How to cite this paper:

Baraa Tariq Hammad, Norziana Jamil, Mohd Ezanee Rusli, & Muhammad Reza Zaba. (2017). DLP sponge construction for authenticated encryption in Zulikha, J. & N. H. Zakaria (Eds.), Proceedings of the 6th International Conference on Computing & Informatics (pp 714-721). Sintok: School of Computing.

DLP SPONGE CONSTRUCTION FOR AUTHENTICATED ENCRYPTION

Baraa Tariq Hammad[1] Norziana Jamil[1] Mohd Ezanee Rusli [1] and Muhammad Reza Zaba[2]

[1] *Universiti Tenaga Nasional, Malaysia, omrami82@yahoo.com, {Norziana, ezanee}@uniten.edu.my*
[2] *MIMOS Berhad, Malaysia, reza.zaba@mimos.my.*

ABSTRACT: In this paper, we present a new DLP-sponge construction to ensure integrity and privacy. This scheme solves problem related to small keys by introducing a double length construction: $k \approx 2r$. Previous researcher show that the size of key k must be twice of the capacity c which will in turn affect the size of the underlying permutation: $b = c + r$. As c decreases, the bitrate r increases as well. Besides that, our scheme is resistant against most of the generic attacks such as multicollision attack with a complexity of $t.2^{2(c+3)\backslash 2}$. Therefore, our scheme is better than some of the existing Authenticated Encryption (AE) schemes.

Keywords: Authenticated Encryption, Sponge construction, Double length.

INTRODUCTION

Authenticated Encryption AE is a kind of encryption whose goal is to provide both privacy, integrity and authenticity. In this scheme, the sender takes the key and the plaintext to return both ciphertext and Message Authentication Code (MAC). Meanwhile, to decrypt it, the receiver takes the same key and the ciphertext to return either plaintext or MAC. AE combines both encryption and (MAC). The underlying computational cost is less than those of encrypt and MAC. Still, AE schemes are widely used nowadays (Rogaway, Bellare, & Black, 2003).

Three composition methods are considered: (a) Encrypt-and-MAC encrypts the plaintext and appends a Mac; (b) MAC-then-encrypt appends a MAC to the plaintext and then encrypts them together; and (c) Encrypt-then-MAC encrypts the plaintext to get the ciphertext and appends the MAC (Bellare, & Namprempre, 2000).

Examples of AE algorithms are ALE (Bogdanov, Mendel, Regazzoni, Rijmen, & Tischhauser, 2013), FIDES (Bilgin, Bogdanov, Knežević, Mendel & Wang, 2013), LAC (Zhang, Wu, Wang, Wu, & Zhang, 2014) and LADP (Li, Xu, & Li, 2015). Most of these primitives are designed for constrained devices.

Various modes of operation have been designed to gather both privacy and integrity. As Examples XCBC and EAX (Gligor & Donescu, 2001), IAPM (Jutla, 2001), CWC (Kohno, Viega, & Whiting, 2004) and GCM (Lemsitzer, Wolkerstorfer, Felber, & Braendli, 2007). Authenticated encryption modes are categorized as single pass modes or double pass modes. Some modes are also allowed for the authentication of unencrypted associated data and these modes are called AEAD (Authenticated-Encryption with Associated-Data). For example, OCB and EAX are single pass and double pass AEAD schemes, respectively.

In order to have a more secure and efficient construction, a double length (from single length) construction has been considered. Hirose (Hirose, 2006) proposed a Double Block Length Construction with two different block ciphers. The collision resistance of this construction is $2^{n/2}$. Nandi (Nandi, Lee, Sakurai, & Lee, 2005) proposed a 2/3-rate double length compression function which takes n inputs and produces $2n$ outputs. Nandi (Nandi, Lee, Sakurai, & Lee, 2005) proved that

the complexity of collision attack is $2^{2n/3}$, which is more secure than the Merkle-Damg°ard construction. It is important to note that these constructions are designed based on Merkle-Damg°ard construction (which takes n bit input and produces $2n$ output) by using two different compression functions. On the other hand, our construction is based on sponge construction which takes an arbitrary length input and produces a variable length output by using the same compression function without the need of extra hardware. In this paper, general descriptions of the related work are firstly provided. Next, our construction called DLP sponge is explained in Section 3. The security analysis of DLP sponge is given in Section 4 and we conclude our work in Section 5.

RELATED WORK

Too many studies of Authenticated encryption has been declared in the last years. Block cipher modes clearly are the most famous way to provide both integrity and authenticity. Many block cipher modes have been proposed e.g., (Rogaway, Bellare, & Black, 2003) (Bellare, Rogaway, & Wagner, 2004) (Iwata, Tetsu, 2006) and (Iwata, Tetsu, 2008). Sponge construction is an iterative construction designed by Bertoni, G. et.al. (Bertoni, Daemen, Peeters, & Van Assche, 2007) and (Bertoni, Daemen, Peeters, & Van Assche, 2011a) that maps a variable length input to a variable length output. This feature make this construction to be suitable for many applications such as hash function, stream cipher, mask generation function, Message Authentication Code (MAC), and Authenticated Encryption (Borowski, 2013).

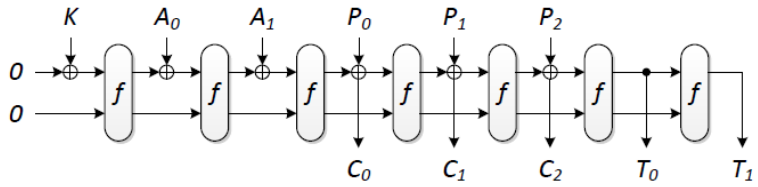

Figure 1: SpongeWrap (Bertoni, Diemen, Peeters, & Van 2011b).

The SpongeWrap construction (Bertoni, Daemen, Peeters, & Van 2011b) is designed for authenticated encryption as shown in Figure 1. Firstly, the key k is initialized and loaded into the state. Next, the header A padded and absorbed into the state. The message M is padded and divided into p blocks, and then the encryption (or decryption) runs in duplex mode (Yalçın, & Kavun, 2012). Another construction utilizing AE has been proposed such as Monkey sponge and Donkey sponge (Bertoni, Daemen, Peeters, & Van Assche, 2012). Both constructions have been designed for lightweight cryptography primitives (Bertoni, Daemen, Peeters, & Van Assche, 2012). The flexibility of choosing the parameters in this construction makes it more suitable to be used in AE. However, when $c = n$, this construction is weaker against generic attack. In order to use this construction for lightweight cryptographic application, we use the internal permutation b in two parallel line without effects on the cost. That's makes our construction more resistance agents generic attacks with $2^{2(c+3)\backslash 2}$ complexity instead of $2^{(c+3)\backslash 2}$.

Duplex Sponge construction is very similar to the design of the sponge construction. The main difference between the two designs is that in the former, there is no squeezing phase before the final digest is produced (Bertoni, Daemen, Peeters, & Van 2011b). It is designed especially for authenticated encryption purpose as shown in Figure 2. The k is deal in one iterated where $|K| = r$. Therefore, in case of small K, the permutation of size b is affected as well. Therefore, in our proposal, we make $|K| = 2r$. Which makes our construction more suitable and flexible to be used when small keys and capacity required.

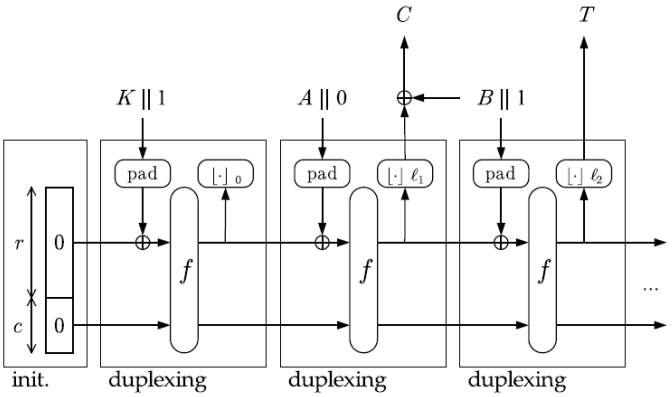

Figure 2: The duplex construction (Bertoni, Daemen, Peeters, & Van 2011b).

The "offset codebook" (OCB) mode that takes a ciphertext of arbitrary length has been reported in (Rogaway, Bellare, & Black, 2003). The message to be encrypted is *M* and the key is *K*. *M* can be any string and the key is just one block. In our construction, the *k* is divided into two sub blocks that work in parallel. According (Rogaway, Bellare, & Black, 2003), both storage and time can be saved if only one block is used. However, the size of the internal permutation is affected. OCB requires an *IV* called nonce. The nonce is needed in both encrypt and decrypt. In our construction, we consider the $c+r = 0$ as "nonce". OCB can provide a stream of cipher on the line as a stream of plaintext. It is unnecessary to know the length of plaintext; therefore, less hardware is needed in our construction.

DLP SPONGE CONSTRUCTION FOR AUTHENTICATED ENCRYPTION

DLP sponge construction was recently introduced by (Baraa, Norziana, Ezanee, Zaba, 2016). It is a construction that enhances the security of sponge construction by having double length and parallel chains going through two processes, namely the absorbing and squeezing processes. To be more specific, two *b*-bit compression functions *f* are used.

The two parallel chains are unchanged during the absorbing process. Subsequently, in the squeezing phase, the compression function *f* takes input from the two chains as *r* and *c* in order to produce the output as shown in Figure 3. The attractive features inherited in the double length and the wide pipe construction serve as the building block of our new design. The encryption process of DLP-Sponge for AE involves:-

- Padding: Append a single "*1*" and many "zeros" until the size of the message *M* been multiply the bit rate *r*.
- Initialization: Initialize the internal state $b = c + r$ to zero. This state is treated as nonce *N*.
- Absorbing phase: In the absorbing phase, *M* is divided into *t* blocks (=*r*). Here, the inputs of *f* and *f`* are $m_i$ and $m_i$`, respectively. It is important to note that the use of same *m* ensures that the outputs of two compression functions *f* and *f`* are similar. The decision of using the same/different compression functions *f, f `*is user-dependent. Meanwhile, the output of encryption is produced as ciphertext *C*.
- Squeezing phase: The internal state is updated and the output *n* is returned as the tag *T* for the message *M*. In the squeezing phase we will use different *r`* (its size is similar to that of Tag *T*) to ensure that the output is produced after one iteration and it is more resistant to pre-image attack. The Decryption Process differs from Encryption in such a way that the input is ciphertext *C* instead of message *M*. The Tag value is subsequently compared with that produced by the Encryption process. The decrypt message will be returned if no error is found.

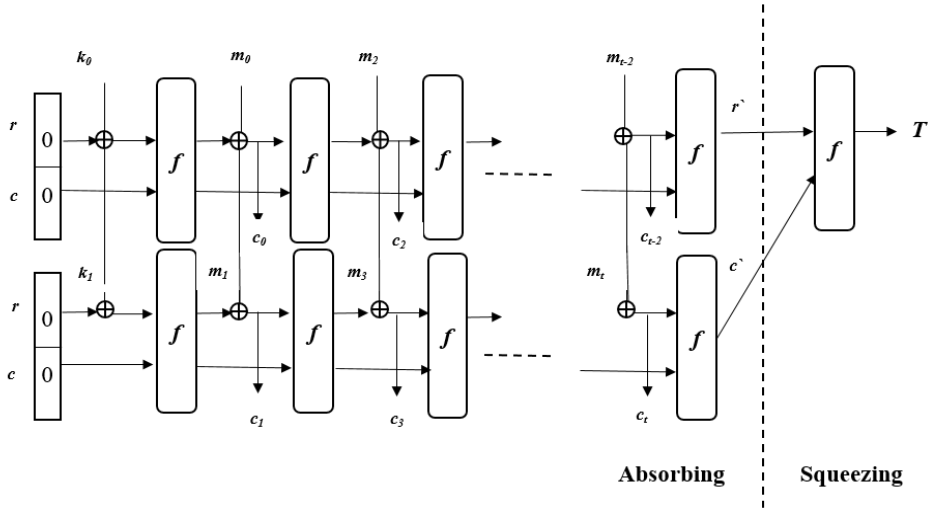

Figure 3: DLP sponge construction as Authenticated Encryption

Algorithm 1 :- DLP-sponge DLP-encrypt[H, M]	**Algorithm 2:-** DLP-sponge DLP-Decrypt[*H, C, T*]
Interface:(*C, T*) = DLP-encrypt(H,M, ℓ) with $H, M \in Z_2^*, \ell \geq 0, C \in Z_2^{\lvert M \rvert} and\ T \in Z_2^{\ell}$ $\boldsymbol{Let}\ H = H_0 \lvert\rvert H_1 \rvert\rvert \ldots \lvert\rvert H_v\ with\ \lvert H_i \rvert = p$ $\boldsymbol{for}\ i < v, \lvert H_v \rvert > p\ and\ \lvert H_v \rvert > 0\ if\ v > 0$ $\boldsymbol{Let}\ M = M_0 \lvert\rvert M_1 \rvert\rvert \ldots \lvert\rvert M_w\ with\ \lvert M_i \rvert = p$ $\boldsymbol{for}\ i < w, \lvert M_w \rvert > p\ and\ \lvert M_w \rvert > 0$ **if** w > 0 **for** $i = 0\ to\ v - 1$ ***do*** $DLP(H_i \lvert\rvert 0, 0)$ $Z = DLP(H_v \lvert\rvert 1, \lvert M_0 \rvert)$ $C = M_0 \oplus Z$ **for** $i = 0\ to\ w - 1$ ***do*** $s = s \oplus (M_i \lvert\rvert 0^{(b-r)})$ $s` = s` \oplus (M_i` \lvert\rvert 0^{(b-r)})$ $s = f(s)$ $s` = f`(s)$ $S = s \lvert\rvert s`$ **end for** $Z = [S]_r$ $C = C \lvert\rvert (M_{i+1} \oplus Z)$ **end for** $Z = DLP(M_w \lvert\rvert 0, p)$ **While** $\lvert Z \rvert < \ell$ ***do*** $Z = Z \lvert\rvert DLP(0, r)$ **end while** $T = \lfloor Z \rfloor_{\ell}$ **return** (*C, T*) **end if**	**Interface :** M = DLP-Decrypt(*H,C, T*)with $H, C, T \in Z_2^{\lvert c \rvert} \cup error$ $\boldsymbol{Let}\ H = H_0 \lvert\rvert H_1 \rvert\rvert \ldots \lvert\rvert H_v\ with\ \lvert H_i \rvert = p$ $\boldsymbol{for}\ i < v, \lvert H_v \rvert \leq p\ and\ \lvert H_v \rvert > 0\ if\ v > 0$ $\boldsymbol{Let}\ C = C_0 \lvert\rvert C_1 \rvert\rvert \ldots \lvert\rvert C_w\ with\ \lvert C_i \rvert = p$ $\boldsymbol{for}\ i < w, \lvert C_w \rvert > p\ and\ \lvert C_w \rvert > 0$ $\boldsymbol{Let}\ T = T_0 \lvert\rvert T_1 \rvert\rvert \ldots \lvert\rvert T_x\ with\ \lvert T_i \rvert = p$ $\boldsymbol{for}\ i < x, \lvert T_x \rvert > p\ and\ \lvert T_x \rvert > 0$ **for** $i = 0\ to\ v - 1$ ***do*** $DLP(H_i \lvert\rvert 0, 0)$ $Z = DLP(H_v \lvert\rvert 1, \lvert C_0 \rvert)$ $M = C_0 \oplus Z$ **for** $i = 0\ to\ w - 1$ ***do*** $s = s \oplus (C_i \lvert\rvert 0^{(b-r)})$ $s` = s` \oplus (C`_i \lvert\rvert 0^{(b-r)})$ $s = f(s)$ $s` = f`(s)$ $S = s \lvert\rvert s`$ **end for** $Z = [S]_r$ $M = M \lvert\rvert (C_{i+1} \oplus Z)$ **end for** $Z = DLP(C_w \lvert\rvert 0, p)$ **While** $\lvert Z \rvert < \ell$ ***do*** $Z = Z \lvert\rvert DLP(0, r)$ **end while** $T = \lfloor Z \rfloor_{\ell}$ **return** ($M_0 \lvert\rvert M_1 \rvert\rvert \ldots M_w$) **else return** Error **end if**

In principle, encryption and authentication processes are not performed concurrently in resource constrained devices due to the limitations of processing power capability and memory. Therefore, authenticated encryption is usually adopted where the same primitive performs both functions such as that shown in (Yalçın, & Kavun, 2012). Algorithm1 and Algorithm 2 have been used for encryption and decryption in DLP sponge, respectively.

Encryption $E= Z_2^k \times (Z_2^*) \rightarrow Z_2^* \times Z_2^t : (K, H, M) \rightarrow (C, T)$
Decryption $D=Z_2^k \times (Z_2^*) \times Z_2^t \rightarrow Z_2^* \cup \{error\}: (K, H, C, T) \rightarrow M \text{ or } error$

In order for DLP sponge to serve as an AE, duplex construction (Bertoni, Daemen, Peeters, & Van 2011b) is firstly performed, followed by the tag production. The challenge is how to deal with small keys, and how and where to use these keys. In the keyed sponge, the indifferentiability limits suggests that the size of key k is twice of the capacity c (Bertoni, Daemen, Peeters, & Van Assche, 2011c). With the underlying permutation $b = c+r$, decreasing capacity will affect the bit rate. Therefore, this motivates us to propose a new construction where $k = 2r$ (independent of the size of b). This makes DLP sponge construction flexible for many applications.

SECURITY ANALYSIS

In this section, the security of DLP sponge against generic attacks is demonstrated. Like random oracle, DLP sponge involves inner collisions. Here, an upper limit for the success probability is imposed in order to distinguish a DLP sponge from a random oracle. This limits includes an attacker that send calls to f and f^{-1} and has the permits to replace a random oracle by a DLP sponge.

Inner Collisions

The two main concepts in DLP sponge construction are state collision and inner collision as shown in Figure 4. A state collision involves a pair of different messages $M \neq M`$ under the same state $S_f[M] = S_f[M`]$. The state collisions acquired during the absorbing phase it could result to similar hash function values $S_f[M] = S_f[M`]$. The squeezing phase produces the same output values $S_f[M \,|0^j] = S_f[M` \,|0^j]$ for all j.

An inner collision involves a pair of two different messages $M \neq M`$ under the same inner state $S_{c,f}[M] = S_{c,f}[M`]$. If a state collision on $M \neq M$ exists, then an inner collision on $M \neq M`$ exists as well. However, the reverse is not true. The state collision for $M \neq M`$ can be produced through inner collisions such that $S_{c,f}[M] = S_{c,f}[M`]$.

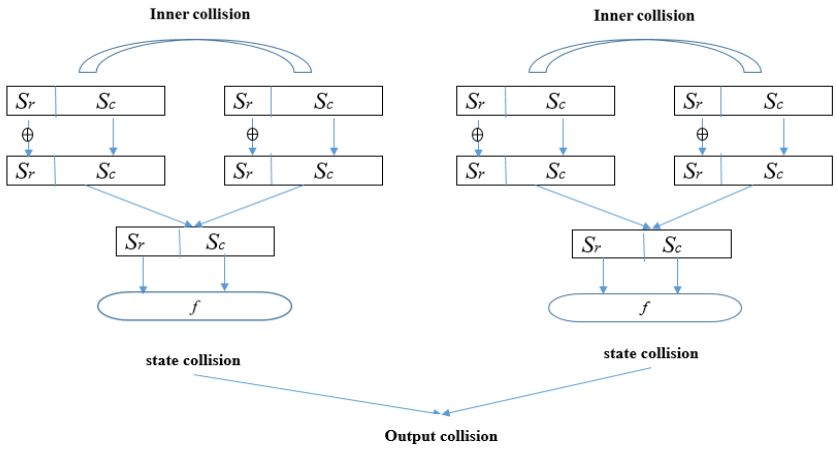

Figure 4. Output collision in DLP sponge

In case that we get inner collision p and q, then we have state collision with $p|a$, $q|b$, for any a and b that satisfy $S_{R,f}[p]+a = S_{R,f}[q]+b$. Then, any pairs of input $p|a|m$, $q|b|m$ lead to an output collision which is distinct of the digest length n. In DLP sponge, the complexity to produce an output collision is $2^{2(c+3)/2}$. In order to produce an output collision, the complexity of random oracle truncated to n bits and sponge construction are $2^{(n+3)/2}$ and $2^{(c+3)/2}$, respectively. So, a sponge construction with $n < c$ and a random oracle truncated to n bits offer similar level of resistance against output collisions. On the other hand, the complexity of DLP sponge in generating output collision is twice of that required by sponge construction.

Many generic attacks have been introduced to iterated hash functions such as Multicollisions (Joux, 2004), second pre-images (Kelsey, & Schneier, 2005) and Herding attack (Kelsey, & Kohno, 2006). These attacks depend on generating inner collision in its success.

Distinguishing a DLP sponge from a Random Oracle RO

Indistinguishability is the ability to differentiate two systems such as concrete construction and ideal systems. An attacker sends calls to both systems and decides the location of the concrete construction. Meanwhile, the concrete construction can replace the ideal system in application without any indication of losing security (Bertoni, Daemen, Peeters, & Van Assche, 2011c).

In this section we follow the distinguishing analysis of sponge construction to prove the security of the DLP sponge (Bertoni, Daemen, Peeters, & Van Assche, 2011a). We distinguish a DLP sponge from a random oracle *RO* for an attacker that does not have direct access to *f*. As shown in Figure 5, the left system combines *F* and DLP sponge *DLP [F]* and the right system is the *RO*. The attacker sends calls to *DLP* because he/she is not able to access *F* directly. This is refereed by *DLP* [*F*]. We refer to the interfaces by *DLP* [*F*] and *RO* by *E*. The interface *E* takes as an input the binary string $M \in \boldsymbol{Z}_2^*$ and an integer ℓ and produce an output of a string $Z \in \boldsymbol{Z}_2^{\ell}$.

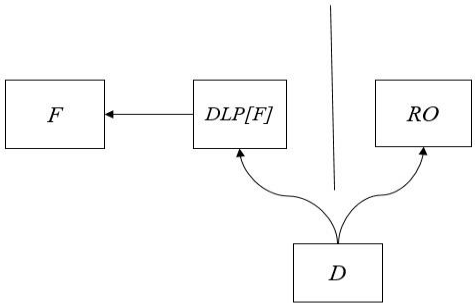

Figure 5. The distinguishing setting

An attacker is given unknown system *D* form distinguish. The success probability of *D* being either *RO* or *DLP* [*F*] can be written as:

$$pr\ (success) = \frac{1}{2} + \frac{1}{2} \sum_{D \in R_{KS}} \left(\Pr\left(D\left[DLP[F]\right]\right) - \Pr(D[RO])\right).$$

The attacker may send calls to *E* of *D*. These calls can be sent appropriate by consecutively asking the first ℓi bits of output for a set of messages $M^{(1)} \ldots M^{(t)}$. By using the response to all the calls that made by the attacker, he/she has to make a decision whether *D* is *RO* or *DLP* [*F*].

We represent to the cost of a calls to *D* by *N*, in case that *D* = *DLP* [*F*] then *N* is the total amount of calls to *F*, *N* is liable on the length of input *M* and the output length ℓ. For example, a query contributes $\lceil |M|+1_r \rceil + \lceil \ell_r \rceil$ to the cost.

The attacker is treated as a primitive *P* that produce 1 if it chooses *D* = *DLP* [*F*] and 0 otherwise. The success probability of the attacker is given by:

$$\frac{1}{2} Pr(P[DLP[F]] = 1) + \frac{1}{2} Pr(P[RO] = 0) = \frac{1}{2} + \frac{1}{2}(Pr(P[DLP[F]] = 1) - Pr(P[RO] = 0)).$$

The rightmost expression is identified the success probability.

$$Adv(P) = |Pr(P[DLP[F]] = 1) - Pr(P[RO] = 0)|.$$

Following the usual convention, the absolute value is taken. The performance of the attacker depends on the Call *C* he/she sends and the decision that been made. For a given sequence of Calls *C*, let $R(C)_{RS}$ denotes the set response sequences for which the attacker *P* guesses *D* is *DLP*[*F*]. Then for *C*, the probability that the attacker will return 1 if he/she addresses *DLP* [*F*] is

$$Pr(P[DLP[F]] = 1) = \sum_{D \in R(Q)_{RS}} Pr(D[DLP[F](C)]).$$

If *D* is *RO* the probability will return.

$$Pr(P[RO] = 1) = \sum_{D \in R(Q)_{RS}} Pr(D[RO(C)]) .$$

The advantage of *C* is

$$Adv(P, C) = \sum_{D \in R(Q)_{RS}} |Pr(D[DLP[F](C)]) \geq Pr(D[RO(C)]).$$

This value can be maximized by taking:

$$R_{KS} = \{D: \left(\Pr\left(D\left[DLP[F]\right]\right) \geq \Pr(D[RO])\right)$$

In that case

$$\sum_{D \in R_{KS}} \left(\Pr\left(D\left[DLP[F]\right]\right) - \Pr(D[RO])\right) = \frac{1}{2}\sum_{D} \left(\Pr\left(D\left[DLP[F]\right]\right) - \Pr(D[RO])\right) .$$

Yielding the following expression:

$$Adv(P, C) = \frac{1}{2} \sum_{D} | \Pr(D[DLP[F](C)]) - \Pr(D[RO(C)]) . \qquad (1)$$

To prove that RO is distinguishing from DLP sponge construction when calling *f* is highest rat by:

$$1 - e^{-\frac{N(N+1)}{2^{2c+1}} + \frac{N(N-1)}{2^{2b+1}}}$$

Let Pr*(IC|C)* refer to the probability that a series of calls, when sent to *DLP[F]* results in an inner collision. If there is no inner collision while the request is being sent, we have Pr(*D*[*DLP*[*F*](*C*)|no *IC*) = Pr(*D*[*RO*(*C*)]). It follows that

$$Pr(D[DLP[F](C)]) = Pr(D[DLP[F](C)] \,|IC)\, Pr(IC|C) + Pr(D[RO(C)])(1 - Pr((IC|C)).$$

Substituting this into Eq. (1) gives:

$$Adv(P, C) = \frac{1}{2} Pr(IC|C) \sum_{D} |Pr(D[DLP[F](C)|IC]) - Pr(D[RO(C)])| .$$

As $\sum_D |Pr(D[DLP[F](C)|IC]) - Pr(D[RO(C)])| \leq 2$, we can upper limit the advantage by $Adv(P, C) \leq Pr(IC|C))$.

Form the above equation, we can see that the success probability for generating an inner collisions is on the right side.

CONCLUSIONS

In this paper, we propose a new construction for Authenticated Encryption, i.e. the DLP sponge. It is designed to resolve issues related to small keys and sponge security when *c* = *n*. The security analysis of DLP sponge subjected to the generic attacks and the existing of inner collision has been performed. Then, an upper limit for the success probability has been introduced to distinguish a DLP sponge from a random oracle.

REFERENCES

Baraa Tareq Hammad, Norziana Jamil, M. Ezanee. R. , M. R. Zaba, (2016), DLP sponge construction, 2016, in press, 5 the International Conference of Business, Economics, Management, Information Technology and Social Science.

Bellare, M. & Namprempre C. (2000, December). Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In International Conference on the Theory and Application of Cryptology and Information Security (pp. 531-545). Springer Berlin Heidelberg.

Bellare, M., Rogaway, P., & Wagner, D. (2004, February). The EAX mode of operation. In International Workshop on Fast Software Encryption (pp. 389-407). Springer Berlin Heidelberg.

Bertoni, G., Daemen, J., Peeters, M., & Van Assche, G. (2007, May). Sponge functions. In ECRYPT hash workshop (Vol. 2007).

Bertoni, G., Daemen, J., Peeters, M., & Van Assche, G. (2011a). Cryptographic sponges. online http://sponge. noekeon. org.

Bertoni, G., Daemen, J., Peeters, M., & Van Assche, G. (2011b, August). Duplexing the sponge: single-pass authenticated encryption and other applications. In International Workshop on Selected Areas in Cryptography (pp. 320-337). Springer Berlin Heidelberg.

Bertoni, G., Daemen, J., Peeters, M., & Van Assche, G. (2011c). On the security of the keyed sponge construction. SKEW.

Bertoni, G., Daemen, J., Peeters, M., & Van Assche, G. (2012). Permutation-based encryption, authentication and authenticated encryption. Directions in Authenticated Ciphers.

Bilgin, B., Bogdanov, A., Knežević, M., Mendel, F. & Wang, Q. (2013, August). Fides: Lightweight authenticated cipher with side-channel resistance for constrained hardware. In International Workshop on Cryptographic Hardware and Embedded Systems (pp. 142-158). Springer Berlin Heidelberg.

Bogdanov, A., Mendel, F., Regazzoni, F., Rijmen, V., & Tischhauser, E. (2013, March). ALE: AES-based lightweight authenticated encryption. In International Workshop on Fast Software Encryption (pp. 447-466). Springer Berlin Heidelberg.

Borowski, M. (2013, October). The sponge construction as a source of secure cryptographic primitives. In Military Communications and Information Systems Conference (MCC), 2013 (pp. 1-5). IEEE.

Gligor, V. D., & Donescu, P. (2001, April). Fast encryption and authentication: XCBC encryption and XECB authentication modes. In International Workshop on Fast Software Encryption (pp. 92-108). Springer Berlin Heidelberg.

Hirose, S. (2006, March). Some plausible constructions of double-block-length hash functions. In International Workshop on Fast Software Encryption (pp. 210-225). Springer Berlin Heidelberg.

Iwata, Tetsu. (2006, March). New blockcipher modes of operation with beyond the birthday bound security. In International Workshop on Fast Software Encryption (pp. 310-327). Springer Berlin Heidelberg.

Iwata, Tetsu. (2008, June). Authenticated encryption mode for beyond the birthday bound security. In International Conference on Cryptology in Africa (pp. 125-142). Springer Berlin Heidelberg.

Joux, A. (2004, August). Multicollisions in iterated hash functions. Application to cascaded constructions. In Annual International Cryptology Conference (pp. 306-316). Springer Berlin Heidelberg.

Jutla, C. S. (2001, May). Encryption modes with almost free message integrity. In International Conference on the Theory and Applications of Cryptographic Techniques (pp. 529-544). Springer Berlin Heidelberg.

Kelsey, J., & Kohno, T. (2006, May). Herding hash functions and the Nostradamus attack. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 183-200). Springer Berlin Heidelberg.

Kelsey, J., & Schneier, B. (2005, May). Second preimages on n-bit hash functions for much less than 2 n work. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 474-490). Springer Berlin Heidelberg.

Kohno, T., Viega, J., & Whiting, D. (2004, February). CWC: A high-performance conventional authenticated encryption mode. In International Workshop on Fast Software Encryption (pp. 408-426). Springer Berlin Heidelberg.

Lemsitzer, S., Wolkerstorfer, J., Felber, N., & Braendli, M. (2007, September). Multi-gigabit GCM-AES architecture optimized for FPGAs. In International Workshop on Cryptographic Hardware and Embedded Systems (pp. 227-238). Springer Berlin Heidelberg.

Li, G., Xu, X., & Li, Q. (2015, July). LADP: A lightweight authentication and delegation protocol for RFID tags. In 2015 Seventh International Conference on Ubiquitous and Future Networks (pp. 860-865). IEEE.

Nandi, M., Lee, W., Sakurai, K., & Lee, S. (2005, February). Security analysis of a 2/3-rate double length compression function in the black-box model. In International Workshop on Fast Software Encryption (pp. 243-254). Springer Berlin Heidelberg.

Rogaway P., Bellare, M. & Black, J. (2003). OCB: A block-cipher mode of operation for efficient authenticated encryption. ACM Transactions on Information and System Security (TISSEC) 6(3) 365-403.

Yalçın, T., & Kavun, E. B. (2012, November). On the implementation aspects of sponge-based authenticated encryption for pervasive devices. In International Conference on Smart Card Research and Advanced Applications (pp. 141-157). Springer Berlin Heidelberg.

Zhang, L., Wu, W., Wang, Y., Wu, S., & Zhang, J. (2014). Lac: A lightweight authenticated encryption cipher. Submitted to the CAESAR competition.