Goldman Sachs Security Assessment

Overview

As part of our ongoing security assessment, we have analyzed the current state of password protection within the organization. The findings indicate significant vulnerabilities due to inadequate hashing algorithms and password policies. This report outlines the current issues, evaluates the level of protection provided, and recommends improvements to enhance security.

Findings

1. **Hashing Algorithm Used:**
   - The current hashing algorithm used to protect passwords is MD5.
2. **Protection Level of the Hashing Mechanism:**
   - **MD5 Weaknesses:**
     - MD5 is considered highly insecure for protecting passwords. It is susceptible to fast brute-force attacks, dictionary attacks, and collision attacks.
     - Modern hardware can compute millions of MD5 hashes per second, making it feasible for attackers to crack passwords within a short period, especially if the passwords are not complex.
     - The lack of salting means that identical passwords produce the same hash, further weakening security by enabling attackers to precompute hashes (rainbow tables).
3. **Controls to Improve Security:** To significantly enhance the security of password storage, the following controls should be implemented:
   - **Use Stronger Hashing Algorithms:** Replace MD5 with more secure hashing algorithms such as bcrypt, scrypt, or Argon2. These algorithms are designed to be computationally intensive, slowing down brute-force and dictionary attacks.
   - **Implement Salting:** Add a unique, random salt to each password before hashing. This ensures that even identical passwords result in different hashes, preventing the use of precomputed hash tables.
   - **Enforce Key Stretching:** Use multiple iterations of the hashing process (as supported by bcrypt and Argon2) to increase the time required to hash each password, thus making brute-force attacks more difficult.
4. **Current Password Policy Analysis:**
   - **Password Complexity:** The majority of passwords analyzed were not complex, often lacking a mix of uppercase letters, lowercase letters, numbers, and special characters.
   - **Password Length:** Based on the ease of cracking, it appears that many passwords are relatively short.
   - **Key Space:** The combination of short, simple passwords and the use of MD5 significantly reduces the effective key space, making it easier for attackers to guess passwords.

5. **Recommendations for Password Policy Improvements:** To strengthen the organization's password policy, the following changes are recommended:
    - **Increase Minimum Password Length:** Set a minimum password length of at least 12 characters to increase the key space and make brute-force attacks more difficult.
    - **Enforce Password Complexity:** Require passwords to include a mix of uppercase letters, lowercase letters, numbers, and special characters. For example, enforce a policy that requires:
        - At least one uppercase letter
        - At least one lowercase letter
        - At least one number
        - At least one special character
    - **Implement Password Expiration:** Regularly require users to update their passwords, while ensuring that old passwords cannot be reused within a certain period.
    - **Monitor and Educate Users:** Conduct regular security awareness training to educate users on the importance of creating strong, unique passwords. Implement tools to help users generate and manage secure passwords.

Conclusion

The current state of password security within the organization is inadequate due to the use of the MD5 hashing algorithm and weak password policies. Immediate action is required to mitigate the risk of password breaches. By adopting stronger hashing algorithms, enhancing password policies, and educating users, the organization can significantly improve its security posture and reduce the likelihood of successful attacks.