International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 12 | Dec - 2021 ISSN: 2582-3930

Password Generation Strategy Based On Hexa-Decimal

M.P.VAISHNNAVE*, R.MANIVANNAN[2], R. RAJESH[3]

1Teaching Fellow, Dept of IT, University College of Engineering Villupuram ,

2Assistant professor, Dept of CSE, Meenakshi Ramaswamy Engineering College, Ariyalur,

3Dept of IT, University College of Engineering Villupuram

Abstract: The importance of password-based authentication in information security cannot be overstated. Passwords may help to improve the information system's security. As a result, administrators devise a variety of password schemes to assist users in strengthening their authentication security. It also improves on previous ways for creating passwords based on mnemonic shapes to identify password susceptibility and boost user privacy. To assist users in creating a safe and strong password, XPass Pass combines the characters generated by our mapping approach and created a 4x4 matrix called X-Matrix that contains all the hexa-decimal digits. These hexa-decimal numbers are binary transformed to integers, and the numbers are used to hunt for UNICODE letters. After all, numbers have been processed and returned, characters have been added and then preprocessed, non-printable characters have been eliminated, and a strong UNICODE encoded password has finally been returned.

Keywords: password generation strategy; mnemonic passwords; Unicode passwords; strong passwords

Introduction:

The user's behavioral model is used to authenticate the user in password-based authentication. Many computer applications, including operating system user login, social account login, online banking, online shopping, and so on, are secured using it[1,2]. Password authentication means that each user has a secret password that is checked by a system to verify their identity. Alphabetic, numeric, and special characters make up the password. The password scheme is designed by system administrators to prevent password cracking or leaking [3].

As a result, most users utilize their Personal Information to establish their account/application password, such as their Father's Name, Mother's Name, pet name, Date of Birth, Place of Birth, and so on, in order to match

the requirements of the password scheme [4]. However, because these passwords are based on their personal information, they can be cracked. However, because knowledge is easy to recall in human nature, random passwords are difficult to remember. To avoid this, we created a Password Generation Strategy based on character Mnemonics that makes it easier to remember passwords and produce secure passwords.

Existing System:

The existing method incorporates the order of letter strokes with password generation to support users in creating strong and memorable passwords. The Alphapwd-based passwords are compared with three leaked password sets in this experiment. The results show that Alphapwd-based passwords are generally resistant to unknown attacks than real password sets [6]. Moreover, a comparison of the passwords generated by Alphapwd, KbCg (Keyboard Change), and SpIns (Special Character Insertion) reveals that Alphapwd password security is better than that of KbCg mnemonic and Alphapwd. SpIns mnemonic password is easier to remember than a password.

The simplest password composition approach simply limits the number of different character sets that can be used in passwords. On the basis of the general password composition method, some researchers have advanced the mnemonic password strategy[7]. The goal of the mnemonic password policy is to assist users in creating secure and easy-to-remember passwords. MnePer, MneEx, MneSchEx, MneGenEx, MnePerEx, and MneYanE have one thing in common: they all allow the user to choose three words or sentences that comprise at least eight words and then replace each one with numbers, letters, and special symbols.

For example, in the MneGenEx strategy, select the sentence "I have two balls and three bats", then “two” =>“2“ (“2" replace “two”), “and”=>“&“, “three”=>“3“, the rest of the words are replaced by the first letter of the word. Then we get the password “Ih2b&3b”. So, most of the A random password can solve the security problem, but the usability is got reduced. In the Opt words, password strategy users need to remember the patterns they choose. the use of their passwords especially when these are not easy to remember. The use of simple patterns will reduce security [8]. So, take simple alphabet writing as an example, the attacker can infer the number of words in the phrase that was used to generate the password.

The basic idea of Alphapwd is as follows:

1. In order to produce a password, the Alphapwd password generating approach requires at least two types of uppercase letters, lowercase letters, numbers, and symbols.

2. By mapping the writing order of the letters on the keyboard, the character represented by the keys is taken as the private key (User Password).

3. The user can choose whether to use all the key symbols as the password or the selection.

4. All they have to do now is remember the mnemonic they chose and where it starts on the keyboard, and then use it to type the password.

Proposed System:

In the proposed system, we introduce the X-Matrix. The X-Matrix is a 4x4 matrix consists of all the hexa-decimal digits from 0-F. Here, Figure.1 shows four default patterns of the X-Matrix, can be used for password generation. Using this matrix, each character in the user's old password will be mapped on the matrix and the respected values will be taken.

C	D	E	F
8	9	A	B
4	5	6	7
0	1	2	3

3	7	B	F
2	6	A	E
1	5	9	D
0	4	8	C

0	4	8	C
1	5	9	D
2	6	A	E
3	7	B	F

0	1	2	3
4	5	6	7
8	9	A	B
C	D	E	F

Figure.1 Default patterns of the X-Matrix Pattern1,2,3,4

Consider the case of a user whose password is "pass." Here, the password string is very small in length, contains repeated characters, and found in the dictionary. Rainbow table approach is a quick way to crack it.. But to strong this password, our system uses the following strategy. First each character in the user password will be mapped as shown in the figure.2 using default X-Matrix pattern 01.

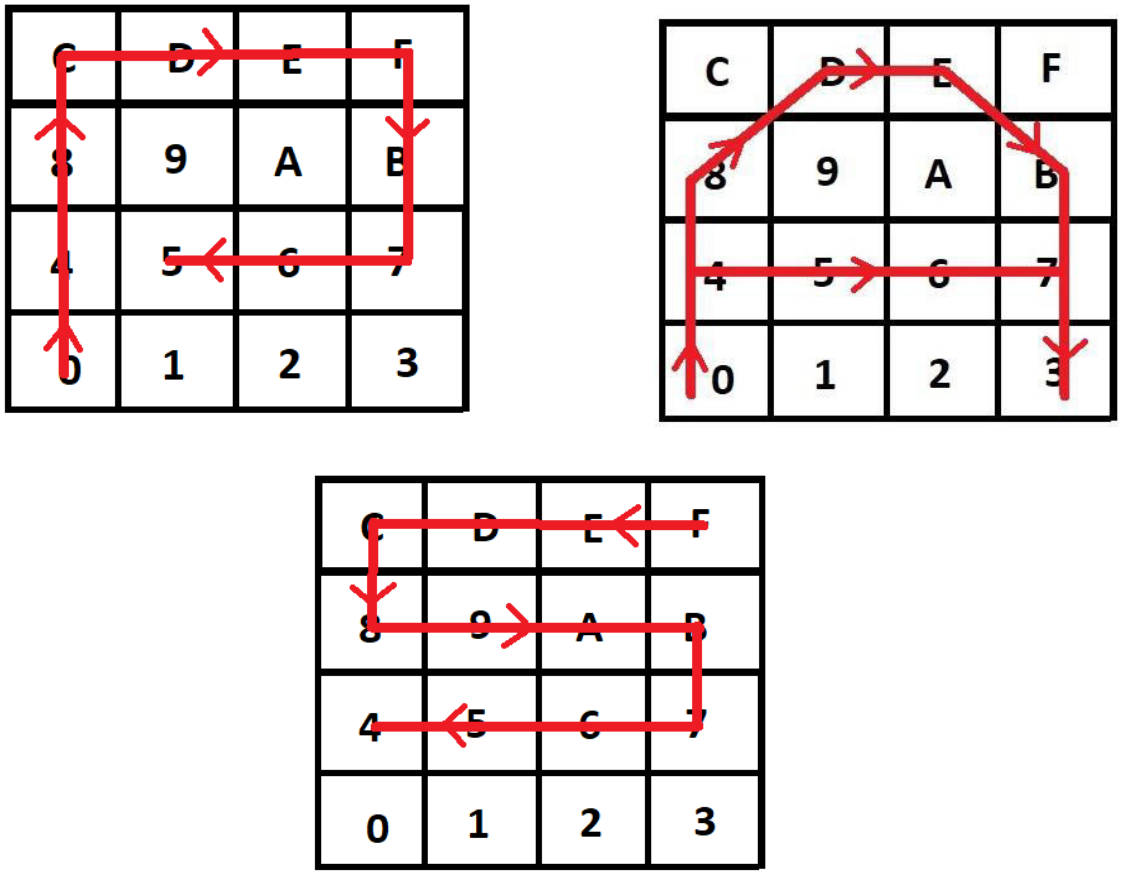


Figure .2 Mapping of 'P','A'& 'S'

The hexa-decimal values that are all appended for each character as shown in the table 1.it consist of mapped value for user password. Table 2 shows the appended Hex-decimal values and all characters respective values are appended.

Table 1: Mapped Values for the user password

Character	Mapped Values
P	048CDEFB765
A	048DEB734567
S	FEDC89AB7654

Table 2: Appended Hexa-decimal value

User Password	Mapped Value
PASS	048CDEFB765048DEB734567FEDC89AB7654

Then the mapped value is divided into separate sub-string with the length of 2 like [04 8C DE FB 76 50 48 DE B7 34 56 7F ED C8 9A B7 65 4]. Now all the hexa-decimal sub-string are converted into their respective integer as shown in the table 3

Table 3 Hexa-decimal to Integer value

Hexa-decimal	Integer
04	4
8C	140
DE	222
FB	251
76	118
50	80
48	72
Hexa-decimal	**Integer**
DE	222
B7	183
34	52
56	86
7F	127

ED	237
C8	200
9A	154
B7	183
65	101
4	4

The character is then generated from the UNICODE table using all of the integer values. For example, the Unicode character found at the 4 place is EOT (End of Transmission) character. Likewise, all the integer's respective Unicode characters are looked up and appended to form a Unicode String and finally all the non-printable characters are removed to form to string that can be used as the new strong password as shown in the table 4. This new password can be used to demonstrate resilience to password assaults like brute force, rainbow tables, and others.

Table 4 Old Passwords to New Password

Old Password	New Password
PASS	ÞûvësEgþÜ«vTþÜ«vT

Results

we have tested all of them with four popular password strength Checker online services named "The Password Meter", Kaspersky Lab, "Password Checker Online" and "Strength Test". "The Password Meter" checks the strength of the password Pattern and gives a score of its strength measure from 0 to 100 including a ranking of four criteria: "Exceptional", "sufficient", "warning" and "failure". If passwords' strengths are good enough, then they will be in "exceptional" and "sufficient" criteria. If the score is not satisfactory, it means that the adversary can compromise the password and it will fall under the "warning"

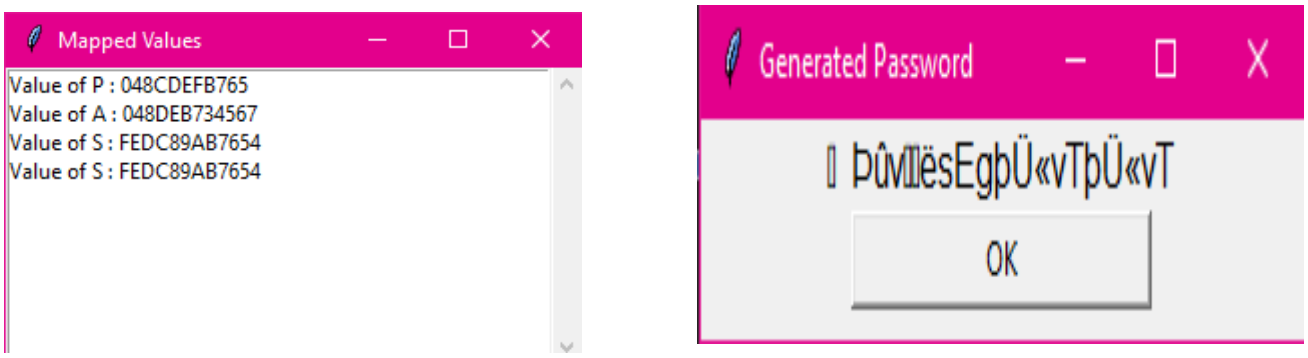


Figure.3 Mapped Values &Generated password

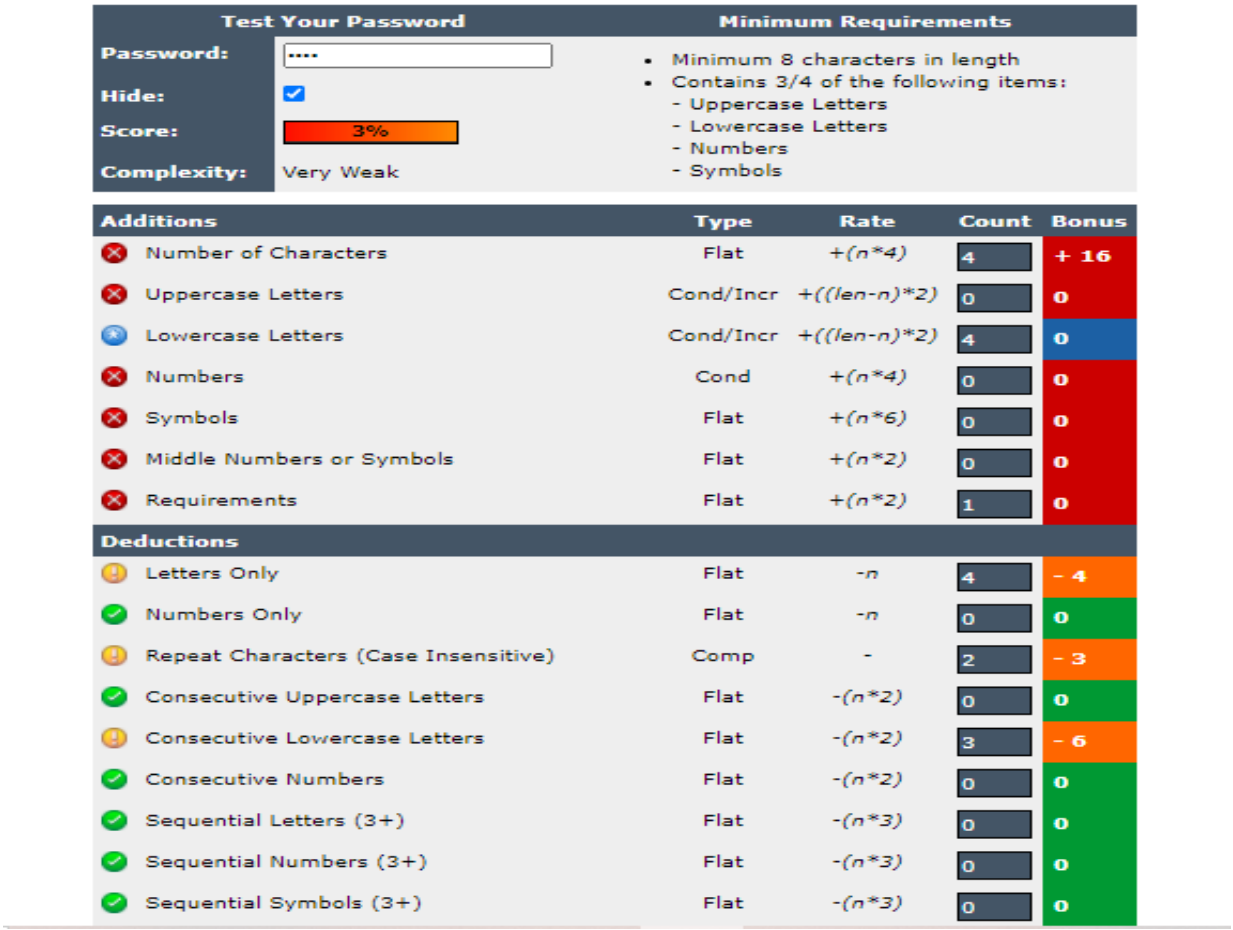
Criterion. Experimenting with "The Password Meter", for our 50 generated passwords we have an average Score of 98.88% which denotes that our system- generated passwords are fairly strong. We can see the experimental results from "The Password Meter" in "Kaspersky Lab" checks the crack ability of passwords using the Brute Force attack to be accomplished on a home Computer.

Using this online service for the 50 passwords from sample input data generated by our system, we have obtained a minimum cracking span of 2 years and maximum 1217000 million years by the Brute Force attack which again proves that our system is not easily penetrable to the Brute Force Attack. "Password Checker Online" checks the crackability of Passwords using both the Brute Force attack (on various Machines: standard desktop PC, Fast Desktop PC, GPU, Fast GPU, Parallel GPUs, Medium Size Botnet) and the Dictionary attack. This checker gives an overall comment on the password depending on its strength. It also provides the cracking time estimation by the Brute Force attack and checks whether it is crack-able under the Dictionary Attack. Overall comment on the password depending on its strength. It also provides the cracking time estimation by the Brute Force attack and checks whether it is crack-able under the Dictionary Attack.

By using this checker, our observation is that, for every password that we generated, the result for the Dictionary attack is "safe". Another inspection is that the Brute Force attack's cracking period for every password is Similar to the results found from "Kaspersky Lab". And the Overall comment for all passwords ranged from medium (with 69% strength) to excellent (with 100% strength). We have also checked the results of the Brute force attack And the Dictionary attack, especially for the passwords of medium strength, and the observation is that those are Safe from both attacks.

For a password whose strength Was 69% by using Brute Force in Medium Size Botnet, it Has been shown crack-able in 14 days. So, this evaluation Also proves that our system-generated passwords are robust,

Maintaining all the password policies and requirement criteria. “Strength Test” checks the entropy measure of a password. The entropy of a password is a measure of its unpredictability. The greater the entropy, the more difficult it is to crack a password. The entropy of a password is commonly stated in bits in cryptography. There is no entropy when a password is already known. On the first try, a password with one bit of entropy can be guessed. When a password's entropy is between 28 and 35 bits, it's considered to be quite weak. It is okay if it is between 36 and 59. (meaning somewhat secured for network and company Usage). When a password has 60 or more entropy, it is considered strong. In this checker, our system-generated Passwords have entropy ranging from 47 (minimum) to 88.2 (maximum). So, with this check, we can also prove that our generated passwords are hard to crack.



Test Your Password		Minimum Requirements
Password:	••••	• Minimum 8 characters in length • Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	☑	
Score:	3%	
Complexity:	Very Weak	

Additions	Type	Rate	Count	Bonus
Number of Characters	Flat	+(n*4)	4	+ 16
Uppercase Letters	Cond/Incr	+((len-n)*2)	0	0
Lowercase Letters	Cond/Incr	+((len-n)*2)	4	0
Numbers	Cond	+(n*4)	0	0
Symbols	Flat	+(n*6)	0	0
Middle Numbers or Symbols	Flat	+(n*2)	0	0
Requirements	Flat	+(n*2)	1	0
Deductions				
Letters Only	Flat	-n	4	- 4
Numbers Only	Flat	-n	0	0
Repeat Characters (Case Insensitive)	Comp	-	2	- 3
Consecutive Uppercase Letters	Flat	-(n*2)	0	0
Consecutive Lowercase Letters	Flat	-(n*2)	3	- 6
Consecutive Numbers	Flat	-(n*2)	0	0
Sequential Letters (3+)	Flat	-(n*3)	0	0
Sequential Numbers (3+)	Flat	-(n*3)	0	0
Sequential Symbols (3+)	Flat	-(n*3)	0	0

Figure.4 Test the password strength

The minimal requirements for password strength are described in Figure.4. The password should be at least 8 characters long to be considered secure. Characters (both upper and lower case), integers (0 to 9) and even special characters (! @ # $ percent & *) must all be included in the password.

Table.5 Password Length and its Cracking rate (in weeks)

Password length	**Estimation in weeks**	
	Cracking rate (regular password without symbol)	**Cracking rate (regular password with symbol)**
8	0	2
9	5	17
10	17	209
Password length	**Cracking rate (regular password without symbol)**	**Cracking rate (regular password with symbol)**
11	521	1721
12	10429	20857

Conclusion:

In this paper, several common password composition strategies are introduced, and their security and usability are analyzed. In order to solve the problem that the password freely chosen by user is often weak, and the password generated by the system is secure but difficult to remember.The password generation technique X-Pass is proposed, which is based on mnemonic shape. It's also been proven that the password created by X-Pass is more resistant to unknown attacks than regularly used genuine passwords. Having a high-security password does not guarantee that the system will be secure; users must also have a strong sense of security

and adopt excellent password habits. More secure and memory-friendly password construction policies, as well as new authentication techniques, are expected to be proposed in the future.

References:

[1] Jianhua Song, Degang Wang, Zhongyue Yun, Xiao Han, (2019) "Alphapwd: A Password Generation Strategy Based on Mnemonic Shape"

[2].M.P.Vaishnnave TE Sankaranarayanan, R Manivannan, K Ramkumar" A Study on Cyber Security for Password Generation", International Journal for Research in Applied Science & Engineering Technology (IJRASET) Volume 8 Issue IX Sep 2020.

[2] Farhana Zaman Glory, Noman Mohammed, Atif U1 Aftab, Olivier Tremblay, (2019) "Strong Password Generation Based on User Inputs"

[3] Mengli Zhang, Gang Zhou, Muhammad Khurram Khan, Saru Kumari, Xuexian Hu, Wenfen Liu (2019) "SPSR-FSPG: A Fast-Simulative Password Set Generation Algorithm"

[4] Abejide Ade-Ibijola, Blessing Ogbuokiri (2019) "Syntactic Generation of Memorable Passwords"

[5] Jun Luo, Jin Deng, Chu Lu, Hong Liu (2019) "Recurrent Neural Network Based Password Generation for Group Attribute Context-Ware Applications"

[6] Mr. Vijay B Gadicha, Dr. A.S.Alvi (2017) "Extensive Approach For Strong Password Generation Using Content-Color Mechanism"

[7] Edward E Kelley, Franco Motika, James B Webb (2007) "Universal Password Generation Method"

[8] Kaspersky Password Tester- https://pasword.kaspersky.com

[9] The Password Meter- https://passwordmeter.com

[10] Password Checker Online- https://howsecureismypassword.net

[11] Strength Test https://my1login.com/resources/password-strength-test