
Utilisez votre Game Boy comme facteur d'authentification OpenID

(sur keycloak)

Qui suis-je ?
Loïc des Rochettes
Consultant, co-fondateur de



Mange et bois keycloak à toutes les sauces
depuis 5ans



Un concept et une
implémentation extensible
et interopérable



OpenID Connect is an interoperable
authentication protocol based on the OAuth
2.0 framework

Keycloak is an open-Source Identity and Access
Management/Single sign-on implementing OpenID
Connect



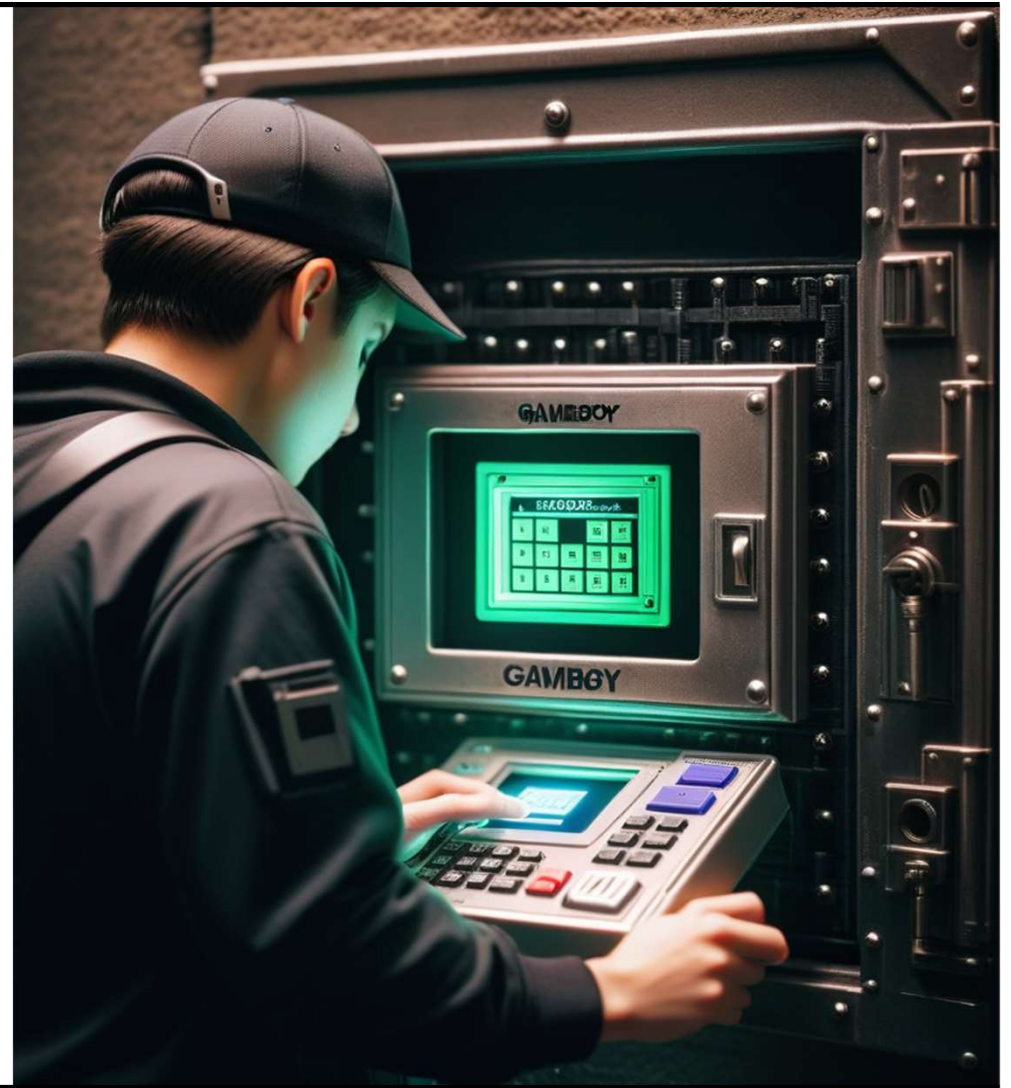
Keycloak -> Authentication SPI

Service Provider Interface : Interface permettant l'extension des capacités de keycloak

Spécifiquement des facteurs d'authentification, par exemple :

E-mail, reCAPTCHA, push notification, etc...

Et pourquoi pas une Gameboy ?



Principes d'authentification

- Une chose que l'on est le seul à ...



Être

- Principalement la biométrie
 - Non applicable dans ce cas
 - Difficile à faire accepter aux utilisateurs
 - Pas aussi infaillible que ce que l'on pense
 - Impossible à renouveler et cas de fuite
-

Connaitre

- Un mot de passe
- Une phrase secrète
- Une information secrète

Le problème avec les mots de passe ?

Le problème avec les questions secrètes ?

Ici, avec la Gameboy c'est une combinaison de touches ?





Posséder

Une clé FIDO

Un compte externe (google, Meta, etc...)


Un smartphone

Une Game Boy ? Sa cartouche ? Sa rom ?
Comment l'identifier de façon unique ?

Keycloak Account Management

localhost:8080/realms/test/account/#/

Navigation privée



Sign in

Welcome to Keycloak account management



Personal info

Manage your basic information

[Personal info](#)



Account security

Control your password and account access

[Signing in](#)

[Device activity](#)

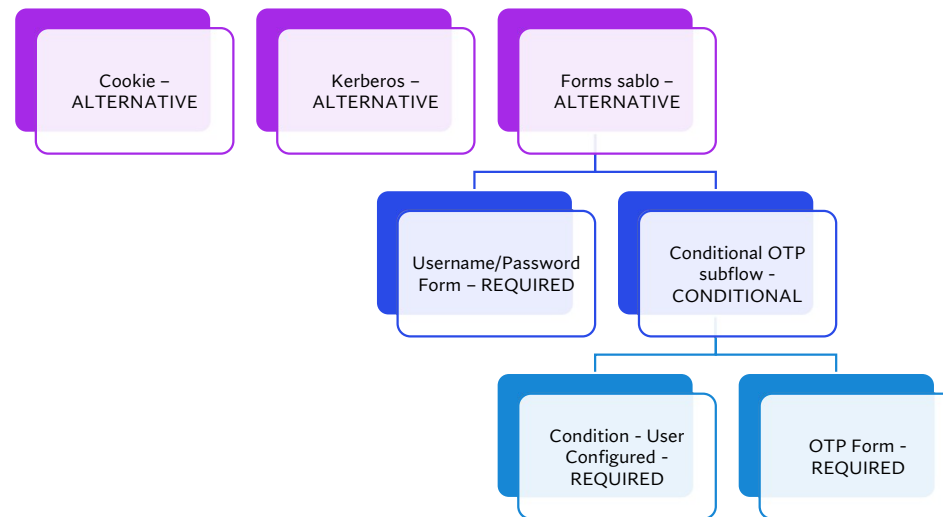


Applications

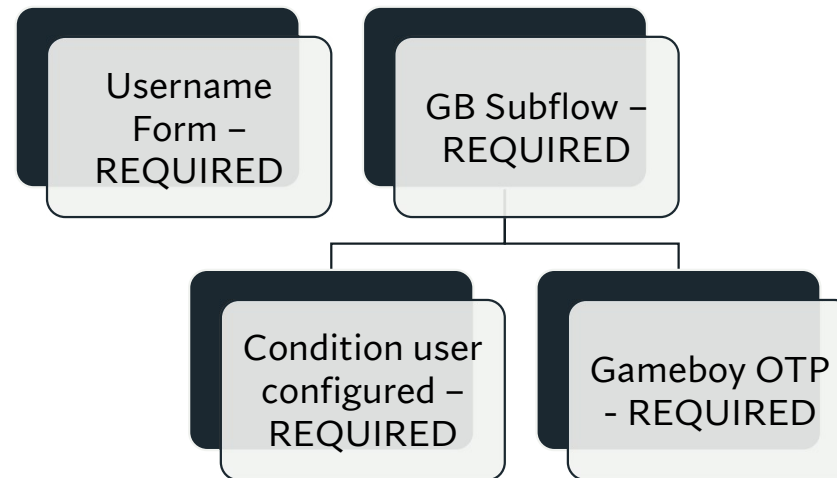
Track and manage your app permission to access your account

[Applications](#)

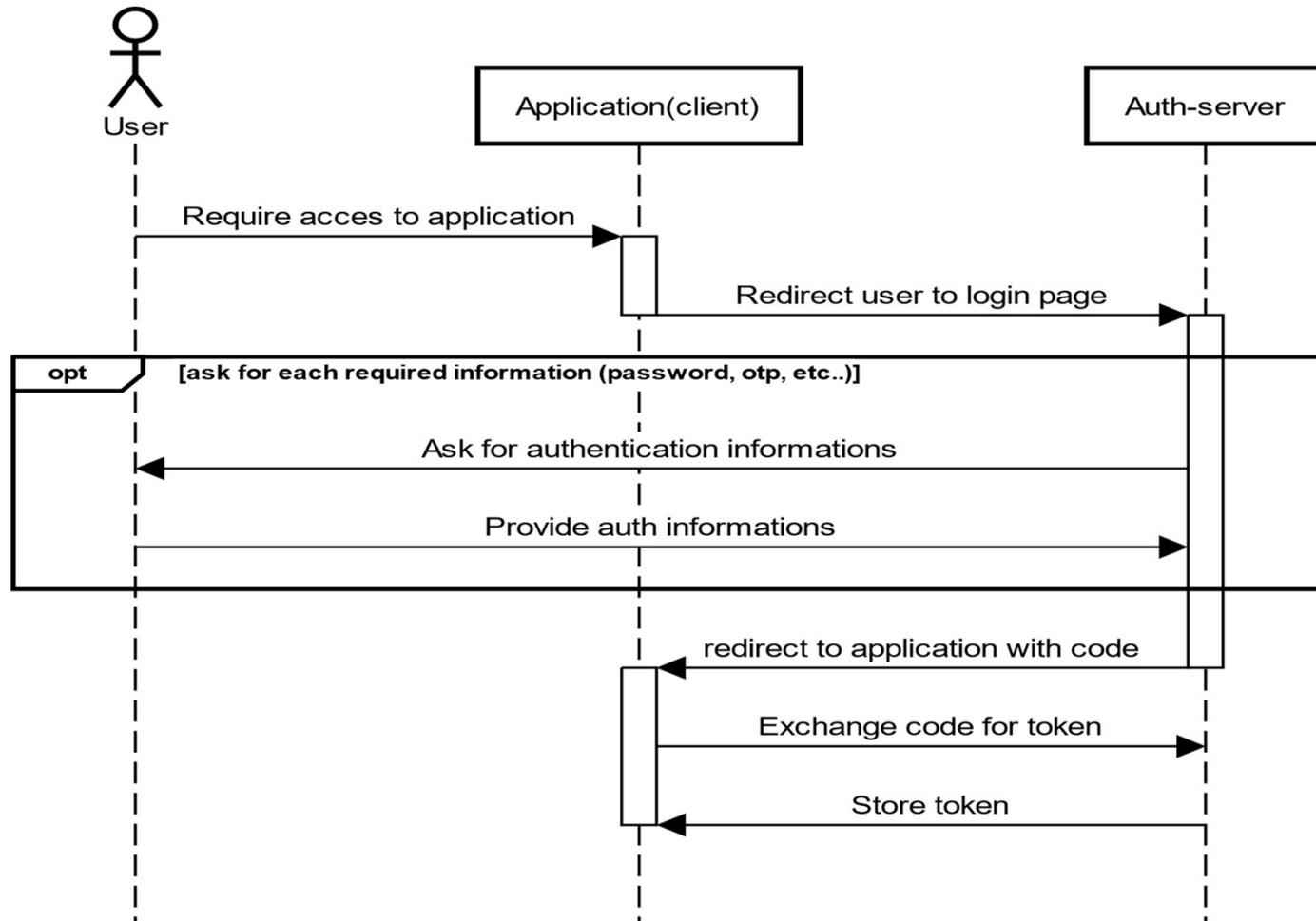
Flux d'authentification par défaut



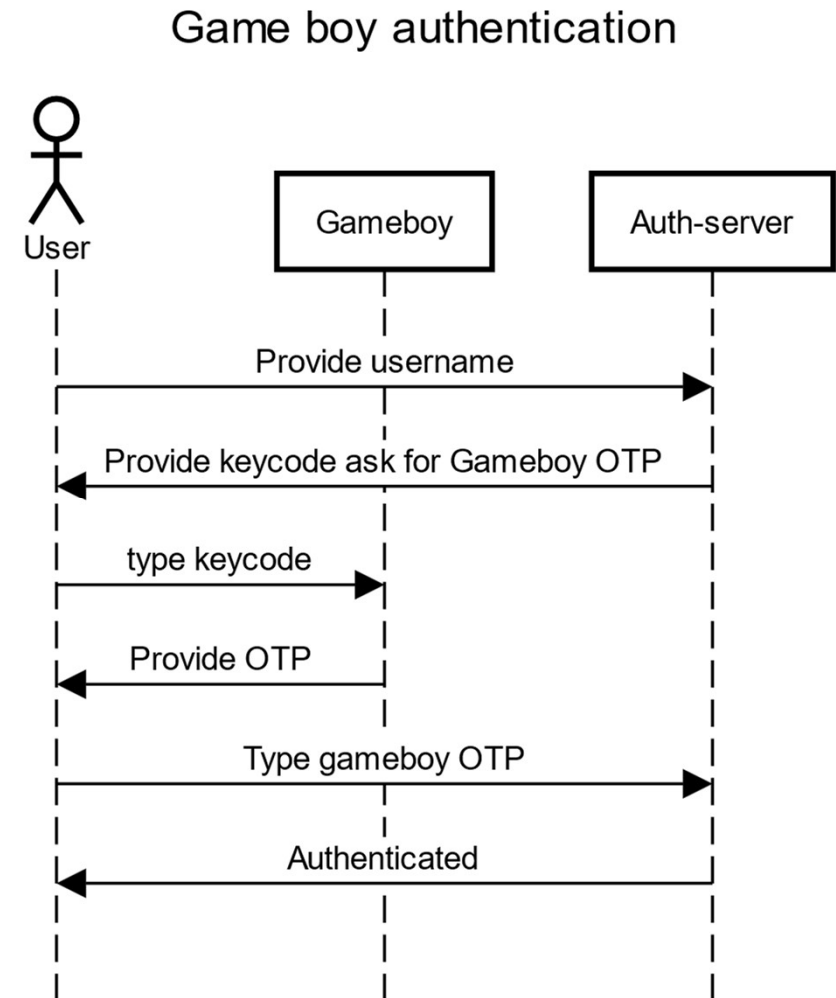
Flux d'authentification Gameboy



Authorization code grant (simplified)



Principe d' authentification basé sur un secret partagé



Algo de hachage

Simple somme de contrôle XOR, suffisant pour la preuve de concept.

La combinaison de touches est convertie en binaire via les codes propres à ceux du registre de la gameboy.

La combinaison de touche est répétée sur toute la longueur du secret et on effectue une somme cumulée (XOR) de l'ensemble sur 3 octets.

Fiabilité/sécurité

- Complexité de la combinaison de touches 6^{10}
 - Complexité du One time password 24bit soit 2^{24}
 - Théoriquement pas si facile que cela a casser sans accès à la rom
-

Intégration dans keycloak

```
public interface Authenticator extends Provider {  
    void authenticate(AuthenticationFlowContext context); //challenge  
    void action(AuthenticationFlowContext context); //Response validation  
    [...]  
}
```

Au delà de la preuve de concept

Un passage en production serait presque possible en générant une rom unique avec un secret fiable pour chaque utilisateur.

Acheter un Gameboy par utilisateur ne sera jamais pertinent... Le but ici était avant tout de vulgariser le concept et les possibilités d'extension du protocole et de keycloak

Il existe déjà des alternatives

Des alternatives existent et doivent être utilisés

Basé sur un principe similaire et production ready

- Clé FIDO
 - One time password (OTP), Authenticator, SMS, mail, etc...
 - Magic link
-

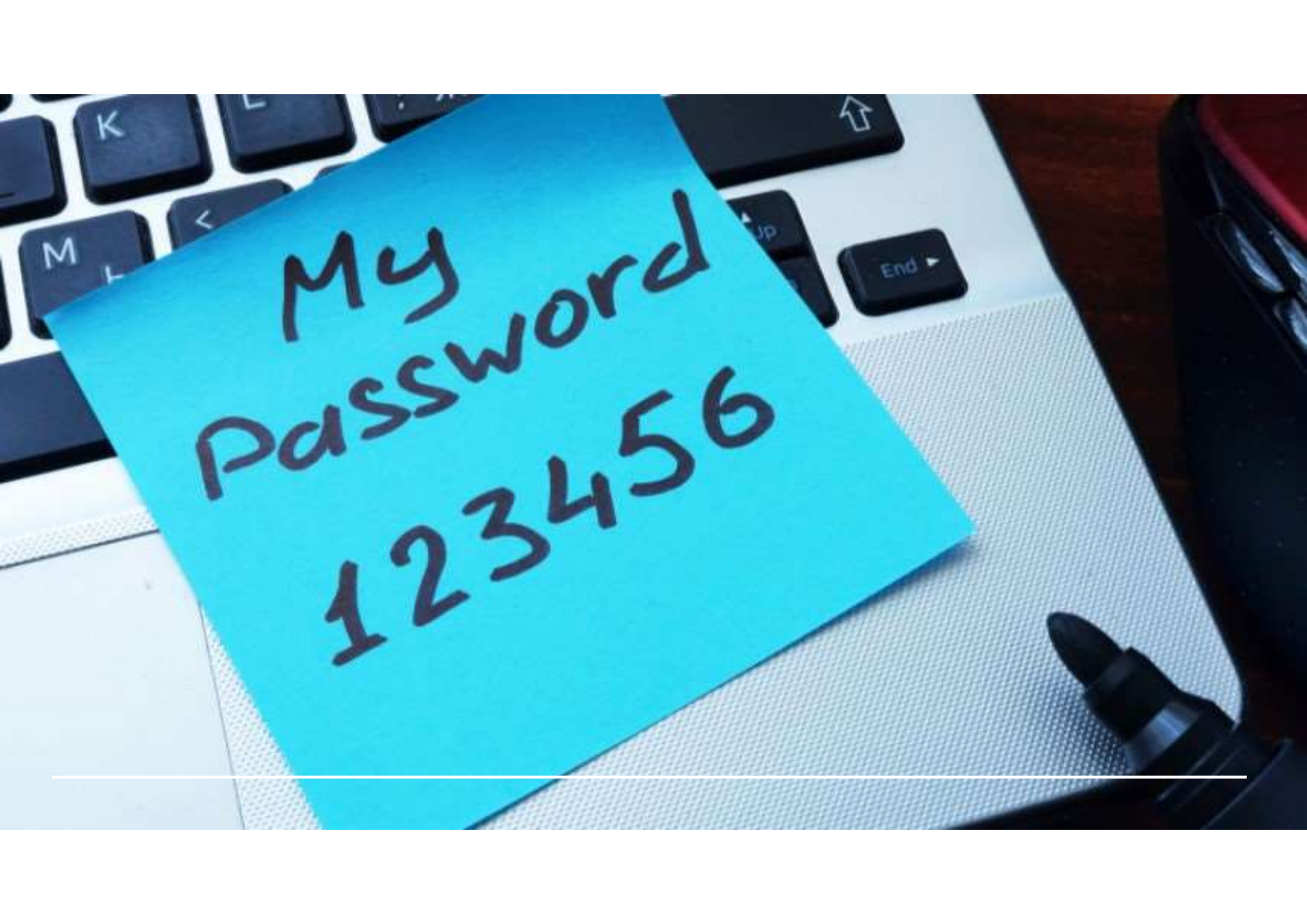
Pour aller plus loin

Avez-vous pensé à proposer une passphrase plutôt qu'un password dans vos conseils aux utilisateurs ? Plus facile à retenir et plus sécurisé.

Utilisez-vous un gestionnaire de mot de passe ? Est-ce que vous en encouragez l'usage ?

Le nerf de la guerre c'est l'expérience utilisateur

Au risque de voir vos super mots de passe finir comme ceci ->

A blue sticky note is placed on a laptop keyboard. The note has the text 'My Password' and '123456' written on it in black marker. The keyboard keys visible include 'K', 'L', 'M', '<', 'End', and 'Up'. A black pen is visible in the bottom right corner.

My
Password
123456

Merci à tous ! Des questions ?



<- Follow me, github repository link coming soon !



<https://x.com/loicdesroc>

<https://please-open.it/>
