Advances in Science, Technology and Engineering Systems Journal
Vol. 2, No. 3, 540-544 (2017)
www.astesj.com
Special Issue on Recent Advances in Engineering Systems

ASTES Journal
ISSN: 2415-6698

An Encryption Key for Secure Authentication: The Dynamic Solution

Zubayr Khalid[*,1], Pritam Paul[1], Khabbab Zakaria[2], Himadri Nath Saha[1]

[1]*Institute of Engineering and Management, Computer Science and Engineering, 700091, India*

[2]*Jadavpur University, Department of Power Engineering, 700098, India*

ARTICLEINFO

Article history:
Received: 06 April, 2017
Accepted: 15 May, 2017
Online: 01 June, 2017

Keywords :
AES
Cryptography
Dynamic Password
EYE
Feistel Block Cipher

ABSTRACT

In modern day technology, the Information Society is at risk. Passwords are a multi-user computer systems usual first line of defence against intrusion. A password may be textual with any combination of alphanumeric characters or biometric or 3-D. But no authentication protocol is fully secured against todays hackers as all of them are Static in type. Dynamic authentication protocol is still a theoretical concept. In this paper, we are focusing on a concept of authentication technique which is actually dynamic in genre, i.e. the password here will change in t time (where t is as small as possible). This technique comprises of both hardware and software part. In this paper, we have covered the idea of generating an efficient algorithm that can work as the final in the Dynamic Password Authentication system. We have used standard deviation within statistics to generalize the possible password which is further secured by Feistel Block Cipher and Advanced Encryption Standard technique (AES), leading and following the said mathematics respectively. In order to allow the system to create variable password in the least time interval possible, we must make sure our process is not much complex.

1 Introduction

This paper is an extension of the research work named Secure Authentication with Dynamic Password presented in the IEEE IEMCON, , Vancouver, Canada, 2016. [1]

When it is the matter with network security, we have to be updated always. As for example, today we simply cannot afford any security system that includes a dilapidated protocol (as for example Caesar Shift Cipher). Security protocols are used to protect several different medium using passwords, which supposed to be known only to the users of the system. Textual password authentication system is the most widely used protocol, though it is not much secure against Shoulder Surfing Attack, Brute Force Attack [2]. Smartcard concept, 3-d password concept are not totally protected either [3], [4]. Biometric system may be the most powerful authentication system ever, but it is much costly [2]. Dynamic password authentication system is the latest system that has come in the field. In this work we have discussed about the cryptographic way to generate the dynamic password for the authentication protocol described in the said conference. [1]

The authentication protocol described in the conference is a combination of both the hardware and software. Here we are imagining a chip which will be used as the key to unlock different ids. According to the main work, we have taken a random textual password e.g. Password. The ASCII values of each letter acts as equivalence to mass where we have used Einstein's Theory on bending Space-Time Continuum because of Mass. Now we have placed the textual password (here Password) on 3-d box which is described by the X-Y-Z coordinates.[1]

Later we have used the concept of the eye. The EYE (technically a sensor) is the only object inside the 3-D box.

The EYE will actually take a random position inside the 3-D box (i.e. the X, Y, Z coordinates of it will be selected randomly). Now the EYE will trace the endings of the shifted parts in the plane by joining them.

[*] Zubayr Khalid, Computer Science and Engineering, Institute of Engineering and Management, +917596872617 & write2zubayr@gmail.com

https://dx.doi.org/10.25046/aj020369

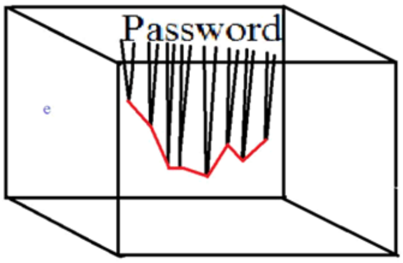


Fig.1: Traced path according as the position of the eye

If we manage to let the eye to change its position in t time (t is as small as possible), then during each time interval we will have a different path as shown.

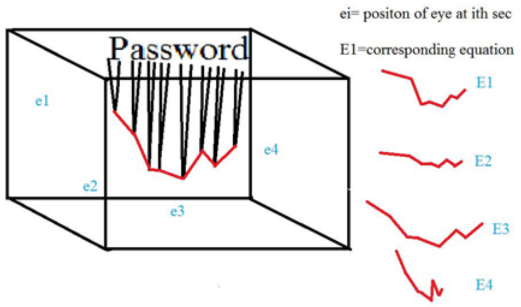


Fig.2: Traced path by the eye

Since at a certain time, the received traced path is the key of the real password, by changing the position of the EYE, and hence by getting different traced path in each time interval t, we gain the dynamically changeable password.

In this paper we have tried to make an algorithm that will be able to convert the traced path (technically a geometric figure) to a combination of binary digits (0 and 1). This expression gained at a certain time will be the password at that time interval. We will try to add some cryptographic concept in the algorithm to make a final defence in the system. The mathematical idea used to convert the geometric figure an expression of standard deviation (in decimal) will be both followed by and lead two different cryptographic concepts the first one is general Feistel Block Cipher concept,[5], [6] and the later one is AES (Advanced Encryption Standard) algorithm.

2 Security with Feistel Block Cipher Model

We will move further assuming that the textual password chosen by the user was Password. Before going to the generation of the password from the traced path, we need to find the coordinate of the points of the path, which we will do in this process. We will follow the general Feistel Block Cipher concept, i.e. at first, we will divide the plaintext into 2 equal parts say L and R. The plaintext will be the ASCII representation of the textual password. Hence we have the plaintext ASCII representation of Password 8097115115119111114100. [5],[6]

Now the string will be divided into 2 halves based on the length of it. If the length of the string is found even, the L, R will be of equal length. If the length of the string is found odd, the length of L will be 1 greater than that of R.

Here we have the length 22. Hence L will be 80971151151 and R will be 19111114100. Each digit in L and R are added and the results will be stored in R‘ and L‘ respectively. The addition of digits in L is 39 and the addition of digits in R is 20. Hence R‘ will store 39 and L‘ will store 20.

We will count (R‘ modulo 2) and (L‘ modulo 2). To find the axis of the traced path, we will follow a concept

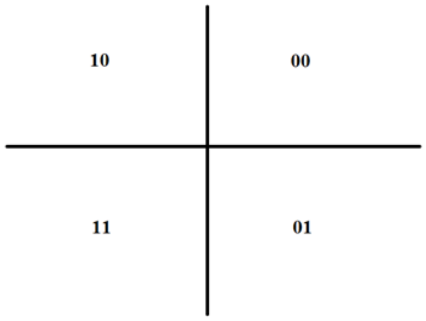
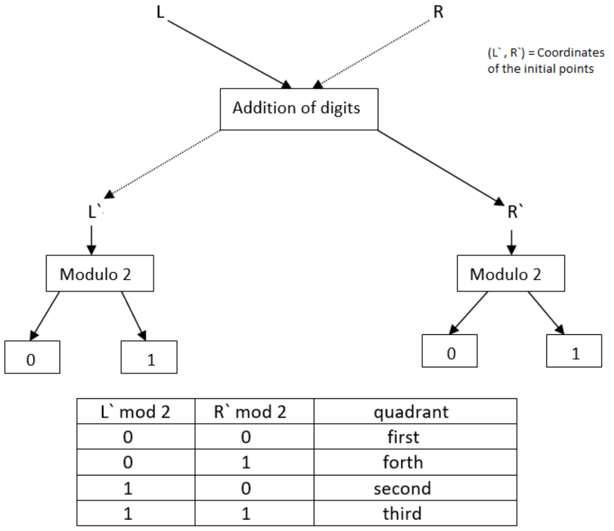


Fig.3: X-Z Coordinate Plane

As for example if (L‘ modulo 2) is 1 and (R‘ modulo 2) is 1 then we will consider that the traced path will be in 3rdquadrant and proceed to the mathematics (hence 0 represents positive quadrant and 1 represents negative quadrant). The values of L‘ and R‘ will be the x-coordinate and z-coordinate of the initial point in the traced path respectively.



L` mod 2	R` mod 2	quadrant
0	0	first
0	1	forth
1	0	second
1	1	third

Fig.4: Feistel Block Cipher Model to get quadrant and initial coordinate

In our example, (L‘ modulo 2) is 0 and (R‘ modulo 2) is 1. Hence the traced expression will be in the 4thquadrant. The initial point will be (20, -39). It should be mentioned that the graph, starting from a specific quadrant, may end in a different quadrant.

A case to be noted that in this concept we havent used Feistel Block Cipher completely. Here we havent used any key and there is no functioning it with L. Instead, we have changed both L and R to R′ and L′ respectively. The swapping of L and R to R′ and L′ is done only once instead of having several rounds.

3 Derivation of the Traced Path

Suppose there are n letters in the password. So, we have n corresponding extreme points in the x-z graph. Here, we connect each 2 consecutive extreme points to get one straight line. Thus, in total we get (n-1) of such straight lines, each having a definite equation f(z,x)=0. The equation of each straight line is provided by the system.

Our job is to find variance (similar to what is used in case of problems with large number of bodies)[7], and thus standard deviation for the distance of each part from x axis (z) for each of (n-1) Straight lines.

$$Variance = \sigma^2 = <z^2> - <z>^2 \tag{1}$$

$< z >= \int_a^b z\rho(x)dx$ where (a, z_1), (b, z_2) are the two extremities of a Straight lines where $b > a$. Here $\rho(x)$ is the probability density such that $\rho(x)dx$ defines the probability of an individual being in the interval of (x,x+dx).

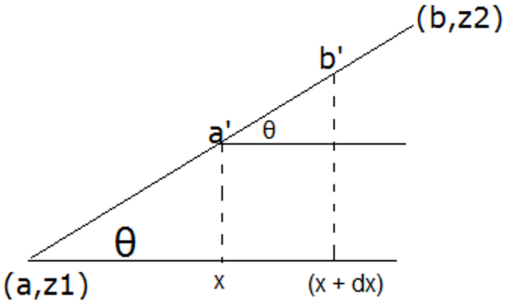


Fig.5: Derivation of traced path

Here, length of $a'b' = dxSec\theta$
Length of $ab = xSec\theta$
So,

$$\rho(x)dx = \frac{dx}{x} \tag{2}$$

Therefore,

$$< z >= \int_a^b \frac{zdx}{x} \tag{3}$$

$$< z^2 >= \int_a^b \frac{z^2dx}{x} \tag{4}$$

Performing the integrations, we can have σ^2
Now, standard deviation= σ
Thus, we will have (n-1) no. of standard deviations for (n-1) of Straight lines.

Let, these standard deviations are σ_1, σ_2,......... $\sigma_{(n-1)}$

Let, $\sigma_k = \Sigma_{i=1}^{n-1} \frac{\sigma_i}{n-1}$, is the standard deviation.

Standard Deviation of those standard deviations is:

$$\sigma_p = \frac{\sqrt{\Sigma_{i=1}^{n-1}(\sigma_k - \sigma_i)^2}}{n-1} \tag{5}$$

σ_p is the password at that moment.

Come to think about it, for a password with n letters, there are $(26+10+30)^n = 66^n$ [26⟶ English letters, 10⟶numbers from 0 to 9, 30⟶special characters] no. of different possibilities, and thus 66^n no. of different combinations of Straight lines in z-x graph. Each of the point in z-x graph can arbitrarily be on each of the 4 coordinates. So, there are 4^npossibilities for each of 66^n different possibilities. Finally, the final standard deviation σ_p depends on the position of average of different σ_i. There are (n-1) of such partitions.

So, probability of a random password, matching the real password be:

$$Probability = \frac{1}{(66\times 4)^n \times (n-1)} = \frac{1}{264^n \times (n-1)} \tag{6}$$

For a 8 digit password, this probability is $\sim 6 \times 10^{-21}$

4 Final Security with AES Algorithm

This process is the final process in the whole system. Here we will use Advanced Encryption Standard (AES) to get the encrypted cipher text [8]. The cipher text will be the actual password in the dynamic password authentication system.

Reasons behind selecting AES encryption for our process:

(i) AES is one of the most efficient and strongest encryption systems.

(ii) Here we are not much concern of any decryption of the encrypted text (i.e. the password). Hence theres no necessity of choosing asymmetric encryption system like RSA. [9]

(iii) Comparing Twofish encryption system and AES encryption system, Twofish slightly leads in terms of security. But AES is faster than Twofish. Since our system gets high security due to the changeable password, we will allow it to change in the minimum time interval possible. Hence we are choosing AES for our authentication system. [10]

According to AES system, we will have a plaintext and key of 128 bits and there will be total 10 rounds-9 regular rounds and 1 final round.

We will get the plaintext from the standard deviation of the previous mathematics.

Let the output from the mathematics is 6.4467. To have a plaintext of 128 bit (i.e. 16 bytes), the output is multiplied with pi (round 16 bytes decimal). Here the output will be:20.25290535989732. Each digit excluding the point (2, 0, 2, 5, 2, 9, 0, 5, 3, 5, 9, 8, 9, 7, 3, 2) is considered and the corresponding binary digit in 8 bit is calculated (i.e. 00000010, 00000000, 00000010, 00000101, etc). Hence we have 128 bit plaintext.

To get the key, the textual password (here Password) is converted to binary. To get a 128 bit (16 byte) key the textual password will have to be of 16 letters. If the textual password is less than 16 letters then there will be a repetition of letters until it reaches 16 letters. If there is more than 16 letters then there will be excise of letters from right to left until it reaches 16 letters. At the end there will be text to binary conversion.

As for example, for the textual password Password since there is 8 letters, after updating we have PasswordPassword. Hence the key will be: 01010000 01100001 01110011 01110011 01110111 01101111 01110010 01100100 01010000 01100001 01110011 01110011 01110111 01101111 01110010 01100100.

Now we will move for AES algorithm.

The algorithm our system will follow can be described in the figure: [11]

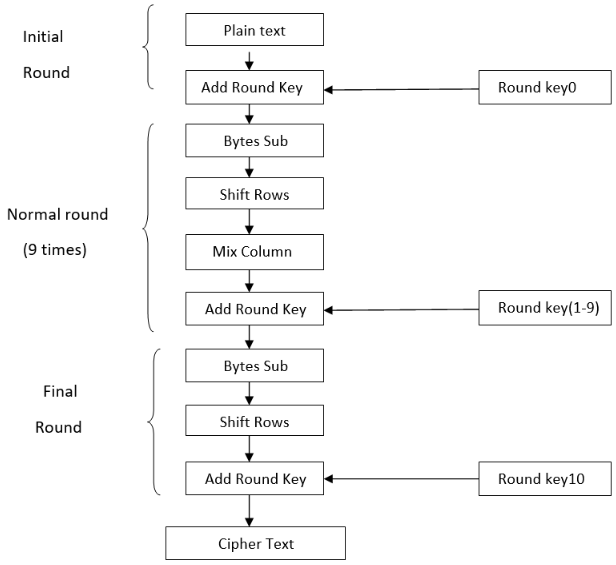


Fig.5: AES Algorithm

The cipher text we get after whole 10 rounds is the actual password of the system. It is to be noted that we havent done anything with decryption since we are not concerned with the decoding of password to original text.Since the password consists of 128 different characters each having 2 possibilities (0,1), the number of different possible passwords is $2^1 28$.

5 Conclusion

In the main work we have already clarified that the concept is secured against most of the common security threats like Brute Force Attack, Timing Attack, Key Logger and Shoulder Surfing Attack. Moreover, it is shown that the Space and Time Complexity of the concept is less than some other common authentication techniques. The whole idea has been developed to make a secure yet simple concept, creating a user-friendly platform. The time interval t between two different traced paths will depend on the time required for the execution of the two encryption algorithms (Feistel Block Cipher and Advanced Encryption Standard technique), along with the completion of the standard deviation. Less is the t, greater is the possibility of having 2 different traced paths in a time interval T, hence greater is the security. The algorithms used in generating the binary password from the graphical traced path are recognised as some secure encryption techniques. Hence they can perform as the final defence as well. At the end we should mention that since we have worked with a completely theoretical concept, it should go through an experimental procedure to detect the flaws and disadvantages.

References

1. Z. Khalid, P. Paul, S. P. Chattopadhyay, A. N. Biswas, Secure Authentication with Dynamic Password, in 2nd IEEE IEMCON and 7th IEEE Annual Information Technology, Electronics and Mobile Communication Conference, Vancouver Canada, 2016.

2. C. Kuo , S. Romanosky , L. F. Cranor , Human Selection of Mnemonic Phrase-based Passwords, [online] available- $https://cups.cs.cmu.edu/soups/$2006/

3. By the editors of Time-Life Books, Computer Security (Understanding Computers),Time-Life Books , 1990.

4. F. A. Alsulaiman, A. El Saddak, A Novel 3D Graphical Password Schema, in Virtual Environments, Human-Computer Interfaces and Measurement Systems, Proceedings of 2006 IEEE International, La Coruna Spain, 2006

5. W. Stallings, Cryptography and Network Security (Fourth Edition), Pearson Education Inc., 2006

   $proceedings/p67_kuo.pdf$, 2006

6. C. De Canniere, A. Biryukov, B. Preneel, An Introduction to Block Cipher Cryptanalysis, Proceeding of The IEEE, Vol. 94, No. 2, 2006

7. D. J. Griffiths, Introduction to Quantum Mechanics, Prentice Hall, Inc. USA, 1995.

8. D. Selent , Advanced Encryption Standard, Rivier Academic Journal, Volume 6, Number 2, 2010

9. P. K. Arya, M. S. Aswal, V. Kumar, Comparative Study of Asymmetric Key Cryptographic Algorithms, International Journal of Computer Science & Communication Networks, Volume 5, Issue 1, 2015

10. S. A. M. Rizvi, S. Z. Hussain, N. Wadhwa, Performance Analysis of AES and Twofish Encryption Schemein International Conference on Communication Systems and Network Technologies, Katra, Jammu, India, 2011.

11. H. Lee, K. Lee, Y. Shin, AES Implementation and Performance Evaluation on 8-bit Microcontrollers, (IJCSIS) International Journal of Computer Science and Information Security, Vol. 6, No. 1, 2009