1 Related Work

1.1 Basic concepts

1.1.1 Message Authentication

The message authentication system aims to provide protection on the integrity of messages. Assume the sender A sends message M to receiver B,the message authentication system be eligible to examine the modification on M. The concept of message authentication system is expressed in Figure 1. The sender uses the message as input to the tag generation system($TG_K(M)$) to generate a short information block called tag. The message is concatenated with tag and transmitted to the receiver. Before the receiver accept the message M, M and its tag T are sent to the verification system($VF_K(M,T)$). If the output of verification system is 1, that means the M and T are not matched, otherwise the message M is accepted by the receiver.

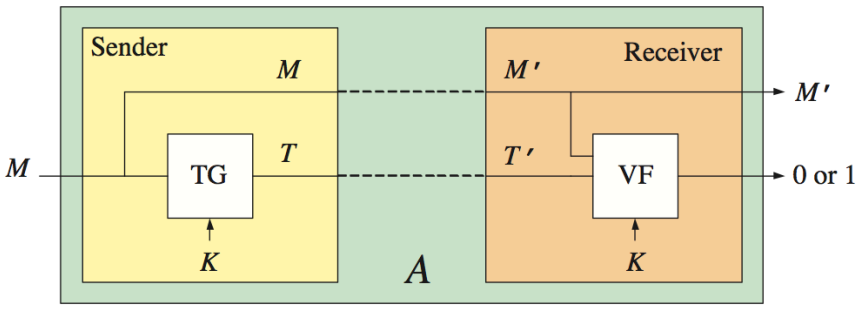

Figure 1: The Concept of Message Authentication System

Common Message Authentication Systems The Message Authentication Code(MAC) is a common message authentication system. The concept of MAC scheme can be seen in Figure 2. In a deterministic MAC scheme, the verification system adopts the same key used in tag generation system. In the verification part of a MAC scheme, the tag T1 of message M is computed and compared with the tag T concatenated with the M. If T1=T then the verification system

output 1 and the receiver accepts M, otherwise the verification system output 0. The early designed MAC schemes are deterministic, which means neither the sender nor the receiver needs to maintain a state used in tag generation. Latter some MAC designs adopt a state maintained by the user in the tag generation, such as the GMAC [26] and Cost-Effective Tag Design [16].

Digital signature is another kind of message authentication system. The signature generation uses a private key while the message verification stage uses public key. The digital signature system can assure non-repudiation of the message protected while MAC schemes can not.

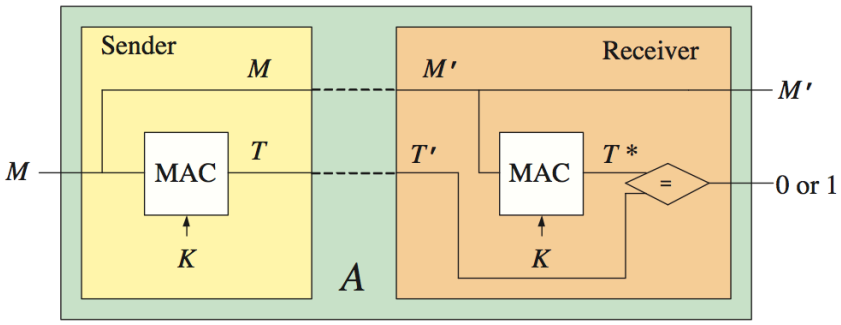

Figure 2: Message Authentication Code(MAC)

1.1.2 The Security of MAC schemes

The forgery attacks When attacking a message authentication system, the adversary try to send a pair(M,T) to the receiver to make VF_k(M,T)=1 while M did not originate with the legal sender. The fake pair(M_f,T_f) that makes VF_k(M_f,T_f)=1 is called a forgery from the adversary. A successful forgery attack indicates that the adversary has made a forgery. The purpose of a message authentication system is preventing the receiver to accept the message from unauthorized senders, such as an adversary. The quantitative property of a secure message authentication system is the low probability for an adversary to make a successful forgery attack with the limited resource.

Chosen-message attacks A strong type of attack that an adversary can conduct on the message authentication system is the adaptive chosen-message attack, marked as uf-cma. When doing uf-cma, the adversary chooses its own input message M and acquires the relative tag T. The adversary try to find the weakness in the design of message authentication system by analyzing the pairs(M,T) of his choice. The uf-cma provides the adversary with the most capability to succeed in the forgery attack. The probability that an adversary conducts a successful forgery attack after limited times of uf-cma is adopted as the basic quantitative security property of a message authentication in cryptography. This fact was also mentioned in [32].

The Security Notions of MAC schemes The formalised quantitative notion of the security of a MAC scheme was introduced by Bellare et al. in [4]. This notion follows the security notion of digital signature introduced in [14]. The successful forgery on a MAC scheme from an adversary A is measured by a experiment called Forgery(MAC,A). In Forgery(MAC,A), the adversary A is provided a black-box access to the tag generation system $\mathrm{TG}_K()$. When $\mathrm{TG}_K()$ takes an input message M_i, it returns tag T_i to A. A conducts uf-cma by keep sending the message queries M_i and observes the relative tag T_i for limited times. On the other hand, A is provided a black-box access to the verification system $\mathrm{VF}_K()$. When A sends a pair(M_j,T_j) to $\mathrm{VF}_K()$, the $\mathrm{VF}_K()$ computes the tag T of M_j and compares T with T_j. If T=T_j then $\mathrm{VF}_K()$=1 otherwise 0. If A sends a pair(M,T) that makes $\mathrm{VF}_K()$ outputs 1 while M has not appeared in the previous queries of uf-cma, then A succeeds a forgery attack and Forgery(MAC,A)=1.

The quantitative security notion of a MAC scheme is forgery probability, expressed as $\mathrm{Forgery}_M AC$=Pr[Forgery(MAC,A)=1].

The Correlation between Security and Randomness Goldreich, Goldwasser, and Micali asserted in [13] that any good pseudorandom function(PRF) is a secure MAC scheme under the quantitative security notion. Bellare, Kilian and Rogaway proved this assertion in [4] saying that if a system behave like a pseudoranom function, this system is a secure MAC scheme if meeting the requirements on domain and range of MAC schemes. Based on these two reduction of security notion, latter researches on security evaluation of MAC schemes posted their focuses on analyzing whether the MAC scheme evaluated behaves like a PRF.

The Randomness of a MAC scheme The definition of PRF was introduced in [13] indicating that PRF could not be distinguished from a ideal random function each bit of whose output was a coin flip. To define how closely a MAC scheme behaves like a PRF, Bellare et al. provided a quantitative notion in [4] named $\mathrm{Adv}^{PRF}_{MAC}()$, which was based on the concept of distinguisher introduced in[13].

Let F_0 and F_1 be two function with a common domain D and a common range R. A distinguisher A for F_0 versus F_1 is an adverary A that has access to a black box named oracle f:D-¿R. After accessing the oracle f, A computes a bit. Assume the function stored in the oracle f is X and A guesses that X is in the oracle, then A computes 1 otherwise 0. The the advantage of A in distinguishing F_0 from F_1 is expressed as Adv_{F1}^{F0}=Pr[f$\xleftarrow{R}$F0:A^{F0}=1]-Pr[f$\xleftarrow{R}$F1:A^{F1}=1]. Pr[f$\xleftarrow{R}$F0:A^{F0}=1] means when the content of oracle f is F0, A guesses that F0 is in oracle then output 1.

We can see that if F0 behaves much like F1, it is hard for A to distinguish between F0 and F1 then Adv_{F1}^{F0} is very small. This case is adopted by Bellare et al. in the quantitative notion of randomness of a MAC scheme. If the randomness of a MAC scheme is good, then the MAC scheme behaves like a PRF and $\mathrm{Adv}_{MAC}^{[PRF]}$ is small.

1.2 The deterministic MAC schemes

Properties of the Deterministic MAC Schemes The early MAC schemes are deterministic systems, which means the sender and receiver do not need to maintain a state for tag generation. The deterministic MAC schemes are symmetric system as the key used by the TG and VF is the same one. Some of the deterministic MAC schemes are designed based on block cipher, keyed cryptographic hash functions or other cryptographic primitives.

1.2.1 Security Evaluation of the Deterministic MAC schemes

The Computational approach of Security evaluation for MAC schemes The approach adopted by researchers in analyzing the security of deterministic MAC schemes can be described as the following procedure: expressed Adv_{MAC}^{PRF} with a equation consist of $\mathrm{Adv}_{E(K)}^{PRP}$, where E(k) represents the block cipher with key k used in the MAC scheme. Hence the quantitative security result Adv_{MAC}^{PRF} is based on the result $\mathrm{Adv}_{E(K)}^{PRP}$, and $\mathrm{Adv}_{E(K)}^{PRP}$ is defined by the computational assumption in cryptography, this approach in computing the quantitative security result is called computational approach by some researchers.

PRP represent pseudorandom permutation, which is introduced by Luby and Rackoff‘in [25] claiming that a good block cipher should behave like a pseudorandom permutation(PRP) whose output should be uniformly and randomly generated and their values should be distinct if the inputs are distinct. According to the assumption in cryptography, the ideal block cipher behaves exactly as pseudorandom permutation. The quantitative security analysis of deterministic MAC schemes are based on this assumption in the calculation of Adv_{MAC}^{PRF}.

The Experiment based Randomness Evaluation According to the computational approach of security analysis on deterministic MAC schemes, the security of a MAC scheme is based the assumption that the block cipher or

other cryptographic primitives(such as cryptographic hash function in HMAC) show high level randomness.

There have been researches on designing tools for randomness evaluation to provide empirical evidence of randomness assumption in security analysis. Early works included DIEHARD suite, Crypt-XS, The NIST Statistic Test Suite(short for the NIST suite) and so on. Soto introduced a randomness testing suite designed by the National Institure of Standard Technology in [38]. The properties of the NIST suite is analyzed in detailed in [34] together with the instruction on using the NIST suite for randomness testing. The majority advantages of NIST suite are:

- The NIST suite covers most hard-to-pass tests for binary sequences mentioned in previous suites(such as DIEHARD or Crypt-XS)
- The purpose of each statistic tests is explained in detail to show what kind of defect of the input is examined.
- NIST suite has been used in candidate selection of Advanced Encryption Standard(AES). The results acquired under the result collection and analysis instruction of NIST suite is convincing.

The NIST suite has be adopted in the candidate selection of Advanced Encryption Standard(AES). The detailed information about selection is discussed in [37, 39].

1.2.2 Iterated MAC schemes

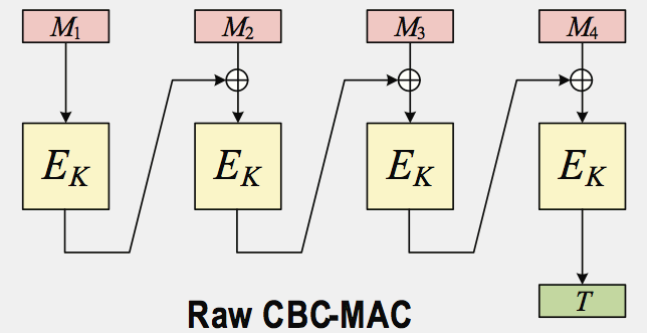

Figure 3: The Raw CBC-MAC

The research on iterated MAC schemes started from the raw CBC-MAC expressed in Figure 3.

Raw CBC-MAC The cryptographic primitive used in the raw CBC-MAC scheme is block cipher. The raw CBC-MAC is a secure MAC for only fixed length inputs. Assume the tag T of message M from CBC-MAC can be expressed as T = CBC-MAC$_k$(M), for another input M$\|$(M$\oplus$T), the tag T1=CBC-MAC(M$\|$(M$\oplus$T))=T. This attack indicates that the raw CBC-MAC is vulnerable if the input length is not fixed. Besides the vulnerability when the input length is not fixed, the raw CBC-MAC scheme suffers birthday attack, means the adversary needs only $2n/2$ input queries to succeed a forgery.

Bellare et al. provided the original security evaluation on raw CBC-MAC[4]. The conclusion is that if the adversary A is allowed to conduct arbitrary fixed length queries for q time, the probability that A succeeds in a forgery after the queries can be expressed with the queries times q and the computational assumption of block cipher used. Their quantitative conclusion showed that the forgery probability is very small.

EMAC The motivation of EMAC design is the problem that the raw CBC-MAC is secure only for fixed length inputs. EMAC is a optimized version of raw CBC-MAC providing security for arbitrary length inputs. The in the EMAC scheme, the final message block in the input is processed by a padding function to ensure that all the input blocks have the same length. Then raw CBC-MAC is applied applies and the result is encrypted by a the same block cipher in raw CBC-MAC with a different key.

Petrank and Rackof provided the original security evaluation of EMAC in [29]. The EMAC is secure for arbitrary length inputs under the security notion if the block cipher meets the related computational assumptions. However, the EMAC still suffers the birthday attack.

1.2.3 CMAC: Optimized CBC-MAC

Hence the raw CBC-MAC is vulnerable if the input length is not fixed, a class of MAC schemes named CMAC are developed to provide security for arbitrary length of inputs. The structure of the variants of CMAC is modeled in Figure 4.

XCBC: three key version of CMAC Black and Rogaway introduced a optimized version of EMAC called XCBC in [7]. There are two motivations of XCBC scheme: providing security for arbitrary length input; eliminate unnecessary padding operation on the input blocks in EMAC.

The XCBC scheme is a refined version of original EMAC. The contributions of XCBC scheme compared with the EMAC are: extending the domain to full domain; the block ciphers used in processing m input blocks is m other than m+1 in EMAC; all the block cipher in XCBC share a same key, the additional two keys are used to do exclusively or operation with the final input block,this design has faster processing speed.

Black and Rogaway provided a original security analysis based on computational approach of XCBC scheme in [7]. Their quantitative result was improved

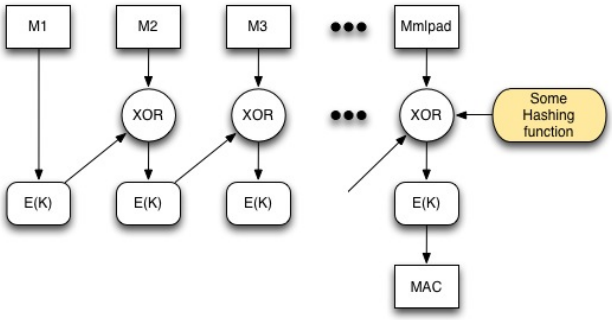

Figure 4: The class of CMAC

by Minematsu and Matsushima in [27]. Why the japanese want to optimize the original security bound? Any flaw in the proof? What is the advantage in the new proof procedure?

TMAC: two key version of CMAC Kurosawa and Iwata designed a optimized version of cmac requiring two different keys in [22] named TMAC. The two different keys used in processing the final input blocks in XCBC are replaced by a keyed hash function with two distinct constants as inputs. This optimization aims to reduce the key cost.

Kurosawa and Iwata provided an original security analysis on TMAC in [22] showing that the same level of security was achieved by TMAC compared with XCBC. The original quantitative security conclusion of TMAC was improved by Minematsu and Matsushima in [27].

OMAC: single key version of CMAC Iwata and Kurosawa introduced a optimization of original XCBC scheme with only one key, named OMAC in [17]. This key is used by the block cipher in the OMAC and the keyed hash function used in TMAC is replaced by a result of Galois field multiplication. OMAC is the variant of CMAC using the least number of keys.

Iwata and Kurosawa provided an original security analysis on OMAC in [17]. This result have no optimized version yet.

other iterated MAC designs Daemen and Rijmen introduced a iterated MAC design named ALRED-MAC in [12]. The motivation of of this work is the limitation of quantitative security result for iterated MAC scheme due to the birthday paradox.

1.2.4 HMAC and MAC design based on keyed hash function

MAC schemes based on cryptographic hash functions are adopted more in Internet community. Bellare, Canetti, and Krawczyk designed two MAC schemes in [2], the nested construction NMAC and the hash based MAC scheme HMAC. The advantage of HMAC compared with MAC schemes based on block cipher is the fast processing speed and simpler in implementation. In this paper, the authors analyzed the security of NMAC and HAMC and provided a quantitative security result based on the security notion that HMAC is secure is the hash function used behaves like a PRF.

1.2.5 Parallel MACs

The motivation of designing MAC schemes in which input message blocks are processed in parallel is the latency of processing when using iterated MAC schemes. This latency becomes more obvious when run the scheme on processors supporting pipeline of parallelism.

XOR MAC scheme Bellare, Guerin and Rogaway introduced a parallel MAC scheme named XOR MAC in [3]. The parallel input processing makes the xor MAC faster the CBC-MAC. In [3], the authors provided a quantitative security conclusion of xor mac that it is security under the security notion of message authentication system if the pseudorandom function used in the design is secure. The security bound of xor mac from [3] expressed a smaller result compared with the result of raw CBC-MAC.

The main weakness of xor mac is the cost of storage for additional input parameters. According the design of xor mac, assume the input message is divided into m blocks and the length of each block is half of the length of block cipher input. Each input block has a index with domain as [1,m]. A input block is the concatenated with the encoder of its index(whose length is same as the input block) to form the input to a block cipher, for example (i$\|$M[i]). Besides the m concatenated input blocks, a nonce formed with random number or counter is used a input to the block cipher. Then the m+1 output blocks from block cipher is xored and the output block of xor operation is concatenated with a random number or counter to form the final tag. We can see that the length of tag from xor-mac is the sum of the length of a output block from block cipher and the length of a random number or counter. This long-length MAC needs additional storage compared with the tag from iterated MAC schemes. On the other hand, the nonce maintained by the user of xor mac is regarded a disadvantage by some researchers.

Raw PMAC Black and Rogaway introduced an parallel MAC scheme named PMAC in [8]. The improvement of PMAC compared with XOR MAC is the less calling of block cipher when generating a tag. Secondly, there is no limitation for the input length when using PMAC. Another benefit brought by PMAC is

that only one key is needed in generating the tag. Other processing operations include bit-level xor and gray code, which are cost-effective and fast processing.

The quantitative security result of PMAC was acquired by Black and Rogaway in [8]. They analysed the probability of internal collision in the arbitrary types of inputs and asserted that the PMAC design is a secure MAC if the block cipher used behaves like a pseudorandom permutation. In the proof from Black and Rogaway, the probability that an adversary can distinguish the PMAC from pseudorandom function is the probability that collsion of internal blocks(the outputs of block cipher) plus the probability of tag collision.

Lee et al. posted attacks on the raw PMAC in [23]. Their research indicated that the raw PMAC design did not bring advantage in security compared with iterated MACs. The raw PMAC suffers from their birthday like attack and the key recover attack.

Tweakable block cipher and Optimized PMAC The concept of tweakable block cipher was formally defined by Liskov, Rivest, and Wagner in [24]. Rogaway introduced a efficient implementation of tweakable block cipher in [31]and adopt this implementation to replace the block cipher used in the original PMAC and OCB authenticated encryption mode. The application of tweakable block cipher in constructing PMAC can enhance the processing speed compared with original PMAC based on Grey code and simplify the structure of the scheme, which also simplifys the security analysis of the scheme .

In [31], Rogaway provided an assertion that the tweakable block cipher XE and XEX behave like a pseudorandom permutation if the block cipher inside is a pseudorandom permutation. Based on this assertion about security of tweakable block cipher, Rogaway gave an quantitative conclusion that the new PMAC based on tweakable block cipher is security under the security notion of message authentication system. This quantitative security result of tweakable block cipher based PMAC was provided in [31]. Minematsu and Matsushima optimzed this result in [27].

The evaluation procedure in [8] is questioned by the latter researchers hence the correlation between collision probabilty and distinguishing probability was not explained . This issue was also pointed in Nandi‘s improved analysis on PMAC in [28]. Nandi also provided a improved quantitative security result of PMAC which is smaller in all the cases than the one in [8] and [27].

iPMAC Sarkar introduce an optimisation on original PMAC in [36]. The galois field multiplication used in the tweakable block cipher introduced is replaced by a technique named tower field representation of Galois field. This replacement enhance the processing speed in software.

Hence the structure of iPMAC is same as the original PMAC except the input masking stage, the security evaluation by the author in [36] followed the approach in [8].

1.3 Non-deterministic message authentication systems

GMAC in GCM McGrew and Viega introduced the Galois/Counter Mode authenticated encryption design and the original security analysis is provided in [26]. The message authentication system in GCM(named GMAC) is a iterated scheme based on the concept of universal hashing. The block processing component used in GMAC is the Galois field multiplication other than the block cipher in deterministic MAC schemes. The motivation of GCM is to design a scheme combining counter mode of operation(CTR) with a message authentication code(the GMAC) to form a efficient and secure authenticated encryption system. One advantage of adopting the Galois field multiplication in GMAC is that it can be made easily in hardware and has efficient performance in software.

In the security analysis of GCM from McGrew and Viega, the GMAC was not discussed alone. The security of GMAC as a message authentication code is based on the fact that the encryption part in GCM is secure and the collision probability of ciphertext blocks is low. On the other hand, the IV should not be reused.

In [18], Iwata, Ohashi and Minematsu provided an optimised analysis for the security of GCM. They pointed out the weakness in the lemma used in forming the bound of the quantitative result of security for GCM with a counter example. This counter example was developed to a distinguishing attack on GCM. Iwata et al. then provided a approach fixing the problem of original lemma and provided the new quantitative security result of GCM.

In [32], Rogaway pointed the weakness of GMAC as a MAC scheme when used alone that the security under adaptive chosen-message attack is not as good as the security of deterministic MAC schemes. On the other hand, GMAC requires the nonce for tag generation and this nonce should be maintained and refreshed by the user of GMAC. The potential issue of this design pointed by Rogaway is the reuse of nonce, which may lead to the collision of the tag.

Handschuh and Preneel introduced key-recovery attacks on universal hash function based MAC algorithms in [15]. They introduced two types of attacks: weak key finding and partial information leaks.

Cost-Effective Tag Design Hong, Guo and Hu introduced a parallel MAC design in [16]. In their design, the inputs are ciphertext blocks(called encrypted cache line in this paper). A nonce is generated by encrypting a (address,counter,random) pair and then used in controlling the tag generation. The pair used in generating the nonce is consist of the virtual address of the ciphertext blocks, a random number and a counter. The inputs are processed in two stages: bit segment swap between 2 input blocks for several rounds and the block rotate shift. The parameters of each stage are segements in the nonce.

The Authenticated Encryption Schemes After various kind of MAC schemes were proved to be secure, a new cryptosystem, the Authentecated Encryption(AE) was introduced The Authenticated to combine the encryption of plaintext blocks and generation of the MAC in a single scheme to provide both con-

fidentiality and integrity. The methodology of constructing a AE scheme was categorized by Bellare in [5]. In this paper, the author claimed that according to the 3 types of modes defined, the most secure one was the Encrypt-then-MAC mode. In Rogawa's works [30] a systematical analysis about AE schemes using associate data was expressed.

Based the notion of proof in [30], the security of several AE schemes in Encrypt-then-MAC mode have been proved, including CCM [19]based on CBC-MAC,EAX[6] based on OMAC, GCM [26] based on universal hashing(new bound revised in [18]), and OCB[33] based on PMAC(new bound revised in [24, 36]). Kasper and Schwabe introduced a implementation of AES that run faster than previous ones and adopted this implementation to construct a faster AES-GCM AE scheme[20].

There were researches on attacks to existing cryptographic algorithms, such as [35, 40, 21]. Some of them introduce the inspiration of revised proof while some others were out of the discussion bound in the proof works. This topic in beyond this paper and will not be discussed in detail.

1.4 Security analysis frameworks and techniques: how to evaluate the security

1.4.1 The computational approach

As mentioned before, researches on the security analysis of deterministic MAC schemes are based on the computational approach. The adversary needs to interact with some oracles(usually the tag generation oracle and verification oracle). In a limited time, the adversary ask arbitrary distinct inputs and observe the related tags. This behavior is usually called adaptive chosen-message attack. After the limited queries, the adversary needs to make a fake (message, tag) pair. If this pair can pass the verification stage, the adversary wins the forgery game. The security notion of a authentication system is defined as: Pr[adversary wins] is extremely low. In the security notion of computational approach, the computation time and probability is of great importance: the adversary can always win a game in an unlimited time while in a limited time with a low probability.

The deterministic MAC schemes are designed based on some cryptographic primitives, such as block cipher. When conducting a computational security evaluation on a message authentication system, the security conclusion is based on the computational assumption that the primitive used is secure. So the computational security evaluation is called reductionist proof due to the notion reduction.

The existing security evaluation through computational approach are done manually. This manually evaluation is regarded as tedious and error prone by some researchers.

1.4.2 The symbolic approach

Besides the computational framework of security evaluation for MAC schemes, there is another framework adopted by some researchers named Dolev-Yao model which is based on the concepts of formal methods. This framework is also called symbolic approach of security evaluation. In the framework of Dolev-Yao model, the cryptographic primitives used in the system is modeled as black boxes and have definitely security. This definitely security restrict the power of the adversary, which means not all the attacks on the MAC scheme can the symbolic approach analyzes.

1.4.3 Use the best of Two Frameworks

As discussed before, each of the two evaluation frameworks has weakness: the computational approach can provide complete and concrete securit result but the evaluation procedure is tedious and error prone, and this issue is reflected by the fact that most of the original quantitative security results of deterministic MAC schemes have been optimized; the symbolic approach is simple and can be conducted automatically while the security notion in symbolic approach is too strict and unrealistic, which means some security properties exist but cannot be evaluated.

Researches have been conducted to span the gap between the two frameworks, aiming to use the tools and methods in symbolic approach to get same level of security results of computational approach. Abadi and Rogaway demonstrated in [1] that it is possible to design security evaluation mechanisms with the advantages from both two frameworks. Cortier, Kremer and Warinschi provided a summary of approaches aiming to adopt the benefits of both symbolic and computational security analysis frameworks in [11]. They categorized the researches on spanning the gap between computational and symbolic security into two aspects: the computational soundness approach and direct approach.

The Computational Soundness Approach According to the discussion in [11], the researches on the computational soundness approach aim to design mechanisms assuring that the security results acquired with techniques from symbolic approach can lead to the security results in the viewpoint of computational approach without significant gap.

The Direct Approach Cortier et al. described the direct approach as "applying symbolic techniques directly to obtain computational security guarantees, without making use of abstract models". The mechanisms in direct approach aim to replace the rules in symbolic framework with the security notion from computational framework to acquire automated security evaluation schemes that produce computational security conclusion. Some mechanisms in the direct approach such as CryptoVerf have been applied to evaluate the security of message authentication systems.

The CryptoVerf System Blanchet designed a automated security analysis tool named CryptoVerf in [10] to evaluate cryptographic systems. Its optimzed version in [9] has been applied in analyzing the security of signature systems(Full-Domain Hash Signature). The motivation of CryptoVerf can be summarized below:

1. The assumptions adopted in the evaluation are computational assumptions used in reductionist proofs

2. The evaluation procedure is based the formal methods to achieve automation.

The computational assumption is more closed to realistic condition than the assumption used in original formal methods(Dolev-Yao model). The motivation of CryptoVerf is taking the advantage of pros in both frameworks to automatically evaluate the security of a cryptographic system in a realistic scenario.

The proof procedure is modeled as a sequence of games. The security of cryptographic system evaluated is modeled as the first game. The computational assumption for the adversary to break is modeled as the final game. If the adversary can never win the final game in the limited resource, the cryptographic system evaluated is assumed to be secure. This idea is similar to the security notion that a MAC scheme is secure if the cryptographic primitives used(such as block cipher and cryptographic hash function) is secure.

There are two inputs required by CryptoVerf: the formal definition of security of cryptographic primitive(such as block) use in the cryptographic system; and the formalization of the cryptographic system in the random oracle model. The CryptoVerf needs to model the cryptographic primitive used and the system itself to a formal. If the behaviour of the primitive has not been modeled, the evaluation results is not accurate. This issue may occur in new designed systems, such as the bit segment swapping and block rotate shift in the Cost-Effective Tag Design[16].

References

[1] Martin Abadi and Phillip Rogaway. Reconciling two views of cryptography (the computational soundness of formal encryption). In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 1872 LNCS, pages 3–22, Sendai, Japan, 2000.

[2] M Bellare, R Canetti, and H Krawczyk. Keying hash functions for message authentication. In *Advances in Cryptology - CRYPTO'96. 16th Annual International Cryptology Conference. Proceedings*, pages 1–15, Berlin, Germany, 1996.

[3] M Bellare, R Guerin, and P Rogaway. XOR MACs: New methods for message authentication using finite pseudorandom functions. In *Advances*

in Cryptology - CRYPTO '95. 15th Annual International Cryptology Conference. Proceedings, pages 15–28, Berlin, Germany, 1995.

[4] M Bellare, J Kilian, and P Rogaway. The security of cipher block chaining. In Yvo Desmedt, editor, *Advances in Cryptology Crypto 94*, volume 839 of *Lecture Notes in Computer Science*, pages 341–358. International Association for Cryptologic Research, Springer-Verlag, 1994.

[5] M Bellare and C Namprempre. Authenticated encryption: relations among notions and analysis of the generic composition paradigm. In *Advances in Cryptology - ASIACRYPT 2000. 6th International Conference on the Theory and Application of Cryptology and Information Security. Proceedings (Lecture Notes in Computer Science Vol.1976)*, pages 531 – 45, Berlin, Germany, 2000.

[6] M Bellare, P Rogaway, and D Wagner. The EAX mode of operation. In *Fast Software Encryption. 11th International Workshop, FSE 2004. Revised Papers (Lecture Notes in Comput. Sci. Vol.3017)*, pages 389–407, 2004.

[7] J Black and P Rogaway. CBC MACs for arbitrary-length messages: the three-key constructions. In *Advances in Cryptology - CRYPTO 2000. 20th Annual International Cryptology Conference. Proceedings (Lecture Notes in Computer Science Vol.1880)*, pages 197–215, Berlin, Germany, 2000.

[8] J Black and P Rogaway. A block-cipher mode of operation for parallelizable message authentication. In *Advances in Cryptology - EUROCRYPT 2002. International Conference on the Theory and Applications of Cryptographic Techniques. Proceedings (Lecture Notes in Computer Science Vol.2332)*, pages 384 —- 97, 2002.

[9] B Blanchet and D Pointcheval. Automated security proofs with sequences of games. In *Advances in Cryptology - CRYPTO 2006. 26th Annual International Cryptology Conference. Proceedings. (Lecture Notes in Computer Science Vol. 4117)*, pages 537 – 54, Berlin, Germany, 2006.

[10] Bruno Blanchet. A computationally sound mechanized prover for security protocols. In *Proceedings - IEEE Symposium on Security and Privacy*, volume 2006, pages 140–154, Berkeley, United states, 2006.

[11] V Cortier, S Kremer, and B Warinschi. A Survey of Symbolic Methods in Computational Analysis of Cryptographic Systems. *J. Autom. Reasoning (Netherlands)*, 46(3-4):225 – 59, 2011.

[12] J Daemen and V Rijmen. A new MAC construction ALRED and a specific instance ALPHA-MAC. In *Fast Software Encryption. 12th International Workshop, FSE 2005. Revised Selected Papers (Lecture Notes in Computer Science Vol. 3557)*, pages 1–17, Berlin, Germany, 2005.

[13] O Goldreich, S Goldwasser, and S Micali. How to construct random functions. *J. Assoc. Comput. Mach. (USA)*, 33(4):792–807, 1986.

[14] S Goldwasser, S Micali, and R L Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.

[15] H Handschuh and B Preneel. Key-recovery attacks on universal hash function based MAC algorithms. In *Advances in Cryptology - CRYPTO 2008. 28th Annual International Cryptology Conference*, pages 144 —- 61, 2008.

[16] Mei Hong, Hui Guo, and Sharon X Hu. A cost-effective tag design for memory data authentication in embedded systems. In *Proceedings of the 2012 International Conference on Compilers, Architectures and Synthesis for Embedded Systems (CASES 2012)*, pages 17–26, 2012.

[17] T Iwata and K Kurosawa. OMAC: one-key CBC MAC. In *Fast Software Encryption. 10th International Workshop, FSE 2003. Revised Papers (Lecture Notes in Comput. Sci. Vol.2887)*, pages 129 —- 53, 2003.

[18] T Iwata, K Ohashi, and K Minematsu. Breaking and Repairing GCM Security Proofs. In *32nd Annual Cryptology Conference. Advances in Cryptology - CRYPTO 2012*, pages 31–49, Berlin, Germany, 2012.

[19] J Jonsson. On the security of CTR+CBC-MAC. In *Selected Areas in Cryptography. 9th Annual International Workshop, SAC 2002. Revised Papers (Lecture Notes in Computer Science Vol.2595)*, pages 76–93, 2003.

[20] E Kasper and P Schwabe. Faster and timing-attack resistant AES-GCM. In *Cryptographic Hardware and Embedded Systems - CHES 2009. Proceedings 11th International Workshop*, pages 1–17, 2009.

[21] Markus G Kuhn. Cipher instruction search attack on the bus-encryption security microcontroller DS5002FP. *IEEE Transactions on Computers*, 47(10):1153–1157, 1998.

[22] K Kurosawa and T Iwata. TMAC: two-key CBC MAC. In *Topics in Cryptology - CT-RSA 2003. Cryptoghraphers' Track at the RSA Conference 2003. Proceedings (Lecture Notes in Computer Science Vol.2612)*, pages 33–49, Berlin, Germany, 2003.

[23] Changhoon Lee, Jongsung Kim, Jaechul Sung, Seokhie Hong, and Sangjin Lee. Forgery and key recovery attacks on PMAC and Mitchell's TMAC variant. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 4058 LNCS, pages 421–431, Melbourne, Australia, 2006.

[24] M. Liskov, R.L. Rivest, and D. Wagner. Tweakable block ciphers. pages 31 – 46, Berlin, Germany, 2002//. tweakable block ciphers;cryptographic primitive;cryptographic key;initialization vector;CBC

mode;OCB mode;primitive block-cipher level;security;tweak block chaining;.

[25] M Luby and C Rackoff. How to construct pseudorandom permutations from pseudorandom functions. *SIAM Journal on Computing*, 17(2):373–386, 1988.

[26] D A McGrew and J Viega. The security and performance of the Galois/counter mode (GCM) of operation. In *Progress in Cryptology - INDOCRYPT 2004. 5th International Conference on Cryptology in India. Proceedings (Lecture Notes in Computer Science Vol.3348)*, pages 343 – 55, Berlin, Germany, 2004.

[27] K Minematsu and T Matsushima. New bounds for PMAC, TMAC, and XCBC. In *Fast Software Encryption. 14th International Workshop, FSE 2007. Revised Selected Papers. (Lecture Notes in Computer Science vol. 4593)*, pages 434 —- 51, 2007.

[28] M Nandi and A Mandal. Improved security analysis of PMAC. *J. Math. Cryptol. (Germany)*, 2(2):149 – 62, 2008.

[29] F Petrank and C Rackoff. CBC MAC for real-time data sources. *J. Cryptol. (USA)*, 13(3):315 – 38, 2000.

[30] P Rogaway. Authenticated-encryption with associated-data. In *Proceedings of the ACM Conference on Computer and Communications Security*, pages 98–107, Washington, DC, United states, 2002.

[31] P Rogaway. Efficient instantiations of tweakable blockciphers and refinements to modes OCB and PMAC. In *Advances in Cryptology-ASIACRYPT 2004. 10th International Conference on the Theory and Application of Cryptology and Information Security. Proceedings (Lecture Notes in Computer Science Vol.3329)*, pages 16–31, Berlin, Germany, 2004.

[32] P Rogaway. Evaluation of some blockcipher modes of operation. 2011.

[33] P Rogaway and J Black. OCB: a block-cipher mode of operation for efficient authenticated encryption. *ACM Trans. Inf. Syst. Secur. (USA)*, 6(3):365–403, 2003.

[34] A Rukhin, J Soto, J Nechvatal, M Smid, and E Barker. A statistical test suite for random and pseudorandom number generators for cryptographic applications. *Nist Special Publication*, (April), 2010.

[35] M-JO Saarinen. Cycling attacks on GCM, GHASH and other polynomial MACs and hashes. In *Fast Software Encryption. 19th International Workshop, FSE 2012. Revised Selected Papers*, pages 216 —- 25, 2012.

[36] P Sarkar. Pseudo-random functions and parallelizable modes of operations of a block cipher. *IEEE Transactions on Information Theory*, 56(8):4025–4037, 2010.

[37] J Soto. Randomness Testing of the Randomness Testing of the Advanced Encryption Standard Candidate Algorithms. pages 0–9, 1999.

[38] J Soto. Statistical testing of random number generators. *Proceedings of the 22nd National Information Systems ...*, 2, 1999.

[39] J Soto and L Bassham. Randomness Testing of the Advanced Encryption Standard Finalist Candidates Randomness Testing of the Advanced Encryption. 2000.

[40] Z Yuan, W Wang, K Jia, G Xu, and X W. New birthday attacks on some MACs based on block ciphers. In *Advances in Cryptology - CRYPTO 2009. Proceedings 29th Annual International Cryptology Conference*, pages 209 – 30, Berlin, Germany, 2009.