Chacha stream cipher implementation for network security in wireless sensor network

F Muhammad, C Ahendyarti and Masjudin
Department of Electrical Engineering, Universitas Sultan Ageng Tirtayasa. Jl. Jendral Soedirman KM 03. Cilegon, Indonesia
E-mail: fadil.muhammad@untirta.ac.id

Abstract. In the 4.0 industrial revolution, technologies are designed to be able to connect to the internet. These technologies should be able to exchange data between them using sensors and others. One of the ways to send it is by the wireless sensor network. However, involving the internet in exchanging data needs security guarantees. Three things must meet in network security, that is confidentiality, integrity, and availability. Cryptography use for the confidentiality of the information that will be sent by the user. In this research will be using the symmetric cryptography for network security on wireless sensor networks. In addition to network security, symmetric cryptography method will be used also consider energy efficiency on a wireless sensor network. Chacha stream cipher as symmetric cryptography will be used as a method of messages encryption. Experiments in this research used NS-3 as a network simulator. Results of the tests show that the purpose method has a computation time of 0.2385 ms and the second-fastest after the rabbit cryptographic process. Chacha LR WPAN simulation process has a computation time of 17.6199 ms.

1. Introduction
Internet of things (IoT) plays a vital role in the development of industry 4.0 [1]. IoT has a central role in progress in several technology sectors such as the development of smart environments (smart city, smart home, smart parking), integrated manufacturing, integrated health services, transportation, environmental monitoring, and the military. A variety of these technologies can be made a system such as applications on smart home, smart city, health information systems, integrated factory systems, and other systems that require the presence of sensors. There are five basic types of IoT technology that are currently being developed successfully: radio frequency identification (RFID), wireless sensor networks (WSN), middleware, cloud computing, and software from IoT applications [2].

WSN allows devices to obtain information data around the environment taken from embedded sensors and transmit that information to other instruments for processing [3]. Unlike personal computers, WSN has limited power, computing, and Random access memory (RAM) when exchanging information. Therefore there are several challenges involved in sending information on WSN. One of the difficulties in sending information by WSN is a guarantee of security for data transmission carried out by WSN devices with limitations on power, computing, and RAM.

There are several standards required for sending data on telecommunications media, namely confidentiality, integrity, and availability or commonly called the CIA Triad [4]. This standard guarantees that information can be appropriately received and cannot be seen or modified by other parties who do not get access to the data. Some areas of IoT such as in the health and military fields, data security is vital.
In this research, WSN will use with symmetric cryptography that will protect the confidentiality of data. The type of symmetric cryptography used is the chacha stream cipher, which is part of The eSTREAM Project [5].

2. Wireless Sensor Network
Unlike conventional sensors that the only function to detect environmental conditions, the wireless sensor network (WSN) has the additional task of communicating with other devices [3]. WSN communicates with other devices without using cable or wireless. In an IoT system, there will usually be more than sensors that will be interconnected. This interconnected sensor requires a network topology so that the sensor can communicate with the other sensor. In a network with multiple sensors, there is a base station/gateway that is intended to pass on to other networks. Gateway on WSN will pass data to the network outside the sensor for further processing.

With the limitations of the power, computing, and memory owned by WSN, need special attention in the implementation of network security in the WSN. As in general network security standards, network security in WSN must also meet several criteria, namely confidentiality, integrity, and availability [6]. Apart from that, there is also some special attention that must be paid attention to WSN network security. An explanation of the standard parameters for WSN network security as follows:

- **Data confidentiality:** In this standard, all data transmitted must be guaranteed confidentiality. Data confidentiality is usually done using cryptographic techniques.
- **Source authentication:** This standard guarantees that the data received originates from the system node and not from another place. In doing authentication, a hash function used as a digital signature.
- **Data integrity:** this standard guarantees that the data sent is data that has not changed data. Data that changes can be corrupt data or data that has been modified by other parties.
- **Availability:** This standard guarantees that each node can transmit data properly and avoid denial-of-service attacks that can result in a full network and WSN nodes unable to communicate.
- **Data freshness:** this condition requires only the latest data to be accepted by the WSN. That way, the old data from each WSN will not be sent back. This way, the WSN will avoid data modifications.
- **Self-organization:** WSN does not have a fixed architecture so that each node and sensor are independent and flexible in managing their respective circumstances.
- **Time synchronization:** with limited resources, each WSN has a mechanism to turn off the device at a particular time.
- **Secure localization:** efficiency in WSN depends on the ability of WSN to find all sensors in the WSN network. In the network, there is usually local security that has been formed.

3. Symmetric Cryptography
The basic concept of cryptography is to change the message we will send in the form of plaintext by using a key by encrypting it into ciphertext and turning it back into plaintext by decryption using the key. With this encryption, every message that will send through a telecommunications network cannot be seen in the contents of the plaintext, but only the ciphertext of the message is visible. When the message has arrived at the recipient, then the message is returned decrypted to see the original plaintext from the sender. In network security, there are two types of cryptography, namely symmetric cryptography and asymmetric cryptography [7]. The difference between symmetric cryptography and asymmetric cryptography is the use of keys when encryption and decryption. In symmetric cryptography, the key used when encryption and decryption is the same key, namely the secret key. In contrast to asymmetric cryptography, which uses different keys, namely the public key for the encryption process and the private key in the decryption process.

The process of encryption and decryption on symmetric cryptography and asymmetric cryptography can be seen in figure 1.
Figure 1. Symmetric vs asymmetric cryptography.

This research uses symmetric cryptography for encryption and decryption plaintext on WSN. Symmetric cryptography uses two types of methods, namely block cipher and stream cipher. The main difference between a block cipher and stream cipher are the methods for converting the plaintext into ciphertext. Stream cipher converts the plaintext into ciphertext by taking 1 byte of plaintext at a time, while block cipher taking one block plaintext at a time. Block cipher usually uses 64 bits or more, while stream cipher uses 8 bits. A standard algorithm that uses block cipher is AES [8], Serpent [9], RC5 [10] etc. Several algorithms that use a stream cipher method, which is also part of the eSTREAM project is Salsa [11], Sosemanuk [12], HC128 [13], Rabbit[14]. This research uses Chacha stream cipher[5], which is also a variant of salsa stream ciphers family.

4. Chacha Stream Cipher

Chacha is a stream cipher based on the 8-round cipher Salsa20/8. Salsa20 was designed in 2005 and submitted to eSTREAM project. Like Salsa stream cipher, there are three basic operations on the Chacha stream cipher. These operations are 32-bit addition operation, 32-bit XOR operation, and constant-distance 32-bit rotation[11]. There is no S-box lookup in Chacha operation. One of the differences between Chacha and Salsa is the initial state matrix. Both Salsa and Chacha state of encryption are stored with 12 32-bit word values within a 4X4 matrix. Salsa initial state is made up of eighth words of a key, four words of input, and four constant words. The initial state of Salsa and Chacha can be seen in figure 2 and figure 3.

| Cons | Key | Key | Key |
|------|-----|-----|-----|
| Key  | Cons| Input| Input|
| Input| Input| Cons | Key |
| Key  | Key | Key | Cons |

Figure 2. Salsa initial state.

| Cons | Cons | Cons | Cons |
|------|------|------|------|
| Key  | Key  | Key  | Key  |
| Key  | Key  | Key  | Key  |
| Input| Input| Input| Input|

Figure 3. Chacha initial state.

Like Salsa, Chacha uses four addition, four XOR, and four rotation s to invertibly four 32-bit state words. Chacha change the Salsa quarter round QR (a, b, c, d) with:
Then a double round in Chacha is:

```plaintext
// Odd round
QR(0, 4,  8, 12)  // 1st column
QR(1, 5,  9, 13)  // 2nd column
QR(2, 6, 10, 14)  // 3rd column
QR(3, 7, 11, 15)  // 4th column
// Even round
QR(0, 5, 10, 15)  // diagonal 1 (main diagonal)
QR(1, 6, 11, 12)  // diagonal 2
QR(2, 7,  8, 13)  // diagonal 3
QR(3, 4,  9, 14)  // diagonal 4
```

Chacha has been selected for implementation of TLS/SSL on Google’s Website. Implementation Chacha on TLS has been published in RFC 7539. OpenSSH added Chacha stream cipher to SSH after Google’s implement Chacha to TLS. Chacha also used by WireGuard VPN protocol.

5. Proposed Method
This research uses Chacha stream cipher for confidentiality on WSN IEEE 802.15.4 or Low Rate Wireless Personal Area Network(LR WPAN). Simulation on LR WPAN will use Network Simulator 3 (NS3) with a two-node sensor for sending and receive data. Node n1 will encrypt and transfer data to node n2. After receiving data, node n2 will decrypt the message sent by node n1. Before transfer data between node n1 and node n2, NS3 will generate a key for the encryption and decryption process. The data encryption and decryption process will use the Chacha20 method with the same key. In the simulation, the distance between node 1 and node 2 is 10 meter. Simulation process can be seen in figure 4.

6. Experiment and Analysis
In this experiment research, we compared chacha stream ciphers as the proposed method with other stream cipher methods for confidentiality on WSN IEEE 802.15.4. The stream cipher that we compare is part of the eSTREAM project that is Salsa, HC128, Rabbit, and Sosemanuk. We did the calculation of computational time on each stream cipher. Each stream cipher will be calculated the computation time during the encryption and decryption process and computation time when simulating the LR-WPAN using NS3. Cryptography process calculation time and LR-WPAN simulation calculation time can be seen in table 1.
### Table 1. Calculation time of the cryptography process and LR WPAN simulation process.

| Cryptography process (ms) | LR WPAN Simulation Process (ms) |
|---------------------------|---------------------------------|
| Chacha                    | 0.2385                          |
| Salsa                     | 0.2704                          |
| HC128                     | 0.2539                          |
| Rabbit                    | 0.2171                          |
| Sosemanuk                 | 0.2735                          |

As seen in table 1, the chacha cryptographic process has a computation time of 0.2385 ms and the second-fastest after the rabbit cryptographic process. Chacha LR WPAN simulation process has a computation time of 17.6199 ms. The computational process of chacha is not much different from Salsa and other eSTREAM project cryptography. Compared to salsa, the cryptographic process of chacha is faster. That's because of Chacha quarter round diffuses changes more quickly. However, in this experiment, the computation time on the chacha stream is slightly slower than Rabbit.

In this experiment, the result of the all stream cipher cryptography process and simulation process not much different. This experiment is made similar to the WSN conditions, which have a relatively small amount of plaintext. This result can be different if the length of the plaintext is different from the experiment. This because each stream cipher has different characteristics. Thus the number of plaintexts can affect the performance of each stream cipher.

### 7. Conclusion and Future Works

In the 4.0 industrial revolution, technologies are designed to be able to connect to the internet. These technologies should be able to exchange data between them using sensors and others. One of the ways to send it is by the wireless sensor network. Security is one of the critical components of data transmission. Confidentiality of network security can use cryptographic techniques. In this research, Chacha stream ciphers are used as cryptographic techniques. Chacha stream ciphers are used because they have a low computation that is suitable for WSN. Research result shows that the Chacha cryptographic process has a computation time of 0.2385 ms and the second-fastest after the rabbit cryptographic process. Chacha LR WPAN simulation process has a computation time of 17.6199 ms. The computational process of chacha is not much different from Salsa and other eSTREAM project cryptography.

For further research, authentication and hybrid cryptography might be implemented to improve data security of WSN. Also needed some network security attacks to test data security methods. Moreover, experiments perform are still at the level of simulation. In the future, it is necessary to do experiments on WSN devices.

### Acknowledgments

This work gets support by a grant of Internal Faculty of Engineering UNTIRTA year 2019 entitled *Pengembangan Hybrid Cryptography Pada Keamanan Jaringan Wireless Sensor Network (WSN).*

### References

[1] Hermann M, Pentek T and Otto B 2016 Design Principles for Industrie 4.0 Scenarios *Proceedings of the 2016 49th Hawaii International Conference on System Sciences (HICSS)* HICSS ’16 (Washington, DC, USA: IEEE Computer Society) pp 3928–37

[2] Lee I and Lee K 2015 The Internet of Things (IoT): Applications, investments, and challenges for enterprises *Bus. Horiz.* 58 431–40

[3] Dargie W and Poellabauer C 2011 *Fundamentals of Wireless Sensor Networks: Theory and*
Practice

[4] International Organization for Standardization 2013 INTERNATIONAL STANDARD ISO / IEC Information technology — Security techniques — Information security management systems — Requirements Inf. Technol. — Secur. Tech. — Inf. Secur. Manag. Syst. — Requir. 2014 38

[5] Bernstein D J 2008 ChaCha , a variant of Salsa20 1–6
[6] Burhanuddin M, Mohammed A, Ismail R, Hameed M E, Kareem A N and Basiron H 2018 A Review on Security Challenges and Features in Wireless Sensor Networks: IoT Perspective J. Telecommun. Electron. Comput. Eng. 10 17–21

[7] Stallings W 2012 Cryptology and Network Security vol 7712
[8] Daemen J and Rijmen V 2002 The Design of Rijndael (Berlin, Heidelberg: Springer-Verlag)
[9] Anderson R and Knudsen L Serpent : A Proposal for the Advanced Encryption Standard 1–23
[10] Rivest R L 2 A Parameterized Family of Encryption Algorithms
[11] Bernstein D J 2007 The Salsa20 family of stream ciphers
[12] Berbain C, Billet O, Canteaut A, Courtois N and Gilbert H 2002 Sosemanuk , a fast software-oriented stream cipher * 16–8
[13] Wu H The Stream Cipher HC-128
[14] Boesgaard M, Zenner E and Christensen T The Stream Cipher Rabbit