International Journal of Soft Computing 12 (3): 142-147, 2017
ISSN: 1816-9503

Ensure Integrity and Authenticity Using Extended Dynamic Chaotic Based Hash Function for Mobile Data Security

[1]B. Madhuravani and [2]D.S.R. Murthy
[1]MLR Institute of Technology, Dundigal, Hyderabad, Telangana, India
[2]Geethanjali College of Engineering and Technology, Keesara, Hyderabad, Telangana, India

Abstract: Today with the progression of web and innovation security of data has turned into the prime worry in versatile and online applications. Broad measure of research has been done since years to give secure and solid hash capacities for data trade. Chaos based hash capacities have picked up a considerable measure of fascination by the researchers because of its non-linearity, irregularity and unusual outcomes. Different chaos based hash capacities have been actualized in the previous decade to accomplish privacy, uprightness and confirmation. In any case, the majority of the conventional chaos based hash functions are processed in sequential approach with a single dimensional array which limits their execution speed and execution in the versatile figuring applications. To beat these issues a novel parallel chaos hashing model is proposed in this study. This model extends the traditional dynamic chaotic system as extended DCS and integrates with a hash function to generate an n-bit hash value. This model gives more security, high calculation speed; restrain memory assets and less calculation overhead in the standalone and mobile applications.

Key words: DCS, extends DC, chaotic map, hash function, message digest, integrity, authentication

INTRODUCTION

Message process is a huge cryptographic calculation which has its application in advanced mark confirmation, message verification code, computerized steganography, computerized time stamping and so on. In the year 1995, national institute of standards and technology presented this calculation. Later different research endeavors are made to coordinate SHA with different ways to deal with accomplish augmented security and honesty. Hash work acknowledges variable-length message as information string and creates settled length process as yield in the wake of preparing. SHA can't counteract assaults and impacts of hash qualities which are the significant issue of this calculation. These calculations are not extremely effective for portable applications which are powerful in nature. SHA is additionally classified into various calculations SHA, MD5 and so on. Like other hash capacities, HA hinks of the message having self-assertive info length however a process of settled length. The first messages are changed over into pieces of little settled sizes.

For guaranteeing the respectability and legitimacy, the vast majority of the advanced applications for example, computerized reports, electronic mail, office mechanization and electronic assets exchange were actualized utilizing message process as a security parameter. Hash capacities are utilized as the essential part in different security conventions like TLS, SSL and S-MIME. Additionally, hash capacity is viewed as the center a portion of computerized mark and has picked up a great deal of consideration among the different specialists.

Confusion based hash capacities have picked up a great deal of fascination by the specialists in the field of distributed computing and versatile registering. Because of the restricted processing power, disorderly circles will get to be non-occasional. A large portion of the conventional turmoil based hash capacities are prepared in successive model which confines their execution speed and execution on the versatile figuring. Atighehchi and Muntean (2013) a parallel keyed hash work utilizing the disorder based calculation is proposed. The impediments in the parallel tumult model were displayed Ganesan *et al*. (2008). Different trial comes about have been performed on subjective messages lastly presumed that the parallel riotous hash capacity is not secured against the factual assaults.

To beat this issue, complex clamorous based hashing systems are produced which are non-straight, irregular and element in nature. It can be spoken to by either discrete or ceaseless frameworks. Henon mapping and strategic mapping can be arranged under discrete while lorenz and rossler framework goes under persistent. This framework is profoundly receptive to beginning conditions. Lyapunov examples are one of the imperative

Corresponding Author: B. Madhuravani, MLR Institute of Technology, Dundigal, Hyderabad, Telangana, India

segments of disordered framework which chooses whether the given framework is turbulent or not. Our proposed plot includes different consistent capacities and is created by complex fleeting conduct and gives high affectability to the underlying limitations of the high-dimensional riotous maps. The principle target of a turbulent hashing framework is the merging property in the complex disordered frameworks. The components in charge of this merging are:

- Both are deterministic
- Both are unpredictable and not unsurprising
- In disorderly framework

A little change in the underlying conditions can influence and mirror an immense change in the yield. Thus in hashing a minor change in the key or plain content will adjust the hashing yield as it were.

Chaotic hash functions: Chaotic hash functions can be classified in various ways according to their riotous plan or multi-threading capacity. Disorganized maps as base component of confused hash capacities are more describing for arranging them. In this way we received a few classes from (Kocarev and Lian, 2011) and included different classifications as takes after (Table 1):

- Simple map-based hash functions
- Complex map-based hash functions
- New chaotic system based hash functions
- Parallel and complex structured chaotic hash functions
- Conventional hash functions with chaotic modifications

Most simple chaotic based hash functions Yi (2005) and Amin *et al*. (2009) utilize a straightforward calculation with one dimensional clamorous maps in correlation with different classes. Yi (2005) a hash work utilizing tent guide and another variable like ais proposed. Yield arrangement of tent guide with a merkle-damgard plan is utilized to acquire the final hash esteem. Issues of the tent guide for example, hash work outline. Amin *et al*. (2009) Tent guide with a straightforward XOR plan is proposed. Any hash work that produces yield estimate under 256 bits is not secure; this outline with 128 bits in length

Table 1: Comparison on chaotic maps

Chaotic maps	Mathematical models	Merits	Demerits
Tent map	$f_\mu = \mu \min$	Faster: simple one dimension chaotic map	Non chaotic input rages are also carried to hash function design
Logistic map	$X_{n+1} = R\, X_n\, (1\text{-}X_n)$	Simple one dimension chaotic map	The statistical complexity Decreases as the control parameter increases No multithreading ability on data and computation level is considered
Sine map	$x_{n+1} = \lambda \sin(\Pi x_n)$	Simple one dimension chaotic map	No auto correlation and cross correlation
Lorenz map	$x = a(y\text{-}x)$ $y = cx\text{-}xz\text{-}y$ $z = xy\text{-}bz$	The Lorenz system is nonlinear, non-periodic, three-dimensional and deterministic	The experimental results show that the secret key can be reconstructed after one pair of known-plaintext/cipher text attacks. Furthermore, the effect of changing one bit in the plain image is a change in only one bit at the same position in an encrypted image
Chen system	$x^* = a(y_0\text{-}x_0)$ $y^* = (c\text{-}a)\, x_0\text{-}x0\text{-}x_0z_0+cy_0$ $z^* = x_0y_0\text{-}bz_0$	This scheme is fast and secure according to simulation results and large size of key space, respectively	Generates random number sequence which is more random in comparison with the sequence that was generated by logistic map in. The second drawback is overcome by setting the parameter of chen map using the last one byte of encrypted plaintext after every iteration that leads to a higher sensitivity of encrypted image to the plain one
Chebyshev chaotic map	$T_k(x) = \cos(k\cos^{-1}x)$	Applied to enhance anonymous message transmission, enhance the randomization and forward security of the interactions and session freshness is achieved to against a typical attack such as replay attack	$T_k(x)$ is not hard to invert when k is relatively small
Quadratic map	$X_{n+1} = r\text{–}(X_n)^2$	A very slight change of the initial value x_0 can lead to a significantly different behavior of the map	Not robust as limited value of parameter r
Dynamic Chaotic System (DCS)	$X_{n+1} = Cl_{\alpha,\beta}(w\infty,$ $w\,\beta, r, X_n, r)$	Complex map with multi-dimensional chaotic system	The distribution of r with respect to X_n is are easy predicted using the statistical and frequency attacks

hash esteem empowers assailant to utilize comprehensive assault for discovering impacts or second preimage with calculation many-sided quality of $<2^{64}$ because of birthday assault by Kocarev and Lian (2011). Displayed hash work by Amin *et al.* (2009) has been effectively assaulted. Both of this outlines utilize straightforward and one dimensional maps that makes them quick however no multi-threading capacity on information and calculation level is considered. Complex guide based hash capacities use multi-dimensional clamorous frameworks in their hashing calculation. Hash elements of Wang *et al.* (2008) Akhshani *et al.* (2009) and Akhavan *et al.* (2013) are in this classification. Higher measurement of utilized disordered maps and frameworks require network augmentation and increments that are slower contrasted with other tumultuous frameworks while memory required for registering higher measurement maps is likewise another bottleneck.

Previous works on chaotic hash function for the most part focus on statistical investigation, cryptanalysis and irregular conduct of yield hash values while uses of hash functions for example, crypto-currency and spam identification require more parameterization over its speed, security and yield hash value. Facilitate more must of past outlines utilize uncertain plans and has been effectively assaulted that makes this exploration territory still open to new plan and altered confused framework plans.

MATERIALS AND METHODS

Proposed model: Due to the non-linear features of dynamic chaotic system, conventional riotous maps for example, quadratic-map with sine outline, outline been broadly connected in the constant applications. In our research, we proposed a novel hashing model in view of extended dynamic chaotic system with parallel approach. The recommended model is quick and exact as far as speed and security is concern. In this model, various confused maps are coordinated as a solitary disorganized framework to produce a n-bit process esteem for a given info content M where n is any 32-bit esteem it is by and large chosen from the arrangement of 128, 256, 512 and 1024 piece esteem.

In this segment, we depict our model to extended DCS that defeats the issues of parallel clamorous frameworks. Conventional parallel confused frameworks neglected to instate the parameters to the disorganized maps. Likewise, instatement parameters are defenseless against a few surely understood assaults for example, change, dissemination and factual assaults.

Extended DCS: Traditional DCS (Dynamic Chaotic System) (Abdoun *et al.*, 2015) uses weighted parameters with the range 0-1. Also, existing DCS has uniform distribution over all the possible values of X_n and r while

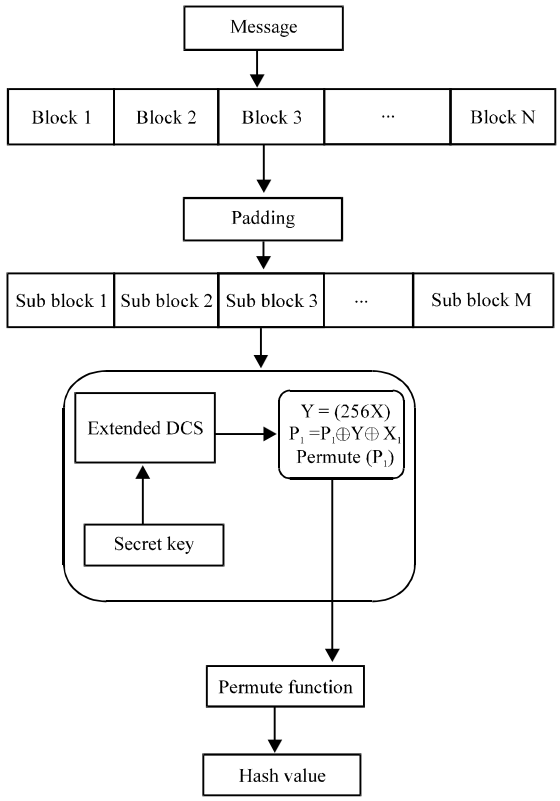

Fig. 1: Proposed model

two constant parameters are fixed as constant ($w_\alpha\alpha$ and $w_\beta\beta$). Since, r of DCS with logistic map is in the range of (0.16), the distribution of r with respect to X_n are easily predicted using the statistical and frequency attacks. To overcome these issues we extended the traditional DCS (Abdoun *et al.*, 2015) scheme with a non-linear chaotic map as a third parameter to improve the security in the mobile computing applications. In this extended model we used three weighted components α, β, γ within range (0.1) using the following Eq. 1:

$$EDCS = X_{n+1}(\alpha, \beta, \gamma) = w_\alpha\alpha(r.Ax_n^2 + w_\beta\beta)(16 - r)\,Ax_n^2) + w_\gamma\gamma\left(\frac{(w_\alpha\alpha + w_\beta\beta)}{2}\right)Ax_n^2 \quad (1)$$

The secret message M is divided in to chunks of n-blocks, (B_1, B_2, ..., B_n), each with r-length. Append padding bits $(10000...00)_2$ with length 'q' at the end of the message M. After padding, each block is again divided into 'm' sub-blocks, each with 32-bit length and it is represented as P_1, P_2.... P_m. Here, secret key is generated dynamically using the client mobile/PC's processor-id. In this system we have introduced an extended DCS. In each round function, iterative multi-chaotic system generates output to the transformation box (Fig. 1).

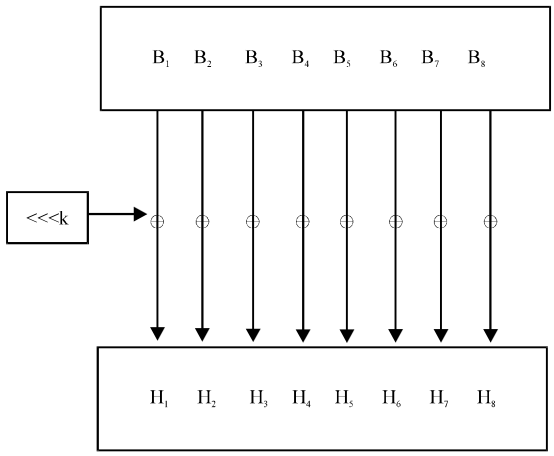

Fig. 2: Permute function

In the transformation box, input sub-blocks , chaotic output and scaling value are used to perform XOR operation and then permutate operation. Permutate operation divides the input XOR value into four 8-bit blocks to generate hash value as shown in Fig. 2.

Algorithm:

Input: M(Input message),Secret key, initialization parameters.
Step 1: Read a message M.
Step 2: If message size is not multiple of 'n' then
Append the bit sequence 1000…000 at the end of the message.
Divide the message into blocks of length n as .
Step 3: After padding, each block is again divided into 'm' sub-blocks,
Step 4: Secret key is generated using the client's mobile/PC processor-id as S. Generated secret key is initialized as X0 for exended DCS.
Step 5: In each round function, iterative chaotic system generates output Xi to the transformation box as shown in Fig. 1.
Step 6: In the transformation box, the following operations are performed on the Y and chaotic output.
$Y = \lfloor 256x \rfloor$
$P_i^1 = P_i \oplus Y \oplus X_i$
Permute (P_i^1)
Step 8: Generates Hash value as
H_i = Permute (P_i)
Hash = $H_1+H_2+H_3, ..., H_m$

RESULTS AND DISCUSSION

Performance analysis of a dedicated hash function can be categorized as:

- Statistical
- Hashing speed
- Cryptanalysis

Statistical analysis: Main parameters are Bi, B, P, ΔB and ΔP. Bi is hamming distance between two values that is calculated by XOR operation between two values and counting of 1's in result:

Table 2: Number of sample

Variables	256	512	1024	2048
B (min)	115.00	108.00	118.00	113.00
Avg. (B)	135.75	132.87	136.75	131.87
P (%)	50.11	50.12	50.18	50.28
ΔB	5.58	5.60	5.61	5.62
ΔP	4.16	4.18	4.29	4.40

Table 3: Hash function

Hash function	P (%)	ΔB	ΔP
MD5 (Rivest)	50.02	5.66	4.42
Xiao *et al.* (2005)	49.88	5.79	4.52
Xiao *et al.* (2009)	50.01	5.72	4.47
Li *et al.* (2012)	49.58	5.76	4.50
Wang *et al.* (2008)	50.12	5.77	4.51
Teh *et al.* (2015)	50.01	5.61	4.38
Chenaghlu *et al.* (2016)	50.09	5.63	4.41
This work	50.28	5.62	4.40

Table 4: Message size (MB)

Message size (MB)	T_R	T_P	Speedup
50	1.9868	0.9746	2.0386
100	3.6792	1.7599	2.0906
150	5.2690	2.2493	2.3425
200	8.2798	3.2963	2.5118
250	9.6920	4.1992	2.3081

$$B = \frac{1}{N}\sum_{i=1}^{N} B_i$$

Probability of humming distance is defined as:

$$P = (\frac{B}{1}) \times 100\%$$

In spite of the fact that P a B are great parameters to decide measurable execution of a hash work, they can have expansive standard deviation. Bigger standard deviations on B and P indicates terrible measurable execution:

$$\Delta B = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(B_i - \overline{B})^2}$$

$$\Delta P = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(\frac{B_i}{1} - P)^2} \times 100\%$$

Statistical analysis of "yahoonews" text datasets using proposed model Table 2. Compared statistical analysis of other algorithms with this work for 2048 number of samples Table 3.

Hashing speed: For a 250 MB irregular information, hashing speed in threading mode is 93.34 MB/s with 8 strings while in general mode (one string) it is 29.53 MB/s Table 4. Average speed up between threaded and regular mode is 2.2583 compared to 1.7475(Chenaghlu *et al.*, 2016) (Fig. 3).

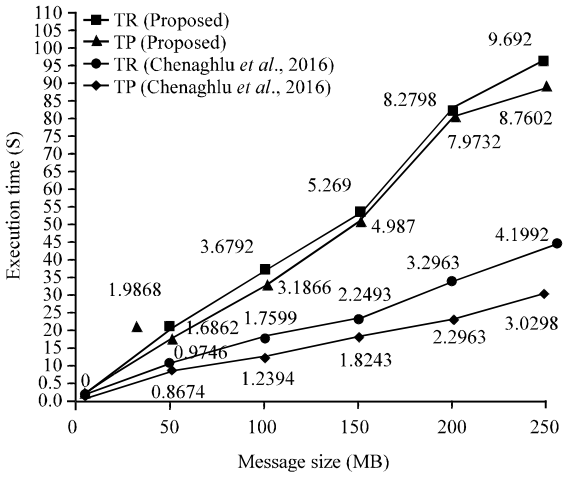

Fig. 3: Hashing speed comparison in threaded and regular mode

Cryptanalysis results

Random-attack: Takes a message M as default message and tries to discover impacts by arbitrarily flipping bits or utilizing piece change. This assault does not require any learning about hash work and can be performed effortlessly. Measurable investigation demonstrated our hash capacity's power against this assault.

Birthday attack: As finding collisions with computing all of possible hash values. As birthday-paradox proves the possibility of a group of 23 randomly chosen people having the same birthday is 50%, same goes for a hash function. Performing this attack needs $2^{n/2}$ computations. Our Proposed hash function produces hash lengths lager than 256 bits thus this attack is not practical.

Exhaustive key search: Can be performed on any figure or calculation that takes a key as info. Length of key can portray capacities security against this assault. Our hash work utilizes equipment distinguishing proof as key (256 piece) which is too huge

Meet-in-the-middle attack: Cannot be applied to our scheme as we use multiple chaotic maps for each round.

Spoofing attack: Is infeasible as proposed model depends on system id to generate a secret key.

CONCLUSION

Security systems ought to be sufficiently hearty to adapt up to substantial web movement. We proposed a novel parallel riotous hashing model with trial comes about whose structure guarantee the arbitrariness, high affectability and impact resistance. This model gives more security, high calculation speed, restrain memory assets and less calculation overhead in the stand alone and mobile application.

REFERENCES

Abdoun, N., S.E. Assad, M.A. Taha, R. Assaf and O. Deforges *et al*., 2015. Hash function based on efficient chaotic neural network. Proceedings of the 10th International Conference on Internet Technology and Secured Transactions (ICITST), December 14-16, 2015, IEEE, London, England, UK., ISBN:978-1-9083-2052-0, pp: 32-37.

Akhavan, A., A. Samsudin and A. Akhshani, 2013. A novel parallel hash function based on 3D chaotic map. EURASIP J. Adv. Signal Process., 2013: 1-12.

Akhshani, A., S. Behnia, A. Akhavan, M.A. Jafarizadeh and H.A. Hassan *et al*., 2009. Hash function based on hierarchy of 2D piecewise nonlinear chaotic maps. Chaos Solitons Fractals, 42: 2405-2412.

Amin, M., O.S. Faragallah and A.A.A. El-Latif, 2009. Chaos-Based Hash Function (CBHF) for cryptographic applications. Chaos Solitons Fractals, 42: 767-772.

Atighehchi, K. and T. Muntean, 2013. Generic parallel cryptography for hashing schemes. Proceedings of the IEEE 12th International Symposium on Parallel and Distributed Computing (ISPDC), June 27-30, 2013, IEEE, Bucharest, Romania, ISBN: 978-1-4799-2967-2, pp: 201-208.

Chenaghlu, M.A., S. Jamali and N.N. Khasmakhi, 2016. A novel keyed parallel hashing scheme based on a new chaotic system. Chaos Solitons Fractals, 87: 216-225.

Ganesan, K., I. Singh and M. Narain, 2008. Public key encryption of images and videos in real time using Chebyshev maps. Proceedings of the 5th International Conference on Computer Graphics, Imaging and Visualisation CGIV08, August 26-28, 2008, IEEE, Penang, Malaysia, ISBN: 978-0-7695-3359-9, pp: 211-216.

Kocarev, L. and S. Lian, 2011. Chaos-Based Cryptography: Theory, Algorithms and Applications. Vol. 354, Springer, Berlin, Germany, ISBN:978-3-642-20541-5, Pages: 396.

Li, Y., D. Xiao and S. Deng, 2012. Keyed hash function based on a dynamic lookup table of functions. Inf. Sci., 214: 56-75.

Teh, J.S., A. Samsudin and A. Akhavan, 2015. Parallel chaotic hash function based on the shuffle-exchange network. Nonlinear Dyn., 81: 1067-1079.

Wang, Y., X. Liao, D. Xiao and K.W. Wong, 2008. One-way hash function construction based on 2D coupled map lattices. Inf. Sci., 178: 1391-1406.

Xiao, D., X. Liao and S. Deng, 2005. One-way hash function construction based on the chaotic map with changeable-parameter. Chaos Solitons Fractals, 24: 65-71.

Xiao, D., X. Liao and Y. Wang, 2009. Parallel keyed hash function construction based on chaotic neural network. Neurocomputing, 72: 2288-2296.

Yi, X., 2005. Hash function based on chaotic tent maps. IEEE. Trans. Circuits Syst. Express Briefs, 52: 354-357.