

HOW TO MANAGE PASSWORDS IN YOUR APPLICATION

DO NOT RESTRICT MAX PASSWORD LENGTH

More chars = higher entropy!
=> Increased safety.
Encourage the usage of
Passphrases!



AVOID HARD CHAR REPETITION RESTRICTION

Multiple characters in a row are
not always a bad password!
Inform the user about it.
Combine with Strength Meter.



DO NOT ENFORCE SPECIAL CHAR USAGE

Get entropy through length not
just special chars or uppercase/
lowercase enforcements.
Still encourage its usage and hint
users via Strength Meter.



REQUIRE 8-12 CHARS MINIMUM

- Depending on your audience:
Require at least 8 chars for a
password. If the target audience
is more tech savvy you can opt
to 12 chars as well.

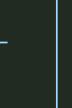


USE TLS/SSL (OBVIOUSLY)

- Transport Encryption is a must!
This is a no brainer but still
important to mention.

DO NOT ENFORCE PLACE OF CHARS

Some Services enforce some
type of characters to be at a
certain place (like the password
should not start with a number).
Don't do that!



USE A STRENGTH METER

- Guide and encourage the user
to use a safer password instead
of enforcing seemingly random
rules!



DO NOT PREVENT PASSWORD C&P

Passwords can be copied from a
Password Manager or pasted
into it. Do not prevent (SSP) the
user from doing that!
It will Make the passwords safer!



ALLOW MOST ASCII CHARACTERS

- Do not restrict the allowed
characters too much. At least
allow space, dash and most
special chars. This helps with
complexity, especially for
random passwords.



NO PASSWORD EXPIRATION

- Refrain from enforcing a
password change on a regular
basis (password expiration)
unless there's good reason to do
so! The user should be able to
change the password at any
time, though!



FORGOT PASSWORD PROCESS SAFE + EASY

You should have an easy to use and safely implemented "Forgot Password" process without sending Passwords in plaintext via email.

ENCOURAGE PASSWORD MANAGERS

Consider encouraging the usage of password managers! You might even publish a list of recommended Managers.

RESTRICT FAILED LOGIN ATTEMPTS

Consider restricting failed login attempts, i.e. lock the login for a couple of (random) minutes after 3 failed attempts.

Completely lock the account after 3 further failed attempts.

Also notify the user via email about it and how they can reactivate (i.e. via "forgot password")!

USE A PASSWORD BLACKLIST

Use a password blacklist and/or Have I been Pwned to check for unsafe passwords.

Combine with Strength Meter.



USE PUBLIC HASH/ KDF ALGORITHMS

- Use modern hash/key derivation function algorithms supporting salts.
i.e. Argon2, bcrypt or scrypt.



USE TWO FACTOR AUTHENTICATION

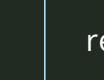
- Two (or multiple) factor authentication is a big plus in app security (correctly implemented).
Can be done via SMS, E-Mail (link/key), App or Fido



DO NOT USE SECURITY QUESTIONS

In the past many services (some still do) used so called security questions for password retrieval or additional security. These questions are often similar and the answers easy to guess for an attacker.

Better use a "forgot password" process + 2FA!



NOTIFY USERS ABOUT BREACHES

Put measures into place to recognize attacks and possible breaches.

Notify your users about breaches, reset the passwords if necessary and let them know so they create new passwords.

© Benedikt Grande

Twitter: @benediktgrande

Website: <https://bgrande.de>

More details at and based on my blog article

<https://bit.ly/3rweZpQ> (German)

Source: <https://bit.ly/3xPQh4W>

Infographic based on canva template

<https://bit.ly/3lsMPs4>