To: Joanna Rycerz

Date: February 7th, 2021

Subject: Solution to the problem given in Task1

Question1: What type of hashing algorithm was used to protect passwords?

Answer: It is either MD5 or MD4 (Raw Hash) hashing algorithm.

Question 2: What level of protection does the mechanism offer for passwords?

Answer:

* MD5 is an “iterative” hash function.
* MD5 is generally a considerable mechanism for storing passwords in production.
* MD5, produces a 128-bit hash.
* MD5 is born out of RSA’s algorithm (defined in Internet RFC).
* MD5 is a utility that can generate a digital signature of a file. MD5 belongs to a family of one-way hash functions called message digest algorithms. The MD5 system is defined in RFC 1321.
* The algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input. It is conjectured that it is computationally infeasible to produce two messages having the same message digest, or to produce any message having a given prespecified target message digest. The MD5 algorithm is intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem such as RSA.

Question 3: What controls could be implemented to make cracking much harder for the hacker in the event of a password database leaking again?

Answer:

* One way of making the password hard to crack is by maintaining credentials from multitude of services in a manager like dashlane because they tend to use varied hashing algorithms & even hashing over hashed passwords [e.g. md5(md5($plaintext)) ] to store and keep the strength high, meeting to the rigidity of a strong case for an algorithm to process.
* Reduce redundancy across services such that in case of a leak out of one service doesn’t make the other passwords vulnerable.
* Use alphanumeric character with special characters.
* Reducing occurrence of an adjective on noun or verb which is an obvious prey to brute force attacks.

Question 4: What can you tell about the organization’s password policy (e.g. password length, key space, etc.)?

Answer: It can be very well determined that the organization's password policy is not up to the mark as:

* The key length is at an average of 11.
* Although they do not allow spaces, the use of special characters is probably resisted to a set of common delimiters like ‘\_’.
* The use of numbers increases the resistance of password by a factor of 10 times the digit appears.
* The lack of capital characters splits the password strength by half.
* Not avoiding the occurrence of English verbs like book, popular, eating, hero, life, John Wick, interest, expert in turn making the password vulnerable to brute force attacks.

Question 5: What would you change in the password policy to make breaking the passwords harder?

Answer:

* Keeping a threshold on length and Caution over use of verbs are nouns or adjectives.
* Not allowing basic personal details to be a part of password like name, date of birth, etc.
* Mandating minimum 3 or 4 special characters and minimum one or two capital letter.
* Applying a hashing algorithm over another, recursively to have a strong hashing function.
* Not allowing sibling credentials to assist the password naming, like name / surname / date of birth / sex.

Thank You,

Vaibhav Mehta