ELSEVIER

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 15 (2011) 2862 – 2868

Procedia Engineering

www.elsevier.com/locate/procedia

Advanced in Control Engineering and Information Science

A Novel Hash Scheme Based on SNP-PLCM

Pengcheng Wei*, Junjian Huang, and Wei Zhang

Department of Computer Science, Chongqing Education College, Chongqing 400067, China

Abstract

By combining the traditional iteration structure of Hash function with the dynamic S-boxes, a novel keyed Hash function is presented. The proposed approach can give a chaotic Hash value by means of the lookup table of functions and chaotic dynamic S-box. Compared with the existing chaotic Hash functions, this method improves computational performance of Hash system by using the chaotic S-box substitution. Theoretical and experimental results show that the proposed method has not only strong one way property, sensitivity to initial conditions and chaotic system's parameters, but also high speed.

Selection and/or peer-review under responsibility of [CEIS 2011]

Keywords: Data Integrality; Chaotic System; Hash Function, S-box;

1. Introduction

Researcher gradually began to study Chaotic Hash function since 1999[1-5]. Chaos as a new method to construct hash function is getting more and more scholars' attentions. K.W. Wong in [6] proposed a method about combination of encryption and Hash. G. Alvarez pointed out the security holes, shortly after the program proposed, but the way of the combination of encryption and Hash is a very unique idea with good promotional value

This article proposed the construction and analysis of K-Hash function based on the chaotic dynamic S-boxes, aiming at the current insufficient of hash function. This method first uses chaotic pseudo-random sequence to construct 8×8 's $S_{8\times8}(\square)$ and 4×4 's $S_{4\times4}(\square)$. Dynamic $S_{8\times8}(\square)$ is for the linear transformation of raw data in order to overcome the multiple iterations of a plaintext block. $S_{4\times4}(\square)$ is to define a dynamic look-up table so that it can decide the replacement order of the plaintext block in the $S_{8\times8}(\square)$ boxes whose purpose is to defend the use of a large number of data before and after replacement

[*] * Corresponding author. Tel.: 86-023-62658040; fax: +86-023-62658040.
E-mail address: wpc75@163.com.

doi:10.1016/j.proeng.2011.08.539

to do statistical analysis. This method can dynamically Adjust the length of hash value without any additional operation. Theoretical analysis and simulation results indicate the algorithm has the good one-way、confusion、initial value and key sensitivity ,with security and efficiency, which is a relatively-easy-to- implement K-Hash function.

2. Construction of chaotic hash function

2.1. The choice of chaotic maps

To take full advantage of piecewise linear chaotic map is a fast operation, we use in [8] proposed chaotic pseudo-random sequence constructed pseudo-random sequence generator which generated by segment number parameter (SNP-PLCM) PLCM. The mathematical expression of interval number parameterization PLCM is

$$x_{n+1}=f(x_n,\mu)=\begin{cases}\dfrac{x_n-i\mu}{\dfrac{\mu}{l}}, & \dfrac{i\mu}{l}\le x_n<\dfrac{(i+1)\mu}{l},\\ \dfrac{x_n-(\mu+\dfrac{i(0.5-\mu)}{l}}{\dfrac{0.5-\mu}{l}}, & \mu+\dfrac{i(0.5-\mu)}{l}\le x_n<\mu+\dfrac{(i+1)(0.5-\mu)}{l},\\ 0, & x_n=0.5\\ f((1-x_n),\mu), & 0.5<x_n<1,\end{cases} \tag{1}$$

Where $i=0,1,2,\cdots,l-1$; $x_n\in[0,1]$, l as the Selected parameters of number interval, the number of the mapping interval is $4\cdot l$.

2.2. Design chaotic pseudo-random sequence generator

Suppose there were two initial values $x_1(0),x_2(0)$ SNP-PLCM: $f_1(\cdot)$ $f_2(\cdot)$, $x_1(i+1)=f_1(x_1(i),\mu_1,l_1)$ and $x_2(i+1)=f_2(x_2(i),\mu_2,l_2)$.

Where u_1,u_2 is the control parameter, $\{x_1(i)\}$ and $\{x_2(i)\}$ are two chaotic orbits, optional initial value x_0 and parameters μ and l , to eliminate the influence of initial value, equation (1) iterate $L\log_{2l}2$ times [9-10].

Define a pseudo-random bit sequence is as follow:

$$k(i)=g\left(x_1(i),x_2(i)\right)=\begin{cases}1, & x_1(i)>x_2(i),\\ Null, & x_1(i)=x_2(i),\\ 0, & x_1(i)=x_2(i).\end{cases} \tag{2}$$

2.3. Design dynamic S-box

Construction of dynamic 8×8 's S-box of Pseudo-random sequence based on chaos is as follow:

□ According to the pseudo-random sequence generator of (2) , SNP-PLCM $f_1(\cdot)$'s and $f_2(\cdot)$'s initial value $x_1(0)$ and $x_2(0)$, interval number parameters l_1 、 l_2 , the level of the control parameters' disturbance L_1 and L_2 , the achieved accuracy of disturbed output sequence L ;

□ Equation (1) iterates at least $L\log_{2l}2$ times.

□ Get 0-1 sequence $T=\{b(i)|i=0,1,2,\cdots\}$ through a pseudo-random sequence generator in (2);

□ Get noise vector $U_k=\{b_{8k},b_{8k+1},\cdots,b_{8k+n}\}$， $k\geq 0$ Due to $T=\{b(i)|i=0,1,2,\cdots\}$;

□ To integers $i,j\in[0,16]$, use noise vector to define two-dimensional vector $S_{8\times 8}[i][j]=U_{ni+j}$, then get 8×8 's S-box $S_{8\times 8}(\square)$.

2.4. Design dynamic lookup table

Although the chaotic dynamic S-box could change as secret key K, chaos degree enhanced by using chaotic characteristics, single chaotic dynamic S-box can not resist statistical analysis attack effectively[12] 。 Therefore, the traditional encryption methods which help to overcome that problem are introduced to the dynamic function lookup table [12]. Defined as Eq (3) ~ (6) the corresponding conversion function.

$$A = f_1(A,B,C,D) = (S_{8\times 8}(\bar{A})\boxed{+}S_{8\times 8}(B) \oplus S_{8\times 8}(C) \oplus S_{8\times 8}(D)) \tag{3}$$

$$B = f_2(A,B,C,D) = (S_{8\times 8}(A) \oplus S_{8\times 8}(\bar{B})\boxed{+}S_{8\times 8}(C) \oplus S_{8\times 8}(D)) \tag{4}$$

$$C = f_3(A,B,C,D) = (S_{8\times 8}(A) \oplus S_{8\times 8}(B) \oplus S_{8\times 8}(\bar{C})\boxed{+}S_{8\times 8}(D)) \tag{5}$$

$$D = f_4(A,B,C,D) = (S_{8\times 8}(\bar{A}) \oplus S_{8\times 8}(B) \oplus S_{8\times 8}(C) \oplus S_{8\times 8}(\bar{D})) \tag{6}$$

Where A、B、C and D are 8-bit register, $\boxed{+}$ for the modular addition operation of 2^8, $\overline{X}$ is the reverse according to the bit, $X\oplus Y$ means according to the different bit or operation, the results are stored in the corresponding register.

Definition 1: If f_a， f_b is equation(3)~(6) 's transfer function, $f_a \circ f_b$ is cascade operation, so define table 1 as the dynamic look-up table.

Table 1. Function lookup table

$S_{4\times 4}(\bullet)$	$F(\square)$	$S_{4\times 4}(\bullet)$	$F(\square)$	$S_{4\times 4}(\bullet)$	$F(\square)$	$S_{4\times 4}(\bullet)$	$F(\square)$
0000	$f_1 \circ f_2 \circ f_4 \circ f_3$	0100	$f_2 \circ f_1 \circ f_3 \circ f_4$	1000	$f_3 \circ f_1 \circ f_4 \circ f_2$	1100	$f_4 \circ f_2 \circ f_1 \circ f_3$
0001	$f_1 \circ f_3 \circ f_2 \circ f_4$	0101	$f_2 \circ f_4 \circ f_1 \circ f_3$	1001	$f_3 \circ f_2 \circ f_4 \circ f_1$	1101	$f_4 \circ f_1 \circ f_2 \circ f_3$
0010	$f_1 \circ f_3 \circ f_4 \circ f_2$	0110	$f_2 \circ f_4 \circ f_3 \circ f_1$	1010	$f_3 \circ f_2 \circ f_4 \circ f_1$	1110	$f_4 \circ f_1 \circ f_3 \circ f_2$
0011	$f_1 \circ f_4 \circ f_2 \circ f_3$	0111	$f_2 \circ f_3 \circ f_1 \circ f_4$	1011	$f_3 \circ f_4 \circ f_2 \circ f_1$	1111	$f_4 \circ f_3 \circ f_1 \circ f_2$

Dynamic Lookup Table offers 16 kinds of optional cascade functions. Different $S_{4\times 4}(\bullet)$'s value has different transformation function $F(\square)$. Compared in a single transformation function, dynamic function lookup table increased complexity of the transformation with little computational cost, increased the difficulty of the attack and is easy to implement software and hardware.

2.5. Design hash function

Assuming message M is the binary sequence, the length of hash value is N ($N=128+32*i$ ， $i=0,1,2\cdots$), $|M| = kN$ ， $k=1,2,\cdots$ ， if M's length $|M|$ is not the integer multiple of N, then fill the last group , filling method is one 1 and several 0, means "1000⋯". The last group also includes the total length of hash function input.

Divided into groups by length N bit to M, note $M=M_1M_2\cdots M_k$, among them, the length of M_i is N bit. Divided into groups by length 32 bit to M_i , note $M_i=m_{i,0}^1m_{i,1}^1\cdots m_{i,31}^1,m_{i,0}^2m_{i,1}^2\cdots m_{i,31}^2,\cdots,m_{i,0}^jm_{i,1}^j\cdots m_{i,31}^j$, $1\le j\le N/32$.

$H_i(M_i)$ Expresses to obtain N bit hash value after through a compression function of N Bit plaintext block M_i . Moreover, $C_{8\times8}$ is a 32-bit counter using to dynamically update $S_{8\times8}[i][j]$. $C_{4\times4}$ is a 8-bit counter using to dynamically update $S_{4\times4}[i][j]$. Figure 1 shows the detailed process description of the algorithm.

```
Input= $\{\mu_1, x_{01}, L_1, \mu_2, x_{02}, L_2, L\}$
step1： $C_{8\times8}=0$， $C_{4\times4}=0$;
Step2： The structure sequence $S_{8\times8}[i][j]$ of 8192 bits 0-1 , which generated by the
        method of Figure 1 ; The first 64-bit structure $S_{4\times4}[i][j]$ ; Initialize $H_i(M_i)$
Step3： for i=1 to i≤k do
    begin
      $M_i = M_i \oplus H_i(M_i)$;
            for j=0 to j≤ $N/32$
            begin
                $A = f_1(m_{i,0}^j \cdots m_{i,7}^j, m_{i,8}^j \cdots m_{i,15}^j, m_{i,16}^j \cdots m_{i,23}^j, m_{i,24}^j \cdots m_{i,31}^j)$ ;
                $B = f_2(m_{i,0}^j \cdots m_{i,7}^j, m_{i,8}^j \cdots m_{i,15}^j, m_{i,16}^j \cdots m_{i,23}^j, m_{i,24}^j \cdots m_{i,31}^j)$ ;
                $C = f_3(m_{i,0}^j \cdots m_{i,7}^j, m_{i,8}^j \cdots m_{i,15}^j, m_{i,16}^j \cdots m_{i,23}^j, m_{i,24}^j \cdots m_{i,31}^j)$ ;
                $D = f_4(m_{i,0}^j \cdots m_{i,7}^j, m_{i,8}^j \cdots m_{i,15}^j, m_{i,16}^j \cdots m_{i,23}^j, m_{i,24}^j \cdots m_{i,31}^j)$ ;
        According to Table 1, the function lookup table, extracts the last bit of $A, B, C, D$
    through the operation of $S_{4\times4}(\bullet)$ . $F(\square)$ calculated according to the value of $S_{4\times4}(\bullet)$ ;
            $H_i(M_i)=H_i(M_i)\|F(\bullet)$ ; //express the connection of two bit strings
            $C_{4\times4}=C_{4\times4}+8$ ;
            if ( $C_{4\times4} \bmod 2^8$ ==0)
        0-1 sequence constructed a new $S_{4\times4}(\bullet)$ through the method of (1)
        end
        $C_{8\times8}=C_{8\times8}+32$ ;
        if ( $C_{8\times8} \bmod 2^{32}$ ==0)
        0-1 sequence constructed a new $S_{8\times8}(\bullet)$ through the method of (2)
    end
```

Figure 1. The Hash Function Algorithm based on chaotic dynamic S-box

3. Hash Function Performance Analysis

A good K-Hash not only has a good one-way, but also has other features, such as Key sensitivity, Key’S any small changes will have a different hash summary; Initial value sensitivity.

3.1. Sensitivity analysis of key

key’s control parameters and initial values within the effective range do different range’s disturbance and obtain the following 6 groups’ key. Controls parameters do small disturbance to obtain the 6 groups’ key are:

$Key_1=\{\mu_1,x_{01},L_1,\mu_2,x_{02},L_2,L\}=\{0.25,0.68,7,0.40,0.34,9,10\}$

$Key_2=\{\mu_1+2*10^{-15},x_{01},L_1,\mu_2-2*10^{-5},x_{02},L_2,L\}$;

$Key_3=\{\mu_1+3*10^{-15},x_{01},L_1,\mu_2-3*10^{-5},x_{02},L_2,L\}$;

$Key_4=\{\mu_1+4*10^{-15},x_{01},L_1,\mu_2-4*10^{-5},x_{02},L_2,L\}$;

$Key_5=\{\mu_1+5*10^{-15},x_{01},L_1,\mu_2-5*10^{-5},x_{02},L_2,L\}$;

$Key_6=\{\mu_1+6*10^{-15},x_{01},L_1,\mu_2-6*10^{-5},x_{02},L_2,L\}$

Initial value do small disturbance to obtain the 6 groups’ key are:

$Key_1=\{\mu_1,x_{01},L_1,\mu_2,x_{02},L_2,L\}=\{0.25,0.68,7,0.40,0.34,9,10\}$

$Key_2=\{\mu_1,x_{01}-2*10^{-15},L_1,\mu_2,x_{02}+2*10^{-15},L_2,L\}$;

$Key_3=\{\mu_1,x_{01}-3*10^{-15},L_1,\mu_2,x_{02}+3*10^{-15},L_2,L\}$;

$Key_4=\{\mu_1,x_{01}-4*10^{-15},L_1,\mu_2,x_{02}+4*10^{-15},L_2,L\}$;

$Key_5=\{\mu_1,x_{01}-5*10^{-15},L_1,\mu_2,x_{02}+5*10^{-15},L_2,L\}$;

$Key_6=\{\mu_1,x_{01}-6*10^{-15},L_1,\mu_2,x_{02}+6*10^{-15},L_2,L\}$.

These 12 groups were used respectively to calculate the128-bit hash value based on chaotic dynamic S-box to obtain the number of bit by the summary of hash before and after disturbance. As illustrated in Figure 2.

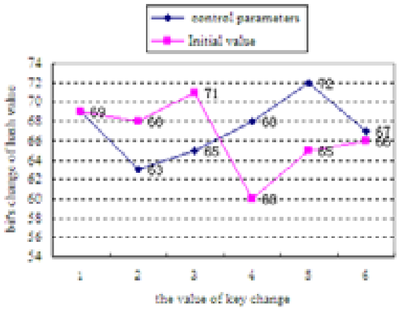

Figure 2. Key sensitivity

As can be seen from Figure 2, By the small disturbance of key control parameters μ_{01} and μ_{02}, the numbers of each bit’s changing of hash value are 68,71,60,65,66, average 66. There is only a difference of 2 compared with the ideal state 64 , about a change in the probability of 50%. Similarly, By the small disturbance of the key’s initial value x_{01} and x_{02}, the numbers of each bit’s changing of hash value are 63, 65,68,72,67, average 67. There is only a difference of 3 compared with the ideal state 64. About the change of probability is 50%. So, hash function based on chaotic dynamic S-box has a very high key sensitivity.

3.2. Sensitivity analysis of the data

For data sensitivity analysis, the following five simulation experiments under different circumstances：

Condition 1 selected the initial text: "*Cryptographic hash functions play a fundamental role in modern cryptography. While related to conventional hash functions commonly used in non-cryptographic computer applications – in both cases, larger domains are mapped to smaller ranges – they differ in several important aspects. Our focus is restricted to cryptographic hash functions (hereafter, simply hash functions), and in particular to their use for data integrity and message authentication.*"。

Condition 2 changed the first character of the initial text C to c.

Condition 3 changed the word "applications" to "function" in the initial text.

Condition 4 changed the full stop to comma in the final text.

Condition 5 added a space in the final text.

Using this algorithm to generate 128-bit hexadecimal value Hash as follows:

Hash value when Condition 1: 7E5454912B6E D86DFA67549317861E6B

Hash value when Condition 2: 65DFD358A0C9C5 DA1B546BCA855F52E A

Hash value when Condition 3: A4908A2783BDF27E5BA A41E325045C 82

Hash value when Condition 4: 4CBC74 D532CD067AD87E FDE4C808E18B

Hash value when Condition 5: 875812D438A842D4F4E2B92D 49831C90

The Hash values are shown in Figure 3.

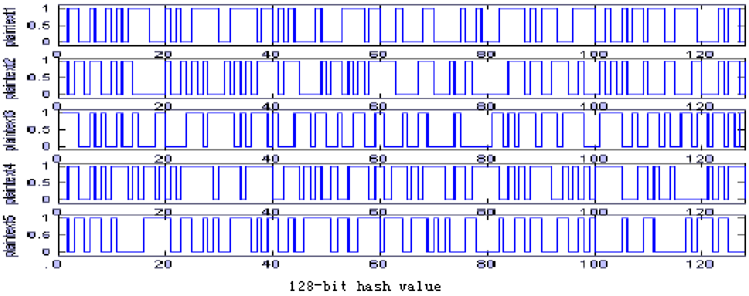

Figure 3. Hash values under different conditions

Simulation results show that, values will cause great changes Hash with small changes in the original data in the case of the same initial key, so it has a high data sensitivity. Hash value is the extremely sensitive function of the original data for each bit.

4. Summarizes

Aiming at the problems in the current e-government information security's data integrity , this article combined with the hash function of traditional structure with the chaotic dynamic S-box to propose a kind of hash function with key based on chaotic dynamic S-box. This algorithm can meet the needs of a various lengths of hash values under hash value as long as the length of N ($N = 128 + 32 * i$, $i = 0,1,2\cdots$). Meanwhile, this algorithm gives consideration to the security and complexity. Theoretical analysis and simulation results show that the algorithm has good statistical properties, collision resistance and flexibility so that can be another good choice for safe hash function.

Acknowledgement

The work described in this paper was supported by the Natural Science Foundation of China (No. 60703035), by Natural Science Foundation Project of CQ CSTC (NO.2009BB2227), the Foundation of Chongqing Education Committee (No. KJ091501,KJ091502, KJ101501, KJ101502).

References

[1]W. Diffie and M. E. Hellman: New Directions in Cryptography [J]. IEEE Transactions on Information Theory, vol. IT-22, pp. 644-654, 1976

[2]L. Kocarev: Chaos-based cryptography: A Brief Overview [J]. IEEE Circuits Mag., Vol.1, No3, pp.6-21, 2001

[3]F. Dachselt and W. Schwarz: Chaos and Cryptography. IEEE Trans. Circuits Sys.1: Fudam. Theory Appl."[J], Vol 48, No12, pp. 1498-1509,2001

[4]R. Schmitz,: Use of Chaotic Dynamical Systems in Cryptography. J. Franklin Inst., Vol. 338, pp.429-441,2001

[5]L. Kocarev: Public-key Encryption Based on Chevyshev Maps [A]. Proc IEEE Symp. Circuits Syst.Vol 3, pp.28-31,2003

[6]Pina Bergamo, Paolo D'Arco, Alfredo De Santis, and Ljupco Kocarev: Security of Pulbic-key Cryptosystems Based on Chebyshev Polynomials[J]. IEEE Tran. On Circuits and System-1:Regular Papers, Vol.52, No.7, PP.1382-1392, 2005

[7]G'erard Maze.: Algebraic Method for Constructing on Way Trapdoor Function[D]. Notre Dram: University of Notre Dame, 2003.

[8]Tomohire Yoshimur and Ttohru Kohda.: Jacobian Elliptic Chebyshey Rational Map [J]. Physical D., Vol.148, No.3-4, pp.242-254,2004

[9]Liu Liang, Liu Yun, Yu hongZhou.: Improvement and Characteristic Research of Chebyshev Polynomials in PKI [J]. Journal of Beijing Jiaotong University, Vol.29, No.5, pp.56-60, 2005

[10]Wang dahu, Wei Xueye, Li qingjiu, Liu yanhong: improvement in public-key encryption and key exchange scheme base on chebyshev polynomials[J]. Journal of the China Railway Socity, Vol28, No.5, pp.95-98,2006

[11]Kohda , Tohru , Fujisaki Hirohi , "Jacobian elliptic. Chebyshev rational maps"[J], Physica D , Vo.148, No.3, pp.242-254, 2001

[12]Wang Dahu.: Research on Nonlinear Theory in Secure Communica[D]. Doctoral Dissertation of Beijing Jiaotong University, 2006