ALPHA-NUMERICAL RANDOM PASSWORD GENERATOR*

BY

VRUSHALI D PATANKAR♣

Student, School of Computer Science and Engineering, Ajeenkya D Y Patil University, Pune, India

Vrushalipatankar@adypu.edu.in

PROF. ARUNA VERMA

Assistant Professor, School of Computer Science and Engineering, Ajeenkya D Y Patil University, Pune, India

Aruna.v@inurture.co.in

ABSTRACT

Computer and internet consumers are constantly growing. As the users increase, there is also a strong need for protection. On computing networks are stored data properties and other useful facts. The correct authorization process for accessing the data is one way to safeguard data properties. The user identity and password system will accomplish this. Password collection is crucial, as the whole permission depends on the password. The password must be strong enough to withstand brute-force attacks and other types of attacks. We'll look at a method for generating random passwords that's strong enough to fend off attacks.

KEYWORDS

Password, Random password, Encryption, Decryption, Security, Symmetric Key.

1. Introduction :

In today's contact method, the password is necessary and unavoidable and provides user data protection. Password is a string sequence for the purpose of authenticating the user's personal identity and of providing or refusing the access to device services. The password not only refuses unauthorized entry to the device but also prevents users previously logging into the system from performing an unauthorized operation.

Unauthorized entry security risk requires more than just one user's risk by his device account. One of the basic ways of user authentication is user ID and password combination. Password is a code word used to permit the user to enter a certain device or programme. The user's identity

* Received 22 September 2021, Accepted 09 October 2021, Published 24 October 2021
♣ Corresponding Author

is password-tested. Injuries will attack the Device and attack data properties where keys are not secure or quickly guessed. The guessing attacks have huge business repercussions, such as a vandal event in 2009, in which Twitter's executive secret was devalued and all internal records of the firm were read. Due to the domain re-use scenario, a new attack model is intended to exploit accounts with a devaluation of a lower-security website and try to re-use the credentials on sensitive websites. The scenario is quite widespread. Duration and diversity add to the domain size of the whole range that makes brute force identification more difficult. Most organizations, for the purposes of composing and using passwords, specify login laws. The categories needed are minimum length and requirement, for instance top and bottom categories, numbers and special characters, forbidden items such as own name, DOB, telephone number, and addresses. Some countries have a national authentication system, which sets user authentication standards for government facilities, including login requirements. The goal is the automatics of the generation of passwords, which provide fundamental safety requirements for design, execution and usage of passwords, by defining an algorithm for the creation of computer resource protective passwords. The algorithm uses random numbers to pick the random password characters. Encryption and decryption mechanisms secure the created password**.**

2. RELATED WORK :

Art Conklin, Glenn Dietrich and Diane Walz addressed the computational paradigm of password-based authentication on many user-based applications. They also shown user identification, using examples of a mixture of user ID and passwords, a basic user authentication scheme, a smart card system where users have a user ID and a password generally. They have spoke about the cognitive capacity of people to recall passwords, possible dangers of low passwords. According to them, compliance with password laws produces more complicated passwords. They even knew why they remembered it. Compliance with the laws of the scheme creates more difficult to find codes.

Manoj Kumar Singh has suggested a way of using the artificial neural network to improve the safer, more effective means of authentication. He covered neural network architecture, learning rules, aim description and authentication method. He has stressed that a neural network with a capacity for intrusion detection should address the password and authentication challenges.

The three random password generating algorithms, ALPHANUM, DICEWARE, and PRONOUNCE3, were analysed by Mikel D. Leonhard and V.N. Venkatakrishnan. They use

measures like safety, memory and affinity to figure out which of the three approaches yields the most appropriate passwords. You used six random login character lengths, including upper case characters, lower case characters and numbers. They also stressed that alterations of passwords profit more from authentication and anonymity over user-selected passwords. Random word lists are generated by the DICEWARE generator. This is focused on the memorization principle and the mental relation needed to keep the password in mind. In PRONOUNCE3 pronounceable words are written in English. This helps to help the user's speech facility to recall the secret.

In order to improve data security, Ayushi suggested a Symmetric Key Cryptographic Algorithm. Secret key methodologies in cryptography are known as stream cyphers or block cyphers. Stream cyphers are operating at a time on one bit and input mechanisms are used. For one block of data at a time, the block cipher utilizes the same key on each block. The same plaintext block is frequently encrypted in the same cypher text when using the same key in a block chip. The identical plaintext is encrypted into different chip text in the upstream cypher. This article covers bit manipulation for encryption and decoding. The data is converted to binar and reversed.In order to improve data protection, new symmetric key algorithms were implemented by Kamini H. Solanki and Chandni R. Patel. Bits for encryption and decryption were used as well. The ASCII character is then translated into binary numbers. Complementing the binary digits. The central constant 10 is multiplied and the product is translated to a hexadecimal value with the supplemented binary digit. You suggested the design and implementation of this algorithm to solve this cost-efficiency problem in order to encrypt a limited number of records.

Section 3 outlines the proposed analysis for random password formation and methods of chip-fixing, section 4 contains an experimental analysis and section 5 discusses conclusion. This article is presented as follows.

3. PROPOSED WORK :

The scheme suggested specifies a modern method for random password production and invents new cryptographic methods for password protection. The revolutionary symmetrical Key Algorithm generates and encrypts a random password. The proposed model offers random development, password encryption, and decryption concepts and guidance. For different uses, including online registration, online tests etc.The process of password encryption and decryption can also be used.

A. Random Password Generator :

The purpose of a random password generator is to generate a secure random password. Random passwords, in general, have several advantages over user-chosen passwords in terms of security and confidentiality. The new process was developed to generate a random password that contains both upper and lower case letters, as well as numerals ranging from 0 to 9, and is of a predetermined length. The alphanumeric algorithm is a straightforward procedure. The random password generator is designed to generate highly security random passwords. Random passwords generally have some advantages over user-selected passwords, which improve safety and security. A random password consisting of both upper and lower case lines as well as digits of 0-9 with a defined length has been developed using the latest technique. The alphanumeric algorithm is a simple one, Creates a pre-determined length random password. The password generator's algorithm selects a random character format and creates a password that is a mix of numbers, lower and upper case, and generates a random password of a predefined length. The password generator algorithm generates a password by selecting a random character from a random character list and combining integers, lowercase letters, and capital letters.

The password's cumulative length is 12. The password with the combination of the lower case character, upper case character and numbers is randomly chosen. The entire scale of the alphabet is 62[10+26+26=62], which shows the number of 10 (0 to 9), the top 26 and the bottom 26. Any character can occur in a password with 62 options. Therefore the number of possible passwords is 62 x 62 x 62 x 62 x 62 x 62=6212

Procedure :

Step 1: Start the procedure
Step 2: Build a random numbered and upper and lower case character set.
Step 3: Password must be 12 long fixed.
Step 4: To produce a password, create Random Password Generator process.
Step 5: Random Password Generator selects a set of three.
Step 6: Every character in a random fixed index location will be restored.
Step 7: Add, one by one, the chosen characters to the index.
Step 8: Password printing.
Step 9: Finishing.

B. **New Password Encryption Method :**

- Cryptography is the procedure by which ordinary text or plain text is converted into a cypher text (decryption). The different algorithms can be classified in many ways. The commonest kinds are I secret key encryption, which is called symmetric key encryption, and ii) public key encryption, also called asymetric key encryption.. The most common types are I The symmetric key cryptographic method is the algorithm we use. A 4 digit binary number (>=1000) is the key used to encrypt the password. This algorithm preserves three different sets of vector values containing letters and numbers from above and below. The algorithm verifies the alphabets of the password of a password that is inserting and determines if the alphabet is an upper / lower case / integer.

  There are three sections of the algorithm. The first element is the password to be converted to the decimal number. Second, binary manipulations are performed. The third element is the conversion of binary to alpha-numerical. Three alphanumeric cases transform into a decimal value in the first step:

  (1) The ASCII alphabet value and constant 50 value are added together, Where the alphabet is the letter in the upper case.

  (2) The constant 20 of the alphabets is removed from the ASCII-alphabet meaning if the alphabet is less in this case

  (3) The constant 10 is then deducted from the ASCII value found if the alphabet is a number.

  The decimal value obtained in the previous procedure is transformed to binary in the second stage, and the binary value is returned. The key breaks down the reversed binary. The key is chosen by the user. The remainder is converted to a binary value, and the quotient is calculated. The remainder of the first three digits, as well as the quotient in the next five digits, are formed.. Ultimately, the resulting binary value is converted to alpha-numerical value.

  Stage 1: Start the operation. Procedure:

  Step 2: Store upper case letters and bottom case letters as separate lists of variable numbers.

  Step 3: Feedback is used as a random password.

  Step 4: Alphabet conversion to decimal notes.

  4.1. Verify the upper case of the alphabet. Find and apply the ASCII value by 50.

  4.2 Lower-case check the alphabet. Select the ASCII value and remove no.

4.3 See Number for the alphabet. Find the value of the ACSII and extract 10 .

Step 5: Transform the resulting decimal value into binary numbers.

Step 6: Binary numbers reverse.

Step 7: Get the user's card.

Step 8: Split the reverse binary number by the key.

Step 9: The remainder should be stored in the first three numbers, and the quotient should be stored in the next five numbers ("the remaining and quotient are no longer than three digits and five digits respectively). If one of them is less than 3 or 5 digits, the requisite 0s (zeroes) must be added to the left side.

Step 10: Transform the binary to decimal

Step 11: Take the decimal value to be the ASCII value and find it encrypted.

4. Example :

Let the randomly generated password character be "B" (Upper-case letter).

1. ASCII is 66 decimal equal of 'B.'
2. Added value ASCII and constant value.
 = 66+50 (ASCII worth 50) = 116
3. 116 is 01110100 binary value. 3.
4. 00101110 will be the reverse of this binary digit.
5. Allow the divider to be 1000 i.e. Key.
6. Split by 1000 00101110 (dividend) (divisor).
7. The rest is 11, and the quotient is 101. The binary numbers are 11000101 after division.
8. Translate 11000101 to 197 decimal.
9. ASCII 197 value is "Å" character, which is saved as encrypted value.

5. New Decryption Method :

Encryption is the mechanism by which plaintext is converted into cypher text. The reversal of encryption as the cypher text is translated into plaintext is decrypted. To decrypt the encrypted password, new form of decryption algorithm must be produced. For both encryption and decryption, symmetric key cryptography uses the same key.

The method of decryption shall be taken as follows: This algorithm holds the individual arrays including the high-casel array list, the bottom caseload array list and the array list of numbers which hold some values of the upper and lower alphabets and the numbers between 0 and 9 The decryption key which must be the same as the password encryption key. The coded password is hexadecimal.

The first move is to decode each character to ASCII. ASCII is translated into binary numbers for any decimal value. Multiply the last 5 digits with the key. With the result provided by the multiplication, the first 3 digits are applied to the chip text. If the result generated is not an 8-bit integer, then a number of zeros to 8-bit is applied to the left. The 8-bit numbers have now been reversed.

Binary digits are translated to a decimal value in the second stage of decryption. The decimal value is verified using various value ranges in each array against the array list. The Upper Case Array ranges from 115 to 140, whereas the Lower Case Array ranges from 77 to 102, with the Upper Case array starting at 38 and ending at 47. The array list can be found by looking at the decimal value set. If the decimal value is in the top case array list, subtract the constant 50 from it This is the alphabet's ASCII value for a random password. If the decimal value is in the lower case array, multiply it by 20. This operation generates an ASCII value and detects the ASCII value alphabet. If the decimal is in a number array list, it is appended to the decimal value. The ASCII value of the character is the result**. The ASCII value is transformed in the final process which produces the decrypted form.**

Procedure:-

Step 1: Start the procedure

Step 2: Convert the ASCII decimal value of each encrypted character.

Step 3: Conversion to binary digits with decimal value.

Step 4: Multiply the key to the final five-digit cypher letter. Stage 5: add the first 3 digits of the text of the cypher to the previous step results.

Step 6: If the result in step 5 is not an 8-bit number, we need to create an 8-bit number.

Step 7: Binary numbers reverse

Step 8: Find the reversed digits decimal number.

Step 9: Keep the Upper Case (115-140 value), Lower Case (77-102 value start), and Number array (74) arrays (values start from 38 to 47). In the top or bottom case sequence or number array, check the resultant decimal. Effectively

9.1 If the decimal value is in the upper case array, subtract a 50 constant from it. This operation generated the ASCII value of any upper case letter in a password.

9.2 Incorporate the 20 constant with the decimal value in the lower case array. This answer is the ASCII value of one of the lower case password alphabets.

9.3 If the decimal value is in an array of numbers, assign a constant of 10 to it. As a consequence, the ASCII value of password numbers is obtained.

Step 10: Assign an ASCII value to the resultant alphabet, one by one adding a decrypted password.

Step 11: Complete the task.

Example:

We get the cypher text as "Å," the ASCII character for 197, after encrypting "B" now we decrypt the chip text in order to obtain the plaintext.

1. By locating the ASCII value, the cypher text Å" is translated to decimal.
2. The 197 binary is 11000101.
3. The result will be 101000 when 00101 is multiplied (last 5 digits) by 1000 (Key).
4. The 101000 answer will be 101110 after the addition of 110 (first three digits of a cypher text).
5. Since the 8-bit number of 101010 is not required, 00101110 must be generated.
6. That will be 01110100 after the reversal of the figure.
7. 01110100 is 116 decimal value.

Therefore 50 should be subtracted from the decimal value "116" in the upper case array list.

8: ASCII significance detection = 116 – 50 (decimo-constant 50) = 66 (ASCII value) 9: Convert the ASCII value to "B" alphabet.

Therefore, the character "B" of the random cypher is encrypted with "Å" and the encoded character is decrypted back to „B" by means of the suggested method of decryption.

6. EXPERIMENTAL STUDY :

A software application tests and demonstrates the idea. Alpha-Numeric Random. Random Password Generator Algorithm is checked for members of the browsing centre of password generation. When new users register at the navigational center, all of their information is saved on the server, and a password is generated and delivered to them through email. When signing in to each login, users will utilize random codes. This offers protection in the server-based world for the typical client in a browsing centre. Whenever the user is logged in, the username and password must be sent. The password given by the user is frequently plaintext, but the database information is an encrypted plain text version. By comparing the password entered by the user with the password already saved, the plain text is converted into encrypted and

decrypted mode. The safety measure is increased by stored passwords when employing the encryption and decryption procedure mentioned in this article.

The suggested approach can also be used to provide the laboratory users with username and password. High schools or other institutions. Students should keep track of their personal information, and the login password can be created using the specified technique. By using a random password generating mechanism, it may be utilized for any application that requires authentication.

7. **Conclusion :**

Practical and usable with great results is the password created by the random alpha-numeric password mechanism shown above. Users choose the password that is associated with themselves and that relates to each case, while the password is chosen manually much of the time. This allows intruders to deploy more attacks as the codes are breached. This condition is avoided by randomly created passwords. One of the downsides could be the difficulty in remembering the random password. However, it is more preferable to compare the security provided by a randomly generated password to that provided by a manually picked password. The amount of encryption and decryption provided here also adds to the security precautions in place. It is cost effective because the encryption and decryption standards are well defined**.**

8. Future Enhancement :

Because any application is password-protected, more research into encrypted automated password generation can be done. The proposed method employs a random character list consisting solely of alphabets and numerical values. Even unusual symbols could be used to improve the password. The password's length can also be increased to make it more secure.

The passwords were chosen at random, which could lead to a new encryption and decryption standard. A significant number of samples will be accessible for experimental examination in the future.

9. REFERENCES :

[1] Art Conklin, Glenn Dietrich, Diane Walz, “Password-Based Authentication: A System Perspective”, Proceedings of the 37th Hawaii International Conference on System Sciences – 2004.

[2] NikCubrilovic, The Anatomy of the Twitter Attack, TechCrunch, July 2009.

[3] Thorsten Brantz and Alex Franz,The Google Web 1T 5-gram corpus, Technical Report

LDC2006T13, Linguistic Data Consortium, 2006.

[4] Mike Bond, Comments on authentication, www.cl.cam.ac.uk/~mkb23/ research/GridsureComments.pdf, 2008.

[5] Manoj Kumar Singh," Password Based a Generalize Robust Security System Design using Neural Network", IJCSI-International Journal of Computer Science Issues, Vol. 4, No. 2, 2009.

[6] Michael D. Leonhard, V. N. Venkatakrishnan, "A Comparative Study of Three Random Password Generators", IEEE EIT 2007 Proceedings.

[7] Ayushi, "A Symmetric Key Cryptographic Algorithm", International Journal of Computer Applications, Volume 1 – No. 15, 2010.

[8] Kamini H. Solanki, Chandni R. Patel, "New Symmetric Key Cryptographic algorithm for Enhancing Security of Data", International Journal of Research in Computer Engineering and Electronics, volume 1, issue 3,Dec 2012.