Toward an Inverse-free Lightweight Encryption Scheme for IoT

HakJu Kim* Kwangjo Kim*

*Department of Computer Science, KAIST.

Abstract

The current IoT (Internet of Things) systems cannot provide the overhead caused by existing encryption scheme for confidentiality. Thus, the development of lightweight encryption schemes is necessary without degrading the system performance. In this paper, we introduce a well-known lightweight encryption scheme, PRINCE, and a method to enrich the message space of PRINCE by introducing XLS (eXtension by Latin Square). We discuss the cryptographic requirements of our inverse-free lightweight encryption scheme and suggest how to extend a PRINCE-like encryption scheme for the secure IoT system.

Keywords: Lightweight encryption, inverse-free, multipermutation, and IoT

I. Introduction

Since Internet was born in the 1980s, the connectivity of human life has been increasing. The improvement of IT (Information Technology) gave the birth of new generation of the connectivity; the connectivity between things as known as IoT (Internet on Things).

Coins have both sides. IoT creates new markets and services via the communication between many things, but the development of IoT has caused new potential threats to the security. While the traditional security issue was restricted to the threat in the cyber-world, the IoT security issue must be extended to the threat in the cyber/physical-world. A successful attack on IoT can endanger human lives[1].

The cryptographic primitives like AES (Advanced Encryption Standard) and SHA-3 (Secure Hash Algorithm) were designed to defend against the cyber-threat, but those primitives would cause an unacceptable overhead to the resource-constrained devices. Therefore, the lightweight cryptographic primitives have been researched recently. The design of a lightweight encryption scheme is essential to the IoT environment, because the encryption scheme is one of the fundamental cryptographic primitives for confidentiality.

In this paper, we aim to design a new lightweight encryption scheme specialized for IoT. Our goal is to make an inverse-free lightweight encryption scheme, which means that the encryption scheme does not require its inverse to decrypt. This paper discusses the requirements of our encryption scheme and extends one well-known lightweight encryption scheme called PRINCE[2].

The remainder of this paper is organized as follows: We introduce PRINCE and XLS[3] in Section 2. The requirements of our inverse-free encryption scheme is discussed in Section 3. We propose the methodology to extend PRINCE in Section 4. Our future work and conclusion are stated in Section 5 and Section 6, respectively.

II. Related work

2.1 PRINCE[2]

PRINCE is a lightweight encryption scheme designed to perform the instantaneous encryption without a warm-up phase at minimum hardware cost. The scheme is a

64-bit block cipher with 128-bit key. The α-reflection property is used to achieve the inverse-free structure; the inverse of PRINCE$_{(k)}$ is equal to PRINCE$_{(k \text{ XOR } \alpha)}$. The inverse-freeness contributes to the low hardware cost of PRINCE. The structure of PRINCE core is illustrated in Figure 1. PRINCE achieves the encryption in one clock cycle at the maximum frequency of 200 MHz, and the unrolled version of the scheme consumes 14-15 times less hardware chip area than AES-128 and 4-7 times less hardware chip area than other lightweight encryption schemes in the literature.

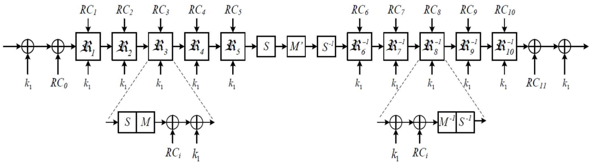
RC1 RC2 RC3 RC4 RC5 RC6 RC7 RC8 RC9 RC10
k1 RC0 k1 k1 k1 k1 k1 k1 k1 k1 k1 k1 RC11 k1
R1 R2 R3 R4 R5 S M' S-1 R6-1 R7-1 R8-1 R9-1 R10-1
S M RCi k1
M-1 S-1 k1 RCi

Figure 1. Structure of PRINCE core[2]

However, PRINCE has a major limitation. The scheme supports 128-bit key size and 64-bit block size only. Most modern encryption schemes support 128-bit block size and the flexible key size.

2.2 XLS[3]

XLS (eXtension by Latin Square) is a way to extend the message space of a cipher. Most block ciphers have the problem of ciphertext expansion, which causes more network overhead. XLS solves the problem using three block cipher calls to encrypt a 2n-1 bit string using an n-bit block cipher. XLS uses a pair of orthogonal latin squares, which are one type of the multipermutation, to prevent the problem of ciphertext expansion. In addition, the implementation of XLS requires three bit-wise XOR, one shift, and one conditional XOR operations. The permutation is involution, so the inverse permutation is not required. Thus, the XLS construction is lightweight.

III. Cryptographic Requirements

Our goal is to design an inverse-free lightweight encryption scheme specialized for IoT. Our encryption scheme, inspired by PRINCE, should be designed to have high execution speed, low hardware chip area usage, and low power consumption. In addition, our scheme should support 128-bit block size, 128/192/256-bit flexible key size, and provable security. This paper focuses on the methodology to extend the block size and the key size of PRINCE. Table 1 shows the summary of the requirements.

Table 1. Summary of the requirements

Cryptographic Requirements	Goal
Security level	Provable Security
Flexible key	128/192/256-bit key
Block size	128-bit
Other requirements	Goal
Execution speed	Faster than PRINCE
Hardware cost	Lower than PRINCE

IV. Extending PRINCE

4.1 Parallel construction

A straightforward method to extend an encryption scheme is to simply connect the scheme in parallel. However, the resulting scheme is only a mode of operation of the original scheme. We illustrate two schematics to increase the block size and the key size of PRINCE in Figure 2.

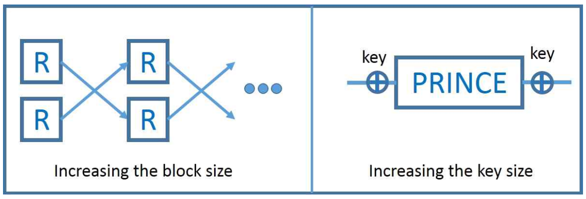

Figure 2. Schematics to increase the block size and the key size of PRINCE

Using the theorem in [2], we believe that we can claim the following theorem if you used the schematics in Figure 2.

Theorem 1. *The advantage of any adversary who makes D queries to the E oracle and T queries to the F/F-1 oracle satisfies*

$$v_X^{CPA}(A) = |\Pr[k \overset{\$}{\leftarrow} F^{\kappa+2n}, F \overset{\$}{\leftarrow} (P_n)^{2^{\ -1}} : A^{\widetilde{F}X_k,F,F^{-1}} = 1]$$

$$- \Pr[\pi \overset{\$}{\leftarrow} P_n, F \overset{\$}{\leftarrow} (P_n)^{2^{\kappa-1}} : A^{\pi,F,F^{-1}} = 1]| \leq DT2^{-(n+\kappa-2)}$$

Parameter n is the block size and k is the key size of the extension of PRINCE. Rest of parameters are explained in [2].

(Proof) Skipped.

4.2 Swapping using Multipermutation

The scheme in Figure 2 is very naïve and inflexible. The orthogonal latin square is a secure mixing function, and is flexible to extend PRINCE to have the block size larger than 128-bit. The implementation of the orthogonal latin square requires three bit-wise XOR, one shift and one conditional XOR operations (dbl), and the implementation of the approximated version requires three bit-wise XOR and one left circular bit-rotation (rol)[3]. Both versions are involutions, so they correspond to the inverse-freeness of PRINCE. The implementation of orthogonal latin square and its approximated version illustrated in Figure 3.

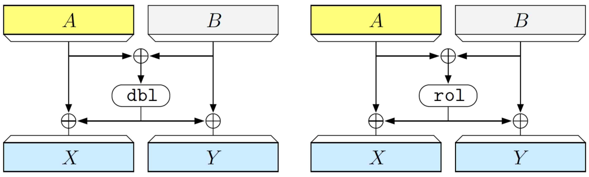

Figure 3. The implementation of orthogonal latin square (left) and its approximated version (right) [3]

We propose the extension of PRINCE using the orthogonal latin square. The orthogonal latin square replaces the simple-swapping in Figure 2. The key is divided into four 64-bit subkeys, which are XORed at each round. The block size of the resulting encryption scheme is 128-bit, and the scheme has the flexible key size of 128/192/256-bit. The use of the orthogonal latin square does not degrade the security level of PRINCE, because the orthogonal latin square is a secure construction[3]. The approximated version is less secure than the original orthogonal latin square, but the difference is negligible[3]. In addition, the approximated version of the orthogonal latin square is more lightweight than the original version. We can have the security- performance trade-off. Although the security proof is not completed, the resulting lightweight encryption scheme meets the cryptographic requirements of our inverse-free lightweight encryption scheme in Table 1. Our extension of PRINCE is illustrated in Figure 4.

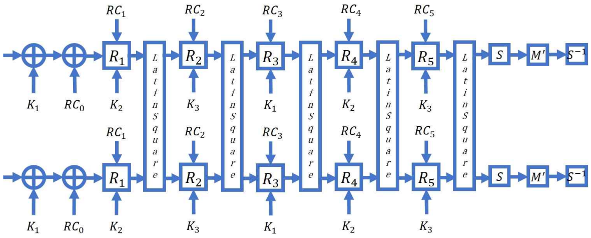

Figure 4. Our extension of PRINCE (first half of the construction)

We bring the definition of a mixing function from [3] and build a conjecture that Theorem 1 still holds for the extension of PRINCE using the orthogonal latin square.

Definition 1. *Fix a set* $S \subseteq 0,1\}^{\geq 1}$, *let* mix: $S^2 \rightarrow S^2$ *be a length-preserving permutation, and let* $\epsilon: N \rightarrow [0,1]$. *We say that* mix *is an* $\epsilon(s)$*-good mixing function if, for all s such that* $\{0,1\}^s \subseteq S$, *we have that*

(1) $\mathrm{mix_L}(\mathrm{A},\cdot)$ is a permutation for all $A \ \ \{0,1\}^s$,

(2) $\mathrm{mix_R}(\cdot,\mathrm{B})$ is a permutation for all $B \in \{0,1\}^s$,

(3) $\Pr[R \overset{\$}{\leftarrow} \{0,1\}^s : C = mix_L(R,B)] \leq \text{for } all\ B, C \in \{0,1\}^s$,

(4) $\Pr[R \overset{\$}{\leftarrow} \{0,1\}^s : C = mix_R(A,R)] \leq \text{for } all\ A, C \in \{0,1\}^s$

Conjecture 1. *Let the multipermutation be a* $\epsilon(s)$ *-good mixing function, Theorem 1 still holds for the extension of PRINCE using the orthogonal latin square.*

4.3 Evaluation

We believe that the security level of our design is at least as strong as the security level of PRINCE. However, our design can be vulnerable against the related key attack proposed in [4].

One orthogonal latin square block consists of five bit-wise operations. There are ten orthogonal latin square blocks in our scheme, so the scheme has increased the performance/cost overhead by 50 bit-wise operations from the original PRINCE. The increased overhead is negligible, but we should design the scheme to have a faster performance and a lower cost than PRINCE.

V. Future work

Our proposed scheme should be revised to resist the related key attack, and a complete security proof of the scheme against all the known attacks including side-channel attacks should be done. The different kinds of the multipermutation and the components of PRINCE should be studied to design a faster and cost-efficient encryption scheme. We will evaluate our design by SoC (System on Chip) later.

VI. Conclusion

IoT is entering human lives quickly. Increased connection between physical-world and cyber-world poses more dangerous threats. The resource-constrained environment of IoT has challenged the cryptographic academia. Our contribution is as follows: First, we have explained the need to develop new lightweight cryptographic primitives specialized for IoT. Second, we have introduced a well-known lightweight encryption scheme called PRINCE and a method to extend the message space of a cipher. Third, we have discussed the requirements of our inverse-free lightweight encryption scheme specialized for IoT. Fourth, we have extended PRINCE using orthogonal latin square to meet the cryptographic requirements.

Acknowledgement

This research was funded by the MSIP (Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2014 [1391104001, Research on Communication Technology using Bio-inspired Algorithm].

[References]

[1] 미래창조과학부: 스마트 안심국가 실현을 위한 사물인터넷(IoT) 정보보호 로드맵 (2014년 10월 31일)

[2] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçın, "PRINCE – A Low-latency Block Cipher for Pervasive Computing Applications," Advances in Cryptology – *ASIACRYPT 2012*. Springer Berlin Heidelberg, 2012, 208-225.

[3] Thomas Ristenpart and Phillip Rogaway. "How to enrich the message space of a cipher." *Fast Software Encryption*. Springer Berlin Heidelberg, 2007.

[4] 주왕호, 안현정, 이옥연, 강주성, 김종성. "최신 경량 블록암호 PRINCE에 대한 향상된 연관키 공격" *정보보호학회논문지 제24권 3호*. 한국정보보호학회, 2014년 6월, 445-451.