REINVENT ENGINEERING

CENTRALE NANTES

Qualité, conception, modélisation

Accéder aux données - API

JY Martin

Novembre 2023

Les différents mécanismes d'accès

- Outils d'administration
- Outils de gestion : ETL, EAI, ESB, ...
- Outils de BI
- **API de connexion**

Plan

La problématique

- Les outils (ETL, BI, ...) ne suffisent pas toujours
- Ces outils, il faut bien les programmer...

=> nécessité de disposer d'un mécanisme, au niveau programmation, pour accéder aux bases de données et d'échange avec elles : API

La problématique

Au niveau de l'API, il faut :

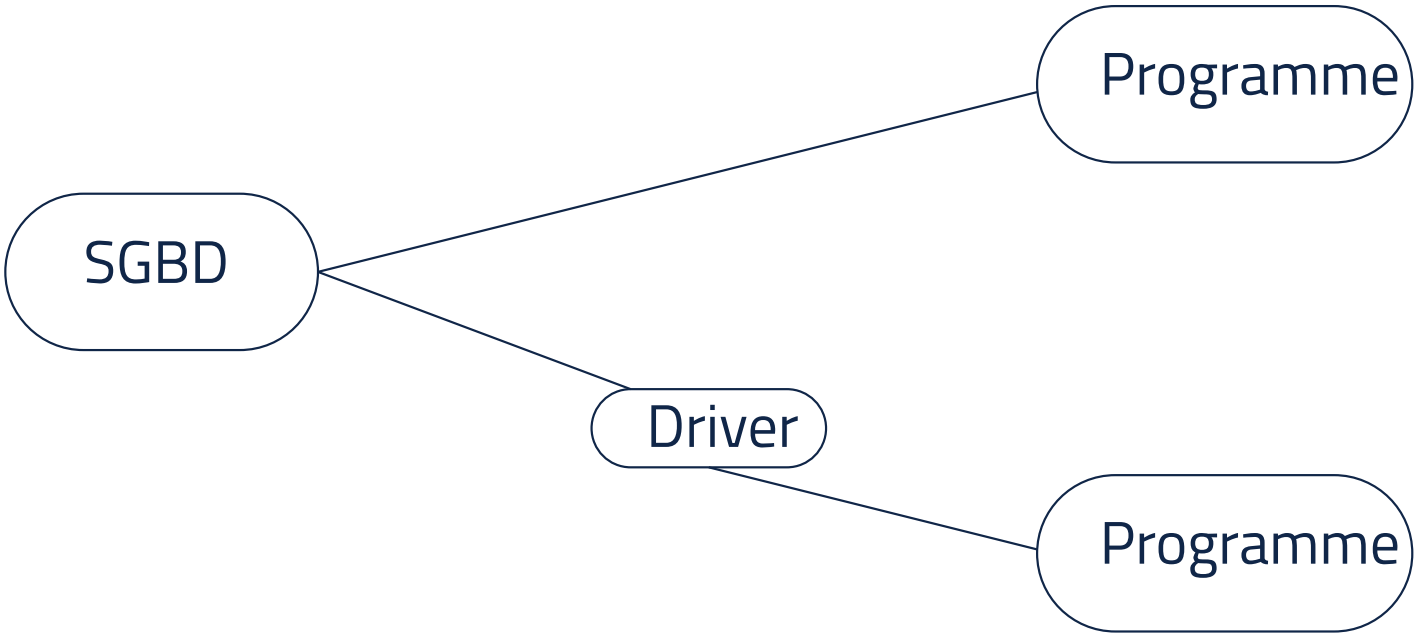
- se connecter au SGBD
- Accéder à la base de données (gestion des droits)
- Sélectionner des informations (gestion des droits)
- Modifier des informations (gestion des droits)

=> S'appuyer sur SQL

2 façon de travailler

- Utilisation directe de l'API
  Ensemble de fonctions et méthodes d'échange avec le SGBD
- Utilisation d'ORM
  Object Relational Mapping
  Utilisation d'un framework qui masque les accès au SGBD

2 façon de procéder



Quelques conseils

- Ne faites pas une connexion / déconnexion à chaque requête
- On ne peut pas faire une requête sur 2 bases de données différentes en même temps
- Par contre il est possible de se connecter en même temps à plusieurs bases de données

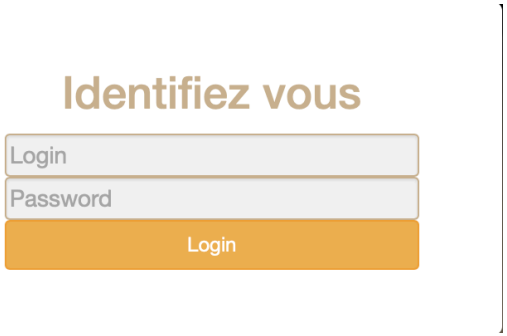
Plan

Principe d'une injection SQL

Injection SQL : utiliser une faille de sécurité lors des requêtes SQL. L'idée est de faire exécuter des requêtes SQL non prévues par l'application.

En général, ce type d'injection est obtenu lors de requêtes SQL mettant en jeu des chaînes de caractères. L'application a une faille de sécurité

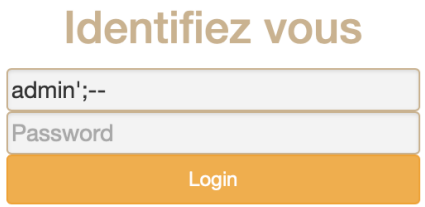
Exemple d'injection SQL



SELECT * FROM Personne WHERE Login='...' AND Password='...'

Exemple d'injection SQL

Imaginons qu'un utilisateur saisisse comme identifiant **admin';–** et que admin soit un identifiant valide mais on ne connait pas le mot de passe



Que devient la requête ?

```
SELECT * FROM Personne WHERE Login='admin';–' AND Password='..'
```

Cette requête fonctionne exactement comme si l'utilisateur avait saisi le bon mot de passe.

Injection SQL

D'où vient le problème ?

Dans la requête SQL, les apostrophes du login n'ont pas été doublées (on parle d'échappement de caractères).

Comment éviter ce type de problème ?

- Traiter les doublements d'apostrophes et espérer qu'il n'existe pas d'autre type de faille de sécurité
- Utiliser les fonctions spécifiques mises à disposition par les APIs et traitant toutes les failles connues.


Success
NEXT EXIT