**DATE : 10 JAN 2024**

**Memo: Assessing Password Security Measures and Proposed Uplifts**

Respected Sir/Madam,

I have conducted an assessment of our organization's password security measures and would like to present my findings along with proposed uplifts to enhance our security posture.

The result and analysis of my finding in context to this assessment are as follows. I have cracked the some of the leaked password using the ***Hashcat*** tool.

e99a18c428cb38d5f260853678922e03 : **abc123**

d8578edf8458ce06fbc5bb76a58c5ca4: **qwerty**

e10adc3949ba59abbe56e057f20f883e : **123456**

75b71aa6842e450f12aca00fdf54c51d : **P455w0rd**

2c9341ca4cf3d87b9e4eb905d6a3ec45 : **Test1234**

5f4dcc3b5aa765d61d8327deb882cf99 : **password**

25f9e794323b453885f5181f1b624d0b : **123456789**

dc647eb65e6711e155375218212b3964 : **Password**

eb61eead90e3b899c6bcbe27ac581660 : **HELLO**

**Hashing Algorithm Used**: After analyzing the password dump provided, it is evident that our organization is using MD5 hashing algorithm to protect passwords. MD5 is a widely known and outdated algorithm, susceptible to brute-force and dictionary attacks.

**Level of Protection**: MD5 hashing offers a low level of protection for passwords due to its vulnerabilities and susceptibility to cracking techniques. With readily available tools like Hashcat, attackers can easily crack passwords hashed with MD5, as demonstrated by the successful cracking of several passwords in the dump.

**Controls to Improve Security**: To make cracking passwords much harder for potential attackers, several controls could be implemented:

* **Upgrade Hashing Algorithm:** Transition from MD5 to more secure algorithms like bcrypt or Argon2, which offer better resistance against brute-force attacks.
* **Implement Salt:** Salting passwords before hashing adds an additional layer of security by creating unique hashes for each user's password, making it harder for attackers to use precomputed hash tables.

**Organization's Password Policy**: From the analysis, it appears that the organization's password policy lacks robustness. There are indications of weak passwords with short lengths and limited character diversity.

**Proposed Changes to Password Policy**: To strengthen our password policy and make breaking passwords harder, I propose the following changes:

* **Enforce Longer Passwords:** Increase the minimum password length requirement to ensure passwords are harder to crack through brute-force attacks.
* **Require Complexity:** Mandate the inclusion of a combination of uppercase letters, lowercase letters, numbers, and special characters to enhance password complexity and entropy.
* **Implement Regular Password Changes:** Enforce periodic password changes to mitigate the risk of compromised passwords being used for prolonged periods.

By implementing these proposed uplifts, we can significantly enhance our organization's password security measures and reduce the risk of successful password cracking attempts.

Thank you for considering these recommendations. Please feel free to reach out if you require further clarification or assistance in implementing these changes.

Best regards,

Prajwal Chapke