Dear Goldman Sachs

I am writing this email regarding the leaked password database to let you know about my findings and to let you know about my suggestions to improve your password policy and help keep your passwords secure in case of another breach.

After the conducted analysis it was determined that your organization uses the MD5 algorithm an outdated hashing algorithm which offers very little protection for hashing passwords. It was also determined that your current password policy is not aligned with the industries best practices allowing users to have short and noncomplex passwords. Seeing that you were using the MD5 hashing algorithm it was very easy to crack the leaked password hashes with hashcat and crackworkstation which is an online hash decoder.

After cracking the passwords, we found that the organizations password policy has a minimum length of 6 characters for passwords and that there is no specific requirement for password creation. Users can use any combination of words and letters to create a password. As a result, I think the following controls could be implemented to increase the overall level of password protection and make cracking much harder in case of another database leak:

1. increase the password length this will increase the computational effort required to crack the passwords.
2. allow for all character types in a password and require at least one non-alphabetic character and usage of ASCII characters because every additional character increases the time it takes to crack a password exponentially and by adding numbers, symbols, upper and lowercase letters to the password it makes it very difficult to brute force. Thus, having a long, complex password is more secure.
3. Avoid common words and character combinations in your password.
4. regularly check current passwords to ensure that they have not been broken.

Seeing that your current password policy is not aligned with the industries best practices things I would suggest improving your password policy are:

1. Using a better hashing algorithm that provides a higher level of protection.
2. implement salting to prevent usage of rainbow tables and implement peppering to make it harder to crack.
3. Having longer and stronger passwords which increases the time it takes to crack a password exponentially.
4. Train your users to follow these policies.

Kind Regards,

Aaron Alvarez

**Passwords Cracked:**

e10adc3949ba59abbe56e057f20f883e md5 123456

25f9e794323b453885f5181f1b624d0b md5 123456789

d8578edf8458ce06fbc5bb76a58c5ca4 md5 qwerty

5f4dcc3b5aa765d61d8327deb882cf99 md5 password

96e79218965eb72c92a549dd5a330112 md5 111111

25d55ad283aa400af464c76d713c07ad md5 12345678

e99a18c428cb38d5f260853678922e03 md5 abc123

fcea920f7412b5da7be0cf42b8c93759 md5 1234567

7c6a180b36896a0a8c02787eeafb0e4c md5 password1

6c569aabbf7775ef8fc570e228c16b98 md5 password!

3f230640b78d7e71ac5514e57935eb69 md5 qazxsw

917eb5e9d6d6bca820922a0c6f7cc28b md5 Pa$$word1

f6a0cb102c62879d397b12b62c092c06 md5 bluered

8d763385e0476ae208f21bc63956f748 md5 moodie00