I. J. Computer Network and Information Security, **2016, 7, 23-30**
Published Online July 2016 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2016.07.04

Password Security: An Analysis of Password Strengths and Vulnerabilities

Katha Chanda
Department of Computer Science and Engineering, Amity School of Engineering and Technology, Noida, Uttar Pradesh, India
E-mail: katha4494@gmail.com

Abstract—Passwords can be used to gain access to specific data, an account, a computer system or a protected space. A single user may have multiple accounts that are protected by passwords. Research shows that users tend to keep same or similar passwords for different accounts with little differences. Once a single password becomes known, a number of accounts can be compromised. This paper deals with password security, a close look at what goes into making a password strong and the difficulty involved in breaking a password. The following sections discuss related work and prove graphically and mathematically the different aspects of password securities, overlooked vulnerabilities and the importance of passwords that are widely ignored. This work describes tests that were carried out to evaluate the resistance of passwords of varying strength against brute force attacks. It also discusses overlooked parameters such as entropy and how it ties in to password strength. This work also discusses the password composition enforcement of different popular websites and then presents a system designed to provide an adaptive and effective measure of password strength. This paper contributes toward minimizing the risk posed by those seeking to expose sensitive digital data. It provides solutions for making password breaking more difficult as well as convinces users to choose and set hard-to-break passwords.

Index Terms—Password, Security, Entropy, Hashing, Password Strength.

I. Introduction

Passwords have been used to grant access to unknown persons since ancient times. Military, spy organizations, high security organizations have seen a rampant use of passwords. Even today, it is not only used to secure computers in the traditional sense but is used to control access to mobile phones, homes, ATMs (automatic teller machines) and many more. Most often, passwords are the only security one enforces to protect an application against unauthorized access and unfortunately many users do not realize the importance of passwords fully. They tend to set short, easy-to-remember passwords which are highly vulnerable to attacks. This paper aims to impress upon users just how vulnerable their private data can be with weak passwords by showing how easy or hard it is to break passwords of different strengths. Of course, with enough computing power and speed, no password is ultimately secure against a brute force attack. Yet with a little vigilance and knowledge, it is easy to make the job significantly harder for potential attackers. Section 2 discusses the existing work on password security. Section 3 analyzes different aspects of password security and sections 4 and 5 discuss the different components that lend strength to the password. A brute force attack is implemented to analyze the times required to break the password. By varying the strength of a password by a few factors, a comparison is made and a definitive conclusion is reached. Section 6 surveys the existing password composition rules enforced by popular passwords in the industry and section 7 describes a password strength checker, PwdStrength that is devised and implemented. The strength of the password is calculated based on a number of factors drawn from earlier conclusions. Table 3 provides a comparison between the output of PwdStrength and a pre existing password strength checker on 22 passwords. The paper concludes with an analysis of the performance of PwdStrength and a short summary on the importance of password security as observed from the preceding sections.

II. Related Work

Significant research has been done with passwords, their security, authentication methods and options beyond passwords. More secure alternatives to passwords exist. But as Herley at al [1] stated in their paper, there are a number of barriers to moving beyond passwords, such as diversity of requirements, user reluctance and usability, individual control of end user systems etc. As of today, alphanumeric passwords are still the most common mode of authentication; hence the focus rests on improving the security of passwords and their authentication. Halderman et al [2] bypass the need to remember multiple passwords for different accounts by using a strengthened hash function to generate high entropy passwords when they are needed. These passwords are protected by a single short master password. Udi Manber[3] implemented a scheme with two salts to prevent guessing attacks on passwords protected with one way functions. So far, most of the existing research focuses on secure management and storage of passwords.

Keith at al [7] presented an empirical study based on the usability of passphrases after a 12 week long experiment. Campbell et al [8] proved that enforcing good password composition rules does not discourage users from setting strong and meaningful passwords. Alain et al [9] used Persuasive Technology as a method to help users chose memorable passwords without forgoing security. Schechter et al [10] propose a method to strengthen user passwords by setting a minimum acceptable false positive rate to prevent statistical guessing attacks. Duggan et al[19] analyzed the password goals for different groups such as students, administrative staff and scientists and observed how password security was related to the sensitivity of their tasks. Kharod et al[20] proposed a new technique that involves the use of hashing, salting and differential masking with a low time complexity to strengthen passwords. Bailey at al[21] studies the fact that users pick passwords of different strengths for different categories of websites; financial accounts have significantly stronger passwords and analyzes the implications of this fact on password research. Despite research on strengthening passwords, data continues to be compromised on a regular basis, prompting the need for better vigilance and stronger passwords from both users as well as organizations. This paper focuses on how organizations as well as individual users can safeguard their data better against malevolent attacks.

III. Aspects of Password Security

There are many aspects to password security that must be considered. These include the manner in which passwords are stored. Secure password storage is crucial in protecting passwords from malicious attacks. Plain text, hashing, salted hashing, rainbow tables are all different methods of storing passwords. Also to be considered are whether the passwords are human generated or computer generated. Computer generated passwords generally possess a higher degree of randomness. Password theft is also an issue to be considered. Password can be stolen through social engineering, brute forcing, keylogging and such. The following subsections explain the various aspects of password security.

A. Password Storage

A password can be made up of characters, numbers and/or special characters. Passwords are mostly case sensitive. Passwords can be entirely numeric. They are called passcodes and are often used as PINs (Personal Identification Numbers) in ATMs and Net banking operations. Passwords are stored online in a number of ways. Some are much more secure than others and some are very vulnerable to attacks. The following section lists a few of the most popular ways.

Plain Text Passwords – This is the simplest form of storing a password. Somewhere on the server of the site, there is a database which stores passwords and usernames in plain text. If the password is 'PassText321' then in the database, the password is stored as 'PassText321'. This is the worst form of storing passwords in terms of security. If the site is hacked and the passwords are stored in human readable form, then all the passwords are immediately compromised. The hacker can read all the passwords with virtually no extra effort.

Encrypted Passwords – Many sites store an encrypted form of the password in the database on their server. Encryption uses a special key to convert the password into a random string of text. The advantage is, without the key, the hacker cannot obtain the passwords. All that can be obtained are the random encrypted strings. The disadvantage is the key is often stored on the same server where the passwords are. So if the server is hacked and the key is retrieved then all the passwords can be decrypted and compromised. The very fact that encryption is reversible, i.e. a message can be coded and decoded poses a security threat.

Hashed Passwords – Hashing is a function that will turn the password into a random long string of letters and numbers. The advantage of hashes over encryption is that hashes are irreversible. Once the password is hashed, there exists no algorithm to change it back to the original password. The hacker would have to hash a number of combinations one-by-one to see which hash matches with the one stored on the server. One way to do this is rainbow tables, which are computationally very fast. Hackers can also use a brute force attack, where every possible combination of letters and numbers are tried, hashed and matched with the hash retrieved from the database. This method can take a very long time and is largely dependent on how powerful the machine is. However today, the computers have become very fast and brute force attacks like John The Ripper can crack passwords quite efficiently. Different types of hashing algorithms like MD5, SHA-1, SHA-256, and SHA-512 exist.

Salted Hashes – To make hashes more secure, 'salt' can be added to the hash. This means that, a random string of characters is either prefixed or postfixed to the password before hashing it. Every password has a different salt. Even if the salts are stored on the database, it will be very complicated cracking the passwords using a rainbow table as the salted passwords are long, complex and unique. Salted hashes can be brute forced but the time taken is significantly longer. Using two salts, one public and one private can also protect the password against offline attacks [3].

B. Human Generated Passwords Vs Randomly Generated Passwords

Passwords can be either human generated or random generated. A random number generator generates a random string of numbers with characters from a pre-defined character set. Each character in the character set has the same probability of being chosen. A pseudorandom number generator (PRNG) generates a random sequence and has applications in cryptography. PRNG numbers are not truly random because it is generated from a small set of initial values. This set is called the PRNG's state and a truly random seed is included within it.

Human generated passwords are never really random. Human generated passwords are usually easy to remember. Humans choose passwords that usually are similar to some element of their lives. Like addresses, birthdates, and names of relatives, or words that are commonly used in everyday life. Passwords like 'abcdefg' or '123456' are also commonly used. With people possessing multiple accounts, it is hard to remember so many different passwords. So, most opt for using short easy-to-remember passwords. This makes human generated passwords more vulnerable and easy to guess [4]. It has also been noted that web users have a tendency to reuse their passwords [4]. If a single password becomes known, then more than one account will be compromised. Since most passwords are human generated, it falls to individual users to make sure the passwords are strong and secure.

C. Password Theft

Passwords can be leaked in a number of ways. An attacker can hack into the database of the site which stores the user credentials and uncover a huge number of passwords. Thefts can also occur on a personal level. A user can write down the password somewhere and it can make its way to malicious hands. Or a user can set a very simple and obvious password that is easy to guess. Social engineering, phishing or keyloggers can also compromise passwords [5]. Passwords can very commonly be uncovered by brute forcing or offline dictionary attacks.

IV. Password Strength

A brute force attack tries every possible combination in a given character set and tries to match it against the original password. So more the number of possible combinations, more the time it will take for the algorithm to generate the guesses. On an average, almost half of the total number of combinations is tried before striking on the right one. The longer it takes to break a password, the stronger it is. So it is logical to conclude that greater the length of a password, the better it can stand against a brute force attack.

Let the length of the password that is to be cracked be N. Let the password consist of only lower case alphabets. This forms the character set. The possible candidates for each character of the password are 26. For a more generic case, let the character set consist of k characters. Then the number of possible passwords can be N^k. So, the length of the password can increase by either increasing N or by increasing k.

If the length of the password is 6 and it is made up of only lower case alphabets then the number of possible passwords is 26^6 which are 308915776. If it were made of upper and lower case characters then the character set size would be 52 and the possibilities would be 52^6, which is 1.9770 x 10^{10}. If the password size is 7 then the possibilities would become 26^7 and 52^7.

To prove that a longer password is indeed more difficult to break than a shorter password, user entered passwords were hashed and then brute forced. Passwords were first hashed by an MD5 hash function. Once the password is hashed, then the combinations are created for a fixed length. Every combination is hashed using the same MD5 hash function and is compared to the hash of the original password. When a match is found, the function terminates. The word whose hash matched the original hash is the correct password. In the worst case scenario, the code will test every single combination before it can find a match. The time taken for each password to be broken is calculated and tabulated.

A. Numeric Tests

The first tests run were for 5 letter passwords. Time to break a single password was calculated and the test was repeated for one hundred different passwords consisting of only lower case alphabets from a-z. The next set of tests was for 6 letter passwords. Again, time required to break a single password was calculated and the test was repeated for one hundred different passwords from a character set of lower case alphabets, a-z. The table shows 20 of the test results. As seen from table 1, the time required to break a six letter password is much higher than a five letter password. And it is also clear from the table that there is more or less a uniform increase in the time. As calculated graphically, the average time increase is 26.

B. Alphanumeric Tests

The next set of tests was run for calculating the time to break 6 letter alphanumeric passwords. Twenty passwords were tested for this. The alphanumeric passwords were compared to twenty randomly selected 6 letters alphabetical passwords and their graphs were computed, which shows how much the password strengthens by adding to its character set. For alphanumeric passwords, the character set becomes 36. Hence for a 6 letter alphanumeric password, number of possibilities is 36^6= 2176782336 and for a 6 letter alphabetic password, number of possibilities is 26^6=308915776.

C. Multiple Case Tests

The next set of tests was run for calculating the time to brute force passwords that comprised of alphabets of both upper and lower case. The character set for multi-case passwords is 52. Twenty random passwords each of 6 letters were tested. These were compared to twenty lower case passwords and their graphs were computed. For each 6 letter multi case password, the number of possibilities is 52^6=19770609664 and for a 6 letter alphabetic password, number of possibilities is 26^6=308915776. The graphical results corroborate the fact, that increasing the character set strengthens the password by a significant amount.

Table 1. Calculated Time to Break 5 Letter, 6 Letter, Alphanumeric and Double Case Passwords.

Sr.no	5 letter password	Time to break	6 letter password	Time to break	Alphanumeric password	Time to break	Double Case password	Time to break
1	Bales	57795.2	abases	1599703	abas34	9944469	Acajou	72887164
2	Candy	58503.7	ballad	1621442	a346be	11524403	Blunts	75462234
3	Delta	52585.6	bennis	1457532	aes3er	10989201	Chough	73235678
4	Egads	56186.9	chinos	1763321	45alze	11031055	Diesel	69984567
5	Feign	55397	daddle	1705889	bes567	10134510	Ethoxy	77567893
6	Garum	47403.85	doting	1514065	045kat	10139948	Flabby	74221345
7	Hoary	68526.15	elects	1557074	bute90	9567085	Gnawed	80556784
8	Igapo	61641.75	fabled	1394745	blips2	10044859	Hector	79556788
9	Lobby	49092.6	glades	1737407	cat101	11071539	Imagos	77564856
10	Maims	60824.15	hacker	1659651	cupola	11116028	Jovial	76554345
11	Nutsy	62828.85	incite	1768656	citco5	11043269	Keener	77908456
12	Peare	60157.1	jinxed	1393465	celt67	11191393	Legmen	72345677
13	Rearm	66847.9	khazen	1613898	delta4	11272714	Macaco	71236578
14	Rough	66346.05	legmen	1398087	5doggy	11167292	Nankin	78665432
15	Skids	67386.45	milady	1623292	death8	11417336	Oafish	69783321
16	Taboo	67245.85	nibble	1642988	dupe33	11597704	Pablum	70112345
17	Thyme	66887.05	odours	1636991	epm4t6	11591159	Quiche	71864579
18	Users	42237.35	phenom	1554071	epm4t6	10855216	Rabato	74556789
19	Xylem	56794.2	quaked	1592733	34egg7	9858146	Sebums	73455675
20	Zonal	61287.9	stomps	1651824	etoph4	10832474	Valued	75338904

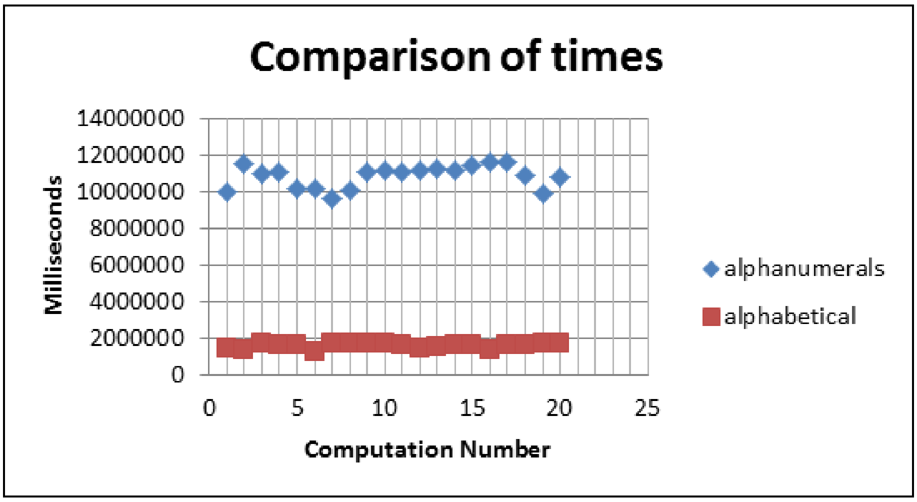

Fig.1. Time Taken to Break 6 Letter Alphabetical Passwords and Alphanumeric Password

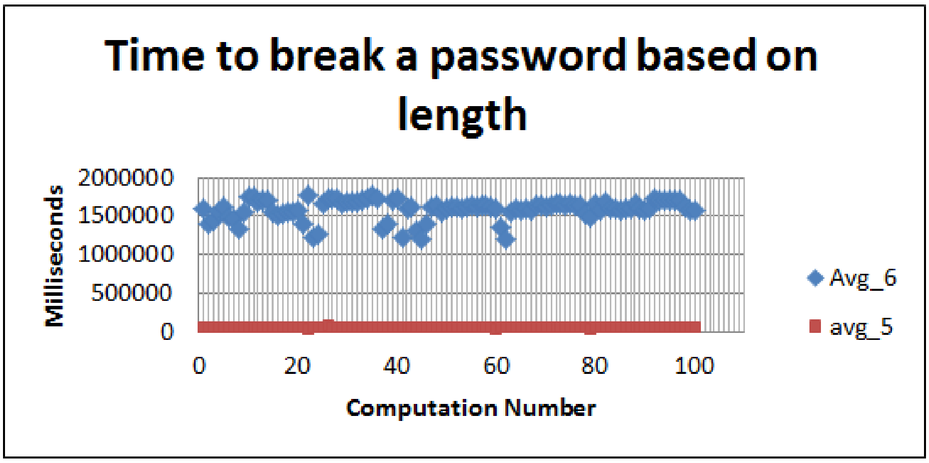

Fig.2. Time Taken to Break 5 and 6 Letter Passwords

V. Entropy

A. *Entropy as Information Content*

Entropy is defined in the context of a probabilistic model. A code that generates a string of “BBBBB…” will have entropy of zero because there is no uncertainty in the next character. It is known that the next character must be ‘B’. If a 256 bit key is randomly generated, then it has 256 bits of entropy. But if every digit is not of equal probability, then the entropy will fail to reflect the true unpredictability. If the key is “cryptography” 50% of the time and a truly random 256 bit key, then the number of entropy is approximately 128 bits but the number of guesses it takes to brute force it may not be 2^{128-1} but 2^{256-1} as half the times, the password can be cracked on the first try and the other times, it needs to guess.

B. *Password Strengths in terms of Entropy*

When it comes to passwords, entropy is used to specify the strength of a password in terms of its information content, measured in bits. A password of m bits strength would need 2^m tries to exhaust all possibilities in a brute force attack. Clearly, the higher the entropy, greater is the strength of the password.

Entropy is given by:

$$H = L* \log_2 N \tag{1}$$

where L is the length of the password and N is the character size.

Let the password be ‘Ast34beta1’ which is chosen out of a 62 size character set. Then the entropy is H= 10 * $\log_{10}62/\log_{10}2$; which is H= 59.541 bits.

Therefore the entropy of a password depends both on length and the number of total possible characters. What increases the entropy per bit more- length or size of character set? From the equation is it clear, that the length of the password matters more.

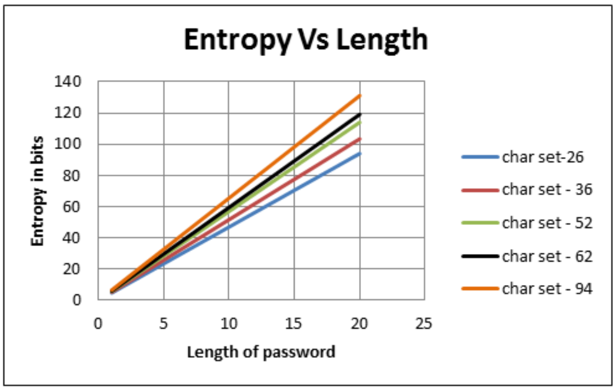

Fig.3. Entropy Vs Length

Table 2. Entropy of passwords of increasing length and fixed character set

Sr. no	Charset-26	Charset-36	Charset-52	Charset-62	Charset-94
1	4.700439718	5.169925001	5.700439718	5.95419631	6.554588852
2	9.400879436	10.33985	11.40087944	11.90839262	13.1091777
3	14.10131915	15.509775	17.10131915	17.86258893	19.66376656
4	18.80175887	20.67970001	22.80175887	23.81678524	26.21835541
5	23.50219859	25.84962501	28.50219859	29.77098155	32.77294426
6	28.20263831	31.019550	34.20263831	35.72517786	39.32753311
7	32.90307803	36.18947501	39.90307803	41.67937417	45.88212196
8	37.60351775	41.35940001	45.60351775	47.63357048	52.43671081
9	42.30395746	46.52932501	51.30395746	53.58776679	58.99129967
10	47.00439718	51.69925001	57.00439718	59.5419631	65.54588852
11	51.7048369	56.86917502	62.7048369	65.49615941	72.10047737
12	56.40527662	62.03910002	68.40527662	71.45035572	78.65506622
13	61.10571634	67.20902502	74.10571634	77.40455204	85.20965507
14	65.80615605	72.37895002	79.80615605	83.35874835	91.76424392
15	70.50659577	77.54887502	85.50659577	89.31294466	98.31883278
16	75.20703549	82.71880002	91.20703549	95.26714097	104.8734216
17	79.90747521	87.88872502	96.90747521	101.2213373	111.4280105
18	84.60791493	93.05865003	102.6079149	107.1755336	117.9825993
19	89.30835464	98.22857503	108.3083546	113.1297299	124.5371882
20	94.00879436	103.3985	114.0087944	119.0839262	131.091777

VI. Strong Passwords in the Industry

The significance of strong passwords has been sufficiently discussed in the preceding sections. It is obvious that apart from the security measures the organization takes to secure their user’s data, it is also the responsibility of the user to ensure that their passwords are strong. Users can be forced to introduce some measure of complexity to their passwords by enforcing some necessary rules. The user must conform to these rules while choosing their new password at the time of registering. The research in [6] proves that people choose weaker passwords for sites which employ lax rules while registering a new account is sufficiently protected. By going through the login/register/sign-up pages of the following web giants, it was possible to gain sufficient data to understand what rules they insist their customers follow when they create a new account.

Ebay.com [11] – ebay.com enforces the following rules for passwords –

- Minimum of six and a maximum of 20 characters
- At least one number and/or a special symbol
- Must be case sensitive. That is, must contain both uppercase and lowercase letters
- Passwords are categorized as ‘weak’, ‘medium’ or ‘strong’. The user is notified if the password is ‘invalid’ or ‘too short’. The password is classified as ‘medium’ or ‘weak’ unless alphabets, numbers and special symbols are used. To be classified as ‘strong’, the password must not only consist of alphabets (both upper and lower), numerals and special symbols but must also have a length greater than 6. Password of length six with all combination of characters gives a ‘medium’ rating but making it of length seven or more makes it strong.

Amazon.com [12] – amazon.com enforces the following rules for a password while registration of a user account

- Must have a minimum of 6 letters
- Must be a combination of upper and lower case and/or a combination of letters and numbers.

Flipkart.com [13] – flipkart.com enforces the following rules on passwords during registration

- It must have a minimum of four characters

Facebook.com [14] – facebook.com enforces the following conditions on passwords while registration of a new user.

- Must have a minimum of 6 characters with respect to length.

Adobe.com [15] – adobe.com enforces the following conditions on passwords while registration of a new user.

- Must have a minimum of 6 characters with respect to length.

Hotmail.com [16] – hotmail.com enforces the following conditions on passwords while registration of a new user.

- Must have a minimum of 8 characters with respect to length.
- It must contain any two of the following- upper case letters/ lower case letters/ numbers/ special symbols.

After a short analysis of the rules which the sites mentioned above enforce, the author concludes that flipkart.com has the least password security. The strongest rules enforced are by ebay.com followed by hotmail.com. Their restrictions force users to set passwords that are naturally hard to brute force.

VII. Password Strength Checker

A. *Design and Description*

This paper designs and implements a password strength checker called PwdStrength. It scores the user entered password against a number of factors and returns the score along with the classification of ‘weak’, ‘fair’, ‘strong’ or ‘invalid’. These factors have been determined from the analysis in the previous sections. The five factors are:

Length: As discussed above, the length of a password can be the strongest deterrent to a brute force attack. If the length is large enough, it can even render useless a rainbow table.

Character set: By increasing the size of the character set, the number of possibilities or guesses that the computer will have to make to chance upon the correct password increases. Like discussed before, the total number of possible tries a computer can make for a password is (size of character set)$^{\text{length of password}}$.

Entropy: In information theory, Entropy measures the uncertainty in a random variable. More the entropy, more the uncertainty, hence lesser are the chances of guessing.

Predictability: – It has been observed that in an effort to keep easy-to-remember password, people tend to use alphabets and numbers in order. That is, “abcd” or “345” etc. If the hacker is aware that the last 2 or 3 digits are numbers, then the chances are that the numbers will be in order. This will lessens the password strength because it becomes much easier to guess a set of numbers in order.

Commonness: When users set passwords, they tend to set common passwords. If not the name of a close family member or a place, it is often something like ‘password’, ‘password123’, or ‘abcdef4567’ etc. If the password is something very common then the attacker may run a dictionary attack using a dictionary of common words list or a phrase book and there’s a chance that the attack is successful. The code checks the password entered against a list of 10000 most common passwords of 2014. Many vendors generate these lists, however the most comprehensive list was found at xato.net [17].

B. *The Scoring System*

The password is scored out of ten on each of the following factors mentioned in the previous sections. The 5 scores are totaled and averaged to give the final score out of 10. The scoring system for each factor is as follows:

If length >12: Score = 10

- If length>= 10 and length < 12: Score = 8
- If length>= 8 and length < 10: Score = 6
- If length>= 6 and length < 8: Score = 4
- If length < 6: No score
- If character set = 26; score = 2
- If character set = 36; score = 4
- If character set = 52; score = 6
- If character set = 62 or 84; score = 8
- If character set = 94; score = 10

For the weakest password that is not invalid, the entropy is found to be approximately 28. For a very complex password with a length of 12 or more, the entropy is found to be over 85. Therefore, this range of entropy has been divided into categories and scored accordingly.

- If entropy<=28; score =2
- If entropy >28 and entropy <=47; score =4
- If entropy >47 and entropy <=66; score =6
- If entropy >66 and entropy <=85; score =8
- If entropy >85; score = 10

For every three letters or number that is in order, one point is deducted from 8. For the first three in-order characters, one point is deducted from 8. After that, for every consecutive character in order, another is removed. Hence ‘12345’ scores an 8-3=5 because ‘123’ warrants -1, ‘234’ warrants -2, ‘345’ warrants -3. However ‘ab’ won’t result in any negative score as many words contain two in-order characters such as ‘**ab**solute’ or ‘**ef**fort’. If however, there are more than 10 such consecutive triplets, then the password is scored a 0 on predictability.

If the password is found among the top ten thousand common passwords, it is awarded a score of 5 out of 10. And if not, it is awarded 8 out of 10. If the password is commonly found, a warning is displayed saying that it is a common password.

On these 5 factors, the password is marked on a scale of 10 separately. Then it is averaged, which gives us the result out of 10. If the final score is less than 4, the password is termed ‘WEAK’. If the score is equal to or above 8, it is classified as ‘STRONG’. Any score in between, the password is termed as ‘FAIR’.

C. Results and Discussion

The The PwdStrength was tested for a number of passwords and their results tabulated. A number of computer users were asked to volunteer passwords they are likely to keep should they open up new accounts. The same passwords were tested by The Password Meter [18], a popular online strength checking website. The Password Meter takes into consideration the character set of the password, the length, consecutive letters, numbers and repeated characters. It does not check for entropy or whether the password is a common one or not. The output of the code is presented in Table 3 below.

Table 3. Output of PwdStrength vs Output of Password Meter

Sr.	**PwdStrength**	Score	Category	**PasswordMeter**	Score @ PasswordMeter
1	Kathachanda	5.0	FAIR	kathachanda	Very Weak: 11%
2	TestingJava123	8.0	STRONG	TestingJava123	Very Strong: 92%
3	HELLO	-	Invalid password	HELLO	Very Weak: 4%
4	Password	4.0	WEAK: Common!	Password	Weak: 26%
5	Cheryl	4.0	WEAK: Common!	Cheryl	Weak: 22%
6	Password!Security	5.0	FAIR	Password!Security	Very Strong: 95%
7	DrJekyll1234Hyde	8.0	STRONG	DrJekyll1234Hyde	Very Strong: 100%
8	Testing1234567	7.0	FAIR	Testing1234567	Very Strong: 100%
9	Amityuniversity	6.0	FAIR	Amityuniversity	Good: 48%
10	camp@#*	4.0	WEAK	camp@#*	Good: 50%
11	4567!#$	3.0	WEAK	4567!#$	Strong: 60%
12	HarryPotter23	8.0	STRONG	HarryPotter23	Very Strong:83%
13	jeromealpha45	6.0	FAIR	jeromealpha45	Good: 43%
14	Tedious$affair	6.0	FAIR	Tedious$affair	Very Strong: 81%
15	Just1Got#Home	9.0	STRONG	Just1Got#Home	Very Strong: 95%
16	JokesterTell321	8.0	STRONG	JokesterTell321	Very Strong: 95%
17	Blake123	7.0	FAIR	Blake123	Strong: 65%
18	annie12	5.0	FAIR	annie12	Weak: 30%
19	Tennis	4.0	WEAK	tennis	Very Weak: 6%
20	fireman56	6.0	FAIR	fireman56	Weak: 36%
21	TellMeWhy65	8.0	STRONG	TellMeWhy65	Strong: 79%
22	NoGood@Food	8.0	STRONG	NoGood@Food	Strong:74%

The advantage of PwdStrength is that it can be frequently updated with respect to the common passwords list. Since every year, new lists are published with the weakest passwords, the code can be maintained up-to-date at all times. Also, as new passwords are added to the old list, a user’s current password may become too easy to guess. Every time the list is significantly updated, the sites can issue a warning to the users to change their password for security reasons. This makes the algorithm efficient and adaptive as it constantly keeps track of recently popular passwords and rejects them for new users.

VIII. Conclusion

It is abundantly clear from this paper, the importance that should be attached to passwords. The ease with which passwords can be broken and data can be compromised has also been clearly explained. Sites should take effective measures to make sure that their user's data is sufficiently secured by ensuring that the correct scheme is employed to protect against hacks. The role played by users in securing their data is also emphasized. The algorithm explained in the previous section can help force users to employ passwords which are complex and difficult to break. If employed and improved upon further, it would go a long way towards making digital data much more secure. The threat of data being compromised will always exist. Risk may be minimized but it can never be eliminated. To that effect, it is always better to be safe than sorry.

References

[1] Herley, Cormac, Paul C. van Oorschot, and Andrew S. Patrick. "Passwords: If we're so smart, why are we still using them?" *Financial Cryptography and Data Security*. Springer Berlin Heidelberg, 2009. 230-237.

[2] Halderman, J. Alex, Brent Waters, and Edward W. Felten. "A convenient method for securely managing passwords." *Proceedings of the 14th international conference on World Wide Web*. ACM, 2005.

[3] Manber, Udi. "A simple scheme to make passwords based on one-way functions much harder to crack." *Computers & Security* 15.2 (1996): 171-176.

[4] Yan, Jianxin, Alan Blackwell, Ross Anderson, and Alasdair Grant. "The memorability and security of passwords: some empirical results." *Technical Report-University of Cambridge Computer Laboratory* (2000): 1.

[5] Gayathiri Charathsandran, "Text Password Survey: Transition from First Generation to Second Generation" unpublished.

[6] Florêncio, D., and C. Herley. "A Large-Scale Study of Web Password Habits in Proc." (2007).

[7] Mark Keith, Benjamin Shao, Paul John Steinbart, The usability of passphrases for authentication: An empirical field study, *International Journal of Human-Computer Studies*, v.65 n.1, January, 2007, p.17-28.

[8] Campbell, John, Dale Kleeman, and Wanli Ma. "The good and not so good of enforcing password composition rules." *Information Systems Security* 16.1 (2007): 2-8.

[9] Alain Forget, Sonia Chiasson, and Robert Biddle. 2007. Helping users create better passwords: is this the right approach?. In *Proceedings of the 3rd symposium on Usable privacy and security* (SOUPS '07). ACM, New York, NY, USA, 151-152.

[10] Schechter, Stuart, Cormac Herley, and Michael Mitzenmacher. "Popularity is everything: A new approach to protecting passwords from statistical-guessing attacks." *Proceedings of the 5th USENIX conference on Hot topics in security*. USENIX Association, 2010.

[11] Ebay.com. www.ebay.com

[12] Amazon.com. www.amazon.com

[13] Flipkart.com www.flipkart.com

[14] Facebook.com www.facebook.com

[15] Adobe.com www.adobe.com

[16] Hotmail.com www.hotmail.com

[17] 10,000 Most Common Passwords List. Available: https://xato.net/passwords/more-top-worst-passwords

[18] Password strength. Available: http://www.passwordmeter.com

[19] Duggan, Geoffrey B., Hilary Johnson, and Beate Grawemeyer. "Rational security: Modelling everyday password use." *International journal of human-computer studies* 70.6 (2012): 415-431.

[20] Kharod, Seema, Nidhi Sharma, and Alok Sharma. "An improved hashing based password security scheme using salting and differential masking." *Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), 2015 4th International Conference on*. IEEE, 2015.

[21] Bailey, Daniel V., Markus Dürmuth, and Christof Paar. "Statistics on Password Re-use and Adaptive Strength for Financial Accounts." *Security and Cryptography for Networks*. Springer International Publishing, 2014. 218-235.

Authors' Profiles

Ms Katha Chanda is a final student of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University, Noida. She is currently pursuing her Bachelor of Technology in CSE from the aforementioned university and expects to graduate in June 2016.

Currently, she is a visiting research student at Singapore University of Technology and Design, Singapore, researching on classification of audio files. Her research interests include computer security and machine leaning. Her past work involves a proposal for a hybrid botnet detection framework which was subsequently published.

How to cite this paper: Katha Chanda,"Password Security: An Analysis of Password Strengths and Vulnerabilities", International Journal of Computer Network and Information Security(IJCNIS), Vol.8, No.7, pp.23-30, 2016.DOI: 10.5815/ijcnis.2016.07.04