Hindawi Publishing Corporation
Journal of Sensors
Volume 2016, Article ID 8763528, 10 pages
http://dx.doi.org/10.1155/2016/8763528

Research Article

An Ultra-Lightweight Encryption Scheme in Underwater Acoustic Networks

Chunyan Peng,[1] Xiujuan Du,[1] Keqin Li,[2] and Meiju Li[1]

[1]*School of Computer Science, Qinghai Normal University, Xining, Qinghai 810008, China*
[2]*Department of Computer Science, State University of New York, New Paltz, NY 12561, USA*

Correspondence should be addressed to Xiujuan Du; 124111397@qq.com

Received 2 December 2015; Revised 14 February 2016; Accepted 18 February 2016

Academic Editor: Jian-Nong Cao

We tackle a fundamental security problem in underwater acoustic networks (UANs). The S-box in the existing block encryption algorithm is more energy consuming and unsuitable for resources-constrained UANs. In this paper, instead of S-box, we present a lightweight, 8-round iteration block cipher algorithm for UANs communication based on chaotic theory and increase the key space by changing the number of iteration round. We further propose secure network architecture of UANs. By analysis, our algorithm can resist brute-force searches and adversarial attacks. Simulation results show that, compared with traditional AES-128 and PRESENT algorithms, our cryptographic algorithm can make a good trade-off between security and overhead, has better energy efficiency, and applies to UANs.

1. Introduction

Recently, wireless sensor networks (WSNs) have emerged as an exceedingly powerful technique for a sea of applications, including control, monitoring, measurement, and surveillance [1–3]. Underwater acoustic networks (UANs) are a novel type of underwater network systems with the emphasis on effectively safeguarding the national marine rights and interests. UANs have been applied to many fields, for example, to monitor underwater environment, explore underwater resource, collect oceanic data, and prevent disaster.

Even though UANs share a slice of common properties with terrestrial sensor networks, such as a large number of nodes and limited power energy, UANs are significantly different from terrestrial sensor networks in a multitude of aspects: narrow bandwidth, long propagation delays, node passive mobility, and high error probability. In the following, we present the unique features of UANs and discuss the challenges in designing secure algorithm.

(A) Unique Features of UANs. A UAN is significantly different from any ground-based sensor network in terms of the following aspects.

(1) Acoustic Communication. Terrestrial wireless sensor networks use radio frequency (RF), laser, and radio waves to transmit data, but RF signal at a node's maximum transmission power is not able to spread more than 1 m in underwater environment [4, 5], and laser and radio waves cannot satisfy long distance communication in water either. We usually use acoustic to implement the communication of UANs [6–10] because the attenuation of acoustic communication is smaller than laser or radio, which can meet with the long distance transmission.

(2) Lower Bandwidth. RF communication propagates at $3 * 10^8$ m/s, but sound propagates at 1500 m/s in the underwater, which is lower than the speed of RF; for the bandwidth, underwater acoustic communication system is up to about 40 kbps for the existing modem product [11–13].

(3) Higher Bit-Error. Underwater acoustic communication channels are influenced by many factors such as path loss, noise, multipath, and Doppler spread. All of these factors give rise to higher bit-error. Moreover, sensors nodes are more vulnerable in harsh underwater environments. Compared with terrestrial sensor networks, underwater sensor networks have higher node-failure rate.

(4) Energy Efficiency. The same as in terrestrial sensor networks, saving energy is a major concern in UANs. Underwater sensor nodes are usually powered by batteries, which are even harder to recharge or replace in harsh underwater environments. UANs suffer from rigid resource constraints, such as limited battery life and computational power.

(5) More Vulnerability to Attacks. Acoustic communication has limited bandwidth due to long propagation delay and low data transfer rates. An underwater channel can be interrupted easily during transmission due to amplitude modulation and multipath occurrence. Additionally, the underwater acoustic channel is an open environment that makes UANs more vulnerable to jamming attacks or DoS attacks.

These characteristics of UANs make the existing work in terrestrial sensor networks unsuitable for UANs and bring about an army of challenges for its security. UANs also require security mechanisms and algorithms to maintain data confidentiality and integrity. Before sending data to the next layer, there is no doubt that the application layer needs to encrypt the payload for information security.

(B) Cryptographic Challenges in UANs. Security mechanisms are widely studied in terrestrial networks, and various defense mechanisms have been developed as safeguards. Due to the difference in communication mediums and physical environments, the existing security technology for terrestrial wireless sensor networks cannot be directly applied to UANs [14–16]. However, limited work has been performed on developing secure communication mechanisms and techniques to protect underwater networks so far.

These new features bring about many challenges to the design of UANs' cryptographic algorithm. Given the constrained energy, computation, and communication capabilities of UANs and the characteristics of the aqueous environments, secure communication techniques are required. In UANs, sensed information must be processed and managed safely. Before the application layer sending data to the next layer, we can encrypt the data and transmit the ciphertext to the sublayer. When we design a new encrypted mechanism, we should consider making use of a lightweight cipher for the mechanism. Our main contribution is to solve the security problems due to various means of attacks and security threats in the application layer, such as stealing, tampering, and other security issues. In this paper, we will discuss UAN security issues and describe a new ultra-lightweight block cipher which is fitter for UANs.

The paper provides the following contributions. First of all, we present a practical and efficient solution to encrypt the plaintext during UANs' communication. The proposed security scenario of UANs can protect end-to-end confidentiality and integrity and support both the one-to-one and one-to-many communication situations. Even though the topology of the whole network has been changed due to nodes mobility, joining, and leaving, it still allows secure reconfiguration. Second, our provided encryption algorithm is indeed valid by means of experiments and analysis with real data. The encryption algorithm introduces limited communication overhead and less energy consuming. To the best of our knowledge, this is the first secure encryption algorithm that has been implemented in UANs. The paper has proved that the security algorithm is suitable for an underwater acoustic networking environment by simulations.

The remainder of this paper is organized as follows. In Section 2, we consider underwater security field related works. In Section 3, we examine the structure of UANs and analyze the security scenario. Section 4 provides a new lightweight encryption algorithm for UANs. In Section 5, we analyze the security of our algorithm and contrast the performance of our algorithm with traditional algorithms in aspects of storage and computation cost. In Section 6, we conclude our paper and outline future works.

2. Related Works

Wireless sensor nodes are lower power devices which are highly constrained in terms of communication bandwidth and propagation delays. Battery life is sensor nodes' main limitation because they require considerable energy to maintain all kinds of communications. The mobile nature of sensor nodes in the aqueous environment also makes the acoustic transmission mechanisms less reliable and more energy-demanding. The security of UANs has been an increasing serious problem, but limited work has been conducted on studying security mechanisms in UANs. Research on UANs security continues to be still in its nascent stages owing to various restrictions. However, the necessity of security technology for UANs is raised rapidly in order to make the underwater communication more secure. In this section, we present a few related works in security-related technologies of UANs in the following paragraphs.

In [17], the authors focused on UANs security issues. They analyzed UANs and their characteristics. The application environments of UANs were studied, and the goals and challenges of UAN security were investigated. The performances of WSNs and ad hoc sensor networks (ASNs) were compared by the contributors. If some key nodes are damaged, the whole network cannot work normally. Therefore, the node's security remains essential. UANs cannot directly use the security protocols suitable for WSNs and ASNs [18–20]. Therefore, the authors provided a security protocol that can be suitable for UANs. Based on the above studies, security threats are classified according to their potential attacks on UANs. Therefore, the authors harbored the idea that corresponding countermeasures against those threats must be taken into consideration, but they did not provide any specific measures.

Cong et al. analyzed the threats and attacks on UANs security in [21]. Sensor nodes can be easily intercepted by an enemy and are at risk of information packet tampering. Owing to the characteristics of UANs and underwater channels, UANs are vulnerable to malicious attacks. A layered security system has a host of limits against a blended attack, and in order to overcome these limitations in UANs, the authors suggested that security mechanism is necessary and designed layered security structure. However, they did not

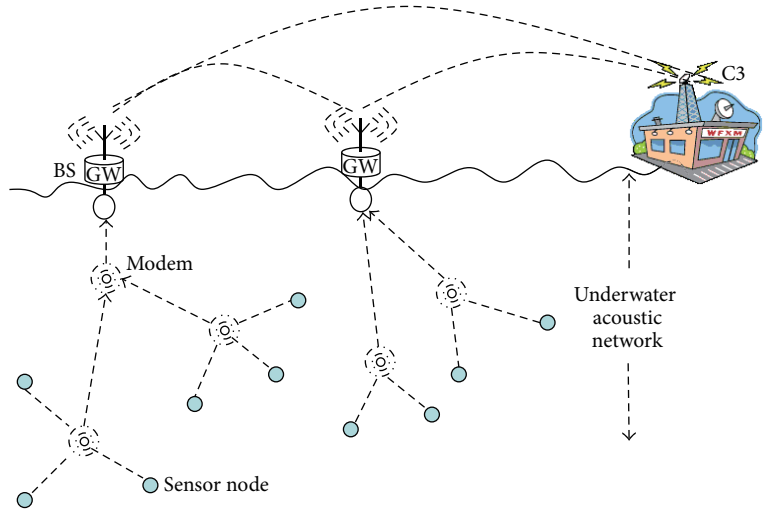

Figure 1: Underwater acoustic network structure.

address how to carry it out, nor did they provide any efficient algorithms.

Dini and Duca faced the problem of secure cooperation among underwater acoustic vehicles and then proposed a cryptographic suite able to reduce at the minimum message overhead in [22]. They presented a secure communication suite that can protect end-to-end communication. The cryptographic suite has provided vehicles authentication, confidentiality, and integrity of messages and key management. A prototype has been implemented and preliminary performance evaluation tests are shown. To avoid ciphertext size expansion, a ciphertext stealing (CTS) mode was used. The mode is used in a block cipher and handles plaintext without limiting its length and generates ciphertext with the same length as the plaintext.

In [23], the authors discussed applicable security algorithms that are suitable for UANs when considering a protocol stack and when sending data to its sublayer. They suggested that a transmitter and receiver should use the symmetric key for data encryption and decryption, which was also recommended in [24] by Wei et al. To avoid increasing in data, a hash algorithm and block encryption was performed in [23].

In [17, 21], the authors stated security issues and basic requirements, but encryption was not considered the foundation of security. However, published paper [22] considered the encryption scheme as a proposed concept, and the UANs project had proposed an encryption method that can be applied. In [23], the authors discussed an applicable security algorithm. However, they only explored security research trends and the security issues of UANs and mentioned a block cipher algorithm such as SEED or ARIR, but they have not discussed how to implement the algorithm. In this paper, we determine an appropriate cryptographic algorithm for basic UANs. Though the security of UANs has been an increasing serious problem, the majority of research on the security of UANs has still been in the theoretical study and simulation stage because the hardware and network technology has been not intensely mature.

In this paper, we present an ultra-lightweight encryption scheme to encrypt the contents of communication in the underwater networks. As far as we know, our provided encryption algorithm is the first algorithm that has been implemented in the application layer of UANs. It can guarantee the confidentiality and integrity from source node to destination node. The network has still been securely reconfigured if there is few nodes' mobility, joining, or leaving. Furthermore, the algorithm is really effective because we have proved by experiment with real data analysis that our encryption algorithm can be applied to the underwater environment. By the comparison and analysis of simulations, our scheme has better performance and it can save more energy and capacity.

3. Overview Securities of UANs

3.1. Network Architecture. We consider an underwater acoustic network with the aim of protecting all of the assets in underwater environments by means of a set of AquaSeNT OFDM (Orthogonal Frequency Divisions Multiplexing) modems. The network structure of UANs is illustrated in Figure 1.

The underwater network is composed of Base Station (BS), command and control center (C3), many AquaSeNT OFDM modems, and many different sensor nodes. The Base Station (BS) acts as a gateway between C3 and AquaSeNT modem. The BS may be a powerful underwater acoustic vehicle located under the sea surface. BS has a radio antenna with which it can communicate with the C3 by traditional radio frequency signals. The BS is located in the sea surface and it can be charged up from C3, because it commonly is connected to C3 through a cable. The C3 is a land station which acts as a command and control center for the physical defense of the asset. AquaSeNT OFDM modem [25] is

an acoustic modem that offers high data rate communications for underwater applications. Every modem is equipped with a number of sensors that are allowed to both sense the state of surrounding waters and detect the presence of targets in the neighborhood. The modem sends out data in packets. The modem will divide a packet into multiple data blocks, and the size of the block is 64 bits.

The hardware platform of AquaSeNT OFDM modem is constituted by transducers, an acoustic modem (Benthos Modem, OFDM modem), and a microcontroller. The software platform of AquaSeNT OFDM modem can run the embedded Linux operating system, network protocol stack, and applications. The speed of microcontroller is up to 600 MHZ and its memory includes 128 M RAM and 32 M flash. In this respect, compared with the land-based sensor networks, the acoustic modem has much more computational and storage resources because the underwater nodes are very expensive and not frequent to be replaced once they have been put to use. Nevertheless one of the most crucial constraints is the lower energy consuming because of the limited battery power, which is the same as land-based sensor nodes. Saving energy becomes more critical for AquaSeNT OFDM because solar and wind energy are not available. In our simulation, we suppose quite a few of modems are located in a limited underwater range and they are in a broadcast domain. Power consumption is necessarily considered in the design. Also, we have paid attention to the memory requirement and the computation complexity in the interest of saving energy.

3.2. Security Scenario of UANs. We organize modems and BS in a group G. Each modem can perform the following operations: (a) joining G; (b) sending messages to others in G; (c) receiving messages from others in G; (d) leaving G. In joining operation, an AquaSeNT modem can be registered in a group G. Once it has joined G, the AquaSeNT modem can send messages to another node and can also receive messages from the other members of the group G. The leaving operation allows an AquaSeNT modem to be unregistered in a group G. If the modem sends messages to C3, BS can relay messages between the modem and C3, which is one-to-one situation. In the one-to-many situation the modem can directly broadcast a message to the BS or other members in the same group in principle. However, because the BS is not resource constrained and the power required to send a message in one-to-many situation decays with a power of distance greater than in one-to-one scenario, the power required to reach all destinations would be so large that the modem would be exhausted very quickly. For the purpose of saving resources, when an AquaSeNT modem wants to broadcast a message, it will send the message to the BS which acts as a relay that can broadcast the message to all the members in a group so as to save some energy for AquaSeNT modem. In order to ensure that the communication is secure between underwater modem nodes, data encryption is the primary method to support confidentiality and integrity, but encryption in an underwater acoustic network is more challenging due to the severe limitations of the underwater environment. Therefore, the encryption algorithm must be lightweight and only use less memory and lower processor cost to perform the encryption algorithm. That is, we must make a well trade-off between the confidentiality and the overhead of algorithm, so we provide an ultra-lightweight encryption method that is fit for UANs.

BS has communicated with C3 usually by the wireless radio transmission, and it can be supplied with the energy continually, so BS can take the traditional methods to transmit the information [26, 27]. Every AquaSeNT modem will store the group ID Kg and its own ID Ks; BS will store the group ID Kg and all nodes' IDs Ks_i $(i = 0, 1, \ldots, n)$ in the same group (n is the number of nodes in the same group). Firstly, the new node will send its group ID Kg' to the next modem or BS. Upon obtaining Kg' in the next hop, the next node can compare the received Kg' to the key Kg stored in its memory. If they are the same then they will set up a link. The node will directly transmit encrypted data to the next hop in the same group, the middle node need not decrypt the data received from the original and continue sending it to BS, and, at last, the cipher will be decrypted in BS. Otherwise, if $Kg' \neq Kg$, then the middle node will drop the key and refuse to forward the data. Here, we do not provide more details concerning the cooperation methods of the nodes because we are interested in the encryption algorithm in this paper. The encrypted algorithm in underwater would satisfy the following characteristics: (1) adapting to the underwater transmission; (2) keeping the overhead of algorithm as small as possible; (3) relatively simple configuration and deployment to minimize the cost of energy; (4) ensuring the security. In order to meet these requirements, we employ the block cipher that uses the operations such as logistic map, XOR, and shifting. Our algorithm is a typical block cipher and iteration algorithm which is used in AES [28], PRESENT [29], and KLEIN [19]. However, our algorithm has revised the S-box so that it avoids the process of permutation and confusion. A general description of our algorithm encryption routine is described in Figure 2.

4. The Encryption Algorithm for UANs

In this section, the design principles of our algorithm will be addressed, and then we will discuss the detailed process of encryption and decryption.

4.1. Algorithm Principle

4.1.1. Logistic Map and the Generation of Chaos. Cryptography has certain unique mathematical requirements: diffusion, confusion, and dependence on keys. These properties are readily satisfied by chaotic functions by their sensitive dependence on initial conditions (function parameters), topological transitivity, and ergodicity [30]. Chaos theory is acted as a favorable, attractive option for cryptography. Therefore, chaos theory has been successfully applied to cryptography for a few years.

Logistic map is a discrete chaotic system, and the mapping relation is given by

$$x_{n+1} = \mu x_n \left(1 - x_n\right). \tag{1}$$

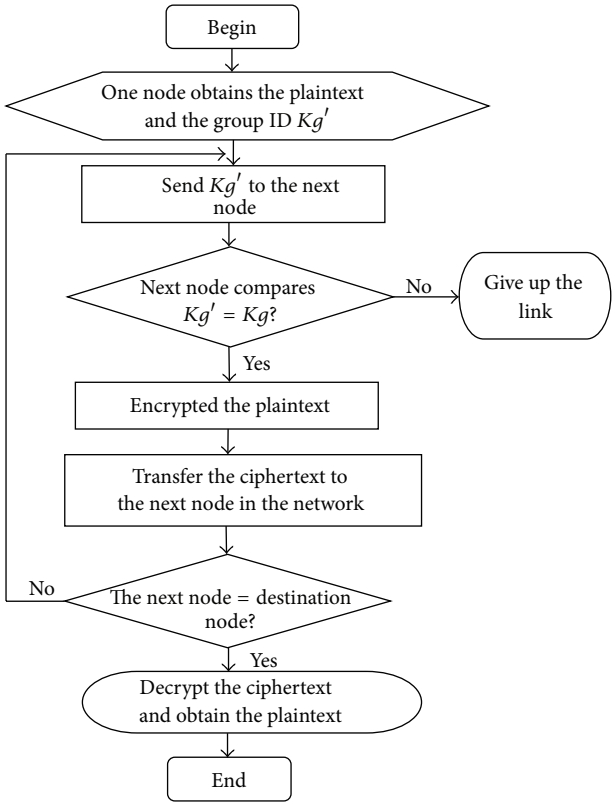

Figure 2: Our algorithm routine.

Here, μ is parameter; we can let $\mu \in (0, 4]$; when $x_0 \in (0, 1)$, no matter how many rounds of iteration there are, we will invariably get $x_n \in [0, 1]$. It was pointed out, in the study of chaotic dynamic system, that when $3.56994\cdots < \mu \leq 4$, the generated values by logistic map will present the state of pseudorandom distribution. To use x_n in the chaos orbit in encryption algorithm, we need to convert x_n into a number denoted by binary. We will address how to change it in Section 4.2.1.

4.1.2. Logic Operation. In this algorithm, in order to reduce the overhead of computation and storage of nodes, the basic operation will be carried out via the most simple logic operation: "bitwise exclusive-or (XOR)." Almost all of the micro processing systems including underwater sensor nodes have supported the operation XOR, so running the operation needs no more additional resources. The operation and reduction properties of "bitwise XOR" for n-bit codes are given by

$$\begin{aligned} b_i \oplus (a_i \oplus b_i) &= a_i \quad (i = 0, 1, \ldots, n-1), \\ a_i \oplus (a_i \oplus b_i) &= b_i \quad (i = 0, 1, \ldots, n-1). \end{aligned} \tag{2}$$

Here, a_i or b_i is one binary bit and the sum of the binary code bits is n. The symbol "$\oplus$" denotes the operation XOR.

4.2. The Process of Encryption. Now that high energy efficiency is the major objective for nodes of underwater sensor networks, the process of encryption cannot contain too complex operations. Underwater nodes are commonly sparsely deployed, and the plaintext data transmitted are mostly attribute information about underwater environments. Most of data are from sensor nodes to the BS, only little control information is from the BS to sensor nodes, and they are usually short. The packet needs to be encrypted and should not be very long either; otherwise its length will be substantially increased after being encrypted, which will increase energy consuming of nodes. The design of block cipher mainly includes two processes: substitution and confusion. In the existing block cryptographic algorithm, the S-box was used in the confusion process [31, 32]. However, the S-box requires more storage space and more complex permutation, so S-box is not suitable for UANs. In order to solve the problem, we adopt logistic map instead of S-box to finish the substitution operation of the cryptographic algorithm, which plays an excellent role of substitution and reduces the cost of encryption. Because the storage capacity of the existing underwater modem hardware is very limited and in order to get better balance between overhead and security, each data block takes 8 bytes; the number of encryption rounds is 8; the subkey of each round is 32 bits in our algorithm. We can increase the number of encryption round to expand the key space for the sake of improving the security.

4.2.1. Generating the Round Key. In order to obtain the binary of the key stream, we have to transform the sequence to the binary by analogue/digital (A/D) conversion. We extract every dot x on the chaos orbit which can be denoted by the binary format x_B. The decimal part of the chosen operation is shown in

$$\begin{aligned} x_B = 0.b_1(x)\, b_2(x)\, b_3(x) \cdots b_i(x) \cdots, \\ x \in [0, 1],\ b_i(x) \in \{0, 1\}. \end{aligned} \tag{3}$$

Here, in our encryption algorithm, if we denote the n-round value during the logistic operation by $\sigma^n(x) \in (0, 1)$, after A/D conversion, we can get a uniform distributed and independent pseudorandom sequence B_i^n which can be denoted by

$$B_i^n = \{b_i(\sigma^n(x))\}_{n=0}^{\infty}. \tag{4}$$

We can obtain the basic pseudorandom sequence by the method of Section 4.1.1. In order to achieve better avalanche effect, we will drop the values of logistic iteration before $N_0 = 200$ (here, N_0 is iteration times). In the process of encryption, the plaintext block sums up to 8 bytes, that is, 64 bits. After N_0 times of logistic iterations, the first binary bit is produced by the pseudorandom sequence $\beta = \sigma^{N_0}(x_0)$; then we continue to compute 37 numbers given by formula (4); we obtain $B_i^1 B_i^2 B_i^3 \cdots B_i^{37}$. We change the 32 numbers into two sequences $K_i = B_i^1 B_i^2 B_i^3 \cdots B_i^{32}$ and $A_i = B_i^{33} B_i^{34} \cdots B_i^{37}$ and then compute the decimal value D_i according to the binary of sequence A_i. D_i is the number of cyclic shift to the left.

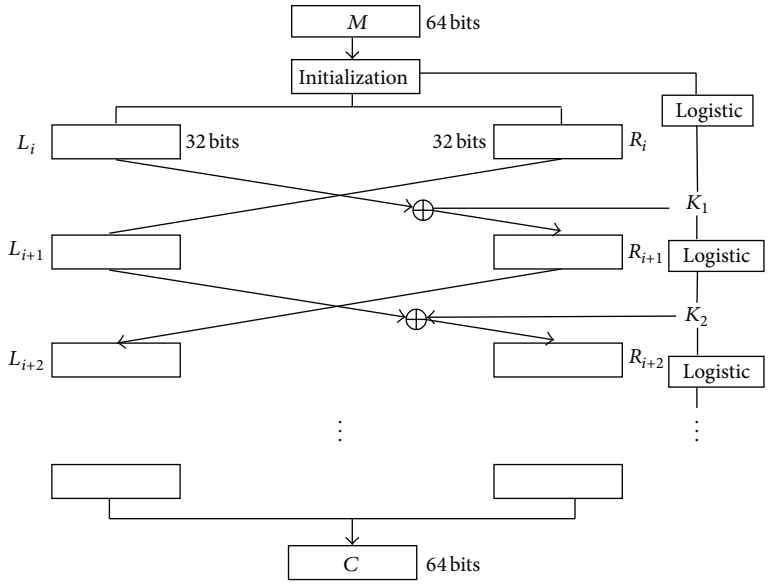

Figure 3: The process of iteration.

4.2.2. Encryption Work

(A) Initialization. At first, the sensor node divides the plaintext into small blocks and then performs the process of initialization. Suppose m is the plaintext which we need to encrypt, m is made of many blocks, the length of each block is 8 bytes, and each block can be denoted by m_i; the relation of m_i and m can be denoted by

$$m = \underbrace{p_0 p_1 p_2, \ldots, p_{l-1}}_{m_0}, \underbrace{p_l, p_{l+1}, \ldots, p_{2l-1}}_{m_1}, p_{2l}, \ldots, \quad (l = 8). \tag{5}$$

Here, the length of P_i is one byte. Furthermore, every block m_i is 8 bytes, which will be looked on as the combination of the left part L_i and the right part R_i; the length of L_i and R_i is 32 bits. L_{i-1} and R_{i-1} indicate the former left part and the former right part, respectively. The 8 rounds of the iteration are similar to Feistel structure, and in order to make the process of decryption easier, we do not exchange L_i and R_i in the last round. We can use (6) in each round of iteration:

$$\begin{aligned} L_i &= R_{i-1}, \\ R_i &= \left(L_{i-1} \lll D_i\right) \oplus K_i, \\ D_i &= D_i'. \end{aligned} \tag{6}$$

Here, the lengths of L_i, R_i, and K_i are all 32 bits, the symbol "$\lll$" denotes the cyclic shift to the left, and K_i is $B_i^1 B_i^2 B_i^3 \cdots B_i^{32}$, which is generated by A/D conversion. D_i is the number that will be shifted to the left; D_i' is the updated number that is shifted to the left in the next round. The entire process of 8-round iterations is shown in Figure 3.

(B) The Process of Producing D_i'. In order to improve the security of the algorithm, D_i need to be relative to the cipher or plaintext. In order to make the decryption operation simple and feasible, we use the cipher feedback to produce the next round key. After the operation of Section 4.2.1, the first communication round key K_i ($i = 1$) is established; the next key K_{i+1} will be produced by new logistic operation, but if the perimeter N_0 is still equal to 200, the produced pseudorandom sequence B_i^n is the same; in order to use different B_i^n for encryption, the value of N_0 is substituted by $N_0 + D_i'$ in the next round iteration. Furthermore, after encryption of each round, if the temporary R_i is denoted by c', we use the formula of $s = D_i \bmod 4$ to decide which bits of cipher will be taken into the operation of D_i' for resisting the chosen-plaintext attack. We can use

$$\begin{aligned} s &= D_i \bmod 4, \\ E_i &= c'_{s*8+0} \parallel c'_{s*8+1} \parallel c'_{s*8+2} \parallel c'_{s*8+3} \parallel c'_{s*8+4}, \\ D_i' &= A_i \oplus E_i \\ N_0 &= N_0 + D_i'. \end{aligned} \tag{7}$$

Here, c_i' is the ith bit of c' and D_i' is relative to the cipher, which is the number of shift in the next round iteration. A_i is binary format of D_i and A_i has five bits. In the decryption, we need use the value of D_i'; accordingly we push D_i' into the stack T.

(C) Combination of Parts of Ciphertext. After the 8-round iteration, we will produce two parts of ciphertext. We can generate the final cipher C after $C = Li \parallel Ri$ ($i = 8$) and the stack T and then transmit (C, T) to another node. A diagram of the encryption algorithm is given by Algorithm 1.

4.3. The Process of Decryption. The receiver will use the decryption algorithm to decipher the information when it received the data (C, T) from the sender. In the process of

Input: $x_0 \in (0,1)$, m, N_0, $\mu \in (3,4]$
Output: C, T
(1) $L_1, R_1 \leftarrow m$
(2) for $i \leftarrow 1$ to 8 do
(3) $x_i \leftarrow \text{logistic}(200 + N_0, \mu)$
(4) $B_i^j \leftarrow x_i$
(5) $K_i \leftarrow (B_i^1 B_i^2 \cdots B_i^{32})$
(6) $D_i \leftarrow (B_i^{33} B_i^{34} \cdots B_i^{37})$
(7) $L_{i+1} \leftarrow R_i$
(8) $R_{i+1} \leftarrow (L_i \lll D_i) \oplus K_i$
(9) $c' \leftarrow R_{i+1}$
(10) $s \leftarrow D_i \% 4$
(11) $D_i' \leftarrow (c'_{s*8+0} \| c'_{s*8+1} \| c'_{s*8+2} \| c'_{s*8+3} \| c'_{s*8+4}) \oplus D_i$
(12) $N_0 \leftarrow N_0 + D_i'$
(14) $T[i] \leftarrow D_i'$
(15) end for
(16) $C \leftarrow L_i \| R_i$
(17) return (C, T)

Algorithm 1: The encryption algorithm.

decryption, the ciphertext C is still a block which has 64 bits. The ciphertext still divided the data block into two parts: the left part and the right part. It is indispensable to pop five bits from the stack T successively which can decide the values of K_i and D_i, and then we can make use of inverse operation to carry out 8 rounds to reverse iterate.

5. Performance Analyses

5.1. Security Analyses. There are four attack models for any efficient adversary on attack encryption scheme: ciphertext-only attack, chosen-plaintext attack, chosen-ciphertext attack, and known-plaintext attack. If an adversary desires to take the measure that is called brute-force attack or exhaustive key search on ciphertext-only attack, it is guaranteed that the key space is wide enough. In this paper, the security algorithm we put forth uses the 64-bit length of each block and sums up to 8-round iteration. The round key K_i is 32 bits. In order to improve the security, a different key is adopted by chaos that is nonlinear operation; that is, the round key K_i is mutative in every round. Even though the current key is intercepted, the adversary cannot decrypt the message. We need to execute 8-round encrypted operation. If an adversary breaks one of K_i illegally, he cannot decrypt the plaintext either. The kind of combination of each round K_i is about 2^{32} and it can actually be broken in time 2^{32*8} if there are 8 rounds in encryption algorithm. We suppose that a powerful computer can search 10^{11} per second. Then we need at least $3.15 * 10^{47}$ years to finish the exhaustive search.

Secondly, it is necessary to obtain D_i that is relative to ciphertext, which can resist the chosen-plaintext attack from any adversary. Consider the following concrete example with our algorithm and suppose the plaintext "8A9BE27CFFB8E961, A2DBE2B4A6E2CD69" will be encrypted; we divide the plaintext into two blocks at first: one is "8A9BE27CFFB8E961" and the other is "A2DBE2B4A6E2CD69." Let $\mu = 3.87691$, $x_0 = 0.1234$, and $N_0 = 200$; the algorithm we provided would work as Tables 1 and 2.

Table 1: Encryption of "8A9BE27CFFB8E961."

Round i	D_i	K_i	c'	
1	19	5FAD7FFF	4C492B20	L_1
2	23	BC1B4010	0CE49C64	R_1
3	27	F6041F67	F466563E	L_2
4	28	8310AAAA	C3DEE36C	R_2
5	4	F4003B17	B26558F8	L_3
6	18	2E3A3680	A38939FB	R_3
7	21	BEFE7EEB	A1E83240	L_4
8	8	DCD88957	55E172F4	R_4

Table 2: Encryption of "A2DBE2B4A6E2CD69."

Round i	D_i	K_i	c'	
1	19	5FAD7FFF	4A086920	L_1
2	23	BC1B4010	08C83176	R_1
3	25	F6041F67	B6900FB5	L_2
4	11	AE1D3150	EF968116	R_2
5	9	82FED26C	A2E1B901	L_3
6	21	CCFF540B	EE22A6DB	R_3
7	17	F6D9DAAD	84DA9F6E	L_4
8	10	6F64BEE2	E5FFD15A	R_4

It is manifested from Tables 1 and 2 that D_i and K_i are entirely different in two tables from the third time even though all the parameters are the same. Consequently, an attacker is unable to obtain the subsequent key stream based on the ciphertexts that have been intercepted by the way of chosen-plaintext attack. We can control the produce of the key stream by plaintext-feedback or ciphertext-feedback. We have adopted the ciphertext-feedback in order to guarantee the synchronousness of encryption and decryption. During the process of encryption, the logistic map will produce the different iteration time D_i through the computation of (7), so, the key stream K_i is very different.

Thirdly, we change only one bit in the plaintext with the same parameters and run the encrypted algorithm, the last character "9" is rewritten into "A" in the plaintext, we can get Table 3 by 8 times encryption, the ciphertext is "1A1AD2FB D757019F," compared with the former ciphertext "84DA9F6EE5FFD15A," every character is different, and there is no identical character. That is, the algorithm is intensely sensitive to the plaintext. This is due to the fact that it is very possible to resist differential attacks.

Shannon pointed out that individuals can decrypt all kinds of ciphertext in the usage of statistical analysis. The statistical value of ciphertext can reflect the basic security level of algorithm. When we encrypt the pure text, although there are many identical characters in the plaintext, the text after being encrypted has no same character and no adversary can derive any meaningful information about the plaintext from the ciphertext as Table 4. When we encrypt "0000000011111111," the cipher is alternated into "412137D7576A104C," which is

Table 3: Encryption of "A2DBE2B4A6E2CD6A."

Round i	D_i	K_i	c'	
1	19	5FAD7FFF	4A086920	L_1
2	23	BC1B4010	09483176	R_1
3	25	F6041F67	B6900FB5	L_2
4	11	AE1D3150	EF96811A	R_2
5	8	82FED26C	12F167DA	L_3
6	10	B55CDE33	EF58B58D	R_3
7	28	BB35C486	1A1AD2FB	L_4
8	31	20FB5B59	D757019F	R_4

Table 4: Encryption of "0000000011111111."

Round i	D_i	K_i	c'	
1	19	5FAD7FFF	5FAD7FFF	L_1
2	12	BC1B4010	43E4BFEF	R_1
3	4	A829B639	52FE49CC	L_2
4	14	CC402773	E3BBF78A	R_2
5	16	67ADCAAF	2E619851	L_3
6	21	A33E74B8	526203C6	R_3
7	25	E37DF4E7	412137D7	L_4
8	29	9D265034	576A104C	R_4

Table 5: Security performance comparison.

	Key size (bit)	Block size (bit)	Round size
PRESENT	80	64	31
AES-128	128	128	10
Blowfish	32–448	64	16
Our algorithm	32	64	8

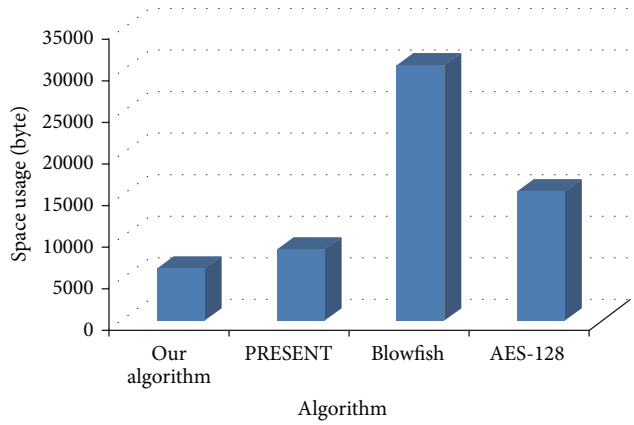

Figure 4: Storage usage.

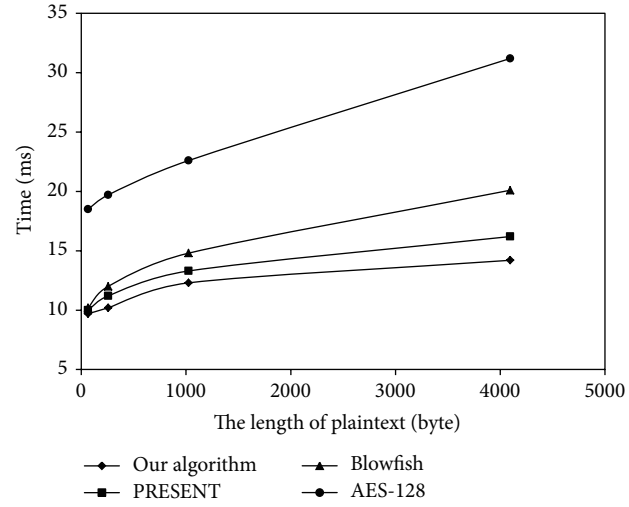

Figure 5: Performance comparison.

extremely different as Table 4. So an important issue to note is that an adversary cannot obtain any useful information of plaintext from the ciphertext.

In this paper we have described the new block cipher. Our goal is to present an ultra-lightweight cipher that offers a level of security commensurate with PRESENT and AES-128. It is a security performance comparison between our proposed scheme and other encrypted schemes as shown in Table 5.

5.2. Storage Cost. Our encryption algorithm has been based on implementation using C++ language (VC++ 6.0). All of the algorithms can settle on the use of a Pentium-4 3.4 GHz machine (running Microsoft Windows XP operation system) as the basis for experiments. The primary goal was to measure the encryption time of algorithms and decryption time is generally the same as encryption time for almost all the algorithms because they are symmetric ciphers; therefore only the encryption times were measured. In implementation of AES-128, the RAM needs 222 bytes and the ROM needs 12568 bytes [32]. Because the block size is 128 bits, the key length is 128 bits. Although PRESENT [29] is more suitable for extremely constrained environments than AES-128, it is still a SP-network and consists of 31 rounds. The block length of PRESENT is 64 bits and two key lengths of 80 and 128 bits are supported. The Blowfish algorithm [33] can encrypt the string with maximum length of 64 bits. Blowfish uses the unsigned longest p-box and S-box, which are needed to store in every node of networks in advance, which is two times more than AES-128. Figure 4 shows the comparison result by comparing the four algorithms in occupying space. Our algorithm obviously has superiority over AES-128, PRESENT, and Blowfish.

5.3. Performance Efficiency. The algorithm uses simple mathematical theories such as XOR, logistic map, and shift operation to be completed [34]. Because the computational algorithm is small, it can run faster. We make use of computer simulation to perform the new algorithm, compared with Blowfish, AES-128, and PRESENT. They are also block cipher of symmetric encryption algorithm. We run the three algorithms 30 times in the same length plaintext, respectively, in the same hardware and software environments. The comparison of total times expended by the four different algorithms is shown in Figure 5 (materials and figures

are available as Supplementary Material available online at http://dx.doi.org/10.1155/2016/8763528).

6. Conclusions

Interest in UANs is increasing, and related studies are also in progress. However, as stated previously, underwater environment is a special environment that has a multitude of restrictions. If we do not consider security requirements in UANs, data can be exposed or a malicious node can attack the system. We have presented the security scenario and an efficient encryption algorithm to protect the confidentiality and the integrity while taking into account the unique characteristic and constraints of the underwater networking environment. Simulations and encryption experiments have shown that the block cryptographic algorithm in application layer introduces limited overhead, and thus it is perfectly adequate for the underwater acoustic communication. So far, we can increase the key space by changing the number of iteration round. For example, when the iteration round is 10, the key space of 2^{10*32} can be expanded 2^{2*32} times of 2^{8*32}. There is no possibility of cracking the round key K_i by means of brute-force attack. The explored algorithm has two characteristics of chaos nonlinear and Feistel structure. It was demonstrated that the algorithm can meet security requirements by security analysis. We completed implementing successfully the encryption and decryption algorithm by programming, which indicated that the proposed mechanism can achieve confidential communication. Future work will continue implementing and evaluating the block symmetric algorithm in real underwater sensor nodes.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work is supported by the National Natural Science Foundation Projects of China (no. 61162003), Qinghai Office of Science and Technology (2015-ZJ-904, 2012-Z-902), the Ministry of Education Chunhui Projects (no. Z2015052), the National Social Science Foundation of China (no. 15XMZ057), and Qinghai Social Science Foundation (no. 2015-ZJ-718).

References

[1] E. H. Callaway, *Wireless Sensor Networks, Architectures and Protocols*, Auerbach Publications, Taylor & Francis, Boca Raton, Fla, USA, 2003.

[2] Q. Liang and X. Cheng, "Underwater acoustic sensor networks: target size detection and performance analysis," *Ad Hoc Networks*, vol. 7, no. 4, pp. 803–808, 2009.

[3] Z.-J. Hu, C.-M. Wang, Y.-P. Zhu et al., "Signal detection for the underwater acoustic voice communication," in *Proceedings of the International Symposium on Test and Measurement*, pp. 1–5, IEEE, Washington, DC, USA, 2003.

[4] J. Preisig, "Acoustic propagation considerations for underwater acoustic communications network development," in *Proceedings of the 1st ACM International Workshop on Underwater Networks (WUWNet '06)*, vol. 11, pp. 1–5, September 2006.

[5] D. B. Kilfoyle and A. B. Baggeroer, "The state of the art in underwater acoustic telemetry," *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 4–27, 2000.

[6] G. Zhongwen, L. Hanjiang, H. Feng, Y. Meng, and L. M. Ni, "Current progress and research issues in underwater sensor networks," *Journal of Computer Research and Development*, vol. 47, no. 3, pp. 377–389, 2010.

[7] A. Caiti and A. Munafo, "Adaptive cooperative algorithms for AUV networks," in *Proceedings of the IEEE International Conference on Communications Workshops (ICC '10)*, pp. 1–5, Capetown, South Africa, May 2010.

[8] I. F. Akyildiz, D. Pompili, and T. Melodia, "State-of-the-art in protocol research for underwater acoustic sensor networks," in *Proceedings of the 1st ACM International Workshop on Underwater Networks (WUWNe '06)*, pp. 7–16, ACM, New York, NY, USA, September 2006.

[9] J. Rice and D. Green, "Underwater acoustic communications and networks for the US Navy's seaweb program," in *Proceedings of the 2nd International Conference on Sensor Technologies and Applications (SENSORCOMM '08)*, pp. 715–722, IEEE, Cap Esterel, France, August 2008.

[10] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "The challenges of building scalable mobile underwater wireless sensor networks for aquatic applications," *IEEE Network*, vol. 20, no. 3, pp. 12–18, 2006.

[11] P. Xie, Z. Zhou, Z. Peng, J.-H. Cui, and Z. Shi, "SDRT: a reliable data transport protocol for underwater sensor networks," *Ad Hoc Networks*, vol. 8, no. 7, pp. 708–722, 2010.

[12] Y. Su, Y. Zhu, H. Mo et al., "UPC-MAC: a power control mac protocol for underwater sensor networks," in *Wireless Algorithms, Systems, and Applications*, pp. 377–390, Springer, Berlin, Germany, 2013.

[13] M. Stojanovic, *Underwater Acoustic Communication: Design Considerations on the Physical Layer*, Wireless on Demand Network Systems and Services, 2008.

[14] S. Y. Shin, J. I. Namgung, and S. H. Park, "SBMAC: smart blocking MAC mechanism for variable UW-ASN (underwater acoustic sensor network) environment," *Sensors*, vol. 10, no. 1, pp. 501–525, 2010.

[15] A. Paul, D. B. Chen, and W. Wang, "Bioinspired mechanisms in wireless Ad Hoc and sensor networks," *Journal of Sensors*, vol. 2015, Article ID 813585, 2 pages, 2015.

[16] N.-Y. Yun, Y.-P. Kim, S. Muminov, J.-Y. Lee, S.-Y. Shin, and S.-H. Park, "Sync MAC protocol to control underwater vehicle based on underwater acoustic communication," in *Proceedings of the IEEE/IFIP 9th International Conference on Embedded and Ubiquitous Computing (EUC '11)*, pp. 452–456, Melbourne, Australia, October 2011.

[17] Y. Dong and P. Liu, "Security considerations of underwater acoustic networks," in *Proceedings of the 20th International Congress on Acoustics (ICA '10)*, pp. 274–277, Sydney, Australia, August 2010.

[18] C. Parr, A. Poschmann, and M. J. B. Robshaw, "New designs in lightweight symmetric encryption," in *RFID Security: Techniques, Protocols and System-on-Chip Design*, P. Kitsos and Y. Zhang, Eds., pp. 349–371, Springer, Heidelberg, Germany, 2008.

[19] Z. Gong, S. Nikova, and Y. W. Law, "KLEIN: a new family of lightweight block ciphers," in *RFID. Security and Privacy: 7th*

International Workshop, RFIDSec 2011, Amherst, USA, June 26–28, 2011, Revised Selected Papers, A. Juels and C. Paar, Eds., vol. 7055 of *Lecture Notes in Computer Science*, pp. 1–18, Springer, Berlin, Germany, 2012.

[20] G. Dini and I. M. Savino, "S2RP: a secure and scalable rekeying protocol for wireless sensor networks," in *Proceedings of the IEEE International Conference on Mobile Ad Hoc and Sensor Sysetems (MASS '06)*, pp. 457–466, Vancouver, Canada, October 2006.

[21] Y. Cong, G. Yang, Z. Wei, and W. Zhou, "Security in underwater sensor network," in *Proceedings of the International Conference on Communications and Mobile Computing (CMC '10)*, pp. 162–168, Shenzhen, China, April 2010.

[22] G. Dini and A. L. Duca, "A cryptographic suite for underwater cooperative applications," in *proceedings of the 16th IEEE Symposium on Computers and Communications (ISCC '11)*, pp. 870–875, Kerkyra, Greece, July 2011.

[23] J. E. Kim, N. Y. Yun, and M. Sardorbek, "Security in underwater acoustic sensor network: focus on suitable encryption mechanisms," in *AsiaSim 2012: Asia Simulation Conference 2012, Shanghai, China, October 27–30, 2012. Proceedings, Part II*, pp. 160–168, Springer, Berlin, Germany, 2012.

[24] Z. Wei, Y. Guang, C. Yanping, and D. Jiajia, "Analysis of security and threat of underwater wireless sensor network topology," in *Proceedings of the International Conference on E-Business and E-Government (ICCEE '10)*, pp. 506–510, Chengdu, China, May 2010.

[25] AquaSeNT OFDM Modem User Manual, http://www.aquasent.com/.

[26] C. Karlof, N. Sastry, and D. Wagner, "TinySec: a link layer security architecture for wireless sensor networks," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 162–175, Baltimore, Md, USA, November 2004.

[27] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Communications of the ACM*, vol. 47, no. 6, pp. 53–57, 2004.

[28] D. A. Osvik, J. W. Bos, D. Stefan, and D. Canright, "Fast software AES encryption," in *Fast Software Encryption: 17th International Workshop, FSE 2010, Seoul, Korea, February 7–10, 2010, Revised Selected Papers*, S. Hong and T. Iwata, Eds., vol. 6147 of *Lecture Notes in Computer Science*, pp. 75–93, Springer, Berlin, Germany, 2010.

[29] A. Bogdanov, L. R. Knudsen, G. Leander et al., "PRESENT: an ultra-lightweight block cipher," in *Cryptographic Hardware and Embedded Systems—CHES 2007*, P. Paillier and I. Verbauwhede, Eds., vol. 4727 of *Lecture Notes in Computer Science*, pp. 450–466, Springer, Heidelberg, Germany, 2007.

[30] G. Alvarez and S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems," *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, vol. 16, no. 8, pp. 2129–2151, 2006.

[31] Y. He and S. Tian, "Block encryption algorithm based on chaotic S-box for wireless sensor network," *Journal of Computer Applications*, vol. 33, no. 4, pp. 1081–1084, 2013.

[32] J. Daemen and V. Rijmen, *The Design of Rijndael: AES-The Advanced Encryption Standard*, Springer, Berlin, Germany, 2002.

[33] Q. C. Zhong and Q.-X. Zhu, "Analysis of Blowfish cryptography," *Journal of Computer Applications*, vol. 27, no. 12, pp. 2939–2941, 2007.

[34] X.-F. Zhang and C.-C. Yin, "Energy harvesting and information transmission protocol in sensors networks," *Journal of Sensors*, vol. 2016, Article ID 9364716, 5 pages, 2016.