSS) measurements. The RSS data will divided into fixed-size blocks. For each block, the algorithm generates an 8-bit syndrome, which represents the integrity of the bit sequence. We calculate these syndromes with simple XOR operations and bit-shifted values. This makes the system quickly find mismatches between the bit sequences generated by the two legitimate parties, commonly known as Alice and Bob. It also limits the amount of information shared over the public channel. Unlike traditional methods that might show long parity or error correction vectors [11 - 13], our approach only reveals the indices of the affected blocks when mismatches are detected. This strategy reduces communication overhead while still keep the bit stream private, which improves the overall privacy of the reconciliation phase. This contribution represents a significant step forward in physical-layer security by providing a statistically efficient and computationally practical solution for secure key generation in resource-constrained IoT systems.

## II. METHODE

Our proposed research consists of five stages, as in **Figure 1**. It is channel sampling, quantization, information reconciliation, privacy amplification, and performance evaluation. These stages are designed to systematically extract, process, and secure secret keys from the physical characteristics of the wireless communication channel. The detailed stage is described below.

### A. Channel Sampling

The channel sampling stage is the fundamental stage of secret key generation. In this stage, Alice and Bob, the legitimate communicating parties, measure and collect wireless channel characteristics, focusing on Received Signal Strength (RSS). This helps them establish a shared random bit for creating a cryptographic key. During this process, Alice and Bob both sample the RSS values from their wireless communication channel at the same time. They use the principle of channel reciprocity, which guarantees that both parties have related measurements RSS value because of the same physical propagation environment. The sampling process usually includes sending pilot signals between the devices. In this research, we use the ping command as a pilot signal in the 2.4 GHz frequency carrier. We follow the approach in reference [14] for sampling the RSS from the wireless communication channel and best ping interval is 110 ms, as in [15].

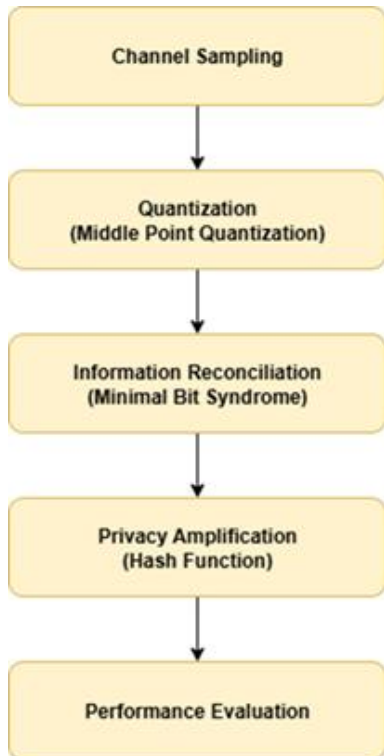


Figure 1. Proposed Research Method

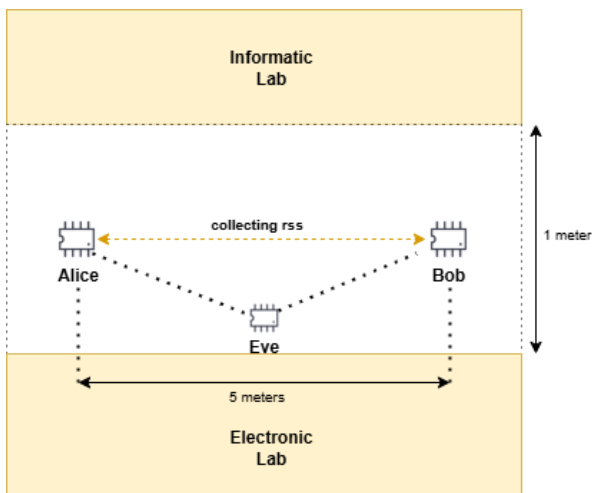


Figure 2. Static Indoor Scenario

The channel sampling process was carried out in two scenarios. The first scenario was carried out in indoor areas with static movement, and the second scenario was carried out in indoor areas with dynamic movement. In an indoor static scenario, Alice and Bob are separated by 5 meters in the indoor area of the alley between classrooms. Alice and Bob are represented by a Raspberry Pi B with a TL-WN722N wireless USB adapter that has a standard wireless network IEEE802.11b/g/n. The frequency used is 2.4 GHz. Alice and Bob did not make any movement until the data sampling reached 4000 data points. In a dynamic scenario, Alice and Bob move, and all of the settings are the same with the static scenario. While Alice and Bob

are sampling the RSS from the communication channel, Eve tries to intercept the communication between Alice and Bob. The details scenario is shown in Figures 2 and 3 and the tools used by every party are shown in Table I. The result for this measurement is RSS data and written as  $R_{a1} \dots R_{b4000}$  on Alice's side,  $R_{b1} \dots R_{b4000}$  on Bob's side and  $R_{e1} \dots R_{e4000}$  on Eve's side.

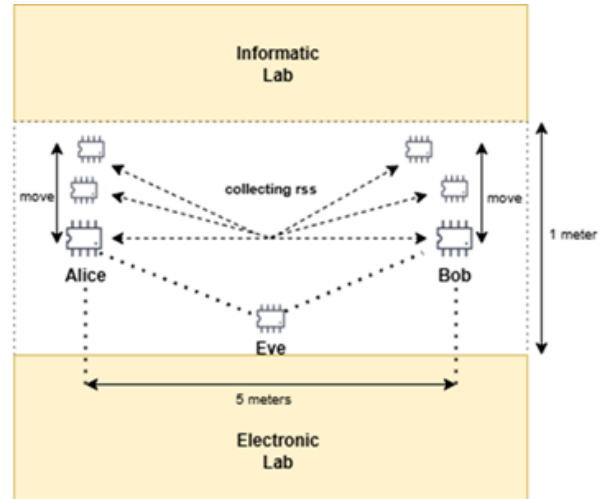


Figure 3. Dynamic Indoor Scenario

Table I. Tools Used in Every Parties/Actor

Actor	Tools	Type
Alice	Raspberry Pi	4B
	Powerbank	Robot
	Wireless USB Adapter	TP Link TL-WN722N
Bob	Raspberry Pi	4B
	Powerbank	Robot
	Wireless USB Adapter	TP Link TL-WN722N
Eve	Raspberry Pi	4B
	Powerbank	Robot
	Wireless USB Adapter	TP Link TL-WN722N

### B. Quantization

The quantization stage represents a critical transformation process in secret key generation, where continuous RSS measurements collected during the channel sampling stage are converted into discrete binary sequences suitable for secret key generation. This research used a middle point quantization technique adopted from [16] and the logic shown in Figure 4. From Figure 4, we can see that the middle point quantization algorithm operates by first computing the arithmetic mean of RSS measurements within predefined temporal blocks, establishing a statistical reference point for the quantization process. The number of intervals,  $n$ , for dividing the RSS into several blocks in this research is 10, 50, 100, 150, 200, and 250. The

algorithm then calculates the middle point threshold for each block by determining the midpoint between the minimum RSS value and the computed mean, creating an adaptive threshold that responds to local channel variations. The quantization decision is then made by comparing each RSS measurement against this dynamically computed threshold. Then, RSS values falling below the middle point threshold are assigned a binary value '1', while measurements exceed the threshold are assigned a binary value '0'. The quantization will be performed on Alice's side and Bob's side. Then we can write the result of the quantization process as  $D_A$  and  $D_B$ . Where  $D_A$  and a sequence of bit data or determined below:

$$D_A = D_{a1} \dots \dots D_{a4000} \quad (1)$$

$$D_B = D_{b1} \dots \dots D_{b4000} \quad (2)$$

### C. Information Reconciliation

The information reconciliation stage uses error correction techniques to eliminate disagreements between quantized sequences generated by legitimate communicating parties (Alice and Bob). This ensures perfect correlation before key generation. In this research, we implement the minimal syndrome algorithm, which offers a better approach to reconciliation. It significantly reduces the information leakage to potential eavesdroppers while maintaining high error correction efficiency [17]. To further improve security against adaptive attacks, the algorithm includes syndrome randomization techniques. It will add random bits into the syndrome calculation, that makes the randomness of the key is high..

On Alice's side, the minimal syndrome algorithm works by dividing the quantized bit sequences,  $D_A$ , into systematic blocks with a defined number of blocks,  $n$ . So every block will have  $b$  number of bits which is:

$$b = \sum D_A / n. \quad (3)$$

After that, Alice must calculate the syndrome vectors as in [18]. The equation shown below:

$$S_A = S_A \text{ XOR } (D_A[b_s = +j] \ll (j \text{ mod } 8)) \quad (4)$$

Where,

$S_A$  : syndrome

$b_s$  : index of starting block

$j$  : iterative variable

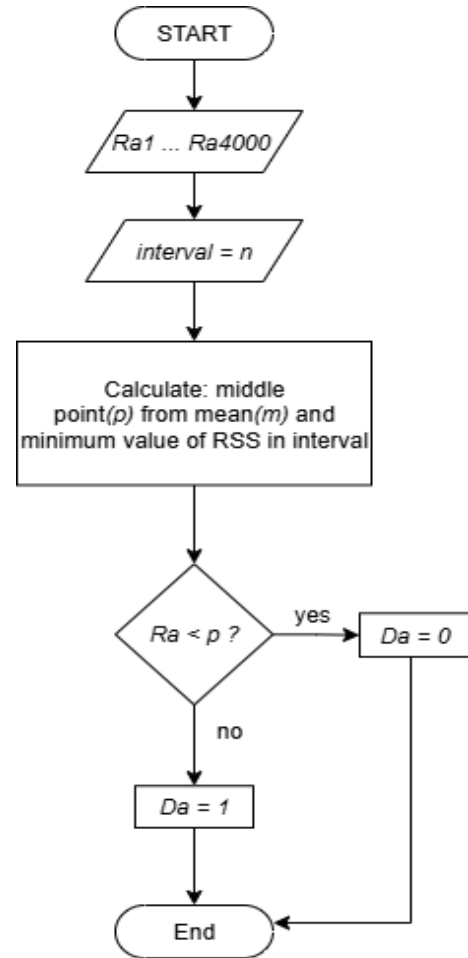


Figure 4. Middle Point Quantization Logic

In this equation (4),  $D_A[b_s = +j]$  refers to the data element at position  $j$  within a block that starts at index  $b_s$ . This value is left-shifted by  $j \text{ mod } 8$  bits. We choose 8 as the length of the bit syndrome. The result of this shift is then XORed with the current syndrome value  $S_A$ , ensuring that each bit in the block contributes nonlinearly to the final syndrome. So can increase the remaining entropy available for key generation [19]. This operation is repeated for each index  $j$  in the block resulting in a single syndrome value that compactly represents the entire block, where  $j$  is:

$$j = n - 1 > j > 0 \quad (5)$$

After that, Alice will ensure that the syndrome is properly divided into expected block bits that allowing interoperability with systems that process data in 8-bit units. The equation (6) shows the operation to extract the least significant 8 bits from the value  $S_A$  and assign it to the  $i$ -th index of the array  $S_A$ . The operation  $AND \ 0xFF$  applies a bitwise mask that clears all bits except the lowest 8 bits, effectively isolating the least significant byte of  $S_A$ .

$$S_A [i] = S_A \text{ AND } 0xFF \quad (6)$$

Then Alice sends only the syndrome information to Bob over the public channel. Alice then received syndrome from Bob. Alice uses the received syndrome to find and correct discrepancies in his corresponding bit sequence through iterative decoding processes.

The reconciliation process continues iteratively until syndrome verification confirms perfect agreement between Alice and Bob's sequences. At that point, both parties have identical bit strings ready for privacy amplification and final key generation. The detailed proposed information reconciliation algorithm is shown in **Algorithm 1**.

#### D. Privacy Amplification

The privacy amplification stage is essential in the secret key generation process, particularly in roles where information may leak to potential eavesdroppers. In this research, we utilize a cryptographic hash function to transform the reconciled bit strings into a shorter, more secure key [15]. This hash function, denoted as  $H()$ , acts on the final agreement of bits between Alice and Bob to create a secure key that is resistant to potential attacks. The output of this hashing process is a key of predetermined length, which effectively reduces the possibility of a successful key recovery by any adversary who may have gained access to the reconciled data. We use 256 bit length key. By applying  $H()$  to the synchronized keys, we ensure that even if an eavesdropper has partial knowledge about the original quantized bit sequences, the output remains secure and unpredictable, resulting in a robust foundation for secure communications in IoT applications [20].

### III. RESULT AND DISCUSSION

#### A. Measurement Result

The result of the channel sampling measurement between Alice and Bob is shown in **Table II**. We need to know the correlation Coefficient on the channel sampling process to make sure that the correlation coefficient from Eve, as an eavesdropper, is low. We use Pearson Correlation to calculate the correlation coefficient [21]. In the static scenario, the correlation between Alice and Bob is relatively high at 0.25, indicating a moderate level of channel reciprocity suitable for secret key generation. In the other hand, the correlations between Eve and Alice (0.0065) and Eve and Bob (0.0097) are extremely low, suggesting that Eve gains minimal information

about the channel characteristics shared by Alice and Bob, which supports the security of the system.

---

#### Algorithm 1: Minimal Bit Syndrome Reconciliation

---

```

INPUT : sequence Bits Data Alice's  $D_A$ ;
OUTPUT: reconciled bit streams with
identical values,  $R$ 
n =10;
b =  $\sum D_A / n$ ;
FOR i = 0 to b-1:
     $b_s = i * n$ 
     $S_A = 0$ 
    FOR j = 0 to n-1:
         $S_A = S_A \text{ XOR } (D_A [b_s + j] \ll (j \text{ mod } 8))$ 
    END FOR
     $S_A [i] = S_A \text{ AND } 0xFF$ 
END FOR

alice_sends( $S_A$ )
alice_receives( $S_B$ )

FOR i = 0 to b-1:
    IF  $S_A [i] \neq S_B [i]$ :
         $E[E_c] = i$ 
         $E_c = E_c + 1$ 
    END IF
END FOR

FOR i = 0 to  $E_c - 1$ :
     $b_i = E[i]$ 
     $b_s = b_i * n$ 
alice_sends( $R_A = D_A [b_s : b_s + n]$ )

Bob adopts Alice's bits for reconciliation.
FOR j = 0 to n-1:
     $R_B [b_s + j] = R_A [b_s + j]$ 
END FOR
END FOR
 $R = R_A = R_B$ 

```

---

Table II. Correlation Coefficient in Channel Sampling

Scenario	User	Correlation Coefficient
static	Alice and Bob	0.25
	Eve and Alice	0.0065
	Eve and Bob	0.0097
dynamic	Alice and Bob	0.018
	Eve and Alice	0.005
	Eve and Bob	0.04

---

However, in the dynamic scenario, the correlation between Alice and Bob drops significantly to 0.018, reflecting the impact of environmental variations on channel stability. While the correlations between Eve and the legitimate users remain low (0.005 with Alice and 0.04 with Bob).

The reduced correlation in Alice and Bob may affect the efficiency and reliability of key generation. Overall, the results demonstrate that while static environments favor secure key establishment due to higher reciprocity and for dynamic conditions pose greater challenges and may require additional techniques to preserve security and reliability.

Figure 5 shows the changes in Received Signal Strength (RSS) across several samples between Alice and Bob, in a static scenario. The RSS values range from about -52 dBm to -62 dBm, indicating a strong and stable signal typical of a

fixed scenario. Both Alice and Bob display closely related RSS patterns. The synchronized changes in their measurements suggest that both parties face similar channel conditions, although minor differences may arise from measurement noise or slight channel asymmetry. But these small variations could cause bit mismatches during quantization and may require reconciliation methods.

Figure 6 presents the RSS (Received Signal Strength) measurements of Alice and Bob across a sample in a dynamic scenario. The RSS values primarily fluctuate between -60 dBm and -67 dBm, the overall trend shows that Alice and Bob's RSS traces remain moderately correlated, indicating that channel reciprocity is still present but with slightly increased differences. These deviations could lead to a higher bit mismatch rate during quantization that required more robust error reconciliation methods.

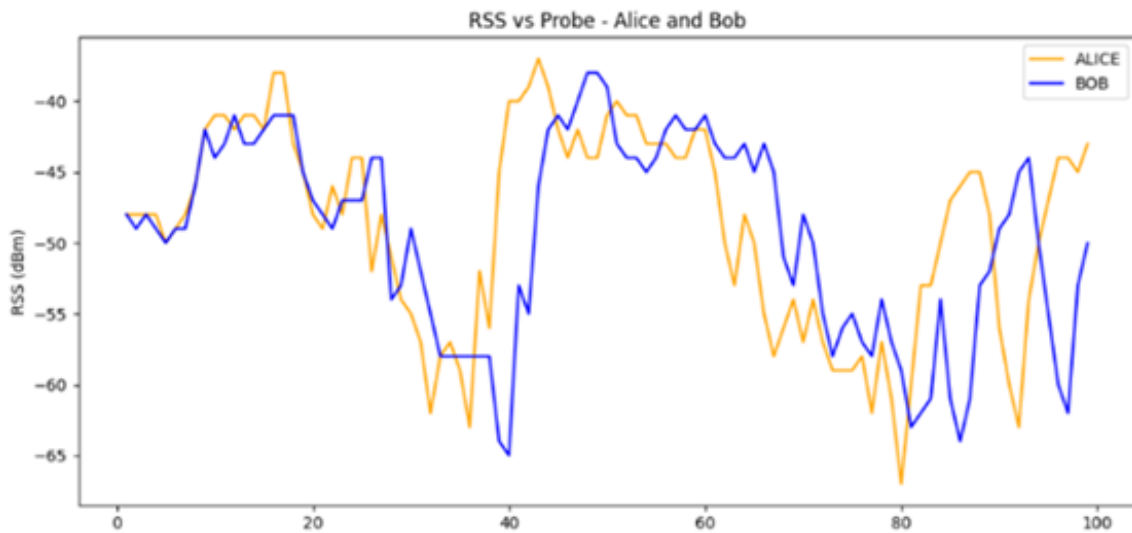


Figure 5. RSS Data Variation in Static Scenario

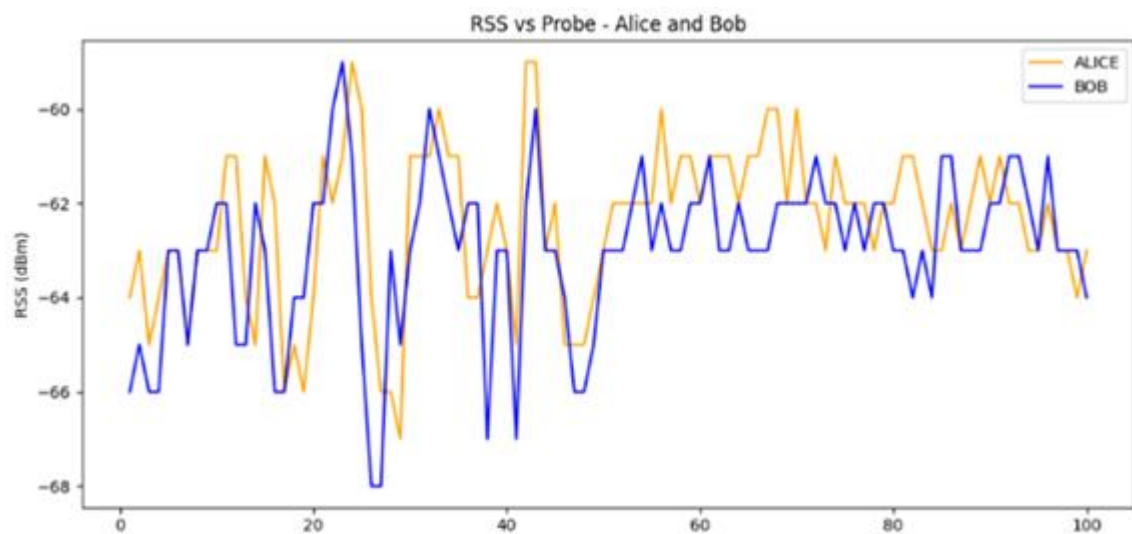


Figure 6. RSS Data Variation in Dynamic Scenario

**B. Performance Evaluation**

In this stage, we will evaluate the performance of the proposed scheme. The metrics are Key Disagreement Rate (KDR), Key Generation Rate (KGR), processing time and NIST test. The details of the performance evaluation are described below.

**1) Key Disagreement Rate**

Key Disagreement Rate (KDR) is the percentage of mismatch bits between the key sequences produced by Alice and Bob after the information reconciliation process. This metric is important for assessing how well the quantization and reconciliation steps work in the physical-layer key generation method. In this research, we measured the KDR by changing the number of blocks used during the key generation process. We specifically looked at block counts of 50, 100, 150, 200, and 250 to see how block size affects the consistency of key bits between Alice and Bob.

The results of this measurement are shown in **Table III**. From **Table III**, we can see that the bits produced during the initial quantization stage still have some mismatches. This is normal because of issues with channel reciprocity and signal noise. But, the data indicates that as the number of interval blocks increases, the number of mismatched bits decreases. This means that smaller block sizes may lead to more consistent bit generation due to less variation in the channel over shorter time periods. We also observed that the dynamic scenario results in fewer mismatched bits compared to the static scenario. This means the quantization algorithm used is more reliable and suits environments with frequent or unpredictable signal changes. In these situations, movement and changing conditions can improve channel randomness and reciprocity.

**Table III.** KDR After Proposed Information Reconciliation Algorithm

Block Number	Mismatch Bit After Quantization		KDR (%) After Information Reconciliation	
	static	dynamic	static	dynamic
	50	314	213	0
100	154	98	0	0
150	171	59	0	0
200	144	49	0	0
250	129	44	0	0

After the quantization phase, all mismatched bits are corrected through the information reconciliation process. In all tested scenarios and block setups, the reconciliation algorithm

successfully removes all mismatches, leading to a KDR of 0%. This result shows that the proposed method ensures that both Alice and Bob can generate identical key sequences. Achieving a consistent 0% KDR demonstrates that the reconciliation protocol is very effective, even with initial quantization errors.

**2) Key Generation Rate**

Key Generation Rate (KGR) refers to the number of bits produced during the secret key generation process. It is an important metric for assessing how well physical layer security mechanisms work. As stated in [16], the KGR is calculated up to the information reconciliation stage. This stage includes quantization and error correction but does not include privacy amplification. In the current setup, the Received Signal Strength (RSS) is sampled every 110 milliseconds. With a total of 4000 RSS samples, the overall time for sampling is about 440 seconds. This duration shows how time-consuming the data acquisition phase can be, especially with a longer sampling interval. Additionally, the processing stage from quantization to reconciliation requires more time based on the number of blocks as shown in **Table IV**. The processing time increases in line with the number of interval blocks because more block used mean more computational demands. However, even with the increased processing time, the KGR remains the same across different interval blocks in both static and dynamic environments, 9.1 milliseconds. This consistency shows that the key generation output is not directly impacted by the block size in the current sampling setup.

**Table IV.** Time Process and Key Generation Rate

Block Number	Quantization and Reconciliation Time (ms)		Key Generation Rate (bit/s)	
	static	dynamic	static	dynamic
	50	0.0196	0.0191	9.2
100	0.0253	0.0253	9.2	9.2
150	0.0313	0.0300	9.2	9.2
200	0.0373	0.0401	9.2	9.2
250	0.0424	0.0420	9.2	9.2

**3) NIST Test**

The National Institute of Standards and Technology (NIST) Statistical Test Suite (STS) is a comprehensive set of statistical methods designed to evaluate the randomness properties of binary sequences generated by a cryptographic algorithm [22]. Randomness is a fundamental requirement in secure communication and cryptography because nonrandom or predictable

sequences can lead to vulnerabilities that compromise key confidentiality and system integrity [23]. So we use NIST test to evaluate the randomness of secret key. NIST test consists of 15 statistical tests but in this research, we only use tests such as approximately entropy, block frequency, longest run, cumulative sum reverse, cumulative sum forward, and frequency. The result of NIST test is shown in **Table V** for dynamic condition and **Table VI** for static condition.

**Table V.** *P-values* of Nist Test with Key Length 256-bit in Dinamic Condition

Parameter	Block length				
	50	100	150	200	250
Approximately Entropy	0.99	0.99	0.99	0.97	0.96
Block Frequency	0.84	0.14	0.2	0.21	0.49
Longest Run	0.42	0.99	0.92	0.81	0.76
Cusum Reverse	0.68	0.57	0.42	0.80	0.94
Cusum Fordward	0.52	0.69	0.52	0.23	0.86
Frequency	0.45	0.90	0.90	0.45	0.71

**Table VI.** *P-values* of Nist Test with Key Length 256-bit in Static Condition

Parameter	Block length				
	50	100	150	200	250
Approximately Entropy	0.97	0.95	0.82	0.98	0.97
Block Frequency	0.69	0.54	0.74	0.42	0.27
Longest Run	0.72	0.97	0.38	0.85	0.93
Cusum Reverse	0.46	0.90	0.30	0.27	0.80
Cusum Fordward	0.42	0.80	0.12	0.63	0.99
Frequency	0.38	0.71	0.70	0.62	0.71

From **Table V** and **Table VI** we can see that the obtained *p-values* for all tests achieve the threshold of 001, indicating that the sequences successfully passed all randomness criteria. The approximately entropy test result shows the consistency *p-values* at 0.96-0.99 in the dynamic scenario and 0.82-0.97 in the static scenario. This indicates that the generated bits have high entropy and have no significant repetitive pattern. The block frequency test results range is between 0.14 and 0.84 in the dynamic scenario and 0.27-0.74 in the static scenario, showing that the local distribution of ones and zeros remains balanced across different block sizes. In the longest run test, *p-values* between 0.42 and 0.99 in the dynamic scenario and 0.27-0.74 in the static scenario indicate that no dominant runs of identical bits occurred and show that the bitstream maintains statistical independence. Then, for cumulative sum tests (forward and reverse), show that *p-values* above 0.2 in dynamic scenarios and above 0.9 in static scenarios demonstrate that there is no directional bias in the accumulation of bits

throughout the sequences. For the frequency (monobit) test, the *p-values* range from 0.45-0.90 in the dynamic scenario and 0.38 to 0.71 in the static scenario, indicating that the proportion of ones and zeros is nearly proportional. Overall, both static and dynamic data meet the NIST randomness standards, validating the effectiveness of the proposed key generation process for secure cryptographic applications, particularly in IoT and physical-layer security systems.

#### 4) Performance Comparison with Other Schemes

The scheme proposed in [16] uses the Clover Filter algorithm to correct bits between Alice and Bob during the information reconciliation process. This algorithm identifies and fixes mismatched bits by using the statistical method and a fixed number of bits for reconciliation information, which is 4800 bits. The existing scheme, as reported in [16], has a Key Generation Rate (KGR) of 9 bits per second. In comparison, the proposed scheme achieves a slightly higher KGR of 9.2 bits per second. While this increase may seem small, it represents a 2.2% improvement in key generation efficiency. This improvement is significant in resource-constrained environments like IoT, where every bit of throughput matters.

For the Key Disagreement Rate (KDR), both the existing and proposed schemes show the same result of 0%. This indicates that the bit correction method in the proposed scheme is as effective as the original scheme. A KDR of 0% means all mismatched bits are successfully corrected during reconciliation, resulting in identical final keys for Alice and Bob. This demonstrates the proposed scheme's reliability and strength, not just in improving efficiency but also in ensuring accuracy in key agreement.

The last comparison is shown in **Figure 7**, which depicts the processing time from channel probing until privacy amplification for different schemes in both static and dynamic scenarios. **ES** stands for the existing scheme in a static scenario, **PS** refers to the proposed scheme in a static scenario, **ED** represents the existing scheme in a dynamic scenario, and **PD** indicates the proposed scheme in a dynamic scenario. The figure clearly illustrates that the proposed scheme consistently improves the existing scheme in processing time across all tested interval lengths. In other words, the proposed method not only improves the key generation rate but also lowers the computational overhead. In the static scenario, the proposed scheme reduces processing time by about 11.6%.

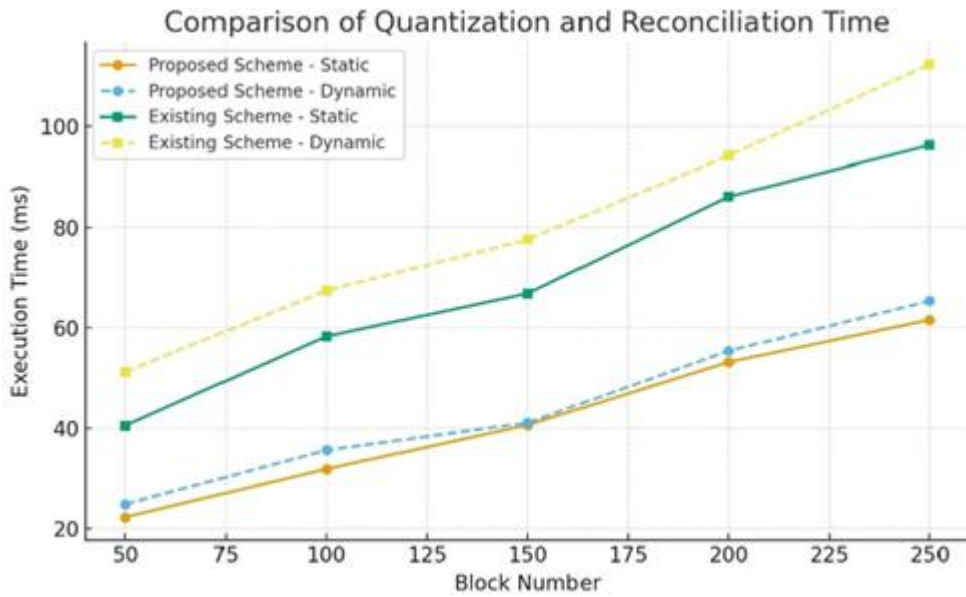


Figure 6. Comparison of Quantization and Reconciliation Time Between the Existing Scheme[ 16] and the Proposed Scheme

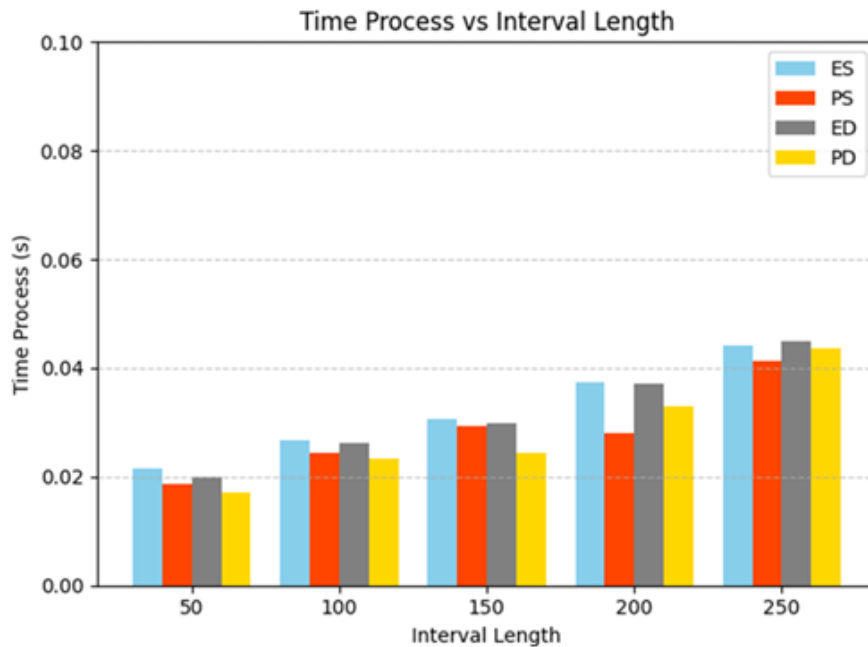


Figure 7. Comparison of Processing Time Between the Existing Scheme[ 16] and the Proposed Scheme

This shows significant efficiency, especially for systems that need quick key establishment without compromising security. In the other hand, in the dynamic scenario, the proposed method lowers processing time by 8.3%, which is also improves the efficiency in processing time in complex wireless environments. These reductions in processing time make the proposed scheme better suited for real-time applications and systems with limited computational resources.

#### IV. CONCLUSION

The proposed key generation scheme shows significant improvements over the existing Clover Filter-based method [11] in terms of KGR and time

processing. It achieves a 2.2% increase in Key Generation Rate (KGR) while keeping the Key Disagreement Rate (KDR) at 0%. This means the proposed scheme guarantees reliable key agreement without losing accuracy. Additionally, there is a decrease in processing time, 11.6% in static scenarios and 8.3% in dynamic scenarios. This highlights how effective the proposed approach is for improving computational efficiency. The proposed scheme also passed the NIST test with *p-value* in all parameter is above 0.01 and indicate that the proposed scheme can produce a strong secret key. Also, the proposed scheme improve approximately 46-50% for processing time during quantization and

reconciliation time. These results suggest that the proposed scheme increases throughput and better suited for resource-limited environments like IoT systems. In the future research, need to improve the channel probing time to reduce the processing time in full stage. Overall, the method provides a secure, lightweight, and practical solution for physical-layer key generation in both static and dynamic wireless settings.

## ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to the Ministry of Higher Education, Science and Technology of the Republic of Indonesia (*Kementerian Pendidikan Tinggi, Sains dan Teknologi / Kemdiktisaintek*) for the research grant support under contract numbers **169/LL7/DT.05.00/PL/2025** and **001/P3M-NSC/VI/2025**. This support has been instrumental in enabling the completion of this research.

## REFERENCES

- [1] D. Canavese, L. Mannella, L. Regano, and C. Basile, "Security at the Edge for Resource-Limited IoT Devices," *Sensors*, vol. 24, no. 2, p. 590, Jan. 2024, doi: 10.3390/s24020590.
- [2] J. Li, Y. Xiao, S. Li, and T. Li, "Designing accountable IoT systems to overcome IoT storage limitation," *Computers & Security*, vol. 148, p. 104118, 2025, doi: 10.1016/j.cose.2024.104118.
- [3] E. Illi, M. Qaraq, S. Althunibat, and A. Alhasanat, "Physical layer security for authentication, confidentiality, and malicious node detection: A paradigm shift in securing IoT networks," *IEEE Communications Surveys & Tutorials*, vol. PP, no. 99, pp. 1–1, Jan. 2023, doi: 10.1109/COMST.2023.3327327.
- [4] X. Wang, M. Waqas, S. Tu, S. Ur Rehman, R. Soua, O. Ur Rehman, S. Anwar, and W. Zhao, "Power maximisation technique for generating secret keys by exploiting physical layer security in wireless communication," *IET Communications*, vol. 14, no. 6, pp. 872–879, Apr. 2020, doi: 10.1049/iet-com.2019.0956.
- [5] J. Wallace, R. Mehmood, R. K. Sharma, and W. Henkel, "Physical-layer key generation and reconciliation," in *Communications in Interference Limited Networks*, B. Błaszczyszyn, M. Kountouris, and H. P. Keeler, Eds. Cham, Switzerland: Springer, 2016, pp. 393–430, doi: 10.1007/978-3-319-22440-4\_17.
- [6] H. Tang, G. Li, T. Guo and A. Hu, "A VT-Code-Based Information Reconciliation Scheme for Secret Key Generation Using RSS," in *IEEE Communications Letters*, vol. 28, no. 4, pp. 783–787, April 2024, doi: 10.1109/LCOMM.2024.3362362
- [7] H.-K. Mao, Y.-C. Qiao, and Q. Li, "High-efficient syndrome-based LDPC reconciliation for quantum key distribution," *Entropy*, vol. 23, no. 11, p. 1440, Nov. 2021, doi: 10.3390/e23111440.
- [8] I. Tsatsaragkos, N. Kanistras and V. Paliouras, "A syndrome-based LDPC decoder with very low error floor," *17th International Conference on Digital Signal Processing (DSP)*, Corfu, Greece, 2011, pp. 1–6, doi: 10.1109/ICDSP.2011.6004950.
- [9] Z. Zhang, G. Li, and A. Hu, "An adaptive information reconciliation protocol for physical-layer based secret key generation," in *Proc. IEEE Vehicular Technology Conf. (VTC-Spring)*, Apr. 2019, pp. 1–5, doi: 10.1109/VTCSpring.2019.8746667.
- [10] D. Dechene, D. E. Luciani, and P. Popovski, "Physical Layer Secret Key Generation with Kalman Filter Detrending," *arXiv preprint arXiv:2305.04540*, 2023
- [11] X. Furqan, J. Hamamreh, and H. Arslan, "Physical Layer Secret-Key Generation Scheme for Transportation Security Sensor Network," *Wireless Personal Communications*, vol. 121, 2021.
- [12] L. Jin, Z. Deng, X. Hu, and Z. Zhang, "Securing NextG Networks with Physical-Layer Key Generation: A Survey," *Security and Safety*, vol. 9, Jan. 2024.
- [13] N. Aldaghri and H. Mahdaviyar, "Physical layer secret key generation in static environments," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3010–3025, 2020.
- [14] Y. Wang, J. Liu, L. Zhang, and X. Hu, "Scramble-Based Secret Key Generation Algorithm in Physical Layer Security," *Mobile Information Systems*, vol. 2022, Article ID 8422490, 2022.
- [15] M. Yuliana, Wirawan, and Suwadi, "A Simple Secret Key Generation by Using a Combination of Pre-Processing Method with a Multilevel Quantization," *Entropy*, 2019, 21, 192.
- [16] C. Nisa., A. Sudarsono., & Y. Mike. "RSS-based Secret Key Establishment using MiddlePoint Quantization and Clover Filter Algorithm". *International Electronics Symposium (IES)*, IEEE, 2020.
- [17] Z. Wan and K. Huang, "Non-reconciliation Secret Keys Based Secure Transmission Scheme Using Polar Codes," 2019 IEEE 5th International Conference on Computer and Communications (ICCC), Chengdu, China, 2019, pp. 1499–1504, doi: 10.1109/ICCC47050.2019.9064302.
- [18] H. Mani, T. Gehring, P. Grabenweger, B. Ömer, C. Pacher, and U. L. Andersen, "Multiedge-type low-density parity-check codes for continuous-variable quantum key distribution," *Physical Review A*, vol. 103, no. 6, p. 062419, 2021.
- [19] P. Treeviriyapab and C.-M. Zhang, "Efficient integration of rate-adaptive reconciliation with syndrome-based error estimation and subblock confirmation for quantum key distribution," *Entropy*, vol. 26, no. 1, p. 53, 2024.
- [20] N. Sharma, H. P. Sultana, R. Singh, and S. Patil, "Secure Hash Authentication in IoT based Applications," *Procedia Computer Science*, vol. 165, pp. 328–335, 2019, doi: 10.1016/j.procs.2020.01.042.
- [21] W. Li and S. Zhao, "Privacy amplification scheme based on composite coding," *arXiv preprint arXiv:2109.07139*, Sep. 2021.
- [22] M. Adil, H. Ullah Khan, M. Arif, M. Shah Nawaz and F. Khan, "New Dimensions for Physical Layer Secret Key Generation: Excursion Lengths-Based Key Generation," in *IEEE Access*, vol. 12, pp. 82972–82983, 2024, doi: 10.1109/ACCESS.2024.3411556.
- [23] R. Upadhyay, S. Singh, V. Trivedi and A. Soni, "Randomness Test for Wireless Physical Layer Key Generation," 2018 International Conference on Advanced Computation and Telecommunication (ICACAT), Bhopal, India, 2018, pp. 1–6, doi: 10.1109/ICACAT.2018.8933725.