After the conducted analysis it was determined that organization uses an outdated password hashing algorithm (MD5) which offers very little protection in the event of a password database leaking. It was also determined that the current password policy is not aligned with industry best practices allowing users to have short passwords (6 characters) and reuse usernames as part of passwords.

As a result of the analysis the following uplifts are proposed to increase the overall level of password protection:

- Use a dedicated password hashing algorithm bcrypt, scrypt or PBKDF2 as this will greatly increase the time needed to crack individual passwords,
- Implement salting to prevent usage of rainbow tables to speed up cracking,
- Increase the minimum password length requirement to 10 characters – this will increase the computational effort required to crack password and will give additional time to change all passwords in the event of the password database being leaked,
- Prevent passwords to be the same as usernames or reused as part of the password – such password combination is easy to check without gaining access to the password database itself.
- It is advised to educate users on creating safe and easy to remember passwords. Having a password policy requiring long passwords with a number of special characters results in user writing passwords down or constantly resetting them. The best way to create a strong and user-friendly password is using passphrases. The best way to create such passwords is to combine a couple of completely random word. It's also advised to use some special characters and numbers as easy to remember substitutions to expand the key space (e.g. mYgranny$cha1rhadstaples)
- Educate users on the benefits of passwords managers. Having a password manager allows having very long and completely random passwords (e.g. M>?{tk6Cfep6BuZ4J)KZWQ8j) without the need to remember/write down. A strong passphrase is still required as a master key for to access the password manager.