Lab 7- Password Cracking

The purpose of this lab is to learn more about passwords and password complexity. For this lab, you will use a web based password analyzing tool at https://www.grc.com/haystack.htm provided by Gibson Research Corporation.

If your password is the “needle” then the ability to hide your password depends on making the “haystack” as big as possible. You will also learn that some of the “truths” about passwords are myths. For example, which of the following two passwords is stronger, more secure, and more difficult to crack?

D0g.....................
PrXyc.N(n4k77#L!eVdAfp9

We have been told that clearly the second password is the better one because it is more secure. It is also impossible to remember. The Gibson Research tool, however, will show that the first password is not only easier to remember, but is 95 times more difficult to crack that the second password.

1. **Read the entire documentation on the web site, you will find it very interesting and informative. The information on this site is part of your course work and will appear on future tests.**

Exercise 1: Using the 10 most common passwords used in the world.

This list was compiled by PCMag.com and is something every “bad guy” has memorized. If your password is among this list, you may as well hand over your wallet or purse to bad guy right now.

1. Enter the password list below and record the Search Space Size, as a power of 10 and Offline Fast Attack Scenario. The first parameter measures the size of the haystack, and the second measures the speed of cracking based on current PC installed cracking tools. **[0.4 Marks]**

Rank	Password	Search Space Size (power of 10)	Offline Fast Attack Scenario
1	password	2.17 x 10^{11}	2.17 seconds
2	123456	1.11 x 10^{6}	0.0000111 seconds
3	qwerty	3.21 x 10^{8}	0.00321 seconds
4	abc123	2.24 x 10^{9}	0.0224 seconds
5	letmein	8.35 x 10^{9}	0.0835 seconds
6	monkey	3.21 x 10^{8}	0.00321 seconds
7	myspace1	2.90 x 10^{12}	29.02 seconds
8	password1	1.04 x 10^{14}	17.41 minutes

9	link182	8.06 x 10^{10}	0.806 seconds
10	dev (My first name)	1.83 x 10^{4}	0.000000183 seconds

Exercise 2: Adding Complexity and Length to Password

2. Now you will analyze how the search space and complexity influence the ability to crack the password. **[0.4 Marks]**

Rank	Password	Search Space Size (power of 10)	Offline Fast Attack Scenario
1	460	1.11 x 10^{3}	0.0000000111 seconds
2	4609	1.11 x 10^{4}	0.000000111 seconds
3	4d6A09	5.77 x 10^{10}	0.577 seconds
4	4d6A09	5.77 x 10^{10}	0.577 seconds
5	4d6A0%9	7.06 x 10^{13}	11.76 minutes
6	SeNeCa	2.02 x 10^{10}	0.202 seconds
7	SeNeCa/	3.24 x 10^{13}	5.41 minutes
8	SeNeCa//	2.76 x 10^{15}	7.66 hours
9	SeNeCa////	1.99 x 10^{19}	6.33 years
10	SeNeCa//??	1.99 x 10^{19}	6.33 years

3. Clearly the “SeNeCa//??” password is easier to remember than “4dA0%9”. What conclusion can your draw from the above Exercise: (write 3-4 sentences to explain your conclusion) **[0.275 Marks]**

➔ The conclusion that I can make from the above exercise is that the password is not said to be strong based on the “Entropy” or “randomness of the characters in the password” but passwords with “Low Entropy of characters” are much stronger. Because, the attacker is totally blind and unknown with how your password looks and what characters it might have. Hence it all cuts down to a guessing game for the attacker, he (the attacker) only needs to know whether the password guess, he made, was an exact match or not ?, which is quite difficult, to predict all of the characters. So, “high entropy” passwords - “4dA0%9”, are easy to crack while doing exhaustive password search, as compared to “low entropy” passwords - “SeNeCa//??”. Moreover, the password - “SeNeCa//??” is also two characters long, than as compared to - “4dA0%9”, which makes it harder to crack or find the password by searching. In addition to this, the password - - “SeNeCa//??” has a lowercase letter, uppercase and special characters in it, so it will take a long time by an attacker to decode or search that password.

Exercise 3: Cracking Hashes.

All operating systems store passwords as hash values, either MD5 or SHA-1. There are various tools designed to steal the password hash value. For these tools to work, however, the hacker needs local access to the machine. (If unauthorized people have local access to a workstation, you have a larger security problem than just passwords). Once he/she has captured the hash values, the value is compared offline to a database of hash values to find a match. If the hacker finds a match to the hash value he\she assumes that must be the password. Take the following passwords in the table below and convert to hash values.

1. Navigate to the web page http://passwordsgenerator.net/md5-hash-generator
2. Enter the following passwords to convert to MD5 hash values. Copy the hash value to the table below. **[0.4 Marks]**
3. Navigate to the web page https://crackstation.net . Read the documentation on the web site.
4. Use your phone or wrist watch to record the approximate time it takes to crack the password hash. (in seconds)
5. Enter the Captcha code and Click Crack Hashes

Rank	Password	MD5 Hash Value	Approximate Cracking Time
1	password	5f4dcc3b5aa765d61d8327deb882cf99	0.001 seconds
2	password1	7c6a180b36896a0a8c02787eeafb0e4c	0.02 seconds
3	Passw0rd	d41e98d1eafa6d6011d3a70f1a5b92f0	0.002 seconds
4	P@ssw0rd	161ebd7d45089b3446ee4e0d86dbcf92	0.04 seconds
5	P@ssw0rd.	4d934e4cde0dce1d9b3ecaf84f5672b2	0.05 seconds
6	P@ssw0rd..	628c98267edfd4766db2be05e3b2105f	0.06 seconds (Actually, it wasn't cracked by the cracker software).

1. What conclusion can you make, from the above exercise, about the optimum, character mix? (write 3-4 sentences to support your answer) **[0.2 Marks]**

➔ For the optimum character mixture, one can actually make a password comprising more of special characters, rather than the lower case characters. Since the attacker is aware about the user's ideology to have lower-case characters more in a password, rather than the special characters, and it's actually logical, because it's easy to remember, for the users. So, the attackers will take this as an advantage for themselves and it will be easy for them to search for the correct match for the password, they will eventually aim for the lower-case characters first and the symbol oriented characters last. However, it is always advisable, to have a password with more special characters than the lower-case ones, since it will force the attacker to have a "full depth" search and it will be difficult for them to search for the correct match and they might give up the search process in the middle, hence making our password more and more secured.

2. What does padding (repetition of a character) do for the hacker and for us? (one sentence) **[0.2 Marks]**

➔ Padding (repetition of a character) can turn a simple-to-crack password into a strong password that is actually more difficult to breach for the hackers, while using the simple password cracking techniques – online and offline fast cracking scenarios, dictionary attacks, and brute force cracking, hence making the password more secure for the users (us).

Grading:

- **LearnName_Lab7_Password.docx** – complete the tables and questions
- submit the lab file using the link on MySeneca

Remember replacing learnname with your name for submission.
Submit using the Lab 7 Submission link under MySeneca\Graded Work