New Algorithm for Generating Complex Password to Be Applied in Several Applications

Hussein L.H.AL.Moteriy[1]Hadab Khalid Obayes[2]

College of Education for Human Sciences, University of Babylon, P.O Box: 4 Babylon –Hilla-Iraq

[1]hus_loia@msn.com[2]hedhabsa@gmail.com

Abstract

The proposed paper present new algorithm for generate complex key or complex password that can be used in the information security, which is the most important filed in computer science today. This algorithm contains many stages; these stages promote formation of an efficient password that requests in many applications such as website, social networks and applications under any operating system. Always this applications needs to change the classic and traditional methods that are supposed responsible of provides maximum security for applications users. The system will produce password has nothing any relation with the initial password and the second important feature is any little change in initial password even if the change was substitutions places lead to different result, thus the proposed system achieves confusion and diffusion.

Keywords: complex password, key generation, information security

1. *Introduction*

The need is increasing day after day to enhance security of many applications, in this applications the user is keen on his/her information thus any weakness in security of applications lead to collapse of confidence in use that applications and that failure in maintaining the secret of information leads to information to be easy prey of any attacking, the applications of network a strong and confidential password is essential, not just for financial sites, but for social networking sites too. With social networking sites like Facebook and Twitter, there is the danger of people faking their way into the site and posting something embarrassing about you or others (Magid 2012).

Here, the authentication plays important role to identify the person who has the right of access to the system, or account.

User authentication methods can be classified broadly into 4 types depending on the elements used for authentication (Wu 1998).

First is the method based on what user knows such as password and PIN (Personal Identification Number).This method has the problem that conjecture is easy and safety for password transferred is not guaranteed due to various crack tools(Hyun-Chul Kim 2008).

Second method is based on the object user owns such as smart card and token. This method provides strong authentication compared to password authentication but has the problem of loss or theft (Eric Cole 2009).

Third is the authentication method based on the information existing in user's body like fingerprint. This method is stronger than the previous two methods in that it uses user's physical information but requires introduction of high priced hardware equipment for application, and since user's physical information need to be used, it can cause offensive feeling to users along with authentication accuracy problem(Eric Cole 2009).

Last method is the authentication based on user's behavioural information such as signature and voice. This method solves offensive feeling problem of authentication that can occur to users. However, in the case of Asians, signature is not that widely used culturally and signature consistency is often not maintained. For reasons as above, this method is largely used for verification of whether requester is the principal or not than for recognition like verification upon electronic payment or log-in. Voice method uses voice flow characteristics to check an individual but voice quality is not always consistent and has noise problem by communication condition (Michael E.Whitman, Herbert J. Mattord 2011)Among such authentication protocols, password method is the most widely used for reason that password is easy to remember and is inexpensive.

2. *The Password And Access Control*

The use of passwords goes back to ancient times when soldiers guarding a location by exchange a password and then only allow a person who knew the password. In modern times, passwords are used to control access to protect computer operating systems, mobile phones, auto teller machine (ATM) machines, and others. A typical computer user may require passwords for many purposes such log in to computer accounts, retrieving e-mail from servers, accessing to files, data bases, networks, web sites, and even reading the morning newspaper online (Eric Cole 2009).

Access control needs identification that can be defined as asserts an identity, the identification plays role protection of an actual individual or application or service into computer system. The entity asserting the identity is commonly called a supplicant.

The access control mechanism checks the user's identity by means of password, passphrase, or other unique authentication code, such as a personal identification number(PIN)(Michael E.Whitman, Herbert J. Mattord 2010).

The password is defined as private word or combination of symbols that only the user should know. One of the biggest debates in security focuses on the complexity of password. A password should be difficult to guess, which means it cannot be a word that is easily associated with the user, such as the name of a spouse, child, or pet. Nor should it be a series of numbers easily associated with the user, such as a phone number, social security number, or birth date. At the same time, the password must be easy to remember(Eric Cole 2009)**.**

The passphrase is defined as a plain –language phrase, typically longer than a password, the virtual password is derived from passphrase. For example while a typical password might be (Kf12earth), a passphrase could be (How To Create And Remember Strong Passwords) from previous *passphrase the virtual password (HTCARSP) is derived***(Eric Cole 2009)**

3. The Password Infrastructure

The technical infrastructure for password of mechanism is built into commonly used computer and network operating system software and is in use unless it has been deliberately disabled, password is stored in a file, or perhaps stored in a database if the application or system deals with a group of users. The problem statement here are

- The attacker could have access to the password file or database in the system or network
- The attacker possible to know the initial password for the user.

4. Stages Of The Algorithm

In the algorithm there are seven stages, the goal of these stages is to produce values in every stage by different tactic and down to create a 36 bytes of binary numbers ,which saved in Database of system as hexadecimal numbers, that is password final.

4.1 First Stage

In this stage the algorithm requires from user to enter two keys, the first one is the user will enter directly (we will call the initial password key (IPK)), the second one is stored in file (we will call the initial password key file (IPKF)) the first key is IPK, and second key is IPKF. IPK must contain at least 6 bytes of symbols and at max 10 bytes of symbols, here the user must remember the first key, after that the user must enter IPKF, it prefer to be different of IPK and must be same size of IPK with at least 6 bytes of symbols and at max 10 bytes of symbols.

In this stage, the algorithm will create protection file contains IPKF, which send to user to keep it. After that, the algorithm will pass IPK and IPKF to second stage without save IPK .The algorithm saved IPKF in file No. 1.

4.2 Second Stage

In this stage, the algorithm will convert the symbols of IPK and IPKF to the offset by ASCII table as decimals numbers, the algorithm create two matrix for each key, the first matrix is horizontal, the second matrix is vertical and pass the matrixes to third stage.

4.3 Third Stage

In this stage the algorithm enter each matrix into different mathematic formal, where the each value of first matrix enter into formal 1

Where

k=1, 2, 3… n k=value index in horizontal matrix

HM= the horizontal matrix , v=value in matrix

|F1 (xk)|= (HM (vk) +k) – (2*k)

In same time the each value of second matrix enter into formal 2

k=1, 2, 3... nk=value index in vertical matrix

VM=the vertical matrix, v=value in matrix

|F2 (xk)|= (VM (vk) +k) – (3*k)

The algorithm will create two matrixes, the first matrix is A which contains results of formal 1 where this matrix is horizontal:

A= [F1(x1), F1(x2), F1(x3) …F1 (xn)]

In same time, the second matrix is B that contains results of formal 2 where this matrix is vertical:

B= [F2(x_1)
F2(x_2)
F2(x_3)
$\vdots$
F2 (x_n)]

The result matrixes A, B pass to next stage.

4.4 Fourth Stage

At this stage, the algorithm will multiplying the matrix B with the reverse matrix A to produce a matrix C, which have dimensions K*K

We know that the dimensions of matrix A, B at least are 6, so we suppose

A= [a1a2a3a4a5a6]

Where reverse of A

A= [$a_6a_5a_4a_3a_2a_1$]

B= [b_1

b_2

b_3

b_4

b_5

b_6]

Therefore, the process of multiplying B with the reverse A produce matrix C at least the dimensions are 6 *6 as

C=[$c_{11}c_{12}c_{13}c_{14}c_{15}c_{16}$

$c_{21}c_{22}c_{23}c_{24}c_{25}c_{26}$

$c_{31}c_{32}c_{33}c_{34}c_{35}c_{36}$

$c_{41}c_{42}c_{43}c_{44}c_{45}c_{46}$

$c_{51}c_{52}c_{53}c_{54}c_{55}c_{56}$

$c_{61}c_{62}c_{63}c_{64}c_{65}c_{66}$]

The algorithm will pass the matrix C to next stage.

4.5 Fifth Stage

At this stage the algorithm will create matrixes D, E after substituted the positions of rows and columns in the matrix C, the first step in this stage is create matrix D which contains in first columns the columns of matrix C which holds even positions, the first column in matrix D is column in matrix C which holds maximum even position, the process continues down to the last column holds the minimum even position which equal 2 in matrix C while it will equal column n in matrix D , the process is continuing to transfer the odd columns from matrix C to matrix D which start at column n+1 in matrix D and begin at column which holds minimum odd position which equal 1 up to maximum odd position in matrix C.

After completion create matrix D, the second step is create matrix E depended on matrix D. Where matrix E contains in first rows the rows of matrix D, which hold seven positions, the first row in matrix E is row in matrix

D which holds maximum even position, the process continues down to the last row holds the minimum even position which equal 2 in matrix D while it will equal row m in matrix E, the process is continuing to transfer the odd rows from matrix D to matrix E which start at row m+1 in matrix E and begin at row which holds minimum odd position in matrix D.

If we suppose the matrix C at least dimensions 6*6

C=[$c_{11}c_{12}$ c_{13} c_{14} c_{15} c_{16}
c_{21} c_{22} c_{23} c_{24} c_{25} c_{26}
c_{31} c_{32} c_{33} c_{34} c_{35} c_{36}
c_{41} c_{42} c_{43} c_{44} c_{45} c_{46}
c_{51} c_{52} c_{53} c_{54} c_{55} c_{56}
c_{61} c_{62} c_{63} c_{64} c_{65} c_{66}]

The matrix D will be

D=[c_{16} c_{14} c_{12} c_{11} c_{13} c_{15}
c_{26} c_{24} c_{22} c_{21} c_{23} c_{25}
c_{36} c_{34} c_{32} c_{31} c_{33} c_{35}
c_{46} c_{44} c_{42} c_{41} c_{43} c_{45}
c_{56} c_{54} $c_{52}c_{51}$ c_{53} c_{55}
c_{66} c_{64} c_{62} c_{61} c_{63} c_{65}]

The matrix E will be:

E=[c_{66} c_{64} c_{62} c_{61} c_{63} c_{65}
c_{46} c_{44} c_{42} c_{41} c_{43} c_{45}
c_{26} c_{24} c_{22} c_{21} c_{23} c_{25}
c_{16} c_{14} c_{12} c_{11} c_{13} c_{15}
c_{36} c_{34} c_{32} c_{31} c_{33} c_{35}
c_{56} c_{54} c_{52} c_{51} c_{53} c_{55}]

The algorithm will pass the matrix E to next stage.

4.6 *Sixth Stage*

At this stage, the algorithm will convert the received matrix into one dimension (one-row and multi columns) therefore the matrix E appears

E=[c_{66} c_{46} c_{26} c_{16} c_{36} c_{56} C_{64} c_{44} c_{24} c_{14} c_{34} c_{54} ……………… $c_{35}c_{55}$]

4.7 Seventh Stage

At this stage the algorithm will convert the elements of matrix E from decimal to binary system therefore this process will create matrix F without representation spaces.

F= [010011100011001011100……………………0001011]

The next process is remove of bits and select others bits to create 36 bytes , the algorithm will select and use one from multi sequences mathematical which start at

S_1= 1,2,3,5,8,13

Next sequences are

S_2=2, 3,5,8,13,21
S_3=3, 4, 7,11,18,29
S_4=4, 5, 9,14,23,37
S_5=5, 6, 11,17,28,45

Where the algorithm use the elements of sequence to remove the bits from matrix F.

The algorithm detects the fit sequence depends on number of elements of matrix A

Where:

Matrix A= 6 elements ⟹ s_1
Matrix A= 7 elements ⟹ s_2
Matrix A= 8 elements ⟹ s_3
Matrix A= 9 elements ⟹ s_4
Matrix A= 10 elements ⟹ s_5

Assuming that the fit sequence is s1

S1= 1 (used to remove one bit), 2(to remove two bits), 3(to remove 3 bits)…

Therefore the process is
1: move 1 bit to file No.2
2: select and save next 8 bits
3: move next 2 bit to file No.2
4: select and save next 8 bits
5: move next 3 bits to file No.2
6: select and save next 8 bits
7: move next 5 bits to file No.2
8: select and save next 8 bits
9: move next 8 bits to file No.2
10: select and save next 8 bits
11: move next 13 bits to file No.2
12: select and save next 8 bits

The algorithm repeats the process 6 times to create 36 bytes.

The matrix that results from the process is matrix G, which consists of elements that selected and stored in the previous process.

In addition, the algorithm will compile bits transmitted in a File No.2 consecutively with the remaining bits of the matrix F after the completion of the process of building the matrix G.

File No.1 that contains IPKF and File No.2 is stored on storage media own user for retrieving user information in the event that the user forgets IPK. Where the algorithm will reverse steps to provide IPK for the user.

The next step is convert each 8 bits from binary system to hexadecimal system

Matrix H will receive the results of the conversion, which consists of digits in hexadecimal system.

Matrix H is the final output of the algorithm.

5. User Authentication

Process of The user authentication starts when the user enter the IPK and up load IPKF to system, the user authentication applies algorithm to generate password final, which compare with database of system that contains password, Here IPK and IPKF must pass through the algorithm as a condition for the comparison and authorization to accept or refuse user.

6. Examples With Discussion

Ex1: The length of IPK is 7 symbols (7 bytes) which is

7Da!-Lq

So the length of IPKF must be 7 symbols (7 bytes) which is

hk2=xR8

The final key (password) is

201B9A69B050FB302EB81C349489F85F21E39E25378B65704213080133626699488196259

Ex2: The length of IPK is 7 symbols (7 bytes) which is

7ma!-Lq

So the length of IPKF must be 7 symbols (7 bytes) which is

hk2=xR8

The final key (password) is

D42CEBAAC938FB302EB8E12424CFC2F90F1CF1F9A93C5B2B029018C0899B13B44A440CB1

In the first and second example there was deliberate in that the change in the initial keys is simple (only replacement m instant of D) in order to see results that showed and proved that the change in final keys is too big.

Ex 3: The length of IPK is 8symbols (8 bytes) which is

T(3/b)z>

So the length of IPKF must be 8 symbols (8 bytes) which is

!E*c=6-1

The final key (password) is

566CD826D6180302032E13F4A121E7190E2D512ECD056FC464DB8A05501C4788137C1B4A

Ex4: The length of IPK is 10 symbols (10 bytes) which is

Pw/l\5A<Yu

So the length of IPKF must be 10 symbols (10 bytes) which is

Tt={q,1;L}

The final key (password) is

33A20820FE078209341208B8E2C279802FE602A15DC6B89A8901F07C18F61A4295E0D81C

The results from apply the algorithm shows in each time generate unique final key at least 72 digits in hexadecimal system.

7. The Powerful Points Of The Algorithm

Systems that apply the algorithm can get on high security as the following points:

i- The access to the password file or database is useless, because the algorithm use the initial password to generate the final password, which not associate with initial password therefore, the initial password must be enter to algorithm mechanism as condition to compare the result of process with password database, if the result is not matching, the system refuses entry, in other words, the authentication fails.

ii- Theft or identify traditional initial password is useless because the algorithm mechanism used two initial password keys, the first one is initial password key (IPK), the second one is initial password key file (IPKF), with IPKF the user will enter the values in the beginning after that no need the user remembers those values, the system will create password file, which will send to user. The user will store IPKF on storage media. No other system can read the contents of IPKF because it is protected.

8. The Features Of The Algorithm

- The initial password IPK is easy to remember.
- The passwords database stored in the system is containing finial passwords not related with initial password (IPK and IPKF).
- Theft the initial password (IPK or IPKF) is useless.
- Theft the final password database is useless.

9. Conclusion

The algorithm contains group of phases, each phase has privacy and the special scenario differs from the subsequent phase, where the target is to create password structure uncommon and non-traditional, in some parts of algorithm there is touch of encryption, the important feature is any simple change in IPK and IPKF lead to big change in final password and no relationship between IPK, IPKF and final password, the algorithm tries to achieve confusion and diffusion. Not to adopt the algorithm on one initial key gave strength and reliability because detect or theft one of initial keys not lead to generate same original final key.

10. References

Arash Habibi Lashkari, Rosli saleh,Samaneh Farmand,Omer Bin Zakaria. "A Wide-range survey on Recall-Based Graphical user Authentications algorithms based on ISO and Attack Patterns." (IJCSIS) International Jouranl of Computers Science and Information security, 3 2009.

Eric Cole, Ronald Krutz,James W.Conley. Network Security. Indiana,USA: Wiley, 2009.

Hyun-Chul Kim, Hong-Woo Lee, Kyung-Seok Lee , Moon-Seog Jun. "A Design of One-Time Password Mechanism using Public Key Infrastructure." Fourth International Conference on Networked Computing and Advanced Information Management. IEEE, 2008. 18~24.

Magid, Larry. How to create and remember Strong pssword. Ohio, 7 12, 2012.

Michael E.Whitman, Herbert J. Mattord. Road Map toInformation Security forIT and Infosec Managers. Bosten: COURSE TECHNOLOGY, 2010.

—. Road to Information Security. Bosten,USA: Course Technology, 2011.

Wu, Thomas. "The Secure Remote Password Protocol." Interent Society Network and Distributed System Security Symposium,Sandiego, 3 1998: 99~98.