




Proposition de sécurisation d'une application

Présentée par Elodie M.



Sommaire



Sécuriser un système

Pourquoi ?

Les protocoles de protection de l'échange de donnée

Hachage / salage

La protection navigateur

Politique des de passes

La sanitization

Session, Token et Cookies

Sécuriser l'authentification

Session

L'accès aux données

Sécurisation de l'API

La journalisation

La stratégie de sauvegarde



Sécuriser un système

RGPD : **R**èglement **G**énéral sur la **P**rotection des **D**onnées

- Moindre privilège
- Réduction de la surface
- Défense en profondeur

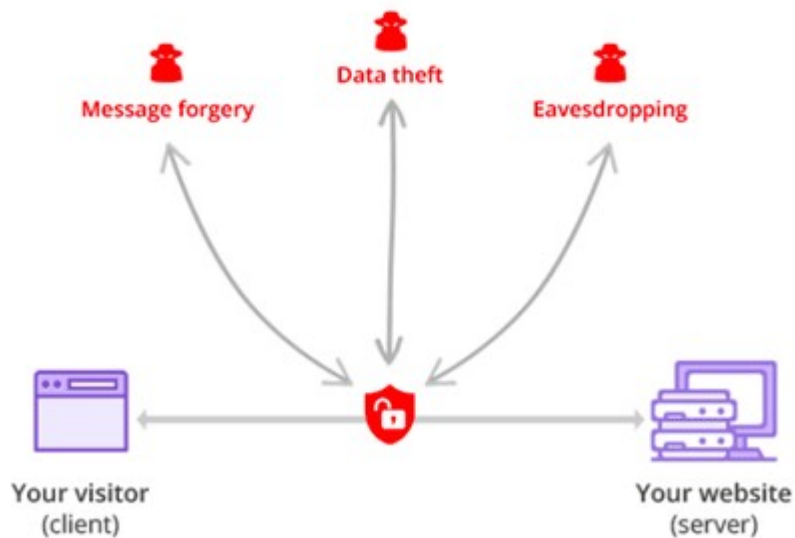


Pourquoi ?

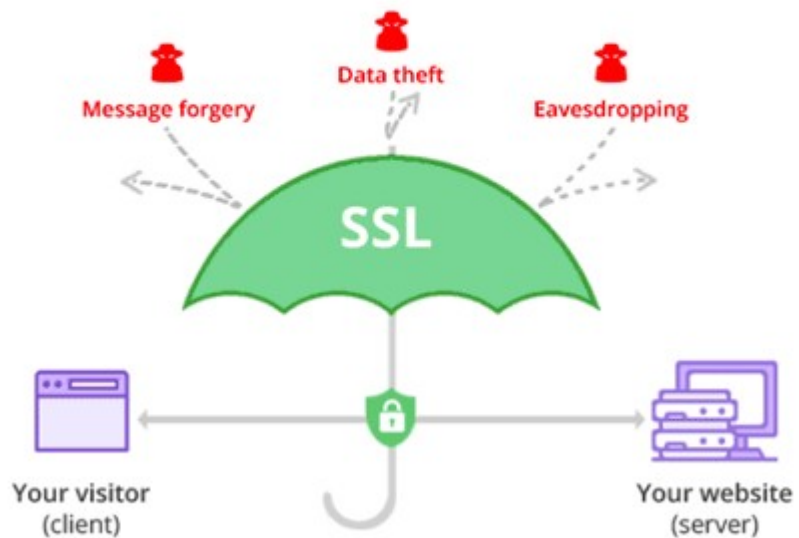
- Il existe de nombreuses failles :
- XSS (Cross-site scripting)
- SQLI (SQL injection)

Les protocoles de protection de l'échange de donnée

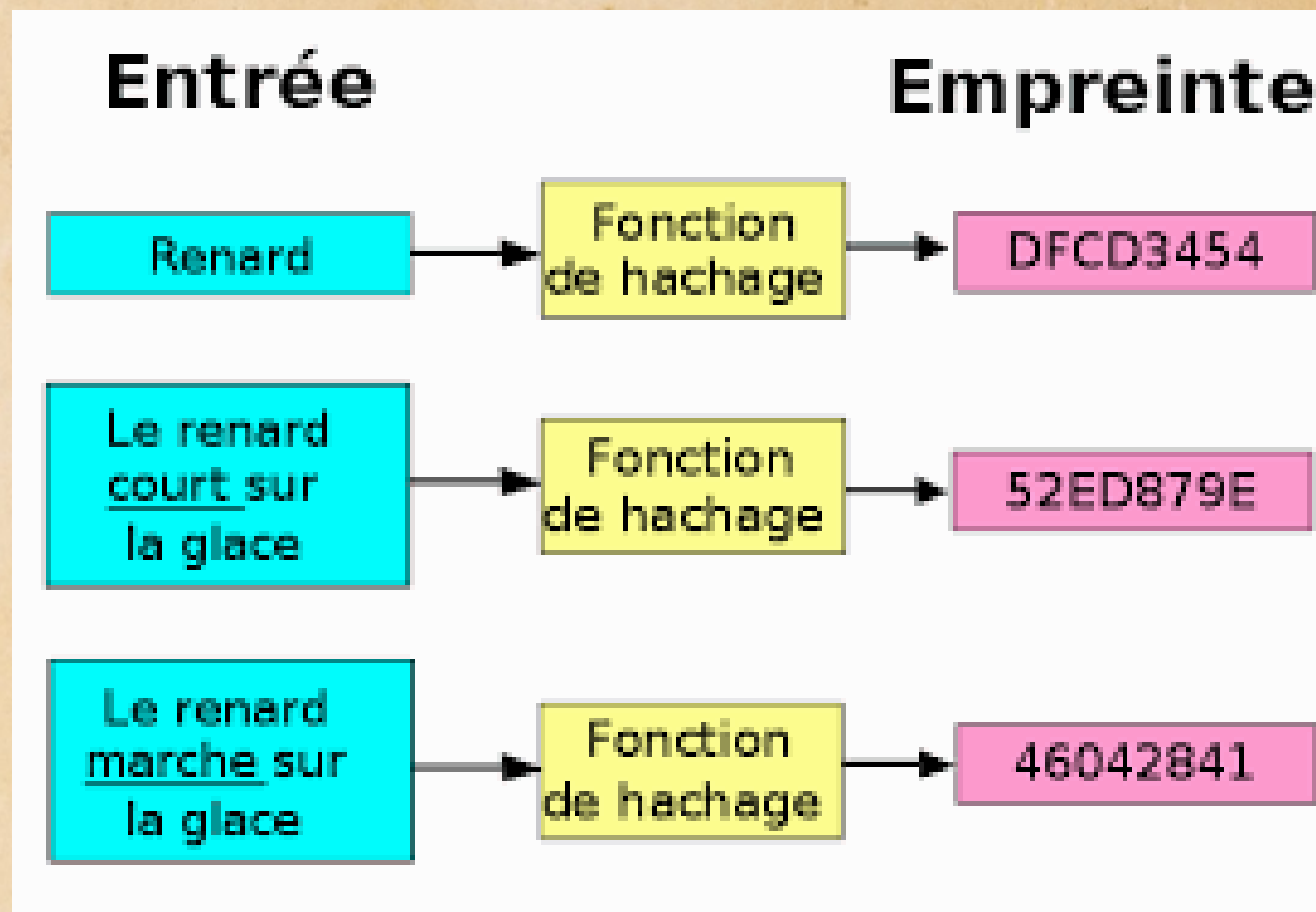
HTTP: No Encryption (no SSL)



HTTPS: Secure Cheap SSL Connection



Hachage, Salage



La protection navigateur

SOP

Same-Origin Policy



CORS

Cross-Origin Resource Sharing

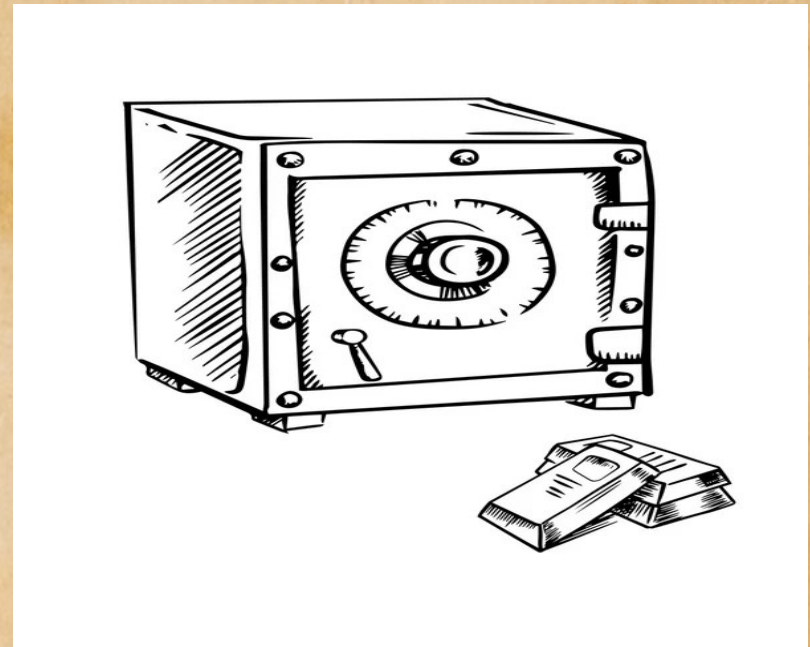


CSP

Content Security Policy



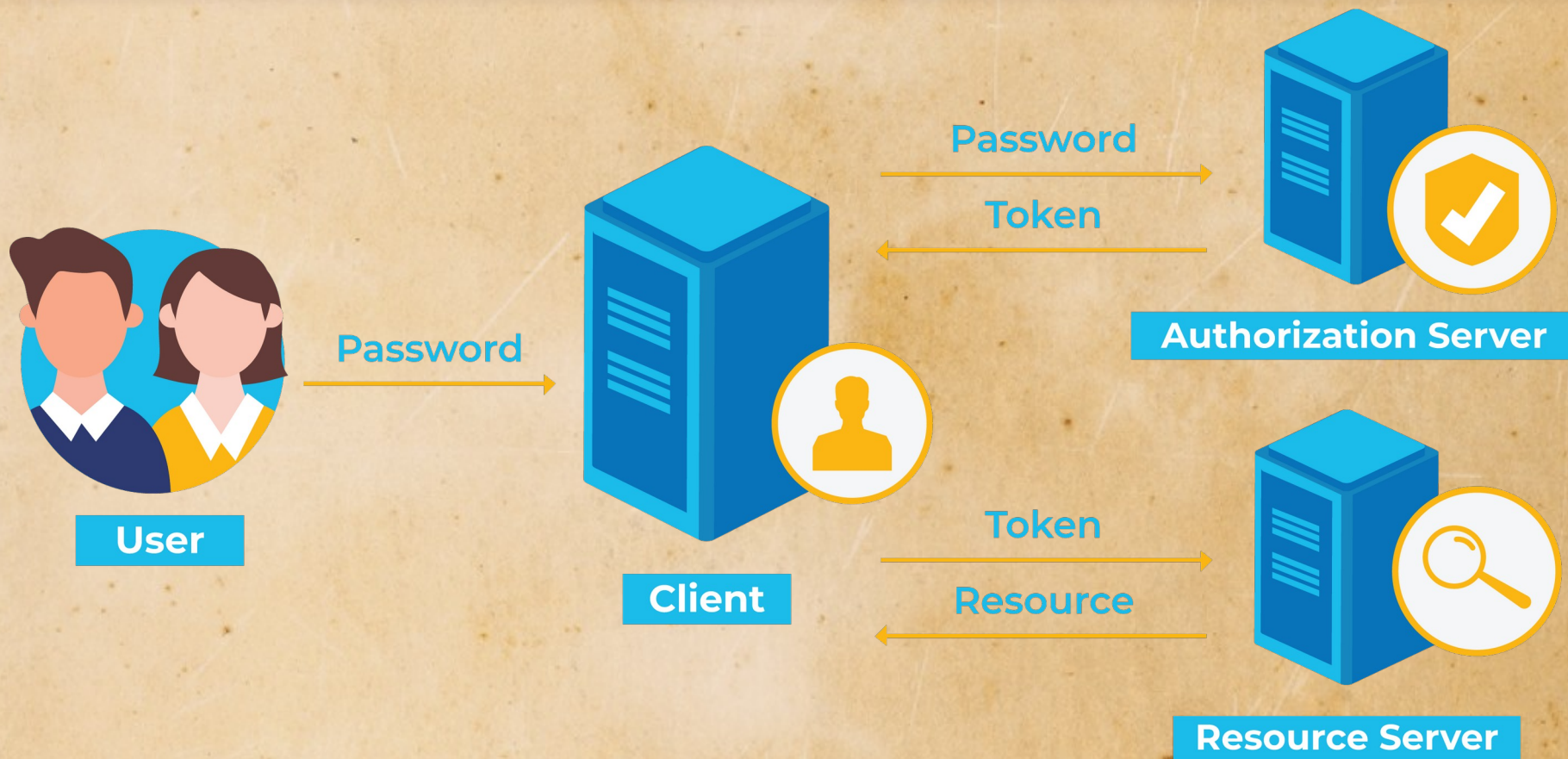
Politique de mots de passe



Sanitization



Tokens access



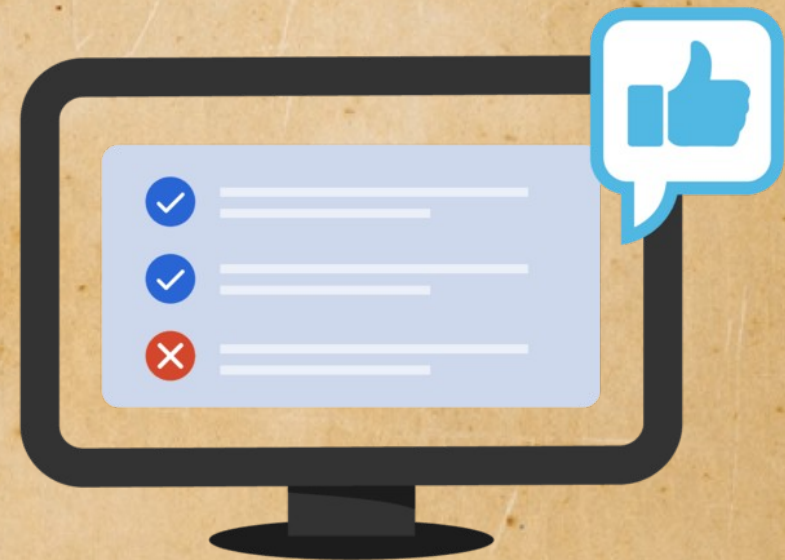
Sécurisation de l'authentification

Authentication



Confirms users
are who they say they are.

Authorization



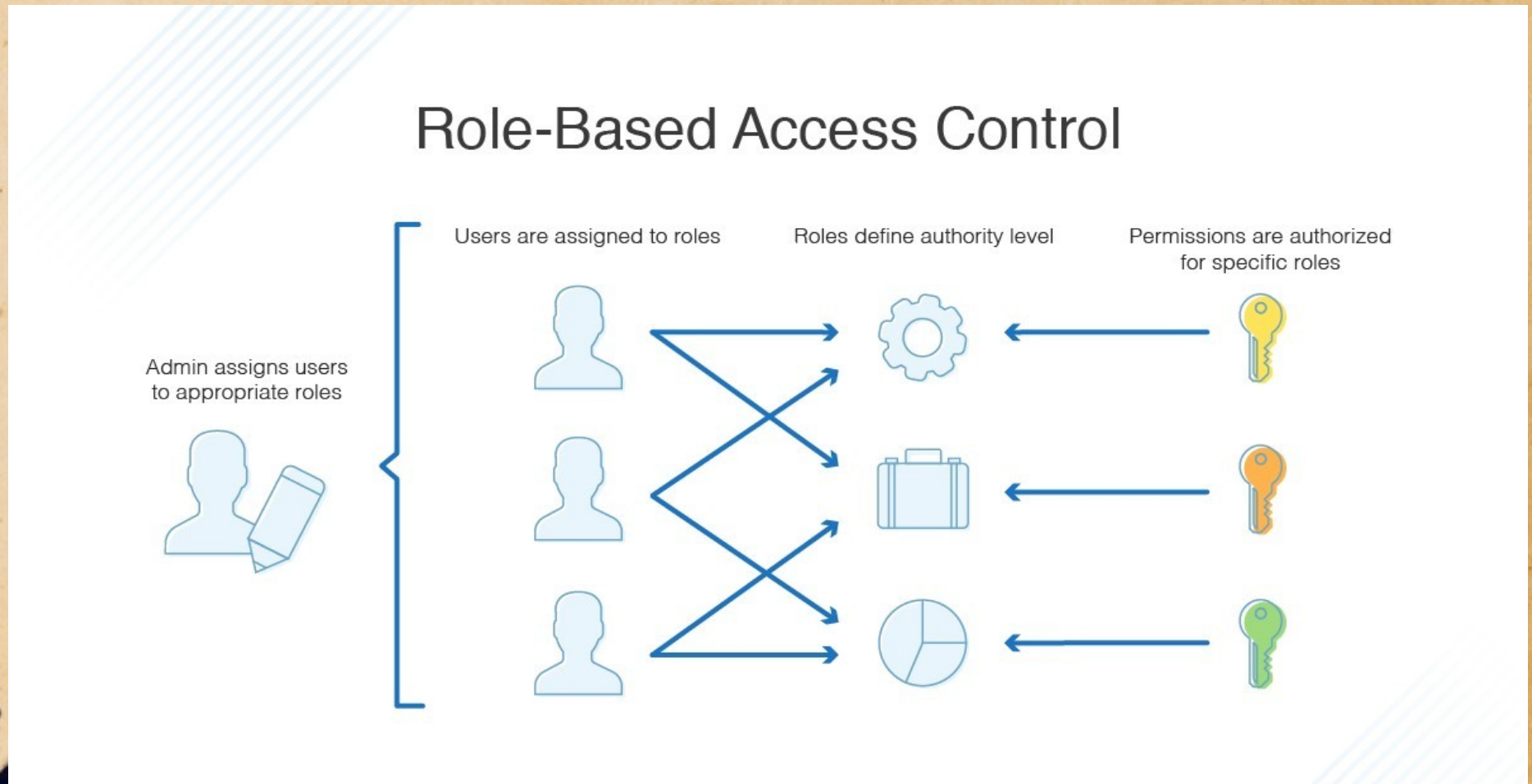
Gives users permission
to access a resource.

Session



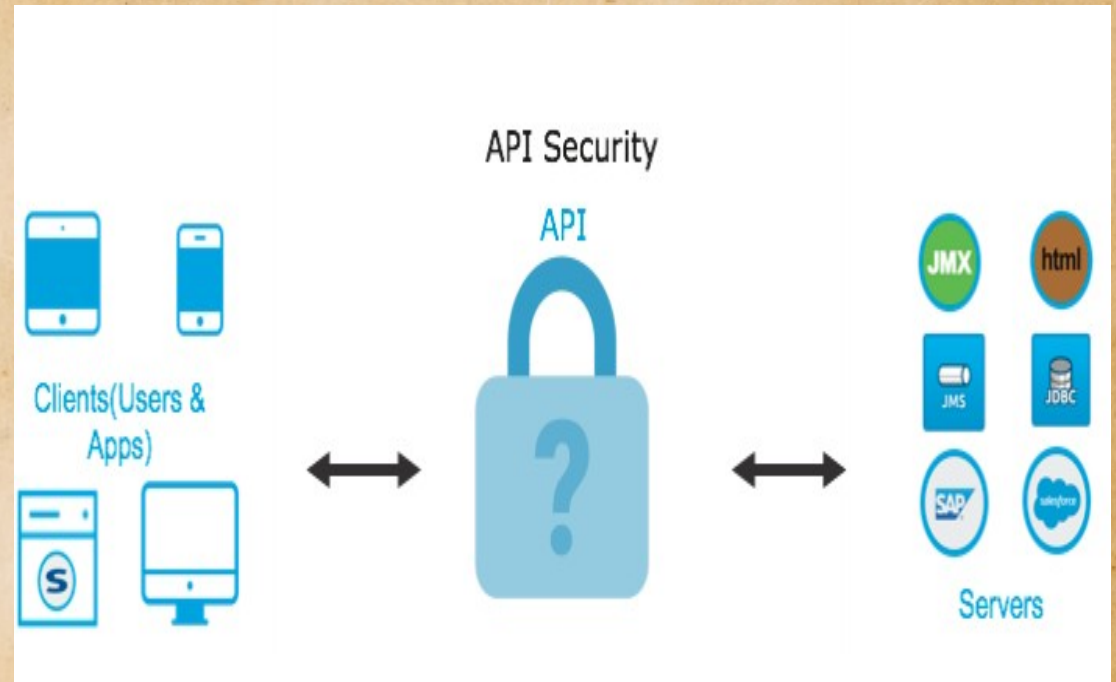
L'accès aux données

- R BAC (Role-Based Access Control)



La sécurisation API (Interface de programmation)

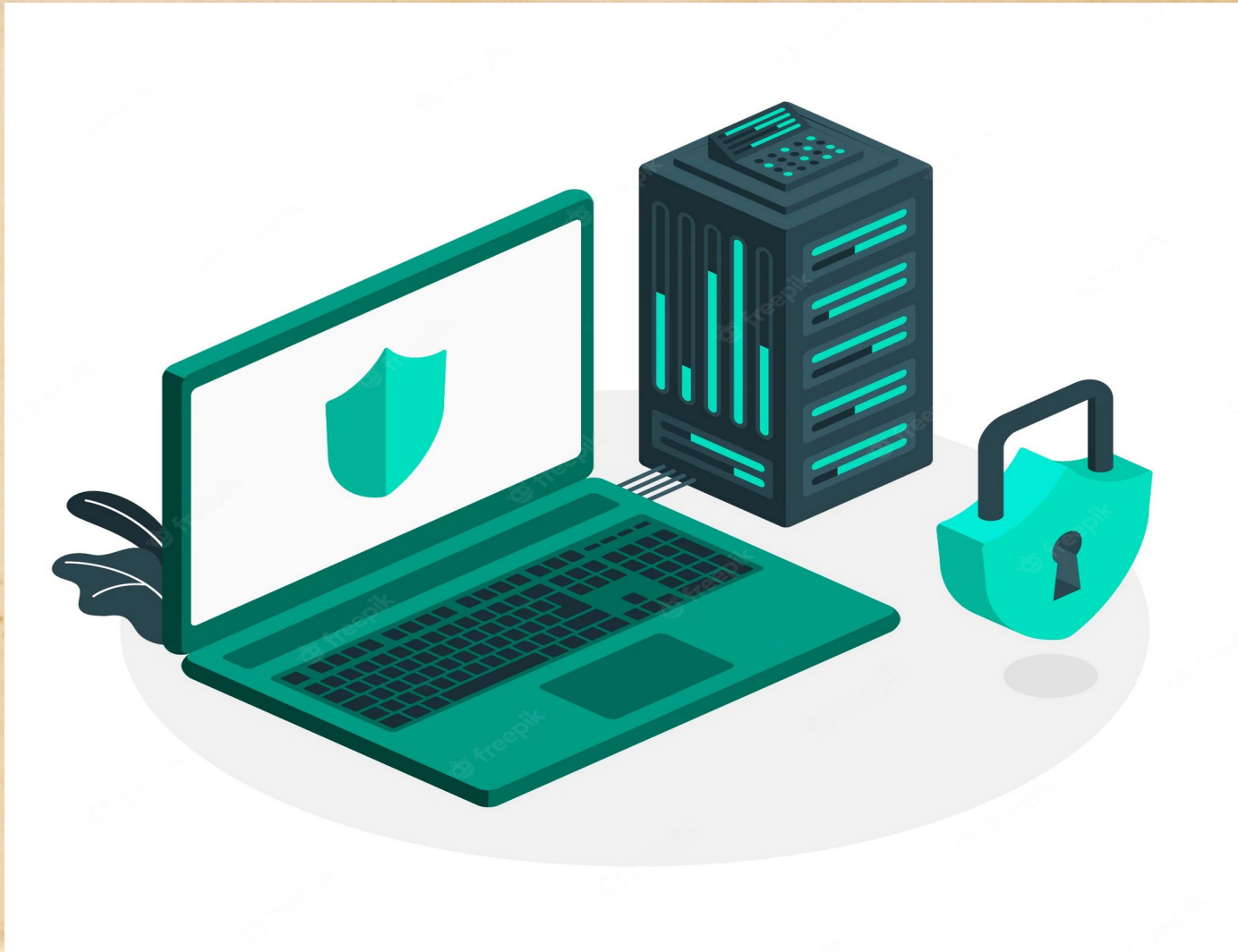
- API STATELESS



Journalisation



La stratégie de sauvegarde



Merci