Another Look at PMAC *

Dayin Wang, Dongdai Lin, and Wenling Wu
(Corresponding author: Dayin Wang)

Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences
Beijing 100080, China
(Email: {wdy, ddlin, wwl}@is.iscas.ac.cn)

(Received Sept. 30, 2005; revised and accepted Nov. 23, 2005 & Jan. 27, 2006)

Abstract

We can view an existing Message Authentication Code (MAC) as a Carter-Wegman MAC in spite of the fact it may not have been designed as one. This will make the analysis easier than it has been when considered from other viewpoints. In this paper, we can look PMAC with two keys as a Carter-Wegman MAC and get a simple security proof for it. Using this viewpoint to look at PMAC, we will learn not only why the PMAC is such constructed, but also a new method of constructing MACs.

Keywords: Carter-Wegman MAC, message authentication code, pseudo-random functions, universal hash family

1 Introduction

A Message Authentication Code (MAC) provides a way to detect whether a message has been tampered with during transmission. The usual model for authentication includes three participants: a transmitter, a receiver and an adversary. The transmitter sends a message over an insecure channel, where the adversary can introduce new messages as well as alter existing ones. Insertion of a new message by the adversary is called impersonation, and modification of an existing message by the adversary is called substitution. In both cases the adversary's goal is to deceive the receiver into believing that the new message is authentic.

In many applications, it is of significant importance that the receiver can verify the integrity of a message. In some cases this is even more important than encryption [8]. The term "MAC" first appeared around 1980 in the ANSI X9.9 standard [1]. From then on, XOR-MAC [3], HMAC [4], XCBC-MAC [9] and so on are proposed in sequence.

The Carter-Wegman MACs are those which use a function from a Universal Hash Family to compress the message M to be MACed. The output of this hash function is then processed cryptographically to produce the tag.

PMAC [6], a fully parallelizable block-cipher mode of operation for message authentication, is deterministic, works for strings of any bit length, employs a single block-cipher key. At first glance the structure of PMAC can't be looked as a Carter-Wegman MAC. But if we use another key to encrypt the block in the last step in PMAC, we can get a typical structure of Carter-Wegman MAC and can get its security proof in Carter-Wegman spirit. In this paper we call the PMAC with two keys as PMACV.

We can view an existing MAC as a Carter-Wegman MAC in spite of the fact it may not have been designed as one. This will make the analysis easier than it has been when considered from other viewpoints. In this paper, we look PMACV as a Carter-Wegman MAC and get a simple security proof. From the proof of the PMACV, we can learn that if we view PMAC as a Carter-Wegman MAC directly, we only need to prove that the collision probability of all the input to PRF is negligible. This is a simple information-theoretic question. Thus we can get a simple proof for PMAC. On the other hand, we learn that we can use block cipher to construct Universal Hash Families from the proof of the PMACV.

The paper is organized as follows: In Section 2, Mathematical Preliminaries are introduced. In Section 3, the definitions of PMAC and PMACV are given. In Section 4, we introduce the definition of Universal Hash Families and how to use them to construct Carter-Wegman MAC. In Section 5, we give the security proof of PMACV by viewing it as Carter-Wegman MAC followed by conclusions in Section 6.

2 Mathematical Preliminaries

The Section is the same as the Section 2 in [6]. We include here for completeness.

If $i \geq 1$ is an integer then $ntz(i)$ is the number of trailing 0-bits in the binary representation of i. So,

*This research is supported by the National Natural Science Foundation of China under Grant No.60373047 and No.90204016; the National Basic Research 973 Program of China under Grant No. 2004CB318004.

for example, $ntz(7) = 0$ and $ntz(8) = 3$. If $A \in \{0,1\}^*$ is a string then $|A|$ denotes its length in bits while $\|A\|_n = \max\{1, \lceil A/n \rceil\}$ denotes its length in n-bit blocks (where the empty string counts as one block). If $A = a_{n-1} \cdots a_1 a_0 \in \{0,1\}^n$ is a string (each $a_i \in \{0,1\}$) then $str2num(A)$ is the number $\sum_{i=0}^{n-1} 2^i a_i$. If $A, B \in \{0,1\}^*$ are equal-length strings than $A \oplus B$ is their bitwise xor. If $A \in \{0,1\}^*$ and $|A| < n$ then $\text{pad}_n(A)$ is the string $A10^{n-|A|-1}$. If $A \in \{0,1\}^n$ then $\text{pad}_n(A) = A$. With n understood we write $\text{pad}(A)$ for $\text{pad}_n(A)$. If $A = a_{n-1}a_{n-2} \cdots a_1 a_0 \in \{0,1\}^n$ then $A \ll 1 = a_{n-2}a_{n-3} \cdots a_1 a_0 0$ is the n-bit string which is the left shift of A by 1 bit while $A \gg 1 = 0a_{n-1}a_{n-2} \cdots a_2 a_1$ is the n-bit string which is the right shift of A by one bit. In pseudo code we write "Partition M into $M[1] \cdots M[m]$" as shorthand for "Let $m = \|M\|_n$ and let $M[1], \cdots, M[m]$ be strings such that $M[1] \cdots M[m] = M$ and $|M[i]| = n$ for $1 \le i < m$.

The field with 2^n points is denoted $GF(2^n)$. We interchangeably think of a point a in $GF(2^n)$ in any of the following ways: (1) as an abstract point in the field; (2) as an n-bit string $a_{n-1} \cdots a_1 a_0 \in \{0,1\}^n$; (3) as a formal polynomial $a(x) = a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$ with binary coefficients; (4) as a nonnegative integer between 0 and $2^n - 1$, where $a \in \{0,1\}^n$corresponds to $str2num(a)$. We write $a(x)$ instead of a if we wish to emphasize that we are thinking of a as a polynomial. To add two points in $GF(2^n)$, take their bitwise xor. We denote this operation by $a \oplus b$. To multiply two points, fix some irreducible polynomial $p(x)$ having binary coefficients and degree n. To be concrete, choose the lexicographically first polynomial among the irreducible degree n polynomials having a minimum number of coefficients. To multiply points $a, b \in GF(2^n)$, which we denote $a \cdot b$, regard a and b as polynomials $a(x) = a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$ and $b(x) = b_{n-1}x^{n-1} + \cdots + b_1 x + b_0$, form their product $c(x)$ where one adds and multiplies coefficients in $GF(2)$, and take the remainder when dividing $c(x)$ by $p(x)$. Note that it is particularly easy to multiply a point $a \in \{0,1\}^n$ by x. We illustrate the method for $n = 128$, where $p(x) = x^{128} + x^7 + x^2 + x + 1$. Then multiplying $a = a_{n-1} \cdots a_1 a_0$ by x yields

$$a \cdot x = \begin{cases} a \ll 1 & \text{if firstbit}(a)\text{=0} \\ (a \ll 1) \oplus 0^{120}10000111 & \text{if firstbit}(a)\text{=1}. \end{cases} \tag{1}$$

It is similarly easy to divide a by x (meaning to multiply a by the multiplicative inverse of x). To illustrate, assume that $n = 128$. Then

$$a \cdot x^{-1} = \begin{cases} a \gg 1 & \text{if firstbit}(a)\text{=0} \\ (a \gg 1) \oplus 10^{120}1000011 & \text{if firstbit}(a)\text{=1}. \end{cases} \tag{2}$$

If $L \in \{0,1\}^n$ and $i \ge -1$, we write $L(i)$ to mean $L \cdot x^i$. To compute $L(-1)$, $L(0)$,$\cdots$,$L(\mu)$, where μ is small, set $L(0) = L$ and then, for $i \in [1 \cdots \mu]$, use Equation (1) to compute $L(i) = L(i-1) \cdot x$ from $L(i-1)$; and use Equation (2) to compute $L(-1)$ from L.

We point out that $huge = x^{-1}$ will be an enormous number (when viewed as a number); in particular, $huge$ starts with a 1 bit, so $huge > 2^{n-1}$. In the security proof this fact is relevant, so there we use $huge$ as a synonym for x^{-1} when this seems to add to clarity.

For any $l \ge 1$, a Gray code is an ordering $\gamma^l = \gamma_0^l \gamma_1^l \cdots \gamma_{2^l-1}^l$ of $\{0,1\}^l$ such that successive points differ (in the Hamming sense) by just one bit. For n a fixed number, PMAC makes use of the "canonical" Gray code $\gamma = \gamma^n$ constructed by $\gamma^1 = 01$ while, for $l > 0$

$$\gamma^{l+1} = 0\gamma_0^l 0\gamma_1^l \cdots 0\gamma_{2^l-2}^l 0\gamma_{2^l-1}^l 1\gamma_{2^l-1}^l 1\gamma_{2^l-2}^l \cdots 1\gamma_1^l 1\gamma_0^l.$$

It is easy to see that γ is a Gray code. What is more, for $1 \le i \le 2^n - 1$, $\gamma_i = \gamma_{i-1} \oplus (0^{n-1}1 \ll ntz(i))$. This makes it easy to compute successive points. Note that $\gamma_1, \gamma_2, \cdots, \gamma_{2^n-1}$ are distinct, different from 0, and $\gamma_i \le 2i$.

Let $L \in \{0,1\}^n$ and consider the problem of successively forming the strings $\gamma_1 \cdot L, \gamma_2 \cdot L, \gamma_3 \cdot L, \cdots, \gamma_m \cdot L$. Of course $\gamma_1 \cdot L = 1 \cdot L = L$. Now, for $i \ge 2$, assume one has already produced $\gamma_{i-1} \cdot L$. Since $\gamma_i = \gamma_{i-1} \oplus (0^{n-1}1 \ll ntz(i))$ we know that $\gamma_i \cdot L = (\gamma_{i-1} \oplus (0^{n-1}1 \ll ntz(i))) \cdot L = (\gamma_{i-1} \cdot L) \oplus (0^{n-1}1 \ll ntz(i)) \cdot L = (\gamma_{i-1} \cdot L) \oplus (L \cdot x^{ntz(i)}) = (\gamma_{i-1} \cdot L) \oplus L(ntz(i))$. That is, the ith word in the sequence $\gamma_1 \cdot L, \gamma_2 \cdot L, \gamma_3 \cdot L, \cdots$ is obtained by xoring the previous word with $L(ntz(i))$.

Algorithm $\text{PMAC}_E(K, M)$
$L \leftarrow E_K(0^n)$
if $|M| > n2^n$ **then return** 0^τ
Partition M into $M[1] \cdots M[m]$
for $i \leftarrow 1$ **to** $m-1$ **do**
  $X[i] \leftarrow M[i] \oplus \gamma_i \cdot L$
  $Y[i] \leftarrow E_K(X[i])$
$\Sigma \leftarrow Y[1] \oplus Y[2] \oplus \cdots \oplus Y[m-1] \oplus \text{pad}(M[m])$
if $|M[m]| = n$ then $X[m] = \Sigma \oplus L \cdot x^{-1}$
  else $X[m] \leftarrow \Sigma$
$Tag = E_K(X[m])[\text{first } \tau \text{ bits}]$
return Tag

Constants $\gamma_1, \gamma_2, \ldots$, the meaning of the multiplication operator, and the meaning of $pad(\cdot)$ are all defined in Section 2. We comment that Line 2 is simply to ensure that PMAC is well-defined even for the highly unrealistic case that $|M| > n2^n$ (by which time our security result becomes vacuous anyway). Alternatively, one may consider PMAC's message space to be strings of length at most $n2^n$ rather than strings of arbitrary length.

The authors of [6] said that PMAC is not a Carter-Wegman MAC. But we say we can view PMAC as a Carter-Wegman MAC in spirt of the fact it may not have been designed as one. In order to see it clearly, we first consider PMACV, a variant of the PMAC, as a Carter-Wegman MAC and give its security proof, and then explain how we can view PMAC. The following is the definition of the PMACV.

Algorithm $\text{PMACV}_E(K_1, K_2, M)$
$L \leftarrow E_{K_1}(0^n)$
if $|M| > n2^n$ **then return** 0^τ
Partition M into $M[1] \cdots M[m]$

for $i \leftarrow 1$ **to** $m-1$ **do**
$X[i] \leftarrow M[i] \oplus \gamma_i \cdot L$
$Y[i] \leftarrow E_{K_1}(X[i])$
$\Sigma \leftarrow Y[1] \oplus Y[2] \oplus \cdots \oplus Y[m-1] \oplus \mathrm{pad}(M[m])$
if $|M[m]| = n$ then $X[m] = \Sigma \oplus L \cdot x^{-1}$
else $X[m] \leftarrow \Sigma$
$Tag = E_{K_2}(X[m])$[first τ bits]
return Tag

From the definitions above, we can see that the only difference between PMAC and PMACV is the number of Key. PMACV change the key in the last step, while PMAC doesn't.

3 Carter-Wegman MAC

Carter-Wegman MAC is based on the universal hashing paradigm introduced by Carter and Wegman [7, 11]. They proposed to hash a given message with a randomly chosen function from a strongly universal family of hash functions, whereafter the output is encrypted with a one-time-pad (OTP) in order to obtain the MAC tag. In the original paper, Wegman and Carter [11] use perfect encryption to produce their MAC. Subsequently Black [5] gave several other variants of these methods for producing a MAC given a universal hash family. In this section, we will simply introduce Universal Hash Families and How to use them to construct Carter-Wegman MACs.

3.1 Universal Hash Families

There are many different variants of Universal Hash Families, and we now present a few of those that will be used later.

In the following discussion and throughout the paper will assume that the domain and range of universal hash functions are finite sets of binary strings and that the range is smaller than the domain.

Definition 1. ***[Carter and Wegman,1979]*** *Fix a domain D and range R. A finite multiset of hash functions $H = \{h : D \longrightarrow R\}$ is said to be* ***Universal*** *if for every $x, y \in D$ where $x \neq y$, $\Pr_{h\in H}[h(x) = h(y)] = 1/|R|$.*

If we relax slightly the requirement that the collision probability be $1/|R|$, we will get the notion of Almost Universal Hash Families in which we allow the collision probability to be some $\epsilon \geq 1/|R|$.

Definition 2. *Let $\epsilon \in R^+$ be a positive number. Fix a domain D and range R. A finite multiset of hash functions $H = \{h : D \longrightarrow R\}$ is said to be $\epsilon - Almost\ Universal(\epsilon - AU)$ if for every $x, y \in D$ where $x \neq y$, $\Pr_{h\in H}[h(x) = h(y)] \leq \epsilon$.*

3.2 From Hash to MAC

In [5], Black give another approach to building a MAC in the Carter-Wegman style. This approach apply a PRF directly to the output of an almost universal hash family. We now give description of it. Let $Rand(n,n)$ denote the set of all functions from $\{0,1\}^n$ to $\{0,1\}^n$. Suppose the shared key between transmitter and receiver is (h, ρ) where $h \in H = \{h : D \to \{0,1\}^n\}$ is a $\epsilon - AU$ and ρ is a randomly-choose function from $Rand(n,n)$. Given a message M, the output of MAC is $\rho(h(M))$. We will get the following theorem regarding the construction.

Theorem 1 (PRF(Hash) as a PRF). *Fix $n \geq 1$. Let $H = \{h : Msg \to \{0,1\}^n\}$ be a family of hash functions. Let $\delta(m)$ be a function such that for all distinct pairs $M, M' \in Msg$,with M and M' at most mn bits long, $\Pr_{h \xleftarrow{R} H}[h(M) = h(M')] \leq \delta(m)$. Let A be an adversary that asks q queries from Msg, each query of length at most mn bits. Then*

$$\Pr[h \xleftarrow{R} H; \rho \xleftarrow{R} Rand(n,n) : A^{\rho(h(\cdot))} = 1] -$$
$$\Pr[g \xleftarrow{R} Rand(Msg,n) : A^{g(\cdot)} = 1] \leq \frac{q^2}{2}\delta(m).$$

This theorem's proof is given in Lemma 6.3.6 in [5]. This theorem tell us this construction $\rho(h(\cdot))$ can not be distinguished with random function family.

4 Security Proof of PMACV

The security proof of PMACV is done by viewing it as Carter-Wegman MAC. Firstly, we divide PMACV into two parts, one is a universal hash family, and the other is a PRF. Then we use theorem 1 to complete the proof. Before giving the definition of the universal hash family in the PMACV, we give a few lemmas that will be used in the proof.

It is often convenient to replace random permutations with random functions, or vice versa in security analysis. Let $Perm(n)$ denote the set of all permutations on $\{0,1\}^n$. The following lemma lets us easily do this. For a proof see Proposition 2.5 in [2].

Lemma 1 (PRF/PRP Switching). *Fix $n \geq 1$. Let A be an adversary that asks at most q queries. Then*

$$\Pr[\pi \xleftarrow{R} Perm(n) : A^{\pi(\cdot)} = 1]$$
$$-\Pr[\rho \xleftarrow{R} Rand(n,n) : \quad A^{\rho(\cdot)} = 1]$$
$$\leq q(q-1)/2^{n+1}.$$

As is customary, we will show the security of our MACs by showing that their information-theoretic versions approximate random functions. As is standard, this will be enough to pass to the complexity-theoretic scenario. Part of the proof is Proposition 2.7 of [2].

Lemma 2 (Inf.Th.PRF $\Longrightarrow$ Comp.Th.PRF). *Fix $n \geq 1$. Let $CONS$ be a construction such that $CONS_{\rho_1,\rho_2}(\cdot) : \{0,1\}^* \to \{0,1\}^n$ for any $\rho_1, \rho_2 \in Rand(n,n)$. Suppose that if $|M| \leq \mu$ then $CONS_{\rho_1,\rho_2}(M)$ depends on the values of ρ_i on at most p points (for $1 \leq i \leq 2$). Let $E : Key \times \{0,1\}^n \to \{0,1\}^n$ be a family of functions. Then*

$$Adv^{prf}_{CONS[E]}(t,q,\mu) \leq Adv^{prf}_{CONS[Perm(n)]}(q,\mu) + 2\cdot Adv^{prf}_{E}(t',p), \text{ and}$$
$$Adv^{mac}_{CONS[E]}(t,q,\mu) \leq Adv^{prf}_{CONS[Perm(n)]}(q,\mu) + 2\cdot Adv^{prf}_{E}(t',p) + \frac{1}{2^n}$$

where $t' = t + O(pn)$.

In PMACV, we can look all the steps before the last encryption as a universal hash family, called PMACH, and prove it is a $\epsilon - AU$. Thus we can prove PMACV is secure in the Carter-Wegman spirit. Now we will give a description of the PMACH before we give its proof.

Algorithm $\mathrm{PMACH}_E(M)$
$L \leftarrow E(0^n)$
if $|M| > n2^n$ **then return** 0^τ
Partition M into $M[1]\cdots M[m]$
for $i \leftarrow 1$ **to** $m-1$ **do**
  $X[i] \leftarrow M[i] \oplus \gamma_i \cdot L$;
  $Y[i] \leftarrow E(X[i])$
$\Sigma \leftarrow Y[1] \oplus Y[2] \oplus \cdots \oplus Y[m-1] \oplus \mathrm{pad}(M[m])$
if $|M[m]| = n$ then $X[m] = \Sigma \oplus L \cdot x^{-1}$
  else $X[m] \leftarrow \Sigma$
return $X[m]$

From the definition above, we can see that the domain of PMACH is $\{0,1\}^*$ and the PMACH hash family's members are selected by the random choice of some block cipher E . As is customary, we will replace the block cipher E by a random function ρ and then we get the following theorem.

Theorem 2. *Fix $l \geq 1$.Then the* PMACH_ρ *on domain* $D(\{0,1\}^*)$ *is* 2^{-n}*–*AU.

Proof. we are required to show that for any two distinct values $M, \overline{M} \in D$, the probability over uniform random choices of functions $\rho \in Rand(n,n)$ that the PMACH_ρ of $M, \overline{M}$ are equal at most 2^{-n}. In other words we must prove that.

$$\Pr[\rho \xleftarrow{R} Rand(n,n) : \mathrm{PMACH}_\rho(M) = \mathrm{PMACH}_\rho(\overline{M})] \leqslant 2^{-n} \quad (3)$$

Due to $\mathrm{PMACH}_\rho(M) = X[m]$, $\mathrm{PMACH}_\rho(\overline{M}) = \overline{X}[\overline{m}]$ and ρ is a random function, we now consider all the cases.

Case 1: Suppose that $|M[m]| < n$ and $|\overline{M}[\overline{m}]| < n$. If $m > \overline{m}$ then $\Pr[X[m] = \overline{X}[\overline{m}]] = \Pr[\Sigma = \overline{\Sigma}] = 2^{-n}$ because of the contribution of $Y[m-1]$ in $\Sigma-$ a random variable that is not used in the definition of $\overline{\Sigma}$. If $m < \overline{m}$ then $\Pr[X[m] = \overline{X}[\overline{m}]] = \Pr[\Sigma = \overline{\Sigma}] = 2^{-n}$ because of the contribution of $\overline{Y}[\overline{m}-1]$ in $\overline{\Sigma}$-a random variable that is not used in the definition of Σ. If $m = \overline{m}$ and there is an $i < m$ such that $M[i] \neq \overline{M}[i]$ then $\Pr[X[m] = \overline{X}[\overline{m}]] = \Pr[\Sigma = \overline{\Sigma}] = 2^{-n}$ because of the contribution of $\overline{Y}[i]$ in $\overline{\Sigma}$-a random variable that is not used in the definition of Σ. If $m = \overline{m}$ then for every $i < m$ we have that $M[i] = \overline{M}[i]$, then, necessarily, $M[m] \neq \overline{M}[m]$. In this case$\Pr[\Sigma = \overline{\Sigma}] = 0$, as the two checksums differ by the nonzero value $\mathrm{pad}(M[m]) \oplus \mathrm{pad}(\overline{M}[m])$.

Case 2: Suppose that $|M[m]| = n$ and $\overline{M}[\overline{m}] = n$. Then $X[m]$ and $\overline{X}[m]$ are offset by the same amount, $huge \cdot L$, so this offset is irrelevant in computing $\Pr[X[m] = \overline{X}[\overline{m}]]$. Proceed as above.

Case 3: Suppose that $|M[m]| < n$ and $\overline{M}[\overline{m}] = n$. Then $\Pr[X[m] = \overline{X}[\overline{m}]] = \Pr[\Sigma = \overline{\Sigma} \oplus huge \cdot L] = 2^{-n}$ since Σ and $\overline{\Sigma}$ are independent of L. similarly, if $|M[m]| = n$ and $\overline{M}[\overline{m}] < n$, then $\Pr[X[m] = \overline{X}[\overline{m}]] = 2^{-n}$.

To sum up,the collision probability of the PMACH is exactly 2^{-n}. □

Lemma 3. *Suppose all distinct pairs M, $\overline{M}$ are at most mn bits long. Then the* PMACH_π *on domain* $D(\{0,1\}^*)$ *is* $\frac{2m^2}{2^n} - AU$*.*

Proof. Using Lemma 1, lets us now replace the random function ρ in Equation (3) by a random permutation π. we get the following equation.

$$\Pr[\pi \xleftarrow{R} Perm(n) : \mathrm{PMACH}_\pi(M) = \mathrm{PMACH}_\pi(\overline{M})] \leqslant \frac{1}{2^n} + \frac{2m(2m-1)}{2^{n+1}} \leqslant \frac{2m^2}{2^n}.$$

□

In the following we prove the security of the PMACV construction.

Theorem 3 (PMACV ≈ Rand). *Fix $n \geq 1$ and let $N = 2^n$. Let A be an adversary which asks at most q queries, each of which is at most mn-bits. Assume $m \leq N/4$. Then*

$$Pr[\pi_1, \pi_2 \xleftarrow{R} Perm(n) : A^{PMACV_{\pi_1,\pi_2}} = 1] - Pr[g \xleftarrow{R} Rand(\{0,1\}^*, n) : A^{g(\cdot)} = 1] \leq \frac{q^2}{2}\cdot\frac{2m^2}{2^n} + \frac{q^2}{2^{n+1}}.$$

Proof. We will first compute a related probability where the final permutation is a random function; this will simplify the analysis. So we are interested in the quantity

$$\Pr[\pi_1 \xleftarrow{R} Perm(n), \rho \xleftarrow{R} Rand(n,n) : A^{\mathrm{PMACV}_{\pi_1,\rho}} = 1] - \Pr[g \xleftarrow{R} Rand(\{0,1\}^*, n) : A^{g(\cdot)} = 1].$$

Due to the $\mathrm{PMACV}_{\pi_1,\rho} = \rho(\mathrm{PMACH}_{\pi_1}(\cdot))$, Lemma 3 gives that $\Pr[\mathrm{PMACH}_\pi(M) = \mathrm{PMACH}_\pi(\overline{M})] \leq \frac{2m^2}{2^n}$. Therefore we may conclude from Theorem 1 that

$$\Pr[\pi_1 \xleftarrow{R} Perm(n), \rho \xleftarrow{R} Rand(n,n) : A^{\rho(\mathrm{PMACH}_{\pi_1}(\cdot))} = 1] - \Pr[g \xleftarrow{R} Rand(\{0,1\}^*, n) : A^{g(\cdot)} = 1] \leq \frac{q^2}{2}\cdot\frac{2m^2}{2^n}.$$

Finally we replaced the PRF ρ in the left-hand probability with a PRP to properly realize the PMACV construction.

Using Lemma 1 this costs us an extra $\frac{q^2}{2^{n+1}}$, and so our final bound is

$$\frac{q^2}{2} \cdot \frac{2m^2}{2^n} + \frac{q^2}{2^{n+1}}.$$

This complete the proof. □

From the Lemma 2 above it is standard to pass from information-theoretic to a complexity-theoretic.

It is a standard result that being secure in the sense of a PRF implies an inability to forge with good probability see [2, 10].

5 Conclusions

In this paper we study the PMAC with two keys. Viewing it as a Carter-Wegman MAC, we effectively eliminates the complexity associated with the adversary's adaptivity and get a simple proof . From this viewpoint we can also view PMAC as a Carter-Wegman MAC directly except that the PRF applying to a universal hash functions is not independent of the universal hash functions. But it is still secure if the collision probability of all the input to PRF is negligible. Thus we can get a simple security proof for PMAC in this way.

On the other hand, we know that we can use block cipher to construct a new Universal Hash Family and then a new MAC from this paper. This is a new method of constructing MACs.

References

[1] ANSI X9.9 (Revised), *American National Standard - Nancial Institution Message Authentication*, Technical Report, ASC X9, (Replaces X9.9 - 1982), 1986.

[2] M. Bellare, J. Kilian, and P. Rogaway, "The security of the cipher block chaining message authentication code", in *Advances in Cryptology - CRYPTO'94*, LNCS 839, Springer-Verlag, pp. 341-358, 1994.

[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*, Springer-Verlag, pp. 15-35, 1995.

[4] M. Bellare, R. Canetti, and H. Krawczyk, "Keying hash functions for message authentication", in *Advances in Cryptology - CRYPTO'96*, Springer-Verlag, pp. 1-19, 1996.

[5] J. Black, "Message authentication codes", Manuscript Available at http://www.cs.colorado.edu/jrblack/papers.html.

[6] J. Black and P. Rogaway, "A block-cipher mode of operation for parallelizable message authentication", in *Advances in Cryptology - EUROCRYPT'2002*, pp. 384-401, 2002.

[7] J. Carter and M. Wegman, "Universal classes of hash functions", *Journal of Computer and System Sciences*, Vol. 18, pp. 143-154, 1979.

[8] N. Ferguson and B. Schneier, *Practical Cryptograhpy*, Wiley, 2003.

[9] V. D. Gligor and P. Donescu, "Fast encryption and authentication: XCBC encryption and XECB authentication modes", in *FSE 2001 (Fast Software Encryption Workshop 2001)*, Springer-Verlag, pp. 92-141, 2002.

[10] O. Goldreich, S. Goldwasser, and S. Micali, "How to construct random functions", *Journal of the ACM*, vol. 33, no.4, pp. 210-217, 1986.

[11] M. Wegmann and J. Carter, "New hash functions and their use in authentication and set equality", *Journal of Computer and System Sciences*, vol. 22, no. 3, pp. 265-279, 1981.

Dayin Wang is now a Ph.D candidate at the State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences. His research interests include Message Authentication codes and Mode of Operation. E-mail address: wdy@is.iscas.ac.cn.

Dongdai Lin is now a full time research professor and deputy director of State Key Labortory of Information Security, Institute of Software of the Chinese Academy of Sciences. He received his B.S. degree in mathematics from Shandong University in 1984, and the M.S. degree and Ph. D degree in coding theory and cryptology at Institute of Systems Science of the Chinese Academy of Sciences in 1987 and 1990 respectively. His current research interests include cryptology, information security, grid computing, mathematics mechanization and symbolic computations.

Wenling Wu is now a professor at the State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences. She received her B.S. degree and M.S. degree in Maths from Northwest University in 1987 and 1990, respectively. She received her Ph.D degree in Cryptography from Xidian University in 1997. From 1998 to 1999 she was a postdoctoral fellow in the Institute of Software, Chinese Academy of Science. Her current research interests include theory of cryptography, mode of operation, block cipher, stream cipher and hash function.