JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 19, No. 3, pp. 579-586, Mar. 2018

부채널 공격 대응을 위한 Rekeying 기법에 관한 연구

짠 송 닷 푹 · 이 창 훈*
서울과학기술대학교 컴퓨터공학과

A Study on Rekeying and Sponged-based Scheme against Side Channel Attacks

Tran Song Dat Phuc · Changhoon Lee*

Department of Computer Science and Engineering, Seoul National University of Science and Technology, Seoul 01811, Korea

[요 약]

SPA(Simple Power Analysis)와 DPA(Differential Power Analysis) 공격은 1999년 Kocheretal.[2]이 소개한 부채널 공격(SCA, Side Channel Attacks)으로 SPA는 공격자가 전력소비 또는 전자기 방사선과 같은 단일 측정 트레이스에 대한 부채널 정보를 수집 및 분석해 키를 유추하고 DPA는 동일한 키로 암호화한 서로 다른 평문과 같은 여러 측정 트레이스에 대한 부채널 정보를 수집하고 이에 대한 차분을 이용해 키를 유추하는 보다 정교한 공격방법이다. SPA와 DPA는 공격자가 수집할 수 있는 부채널 정보를 본질적으로 줄여 대응해야 하기 때문에 SPA와 DPA에 대한 대응에는 많은 어려움이 있다. 본 논문에서는 SPA및 DPA와 같은 수동적 부채널 공격에 대응하기 위한 ISAP[8] 스킴에 대한 안전성에 대해 다루고 있고 기존에 부채널 공격에 대응하기 위한 기법 Rekeying 기법과 스펀지 구조를 다루고 있다. 또한, 본 논문에서는 Rekeying 기법과 스펀지 구조에 기반해 보다 안전한 암호화 및 인증을 제공하는 개선된 ISAP 스킴을 제안하고자 한다.

[Abstract]

Simple Power Analysis(SPA) and Differential Power Analysis(DPA) attacks are Side Channel Attacks(SCA) which were introduced in 1999 by Kocher et al [2]. SPA corresponds to attacks in which an adversary directly recovers key material from the inspection of a single measurement trace (i.e. power consumption or electromagnetic radiation). DPA is a more sophisticated attacks in which the leakage corresponding to different measurement traces (i.e. different plaintexts encrypted under the same key) is combined. Defenses against SPA and DPA are difficult, since they essentially only reduce the signal the adversary is reading, PA and DPA. This paper presents a study on rekeying and sponged-based approach against SCA with current secure schemes. We also propose a fixed ISAP scheme with more secure encryption and authentication based on secure re-keying and sponge functions.

색인어 : Rekeying, 스펀지 기반 구조, ISAP, 부채널 공격, 마스킹

Key word : Rekeying, Sponged-based Construction, ISAP, Side Channel Attacks, Masking

http://dx.doi.org/10.9728/dcs.2018.19.3.579

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-CommercialLicense(http://creativecommons.org/licenses/by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 16 February 2018; **Revised** 22 February 2018
Accepted 25 March 2018

***Corresponding Author; Chanhoon Lee**

Tel: +82-2-970-6712
E-mail: chlee@seoultech.ac.kr

Ⅰ. Introduction

The design of efficient and effective countermeasures against side-channel and fault attacks is a very challenging task. In the early years, the main goal of designers of embedded systems was to engineer systems in such a way that they do not leak side-channel information at all, or to randomize the power consumption by masking techniques. However, over the years it has become more and more clear that such countermeasures are very expensive to implement for settings with high security requirements.

Fresh re-keying is a type of protocol which aims at splitting the task of protecting an encryption/authentication scheme against side-channel attacks in two parts. One part, a re-keying function, has to satisfy a minimum set of properties (such as good diffusion), and is based on an algebraic structure that is easy to protect against side-channel attacks with countermeasures such as masking. The other part, a block cipher, brings resistance against mathematical cryptanalysis, and only has to be secure against single measurement attacks. Since fresh re-keying schemes are cheap and stateless, they are convenient to use in practice and do not require any synchronization between communication parties.

Sponge-based designs is motivated by their suitability to model SPA leakage. Namely, the sponge parameters provide a convenient tool to argue on the side-channel security of keyed sponge constructions given bounded side-channel leakage of the single permutation. The basic idea is to use the sponge parameters to express a construction's capability to cope with the leakage generated by the permutation. Particularly, the sponge parameters are adjusted according to the amount of information an adversary learned about the secret state.

ISAP [1] is a family of symmetric authenticated encryption based on sponge-based construction and fresh re-keying approaches within Encrypt-then-MAC mechanism. ISAP family has two main versions: ISAP-128 and ISAP-128a, which both are designed for 128-bit cryptographic and passive side-channel attacks security, the DPA and limited SPA leakage particularly.

In this paper, we give a study on rekeying and sponged-based approach against side channel attacks in combination with fixing ISAP scheme in construction. An approach in secure rekeying function is presented to apply with more efficient performance.

Ⅱ. Rekeying and Sponged-based Construction

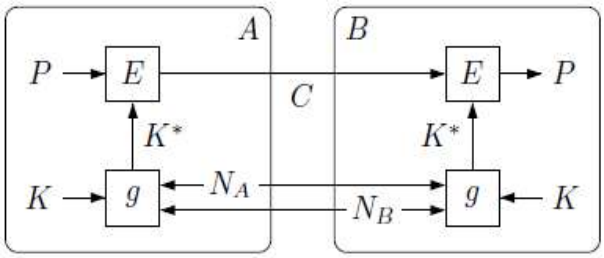


그림 1. 두 사용자 사이에서의 Rekeying scheme
Fig. 1. Rekeying scheme with two parties

Rekeying is a countermeasure to DPA that can be seen to work on protocol level. The idea of frequent re-keying is to prevent DPA on the cryptographic primitive by limiting the number of processed inputs per key. In other words, it limits the data complexity for each key by a small number q that renders DPA on the key infeasible. It is nowadays a common assumption that small data complexities have sufficiently small side-channel leakage and do not allow for successful key recovery from DPA attacks.

On the encryption of every new plaintext P, the block cipher E is provided with a new session key K^*. This session key K^* is derived from a pre-shared master secret K and a nonce N that is randomly generated on the tag. This inherently prevents DPA on the session key K^* of the block cipher E. However, for key derivation it requires a re-keying function g: $(K, N) \rightarrow K^*$ that is easy to protect against both SPA and DPA attacks.

2-1 Basic Rekeying Fuction

The scheme was proposed at AFRICACRYPT 2010, built from a block cipher BC and a rekeying function g. The rekeying function g(k, r) to derive new session keys, and the block cipher E(k^*, m) to encrypt message blocks. First, the rekeying function produces a session key k^* from the master key k and a random nonce r. Second, the plaintext x is encrypted by the fresh key k^* with a block cipher. It can easily turn into a hybrid rekeying by using a counter instead of the random nonce r.

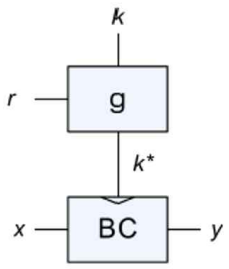


그림 2. 기본 Rekeying 스킴 [5]
Fig. 2. Basic rekeying scheme [5]

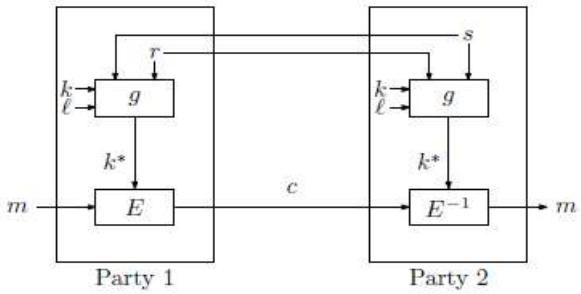


그림 3. 다중 사용자를 위한 Rekeying 스킴 [17]
Fig. 3. Rekeying scheme for multi-party [17]

2-2 ABdalla-Bellare Rekeying Scheme

This scheme [8] is based on a pseudo-random function (PRF) in combination with a hash function at instantiation step. It proves the security when combination of g with a well-chosen compression function PRF. The function g handles with side-channel protection, since the compression function prevent pre-image and collision attack. It also guarantees that an attacker cannot distinguish the output of F from a random sequence, which implies that he cannot recover the key *k* that generated this output. This construction is provably resistant against the collision-based key recovery attack. It can be seen as an extension of the basic rekeying scheme.

However, because of the additional compression function and block cipher, it leads to a large performance overhead for a single encryption. And, implementation is also more expensive than original fresh rekeying scheme.

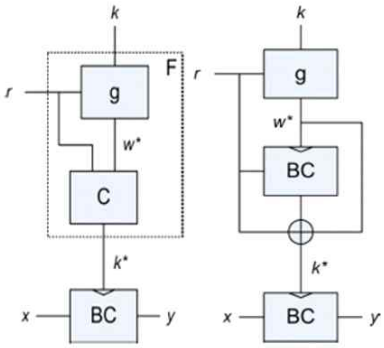


그림 4. Abdalla-Bellare의 Rekeying 스킴
Fig. 4. Abdalla-Bellare rekeying scheme

2-3 Kocher' s Rekeying Scheme

Kocher's rekeying scheme [3] proposed a different way to produce session key. The session key is not derived from a static secret master key and a random nonce. Instead, it uses a tree structure concept to update and assign the secret key as session key. The number of usable session keys k^*_i is determined by the depth of the tree.

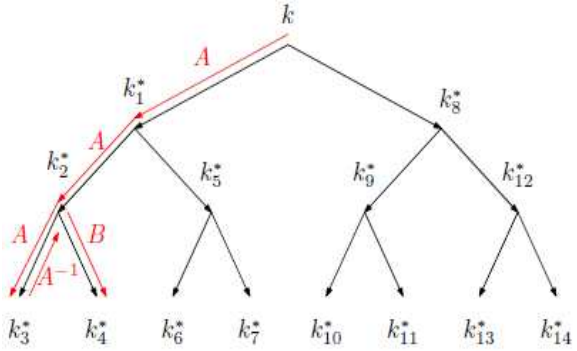


그림 5. Kocher의 Rekeying 스킴
Fig. 5. Kocher's rekeying scheme

The root of the tree is the secret master key *k* and the other vertices represent session keys k^*_i. To traverse through the tree, the functions *A*, *B*, A^{-1}, and $B^{-1}$are used, where A^{-1}, and $B^{-1}$are the inverse functions of A, and B. For instance, $k^*_1 = A(k)$, and $k^*_8 = B(k)$.

2-4 Sponge-based Construction

Sponge functions [18] are a generalization of hash functions and using the latter for generating pseudo-random bits. Beyond hash functions, sponges have been used to build several cryptographic objects from permutations. A recommendation for random number generation using deterministic random bits generators is published by NIST to specify how to implement a PRNG using a hash function, a keyed hash function, a block cipher or an elliptic curve.

The sponge-based constructions provide advantages since it allows to implement a wide range of primitives (hash, MAC, and cipher) with elegant and simple design, obvious state size, no key schedule, and key is injected once. This construction is causing less implementation overhead for decryption, since no inverse building blocks (permutation) are needed. It is useful in limited-resource applications. The confidentiality and integrity of a message can be guaranteed with a single processing pass, without the use of a separate encryption algorithm and a HMAC.

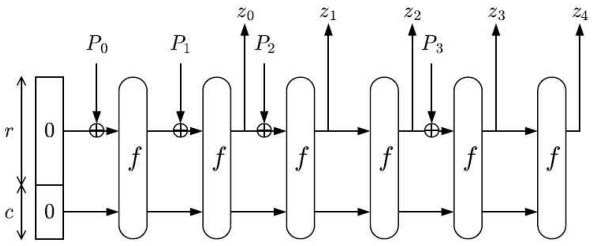


그림 6. 스펀지 구조
Fig. 6. The sponge construction

The sponge construction then proceeds in two phases: the absorbing phase followed by the squeezing phase. In the absorbing phase, the *r*-bit input message blocks are XORed into the first *r* bits of the state, interleaved with applications of the function *f*. When all message blocks are processed, the sponge construction switches to the squeezing phase. In the squeezing phase, the first *r* bits of the state are returned as output blocks, interleaved with applications of the function *f*. The number of output blocks is chosen at will by the user.

Ⅲ. Rekeying and Sponged-based ISAP Scheme

ISAP [1] is a family of symmetric authenticated encryption based on sponge-based construction and fresh re-keying approaches within Encrypt-then-MAC mechanism. ISAP family has two main versions: ISAP-128 and ISAP-128a, which both are designed for 128-bit cryptographic and passive side-channel attacks security, the DPA and limited SPA leakage particularly.

Each member of ISAP family is defined by several different parameters: round numbers *a*, *b* and *c* for permutation p^a, p^b and p^c; and rates r_1, r_2 and r_3. It is using two *k*-bits secret keys K_A and K_E to construct three major building blocks: ISAPENC for encryption, ISAPMAC for authentication and ISAPRK served as a re-keying function for absorbing the secret key K_A.

표 1. ISAP 파라미터
Table 1. ISAP parameters

Member	*k*-bit Security	Rate(bits)			Rounds		
		r_1	r_2	r_3	*a*	*b*	*c*
ISAP-128	128	144	1	144	20	12	12
ISAP-128a	128	144	1	144	16	1	8

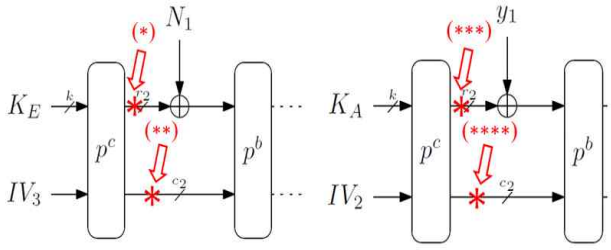
The sponge-based construction ISAPENC encrypts the plaintext to compute the ciphertext as follows. It inputs with the *k*-bit secret key K_E and a constant IV_3 through the *c*-round permutation p^c. Then, the same *k*-bit nonce N is absorbed using a r_2 bits rate and the *b*-round permutation p^b. Finally, the keystream is squeezed using a r_3 bits rate and the *c*-round permutation p^c. The ciphertext *C* is generated by XOR-ing the plaintext *P* and the keystream with the same length of *P*.

The ISAPMAC is a sponge-based suffix MAC using a function *g* as a re-keying function to absorb secret key K_A, instead of an XOR operation for securing the K_A against various passive side channel attacks, such as SPA and DPA. It produces the tag *T* for authentication as follows. First, the *k*-bit nonce *N* and a constant IV_1 are processed, followed by the *a*-round permutation p^a. Then, the associated data $A_{1..s}$ and the ciphertext $C_{1..t}$ are absorbed using the *b*-round permutation p^b. Finally, the secret key K_A is absorbed by the function *g* and the *k*-bit tag *T* is squeezed using the *a*-round permutation p^a.

The function *g* is used as a re-keying function and described by the building block ISAPRK. The process begins with the inputs of the secret key K_A and a constant IV_2 through the *c*-round permutation p^c. Then, the *k*-bit value *y* is absorbed using a r_2 bits rate and the *b*-round permutation p^b. Finally, the K^*_A is squeezed using a *k* bits rate and the *c*-round permutation p^c.

For more detail, refer [1].

Ⅳ. Security of ISAP against Side-channel Attacks



(a.) (b.)

그림 7. (a.) ISAPENC에 대한 DPA 공격 (b.) ISAPRK에 대한 DPA 공격
Fig. 7. (a.) The DPA attacks on ISAPENC (b.) The DPA attacks on ISAPRK

Recover the master key K_A, K_E :
Message block *v*

1. Precomputation:
Repeat *t* times:

a./ Guess a new value for rate r_2 and c_2 by some DPA oracles for leakage information.
b./ Compute $s = p^c(r_2, c_2)$ and save a pair(s, r_2,) in a list L.

2. Queries :
Repeat $t' = 2^n/t$ times:
a./ Request rate r_2 through c-round permutation p^c
b./ If list L contains an entry (s, r_2) for some r_2, return r_2 and s as secret key K_E for encryption and K_A for authentication.

The ISAP scheme is designed to provide not only ability to secure against passive side-channel attacks, especially DPA and SPA but also good performance and low hardware footprint. Compared to other structures leads to overheads and cost increases with the protection order for side-channel attacks resistance, ISAP is using the fresh re-keying scheme within its mechanism to lower overheads and, in combination with the sponge-based construction which also cause less implementation overhead for decryption.

The security of the ISAP scheme is claimed to be secure against DPA and SPA attacks based on the assumption of two secure re-keying function g_1, g_2; and sponge-based constructions within the implementations of encryption, decryption and authentication MAC processes. However, the lack security in the ISAPENC encryption and ISAPRK re-keying structures while only processing with publicly known data inputs may allow for successful key recovery from DPA attacks.

Because of fact that initial values IV_i and k-bits master keys K_A, K_E are all publicly known data (with K = K_A || K_E), the attacker can execute a key recover based on DPA oracle attacks to obtain K_A and K_E. The attack process is as follows.

Ⅴ. Fixed ISAP Scheme with Rekeying Function

5-1 Encryption Process

The encryption inputs with the k-bit secret key K'_E given by rekeying function combined with masking technique to produce session key, and a constant IV_3 through the c-round permutation p_c. Then, the same k-bit nonce N is absorbed by the re-keying function g_1. Finally, the key stream is squeezed using a r_3 bits rate and the c-round permutation p_c. The ciphertext C is generated by XOR-ing the plaintext P and the keystream with the same length of P.

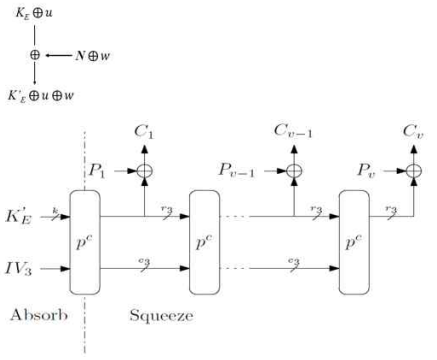


그림 8. 마스킹 기법이 적용한 개선된 암호화 프로세스
Fig. 8. Fixed encryption process with masking technique

m, x variants are variables all masked with same values. All variables are masked by independent random values which makes side-channel attacks are vulnerable to apply.

5-2 Authentication Process with Rekeying Function

The authentication process is same as the ISAPMAC authentication of ISAP scheme since the secret key K_A is absorbed by the secure re-keying function g_2. The function g is also described as re-keying building block. The process begins with the inputs of the random nonce N and a constant IV_2 through the c-round permutation p^c. Then, the k-bit value y is absorbed by the function g_2. Finally, the K^*_A is squeezed using a k bits rate and the c-round permutation p^c.

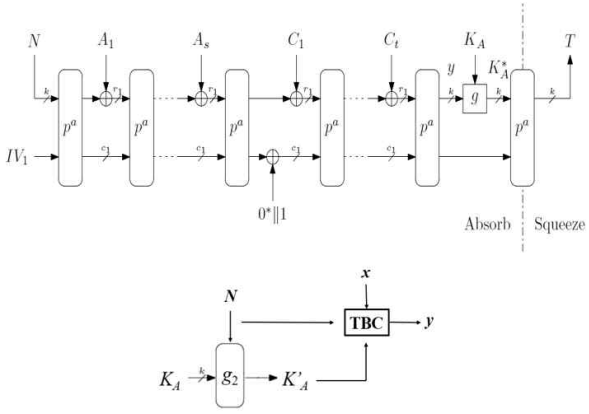


그림 9. TBC(Tweakable Block Cipher)와 결합된 Rekeying 기법이 적용된 인증 프로세스
Fig. 9. Authentication process with rekeying function combined with a TBC(Tweakable Block Cipher)

We get always a different session key K^*_A for different nonce N. For every different value of the tweak, we have

different and independent block cipher instances. So, basically, we just use different block ciphers with different keys, and none of them is used with multiple keys, which makes the side-channel attacks impossible to apply. It increases the size of the list, since an attacker trying to perform the first step of the attack and he need to pre-calculating a list for a set of pairs (N_i; K^*_i), not for a set of K^*_i.

Ⅵ. Security Results

Combine the structure with the tweak cipher with the tweak space: $T = T_0 = \{0, 1\}^{b-k}$ x $\{0, 1\}$ and the masking: $M = (K, X, i) \rightarrow 0^r \| right^{b-r}(P(X \| K))$ that satisfy the security: Adv $(\sigma, p, k) \leq (2\sigma^2/2^b) + (5.5\sigma^2/2^{b-r}) + (p/2^k) + (3\sigma p/2^{b-r})$

Since we use the $(b - r)$ masking of properness, with random secret tweakable permutation, b-bit string, r rate and c capacity of sponge function: $b = r + c$.

As the secure tweakable structure in Rekeying function against side channel attacks properties, we can consider SCREAM and iSCREAM scheme [20].

표 2. SCREAM과 iSCREAM 설계 접근 방법
Table 2. SCREAM and iSCREAM design approaches

	SCREAM	iSCREAM
Side channel resist.	with masking	with masking
Related-Key resist.	optional	optional
Confidentiality	128 bits	128 bits

표 3. SCREAM과 iSCREAM의 성능
Table 3. Performance result of SCREAM and iSCREAM schemes

	ROM words			RAM	Cycle count	
	code	tables	total	words	encrypt	decrypt
AES	1147	512	1659	33	4557	7015
AES furious	800	768	1568	192	3629	4462
Scream-10	1173	2048	3221	80	7646	7672
iScream-12	951	1024	1975	64	8724	8724

The security analysis shows that the good properties, allows generating secure improved bounds on the probability of differential and linear trails. With the tweakable schedule is using bit rotation which prevents the simple trails exploitation with difference by the tweak. The best trails is presented have 12 active S-boxes between one active step and two inactive steps.

Corresponding to ISAP scheme, an implementation of secure GGM construction with AES-128 on an 8-bit micro-controller can be exploited by using several side channel attacks, such as template attacks. An 8-bit micro-controller implementation needs more complicated side channel attacks countermeasures than a parallel implementation of the round function.

Ⅶ. Conclusion

In this paper, we present a study on rekeying and sponged-based approach against side channel attacks with an overview of secure rekeying and sponge functions. We also discussed the security of the ISAP scheme against the side-channel attacks, such as SPA and DPA The fixing solution increases the security level against those attacks in combination between re-keying functions, sponge-based functions with masking technique and tweakable cipher to avoid the publicly known inputs in mechanism. Furthermore, it is expected to get a better performance in design for a lightweight construction in further research.

Acknowledgement

This study was supported by the Research Program funded by the SeoulTech(Seoul National University of Science and Technology).

References

[1] Dobraunig, C., Eichlseder, M., Mangard, S., Mendel, F., Unterluggauer, T, "*ISAP- towards side-channel secure authenticated encryption.*", IACR Trans. Symmetric Cryptol. 2017(1), 80–105, 2017.

[2] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun, *Differential Power Analysis,* In Michael J. Wiener, editor, CRYPTO '99, Vol. 1666 of LNCS, pp. 388–397. Springer, 1999.

[3] Paul Kocher, *Leak Resistant Cryptographic Indexed Key Update*, US Patent 6539092, 2003.

[4] Bart Mennink, Reza Reyhanitabar, and Damian Vizár "*Security of full-state keyed sponge and duplex: Applications to authenticated encryption*," iIn Tetsu Iwata and Jung Hee Cheon, editors, ASIACRYPT 2015, Vol. 9453 of LNCS, pp. 465–489. Springer, 2015.

[5] Marcel Medwed, François-Xavier Standaert, Johann Großschädl, and Francesco Regazzoni, "*Fresh re-keying: Security against side-channel and fault attacks for low-cost*

devices," in Daniel J. Bernstein and Tanja Lange, editors, AFRICACRYPT 2010, volume 6055 of LNCS, pages 279–296. Springer, 2010.

[6] Chang-hoon Lee, "*A Study on Application Method of Crypto-module for Industrial Control System*", in Content Applications and Convergence Technology, Journal of DCS, Vol. 18, No. 5, August. 2017.

[7] Elena Andreeva, Joan Daemen, Bart Mennink, and Gilles Van Assche, "*Security of keyed sponge constructions using a modular proof approach*", In Gregor Leander, editor, FSE 2015, Vol. 9054 of LNCS, pp. 364–384. Springer, 2015.

[8] Abdalla, M., Bellare, M., "*Increasing the lifetime of a key: A comparative analysis of the security of re-keying techniques*", In: Okamoto, T. (ed.) ASIACRYPT 2000. LNCS, vol.1976, pp. 546–559. Springer 2000.

[9] Hee-Sook Kim, "*Smart CCTV Security Service in IoT(Internet of Things) Environment*", in Convergence Content Services, Journal of DCS, Vol. 18, No. 6, October. 2017.

[10] Sonia Belaïd, Fabrizio De Santis, Johann Heyszl, Stefan Mangard, Marcel Medwed, Jörn-Marc Schmidt, François-Xavier Standaert, and Stefan Tillich, "*Towards fresh re-keying with leakage-resilient PRFs: Cipher design principles and analysis*.", J. Cryptographic Engineering, 4(3):157–171, 2014.

[11] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche, "*On the indifferentiability of the sponge construction*", in Nigel P. Smart, editor, EUROCRYPT 2008, volume 4965 of LNCS, pages 181–197. Springer, 2008.

[12] Francesco Berti, François Koeune, Olivier Pereira, Thomas Peters, and François-Xavier Standaert, "*Leakage-resilient and misuse-resistant authenticated encryption*", Cryptology ePrint Archive, Report 2016/996, 2016.

[13] Christoph Dobraunig, François Koeune, Stefan Mangard, Florian Mendel, and François-Xavier Standaert, "*Towards fresh and hybrid re-keying schemes with beyond birthday security*", In Naofumi Homma and Marcel Medwed, editors, CARDIS 2015, Vol. 9514 of LNCS, pp. 225–241. Springer, 2015.

[14] Sebastian Faust, Krzysztof Pietrzak, and Joachim Schipper, "*Practical leakage resilient symmetric cryptography*", in Emmanuel Prouff and Patrick Schaumont, editors, CHES 2012, volume 7428 of LNCS, pages 213–232. Springer, 2012.

[15] Qian Guo and Thomas Johansson, "*A new birthday-type algorithm for attacking the fresh re-keying countermeasure*", Cryptology ePrint Archive, Report 2016/225, 2016.

[16] Stefan Mangard, Elisabeth Oswald, and Thomas Popp, "*Power analysis attacks – Revealing the secrets of smart cards*", Springer, 2007.

[17] Marcel Medwed, Christophe Petit, Francesco Regazzoni, Mathieu Renauld, and François-Xavier Standaert, "*Fresh re-keying II: Securing multiple parties against side-channel and fault attacks*", in Emmanuel Prouff, editor, CARDIS 2011, Vol. 7079 of LNCS, pp. 115–132. Springer, 2011.

[18] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "*Sponge functions*", in Ecrypt Hash Workshop 2007, May 2007, also available as public comment to NIST from http://www.csrc.nist.gov/pki/HashWorkshop/Public_Comments/2007_May.html.

[19] Thomas Unterluggauer, Mario Werner, and Stefan Mangard., "*Side-channel plaintext-recovery attacks on leakage-resilient encryption*.", in DATE, 2017.

[20] V. Grosso, G. Leurent, F.-X. Standaert, K. Varici, F. D. A. Journault, L. Gaspar, and S. Kerckhof, "*SCREAM & iSCREAM Side-Channel Resistant Authenticated Encryption with Masking*;", in CAESAR Competition, 2014.

짠 송 닷 푹(Tran Song Dat Phuc)

2011년 : HCMC University of Technology, Vietnam (Bachelor Degree)
2015년 : Seoul National University of Science and Technology, Depart. of Computer Science and Engineering (Master Degree)

2015년～현 재: Seoul National University of Science and Technology, Depart. of Computer Science and Engineering (Doctoral Candidate)
※ Interests : Cryptography, Network Security, Information Security.

이 창 훈(Changhoon Lee)

2001년 : Hanyang University, Depart. of Mathematics (Bachelor Degree)
2003년 : Korea University, Depart. of Information Management and Security (Master Degree)
2008년 : Korea University, Depart. of Information Management and Security (Doctoral Degree)

2009년～2012년: Hanshin University, Depart. of Computer Engineering (Assistant Professor)
2012년～2015년: Seoul National University of Science and Technology, Depart. of Computer Science and Engineering (Assistant Professor)
2015년～현 재: Seoul National University of Science and Technology, Depart. of Computer Science and Engineering (Associated Professor)
※ Interests : Information Security, Cryptography, Digital Forensics, Computer Theory.