Information Sciences 284 (2014) 157–166

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A scheme for data confidentiality in Cloud-assisted Wireless Body Area Networks

Nguyen Dinh Han [a,d], Longzhe Han [b], Dao Minh Tuan [a], Hoh Peter In [c], Minho Jo [d,*]

[a] *Hung Yen University of Technology and Education, Viet Nam*
[b] *School of Information Engineering, Nanchang Institute of Technology, China*
[c] *College of Information and Communications, Korea University, Seoul, Republic of Korea*
[d] *Department of Computer and Information Science, Korea University, Sejong City, Republic of Korea*

ARTICLE INFO

Article history:
Received 7 August 2013
Received in revised form 25 March 2014
Accepted 29 March 2014
Available online 13 April 2014

Keywords:
Wireless Body Area Networks
Cloud
Security
Data confidentiality
Encryption
Unambiguous language

ABSTRACT

The integration of Wireless Body Area Networks with a cloud computing platform creates a new digital ecosystem with advanced features called Cloud-assisted Wireless Body Area Networks. This ecosystem enables users to globally access e-healthcare services at competitive costs. However, the secure data communications between the cloud and Wireless Body Area Networks are critical because the data is related to users' privacy information. In this paper, we propose the Multi-valued and Ambiguous Scheme to capture data confidentiality in the Cloud-assisted Wireless Body Area Networks since it is the most important issue. The approach combining the scheme with existing encryption schemes provides a general paradigm for deploying applications. The obtained results show that secure data communications between the cloud and Wireless Body Area Networks can be achieved.

1. Introduction

The use of Wireless Body Area Networks (WBANs) is greatly improving healthcare quality nowadays. WBANs have attracted considerable attention because they have a wide range of applications from ubiquitous health monitoring and computer assisted rehabilitation to emergency medical response systems. Other potential applications include interactive gaming, social computing, entertainment, and the military. However, the challenges coming from stringent resource constraints of WBAN devices, and the high demand for both security/privacy and practicality/usability may prevent those applications from being widely deployed [14,22,20]. In reality, security and privacy protection of the data collected by a WBAN, either while stored inside the WBAN or during its transmission out of the WBAN, is a major unsolved problem [11]. A possible way to solve this problem is to exploit the benefits of cloud computing [1,8]. However, the cloud also has its own set of security problems [12,21,10], in that the owner of the data may not have control of where the data is placed [6]. Again, WBANs can in turn help to mitigate security problems with the cloud.

Indeed, the integration of WBANs with cloud computing will create a new system named Cloud-assisted WBANs. This new system provides a cloud computing environment that links different devices from miniaturized sensor nodes to high-performance supercomputers that process the huge amount of data collected from multiple WBANs. Since the challenges of resource constraints of WBAN devices are not a major concern when coupled with cloud computing resources,

[*] Corresponding author. Tel.: +82 44 860 1348.
E-mail address: minhojo@korea.ac.kr (M. Jo).

http://dx.doi.org/10.1016/j.ins.2014.03.126
0020-0255/© 2014 Elsevier Inc. All rights reserved.

WBAN applications can be deployed on Cloud-assisted WBANs at competitive costs. The system also has a feature that enables its users and applications to access its data from anywhere in the world. Therefore, the security and privacy of the data must be protected in the framework of this new system.

In this paper, we propose a Multi-valued and Ambiguous Scheme (MAS) to overcome existing shortcomings in Cloud-assisted WBANs. Our scheme mainly deals with data confidentiality and provides a general paradigm for deploying applications in Cloud-assisted WBANs. The rest of the paper is organized as follows. In Section 2, we recall some basic notions of languages and codes. Concepts regarding cryptosystems and unambiguous languages are also mentioned. Section 3 consists of four subsections. In the first subsection, we give a new method to design cryptosystems as the standard approach to protect data. In this new method, users are able to encode data with a secret key that can only be decoded by the intended receivers. The obtained cryptosystems possess interesting properties such as allow the use of unambiguous languages that are generally not codes. Also, the cryptosystems contain a trapdoor which can be reduced to an undecidable problem. The second subsection presents our proposed scheme for data confidentiality that is suitable for use in Cloud-assisted WBANs. The third subsection is devoted to analyzing security issues concerning our scheme. The remaining subsection is for application of our scheme in Cloud-assisted WBANs. In Section 4, we give our simulation results and discuss them. The final section concludes our work.

2. Notations and basic definitions

We first recall some necessary notions (for more details, we refer to [2]). Let A be a finite alphabet. As usual, A^* is the free monoid of all finite words over A. The empty word is denoted by ε and $A^+ = A^* - \{\varepsilon\}$. The length of the word $w = a_1a_2\cdots a_n$ with $a_i \in A$ is $|w| = n$, $|\varepsilon| = 0$. $A^{\leqslant n} = \{w \in A^* | |w| \leqslant n\}$. A *factorization of a word* $w \in A^*$ *on* X, where $X \subseteq A^*$, is given by the equation $w = u_1u_2\cdots u_n$ where $u_1, u_2, \ldots, u_n \in X$, $n \geqslant 1$. A subset of A^* is called a *language*. A language $X \subseteq A^+$ is a *code* if every word w in A^* has at most one factorization on X. We denote by X^* the submonoid generated by X and $X^* = X^+ \cup \{\varepsilon\}$.

As a general reference for cryptosystems we mention [18], and for the facts concerning the unambiguous languages we refer to [7]. We need also two basic definitions:

Definition 1. *A cryptosystem is a five-tuple* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, *where the following conditions are satisfied*:

1. $\mathcal{P}$ *is a finite set of possible original* (*plain*) *words*
2. $\mathcal{C}$ *is a finite set of possible encoded words*
3. $\mathcal{K}$ *is a finite set of possible keys*
4. *For each* $K \in \mathcal{K}$, *there is an encoding rule* $e_K \in \mathcal{E}$ *and a corresponding decoding rule* $d_K \in \mathcal{D}$. *Each* $e_K : \mathcal{P} \rightarrow \mathcal{C}$ *and* $d_K : \mathcal{C} \rightarrow \mathcal{P}$ *are functions such that* $d_K(e_K(x)) = x$ *for every* $x \in \mathcal{P}$.

Definition 2. *Consider a language* $X \subseteq A^+$ *and a natural number* $k \geqslant 0$. *Then,*

(*i*) *The set X is said to be k-unambiguous if it satisfies the condition: for all* $k \geqslant m \geqslant 1$ *and for all* $x_1, x_2, \ldots, x_k, y_1, y_2, \ldots, y_m \in X$, *if* $x_1x_2\cdots x_k = y_1y_2\cdots y_m$, *then* $k = m$ *and* $x_i = y_i$ *with* $i = 1, \ldots, k$.
In the converse case, if X does not satisfy the above condition, then X is said to be k-ambiguous.

(*ii*) *If there exists the biggest integer k such that X is k-unambiguous, then k is called the unambiguous degree of X. If such an integer does not exist, then X is said to have the unambiguous degree of* ∞.

3. A scheme for data confidentiality in Cloud-assisted WBANs

3.1. Design of multi-valued and ambiguous cryptosystems

In this section, we present a new method to design cryptosystems that have multi-valued and ambiguous properties. The multi-valued property of the cryptosystems can be established using *multi-valued encoding rules*. We will introduce a key technique to apply these encoding rules to unambiguous languages. This technique equips the cryptosystems with the ambiguous property. Consequently, the obtained cryptosystems may avoid any attacks that consider codes as the target, or the area selected for attacks must be on a large scale.

At first, we define the notion of multi-valued morphisms that allow us to establish dedicated multi-valued encoding rules.

Definition 3.

(*i*) *A multi-valued morphism is an injective function* f: $A^* \rightarrow B^*$, *that associates each letter* $a \in A$ *with a subset* X_a *of* B^* *and* $f(a_1a_2\cdots a_n) = f(a_1)f(a_2)\cdots f(a_n)$ *for every* $a_1, a_2, \ldots, a_n \in A$.

(*ii*) *The multi-valued morphism f is called a multi-valued encoding rule if for all* $w, w' \in A^*$, $w \neq w'$ *we have* $f(w) \cap f(w') = \emptyset$.

(*iii*) *The multi-valued morphism f is called a restricted multi-valued encoding rule if there is an integer* $k > 0$ *and for all* $w, w' \in A^{\leqslant k}$, $w \neq w'$ *we have* $f(w) \cap f(w') = \emptyset$.

Remark 1. The encoding procedure deduced from multi-valued encoding rules consists of associating to a word in A^* some encoded words in B^*. However, the fact that f is injective ensures that the encoded words are uniquely decipherable, in order to get the original words back.

The following theoretical results provide necessary and sufficient conditions for a given multi-valued morphism to be a multi-valued encoding rule or a restricted multi-valued encoding rule.

Proposition 1. *Let A and B be finite alphabets. Let* $f: A^* \to B^*$ *be a multi-valued morphism, which associates each letter* $a \in A$ *with a subset* X_a *of* B^**, and an integer* $k > 0$*. Then, for all* $f(a_1), f(a_2), \ldots, f(a_p), f(b_1), f(b_2), \ldots, f(b_q) \subseteq B^*$ *with* $a_i, b_j \in A$, $i = 1, \ldots, p$, $j = 1, \ldots, q$, *we have*

(*i*) *f is a multi-valued encoding rule if and only if the condition* $f(a_1)f(a_2)\cdots f(a_p) \cap f(b_1)f(b_2)\cdots f(b_q) \neq \emptyset$ *implies* $p = q$ *and* $f(a_i) = f(b_i)$ *for* $i = 1, \ldots, p$.

(*ii*) *f is a restricted multi-valued encoding rule if and only if the condition* $f(a_1)f(a_2)\cdots f(a_p) \cap f(b_1)f(b_2)\cdots f(b_q) \neq \emptyset$ *with* $p, q \leqslant k$ *implies* $p = q$ *and* $f(a_i) = f(b_i)$ *for* $i = 1, \ldots, p$.

Proof.

(*i*) ($\Rightarrow$). We assume by a contradiction that there exists $f(a_1)f(a_2)\cdots f(a_p) \cap f(b_1)f(b_2)\cdots f(b_q) \neq \emptyset$ with $p \neq q$ or $f(a_i) \neq f(b_i)$ for some i. Then, we have $f(a_1a_2\cdots a_p) \cap f(b_1b_2\cdots b_q) \neq \emptyset$ with $p \neq q$ or $f(a_i) \neq f(b_i)$ for some i. Now $p \neq q$ and $f(a_i) \neq f(b_i)$ both imply that $a_1a_2\cdots a_p \neq b_1b_2\cdots b_q$. We deduce that there exist two different words $a_1a_2\cdots a_p$ and $b_1b_2\cdots b_q$ in A^* such that $f(a_1a_2\cdots a_p) \cap f(b_1b_2\cdots b_q) \neq \emptyset$. This contradicts the assumption.
($\Leftarrow$). We assume by the contradiction that f is not a multi-valued encoding rule. Then, there exist two different words $a_1a_2\cdots a_p, b_1b_2\cdots b_q \in A^*$ with $a_i, b_j \in A$, $p \neq q$ or $a_i \neq b_i$ such that $f(a_1a_2\cdots a_p) \cap f(b_1b_2\cdots b_q) \neq \emptyset$, or equivalently, $f(a_1)f(a_2)\cdots f(a_p) \cap f(b_1)f(b_2)\cdots f(b_q) \neq \emptyset$. Now $a_i \neq b_i$ implies that $f(a_i) \neq f(b_i)$. Therefore, we have $f(a_1)f(a_2)\cdots f(a_p) \cap f(b_1)f(b_2)\cdots f(b_q) \neq \emptyset$ with $p \neq q$ or $f(a_i) \neq f(b_i)$ for some i, $i = 1, \ldots, p$. This contradicts the assumption.

(*ii*) The proof of (*ii*) is similar to the proof of (*i*). □

Now, to enhance the cryptosystem with the ambiguous property, we use the technique described in the following corollary.

Corollary 1. *Let* $A = \{a_1, a_2, \ldots, a_n\}$ *and let* $k > 0$ *be a positive integer. Consider a language X that has the unambiguous degree k such that X can be partitioned into n subsets* $X_1, X_2, \ldots, X_n$, $X_i \cap X_j = \emptyset$, $\forall i \neq j$, $X_1 \cup X_2 \cup \cdots \cup X_n = X$. *Suppose that* $g: A^* \to X^*$ *is a multi-valued morphism that maps each* $a_i \in A$ *to a subset* X_i *and* $g(ww') = g(w)g(w')$ *for all* $w, w' \in A^{\leqslant k}$. *Then, g is a restricted multi-valued encoding rule.*

Proof. We assume by a contradiction that g is not a restricted multi-valued encoding rule. Then, there exist $w, w' \in A^{\leqslant k}$, $w \neq w'$ such that $g(w) \cap g(w') \neq \emptyset$. This implies that there exists an equation $x_1x_2\cdots x_i = y_1y_2\cdots y_j$ with $x_1 \neq y_1$, $x_i, y_j \in X$, $1 \leqslant i, j \leqslant k$. By definition, X does not have the unambiguous degree k. This contradicts the assumption. Hence, g is a restricted multi-valued encoding rule. □

Remark 2. Corollary 1 obviously provides a sufficient condition to construct multi-valued and ambiguous cryptosystems. Consider a cryptosystem of this sort. We will show that its multi-valued and ambiguous properties are given by g and X respectively. Notice that the encoding procedure using g can encode any word in A^*, obtaining some encoded words (see Remark 1). However, for each encoded word, the decoding procedure gives the unique result only for the case where the length of the original word produced it is less than or equal to k. This fact is due to the ambiguous property of X (i.e. any word of length greater than k in X^* may have more than one factorization on X). Therefore, g and X are considered as secret keys in such a cryptosystem.

Example 1. Let $A = \{u_1, u_2, u_3, u_4, u_5\}$ and consider $X = \{c, ca_1, a_1b_1, b_1a_2, a_2b_2, b_2a_3, a_3b_3, b_3\}$ that has the unambiguous degree $k = 3$. One of the partitions of X is: $X_1 = \{c, a_1b_1\}$, $X_2 = \{ca_1\}$, $X_3 = \{b_1a_2, a_2b_2\}$, $X_4 = \{b_3\}$, $X_5 = \{b_2a_3, a_3b_3\}$, and g is defined by: $g(u_i) \in X_i$, $i = 1, \ldots, 5$. Suppose that the original word is $w = u_2u_3u_5u_4$. Since its length is 4, we divide it into two words $w_1 = u_2u_3u_5$ and $w_2 = u_4$ to guarantee that their length is less than or equal to k.

For g defined above, the encoded words can be: $ca_1b_1a_2b_2a_3$ and b_3, or $ca_1a_2b_2b_2a_3$ and b_3, or $ca_1b_1a_2a_3b_3$ and b_3, or $ca_1a_2b_2a_3b_3$ and b_3.

Decoding the encoded word $ca_1b_1a_2b_2a_3$, we gain three words in X, which are $ca_1 \in X_2$, $b_1a_2 \in X_3$, and $b_2a_3 \in X_5$. Thus, the corresponding original word is $u_2u_3u_5$. The word u_4 is decoded from b_3. Consequently, we have the original word w. Other encoded words can be decoded in the same manner and they all give the original word w.

The ambiguity can happen when we encode a word whose length is greater than k. For instance, in the case where we encode the word $w = u_2u_3u_5u_4$, for g defined above, a possible encoded word is $ca_1b_1a_2b_2a_3b_3$. Then, decoding gives two results: $(c)(a_1b_1)(a_2b_2)(a_3b_3)$ with $c \in X_1$, $a_1b_1 \in X_1$, $a_2b_2 \in X_3$, $a_3b_3 \in X_5$, or $(ca_1)(b_1a_2)(b_2a_3)(b_3)$ with $ca_1 \in X_2$, $b_1a_2 \in X_3$, $b_2a_3 \in X_5$, $b_3 \in X_4$. The corresponding original words are $u_1u_1u_3u_5$ and $u_2u_3u_5u_4$.

3.2. The proposed scheme

By the method introduced in the previous section, we propose a cryptosystem that, in turn, allows us to establish a scheme for data confidentiality as the main result of this section.

Let $A = \{a_1, a_2, \ldots, a_n\}$ be a finite alphabet. Consider a language X which has the unambiguous degree k, $k > 0$ such that X can be partitioned into n subsets $X_1, X_2, \ldots, X_n$, $X_i \cap X_j = \emptyset$, $\forall i \neq j$, $X_1 \cup X_2 \cup \cdots \cup X_n = X$. We denote by X_P the set of possible partitions of X. Then, by Remark 2 we can formulate our cryptosystem as follows.

Schema 1. *A multi-valued and ambiguous cryptosystem*

Let $\mathcal{P} = A^{\leqslant k}, \mathcal{C} = X^*$. $\mathcal{K}$ consists of all injective multi-valued functions g: $A \to X_P = \{X_1, X_2, \ldots, X_n\}$. For each $g \in \mathcal{K}$, define:

$$e_g(x) = w \in g(x),$$

and define

$$d_g(w) = \{y | w \in g(y)\}.$$

Note 1. For any $w \in X^*$, if $w \in g(x)$ then w is considered to be an encoded word of x. Thus, the number of encoded words of any original word can be very large. However, defining the language X with the unambiguous degree k, where k is large enough, depends on firm foundations. By Remark 2, together with g, the language X must be kept secret. Although k is used for decoding, the decoding delay needs to be considered since it impacts on the performance of the cryptosystem.

Let m be some fixed positive integer and let S be a secret bitstring of length m. Consider a secret unambiguous language $X \subseteq \{0,1\}^*$, which has the unambiguous degree k, satisfying the condition: for all $x_1, x_2, \ldots, x_k \in X$, we have $|x_1| + |x_2| + \cdots + |x_k| \leqslant m$. With this X and A defined above, we can define e_g and d_g as in Schema 1. Now, we can describe our scheme for data confidentiality. The scheme consists of two procedures, ENCODE and DECODE, that are presented below.

The procedure ENCODE encodes a word $u \in A^*$, $u = u_1u_2 \cdots u_n$, $u_i \in A$, obtaining m-bit blocks of encoded words. Concretely, we use a while loop to scan the word u from left to right. Then, each m-bit block of the encoded words can be produced using the nested while loop. Indeed, the condition ($count \leqslant k$) and ($|w_j| < m$) guarantees that the length of the word used to produce the block w_j is less than or equal to k, and w_j does not exceed m bits. Depending on the encoding situation, the PAD (w_j) is called to pad w_j in order to gain the m-bit block. Next, the exclusive-or ($\oplus$) of two bitstrings is used to create masks on m-bit blocks constituting the output.

```
procedure ENCODE ($u$)
  $i = 1, j = 10$;
  while $i \leqslant n$ do
    $count = 1$;
    while ($count \leqslant k$) and ($|w_j| < m$) do
      if $|w_j e_g(u_i)| \leqslant m$ then
        $w_j = w_j e_g(u_i)$, $count = count + 1$, $i = i + 1$
      else PAD ($w_j$);
    if $|w_j| < m$ then PAD ($w_j$);
    if $j == 1$ then $w'_j = w_j \oplus S$ else $w'_j = w_{j-1} \oplus w_j$;
    $j = j + 1$;
  return $w = w'_1 w'_2 \ldots w'_{j-1}$
```

The DECODE procedure takes an encoded word w of q m-bit blocks as input, $w = w'_1 w'_2 \ldots w'_q$, $|w'_j| = m$, and produces the original word $u \in A^*$ as output. At first, the m-bit secret key S is used to remove the masks of input blocks. Then, each block is decoded separately. The EXTRACT (w_j, tmp) extracts words in X from w_j, then stores them in the array tmp. Then, the corresponding original words can be obtained from tmp using d_g.

```
procedure DECODE($w$)
  $i = 1, j = 1$;
  while $j \leqslant q$ do
    if $j == 1$ then $w_j = w'_j \oplus S$ else $w_j = w_{j-1} \oplus w'_j$;
    EXTRACT ($w_j$, $tmp$);
    $count = 1$;
    while ($count \leqslant length(tmp)$) do
      $u_i = d_g(tmp[count])$, $count = count + 1$, $i = i + 1$;
    $j = j + 1$;
  return $u = u_1 u_2 \cdots u_{i-1}$
```

Remark 3. It is obvious that g, X and S constitute a secret key. Therefore, our scheme forms a symmetric key system. As a consequence, it is suitable for data confidentiality in WBANs [15,11]. Recently, the Cloud-assisted WBANs integrating WBANs with cloud computing allows WBAN applications to exploit the benefits of cloud computing. However, because of the critical nature of the applications, it is important that the cloud be secure. At present, as pointed in [6], providing a holistic solution to securing the cloud is a difficult task due to the cloud's extensive complexity. In Section 3.4, we will show how the scheme can be applied in Cloud-assisted WBANs.

Example 2. Let $A = \{u_1, u_2, u_3, u_4, u_5\}$ and $B = \{c, a_1, a_2, a_3, b_1, b_2, b_3, b_4\}$. Suppose that B has an equal probability distribution 1/8, then Huffman codes representing c, a_1, a_2, a_3, b_1, b_2, b_3 and b_4 can be 110,001,010,011,100,101,000 and 111, respectively. Consider $X = \{c, ca_1, a_1b_1, b_1a_2, a_2b_2, b_2a_3, a_3, ca_1a_3b_1\} \subseteq B^*$ that has the unambiguous degree $k = 3$. One of the partitions of X is: $X_1 = \{c\}$, $X_2 = \{ca_1, a_1b_1\}$, $X_3 = \{b_1a_2, ca_1a_3b_1\}$, $X_4 = \{a_2b_2\}$, $X_5 = \{b_2a_3, a_3\}$, and g is defined by: $g(u_i) \in X_i$, $i = 1, \ldots, 5$. Let $m = 18$, $S = 101000110110101100$ and suppose that the original word is $u = u_2u_3u_5u_3u_4u_5u_2u_1u_3u_5$. Then, one of the encoded words produced by ENCODE is $w = w'_1 w'_2 w'_3 w'_4$. Conversely, given the encoded word w as input, DECODE gives u as the result. The detailed encoding and decoding processes are given in Tables 1 and 2, respectively. For illustration, in these tables, we use both bitstrings and symbolic notations.

3.3. Security concerns

As the design of our proposed scheme, security considerations are reduced to considering the security of the underlying cryptosystem. We recall that modern cryptography is strongly linked to complexity theory. Existing cryptosystems require (either explicitly or implicitly) the ability to generate instances of hard problems. Such an ability is captured in the definition of one-way functions. Since proving that one-way functions exist is not easier than proving that $P \neq NP$ [4], we assume that one-way functions exist as far as our cryptosystem is concerned.

To support our assumption, we analyze the computational difficulty regarding our cryptosystem in the context of attacks. Our cryptosystem can be subjected to two different types of attacks based on its design.

Case 1: The adversary does not know about X and g (i.e. ciphertext-only attacks). At first, we remark that in Schema 1, for each $g \in \mathcal{K}$, one can verify that $g(A) = X \subseteq B^*$, where $B = \{0,1\}$. This implies that g is surjective. Hence it is a bijection from A onto X. Then, g can be extended to a morphism from A^* into B^*. This fact allows us to establish encoding and decoding procedures from g as mentioned in Remark 2. Next, assume that the adversary possesses an encoded word $g(w)$. Then, he has to construct an algorithm that can produce $g(w)$, or equivalently, construct a morphism h: $A^* \to B^*$ and find a word $w \in A^*$ such that $h(w) = g(w)$. This implies that he has to solve the *Post Correspondence Problem*. It was proven that this problem is undecidable [16,5]. It is still undecidable when the length of w is restricted to a fixed $k \in \mathbb{N}$ [9].

Table 1
The detailed running steps of ENCODE.

i	j	u_i	$e_g(u_i)$	w_j	PAD (w_j)	w_j
1	1	u_2	ca_1	ca_1		
2	1	u_3	b_1a_2	$ca_1b_1a_2$		
3	1	u_5	b_2a_3	$ca_1b_1a_2b_2a_3$		011001010100000111
4	2	u_3	$ca_1a_3b_1$	$ca_1a_3b_1$		
5	2	u_4	a_2b_2	$ca_1a_3b_1a_2b_2$		000000111110111110
6	3	u_5	a_3	a_3		
7	3	u_2	a_1b_1	$a_3a_1b_1$		
8	3	u_1	c	$a_3a_1b_1c$	$a_3\underline{b}_3a_1b_1c\underline{b}_3$	101001010000100101
9	4	u_3	b_1a_2	b_1a_2		
10	4	u_5	b_2a_3	$b_1a_2b_2a_3$	$b_1a_2b_2\underline{b}_3a_3\underline{b}_4$	111010100100101111

Table 2
The detailed running steps of DECODE.

i	j	$count$	w_j	tmp	$tmp[count]$	u_i
1	1	1	110001100010101011	$ca_1b_1a_2b_2a_3$	ca_1	u_2
2	1	2	110001100010101011	$ca_1b_1a_2b_2a_3$	b_1a_2	u_3
3	1	3	110001100010101011	$ca_1b_1a_2b_2a_3$	b_2a_3	u_5
4	2	1	110001011100010101	$ca_1a_3b_1a_2b_2$	$ca_1a_3b_1$	u_3
5	2	2	110001011100010101	$ca_1a_3b_1a_2b_2$	a_2b_2	u_4
6	3	1	011000001100110000	$a_3a_1b_1c$	a_3	u_5
7	3	2	011000001100110000	$a_3a_1b_1c$	a_1b_1	u_2
8	3	3	011000001100110000	$a_3a_1b_1c$	c	u_1
9	4	1	100010101000011111	$b_1a_2b_2a_3$	b_1a_2	u_3
10	4	2	100010101000011111	$b_1a_2b_2a_3$	b_2a_3	u_5

Case 2: The adversary does not know about g (i.e. known/chosen plaintext attacks). In this case, we estimate the number of possible ways to choose g. Suppose that the sizes of A and X are n and m respectively, with $m \geqslant n$. Then, the number of ways to partition X into n non-empty subsets is the Stirling number of the second kind, denoted by $S(m, n)$. It is defined as follows

$$S(m,n) = \frac{1}{n!}\left[\sum_{j=0}^{n}(-1)^{n-j}\binom{n}{j}j^m\right].$$

The number of all injective functions g: $A \rightarrow X_P$ is $n!$. Hence the total number of ways to select g is $S(m, n) \times n!$. For instance, with $n = 5$, $m = 8$ as given in Example 2, we have $S(8, 5) = 1050$ and $5! = 120$. Then, the number of ways to select g is 126,000. Notice that, as n grows, the factorial $n!$ increases faster than all polynomial and exponential functions. Even in the case $S(m, n) = 1$, or equivalently $m = n$, we can always choose n, such as $n = 128$ is large enough for most cryptosystems.

3.4. Application of the proposed scheme

In general, our scheme proposed in the previous section deals with data confidentiality. Thus, it can have a wide range of applications in various fields. For example, the scheme can be used to protect data stored in all kinds of media, design cryptographic protocols in networks, etc. Especially, as mentioned in Remark 3, because our proposed scheme forms a symmetric key system, it is applicable to resource constrained networks such as WBANs. In the following, we give *a paradigm* for use of the scheme in Cloud-assisted WBANs. This paradigm can deal with many security issues in communication between WBANs and the cloud.

In the paradigm, parts limited by the dotted and dashed rectangle represent a traditional encryption scheme which can be symmetric or asymmetric. The innovative parts represent our scheme with two procedures, ENCODE and DECODE.

The ENCODE and DECODE procedures allow sensor nodes to secretly send their data to local servers and vice versa within a WBAN. Then, using encryption schemes, data collected from that WBAN can be securely transmitted from local servers to the cloud and stored in the cloud in a distributive manner. Thus, Cloud-assisted WBANs will enable users to globally access processing and storage infrastructure at competitive costs. Moreover, with the use of existing encryption schemes in cloud computing environments, the problems of *data authentication, data integrity, data freshness, secure localization, availability*, and *secure management*, defined as key security requirements in WBANs [15] can be effectively solved. Therefore, providers of Cloud-assisted WBANs can use the paradigm for deploying applications.

Although WBANs do not need to store and process a massive amount of data, there is a major security problem with the cloud, which is the owner of the data may not have control of where the data is placed. This is because users must utilize the resource allocation and scheduling provided by the cloud [6]. Fortunately, Park et al. [13] suggest a solution for the problem by means of Content Centric WBANs, where each data is given a name. Then, users request the data that they require using its name.

4. Results and discussions

The theoretical results presented so far ensure that our proposed scheme can be used for Cloud-assisted WBANs. However, to apply the scheme in reality we need to evaluate its performance. In this section, we discuss the implementation process of the scheme in detail. We also give some results comparing the performance of our scheme with the performance of existing encryption schemes.

Recall that our scheme takes as input a word in A^*, and produces as output some words in B^*, where A, B are finite alphabets. Here, we consider A and B as finite subsets of $\{0,1\}^*$. For security reasons, the size of A is set to be 128. This means that each element of A is represented by a bitstring of length 7. Then, we have to design B such that it can define the secret language X satisfying the following three conditions:

Table 3
Huffman codes representing elements of B.

Letter	Huffman code	Letter	Huffman code
c	0001	d_9	00100000
a_1	1000	d_{10}	00100001
a_2	1001	d_{11}	00100010
a_3	0100	d_{12}	00100011
b_1	0101	...	...
b_2	1110	...	...
b_3	0110	d_{135}	11111110
a_4	0000	d_{136}	11111111

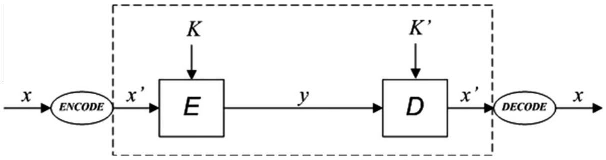

Fig. 1. The paradigm for secure communication in Cloud-assisted WBANs.

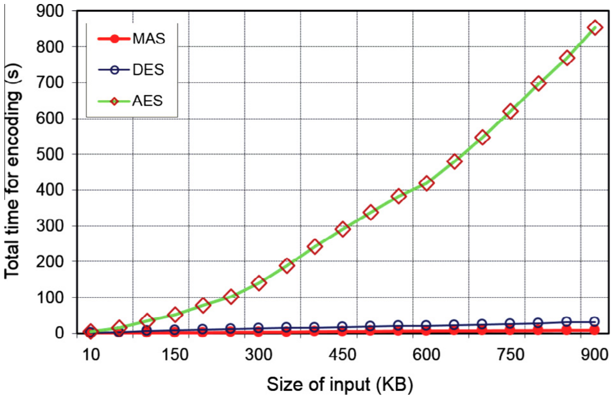

Fig. 2. Total time for encoding of MAS in comparison with DES and AES.

(1) The size of X must be greater than 128 in order to partition X into 128 non-empty subsets.
(2) The average length of words in X is close to 7 (i.e. it is close to the average length of the optimal injective encoding).
(3) X has the unambiguous degree k, $k > 0$.

Furthermore, as a requirement of easy decoding, B should be a *prefix set* (i.e. no element of B is a proper prefix of another element in B). To accomplish these goals, we use (variable-length) Huffman encoding [3] to represent elements of B as shown in Table 3. Note that symbolic letters are used for reference.

Then, the secret language X that satisfies the above conditions is: $X = \{c, ca_1, a_1b_1, b_1a_1, b_1a_2, a_2a_1, a_2b_1, a_2b_2, b_2a_1, b_2a_2, b_2a_3, b_2b_1, a_3a_1, a_3a_2, a_3b_1, a_3b_2, a_3b_3, b_3a_1, b_3a_2, b_3a_3, b_3b_1, b_3b_2, b_3c\} \cup \{d_9, d_{10}, \ldots, d_{129}\}$. By definition, one can verify that X has the unambiguous degree 4. The number of elements of X is 144. In fact, we use 128 elements of B to design X. We reserve the set consisting of all unused elements in B for padding.

Since the unambiguous degree of X is 4, the procedure ENCODE can only produce blocks of length 32 at most. In order to take advantage of unambiguous languages, we must combine blocks in groups. Moreover, for security reasons, the output of ENCODE should be a sequence of 128-bit blocks. This allows it to be secure against typical cryptanalytic attacks such as differential and linear cryptanalysis [19]. Thus, in our case, we group four 32-bit blocks to receive a single 128-bit block. Then, the size of the secret key S is set to be 128. In reality, we have to adjust the ENCODE procedure to conform to these requirements. As a result, the DECODE procedure also needs to be modified (see Fig. 1).

Actually, when implementing the modified DECODE procedure, we face a challenge due to the ambiguity of X (i.e. we must have the ability to extract factorizations on X from a given encoded word). To overcome this challenge, we use a complete finite graph with vertices represent words in X. Edges of the graph represent the concatenations of words in X. If the ambiguity happens with an encoded word, we run the Dijkstra's algorithm [17] to find the shortest factorization constituting that word, and give it as the result. We selected the programming language C-sharp for our simulation. We conducted

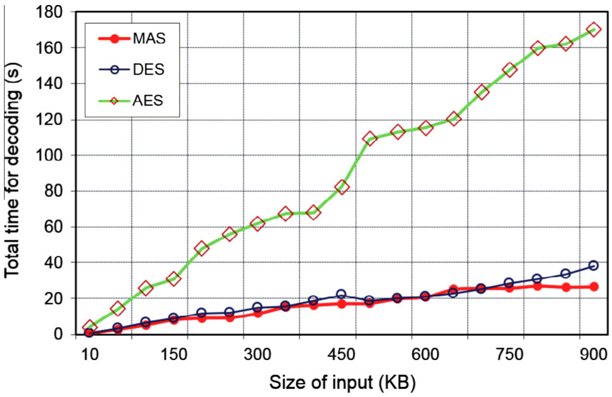

Fig. 3. Total time for decoding of MAS in comparison with DES and AES.

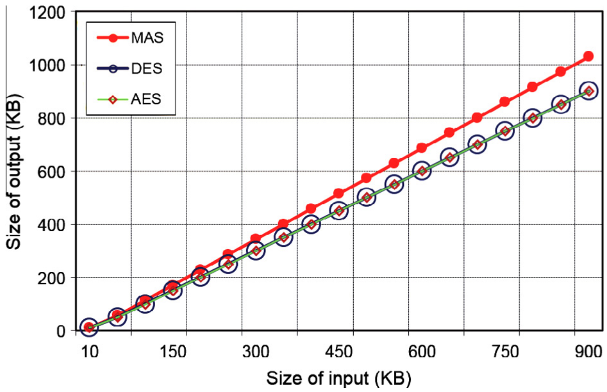

Fig. 4. The relationship between sizes of output and input in MAS, DES, and AES.

experiments to compare the performance of our scheme with two well-known block ciphers, DES and AES. Simulation results show that our scheme performed faster than DES and AES (see Figs. 2 and 3). But, one of the scheme's disadvantages is that the size of output is about 14% bigger (see Fig. 4).

The faster rate in the encoding process is due to the work of function g, that can be implemented as an operation to directly access internal memory. For the decoding process, we have to maintain some finite structures and algorithms to extract words of X. Therefore, the decoding process consumes more time compared to the time consumption of the encoding process.

The fact that bigger output can be explained that the words of language X are normally longer than the original words (i.e. a bitstring of length 7 is encoded into a bitstring of length 8). Since communication devices consume energy in WBANs, this disadvantage of the scheme should be improved.

5. Conclusions

Although the integration of Wireless Body Area Networks with a cloud computing platform enables users to globally access e-healthcare services at competitive costs, it introduces some serious security issues. In which, data confidentiality is the most important issue because the data is related to users' privacy information. In this paper, we proposed a method to solve existing problems with Cloud-assisted Wireless Body Area Networks by considering them in an integrative manner. As a result, a new scheme dealing with data confidentiality has been created which gives us a general paradigm for deploying applications in Cloud-assisted WBANs.

Acknowledgment

The authors acknowledge their great indebtedness to the referees for their valuable comments and suggestions. This research was supported by the Brain Korea 21 Plus Program funded by the Ministry of Education, South Korea.

References

[1] S. Ahn, S. Lee, S. Yoo, D. Park, D. Kim, C. Yoo, Isolation schemes of virtual network platform for cloud computing, KSII Trans. Internet Inform. Syst. 6 (11) (2012) 2764–2783.

[2] J. Berstel, D. Perrin, C. Reutenauer, Codes and Automata, Cambridge University Press, Cambridge, UK, 2010.

[3] E.N. Gilbert, E.F. Moore, Variable-length binary encodings, Bell Syst. Tech. J. 74 (1959) 933–967.

[4] O. Goldreich, Foundations of Cryptography – A Primer, Now Publishers Inc., Hanover, MA 02339, USA, 2005.

[5] V. Halava, T. Harju, Some new results on post correspondence problem and its modifications, TUCS Technical Report 388, January 2001, pp. 1–12.

[6] K. Hamlen, M. Kantarcioglu, L. Khan, B. Thuraisingham, Security issues for cloud computing, Int. J. Inform. Security Privacy 4 (2) (2010) 39–51.

[7] N.D. Han, H.N. Vinh, P.T. Huy, An extension of codes by unambiguity of languages, in: Jeng-Shyang Pan, Kebin Jia (Eds.), Proceedings of the Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IEEE Computer Society, 2012, pp. 490–493.

[8] L. Hu, O.M. Dung, Q. Liu, T. Han, Y. Sun, Integration of wireless body area networks (wbans) and wan, wimax and lte, KSII Trans. Internet Inform. Syst. 7 (5) (2013) 980–997.

[9] G. Lallement, Semigroups and Combinational Applications, John Wiley and Sons, 1979.

[10] C.-T. Li, C.-C. Lee, C.-Y. Weng, C.-I. Fan, An extended multi-server-based user authentication and key agreement scheme with user anonymity, KSII Trans. Internet Inform. Syst. 7 (1) (2013) 119–131.

[11] M. Li, W. Lou, K. Ren, Data security and privacy in wireless body area networks, IEEE Wireless Commun. 17 (1) (2010) 51–58.

[12] Q. Liu, G. Wang, J. Wu, Time-based proxy re-encryption scheme for secure data sharing in a cloud environment, Inform. Sci. 258 (2014) (2014) 355–370.

[13] Y. Park, D. Kim, M. Jo, H.P. In, Content-centric wbans for bio medical service, in: Proceedings of the 3rd International Conference on Internet (ICONI), Korean Society for Internet Information (KSII), December 2011, pp. 155–158.

[14] S. Rezvani, S.A. Ghorashi, A novel wban mac protocol with improved energy consumption and data rate, KSII Trans. Internet Inform. Syst. 6 (9) (2012) 2302–2322.

[15] S. Saleem, S. Ullah, K.S. Kwak, A study of ieee 802.15.4 security framework for wireless body area networks, Sensors 11 (2) (2011) 1383–1395.

[16] A. Salomaa, Computation and Automata, Cambridge University Press, 1985.

[17] R. Sedgewick, Algorithms in C++, Part 5: Graph Algorithms, Addition-Wesley, Pearson Education, Inc., USA, 2002.

[18] D.R. Stinson, Cryptography: Theory and Practice, CRC Press, Inc., Florida, 1995.

[19] H.C.V. Tilborg, Encyclopedia of Cryptography and Security, Springer Science+Business Media, Inc., New York, NY 10013, USA, 2005.

[20] A. Wang, X. Zheng, Z. Wang, Power analysis attacks and countermeasures on ntru-based wireless body area networks, KSII Trans. Internet Inform. Syst. 7 (5) (2013) 1094–1107.

[21] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, A.V. Vasilakos, Security and privacy for storage and computation in cloud computing, Inform. Sci. 258 (2014) (2014) 371–386.

[22] X. Zhang, Y. Xia, S. Luo, Energy-aware management in wireless body area network system, KSII Trans. Internet Inform. Syst. 7 (5) (2013) 949–966.

Minho Jo is professor of the Department of Computer and Information Science, Korea University, Sejong City, South Korea. He received his Ph.D. in Department of Industrial and Systems Engineering, Lehigh University, USA in 1994. He is the Founder and Editor-in-Chief of the KSII Transactions on Internet and Information Systems. He is an Editor of IEEE Network and an Editor of IEEE Wireless Communications. He is now Vice President of Institute of Electronics and Information Engineers (IEIE), and Vice President of Korea Information Processing Society (KIPS). Areas of his current interests include cognitive radio, mobile cloud computing, 5G wireless communications, network algorithms, optimization and probability in networks, network security, wireless communications, and mobile computing.

Hoh Peter In is a professor in the Dept. of Computer Science at Korea University, Seoul. He received his Ph.D. in Computer Science from University of Southern California (USC). He was an Assistant Professor at Texas A& M University. His primary research interests are multimedia communications, wireless networks, content-centric networks, embedded software engineering, social media platform and service, and software security management. He earned the most influential paper award for 10 years in ICRE 2006. He published over 100 research papers.

Nguyen Dinh Han is a research professor in Department of Computer and Information Science, Korea University, Sejong City. He received his Ph.D. in Computer Science from Ha Noi University of Science and Technology in 2013. He has worked as a senior researcher at Hung Yen University of Technology and Education since 2007. His current research areas are the theory of code and applications, computer and network security, wireless communications, and cloud computing.

Longzhe Han received his Ph.D. from College of Information and Communications, Korea University, Seoul in 2013. His research interests are cognitive radio networks, information centric networks, network security, multimedia communications, wireless networks and embedded software engineering.

Dao Minh Tuan is a researcher at Hung Yen University of Technology and Education. His research interests include information hiding and security, secure programming languages design and implementation, and mathematical foundation for computer science.