

# Mécanismes d'authentification

Matthieu Nicolas

20/09/2016

## 1 Authentification locale

- Principe
- Architecture
- Avantages vs Inconvénients
- Bonnes pratiques
- Frameworks & Librairies

## 2 Authentification déléguée

- Principe
- Architecture
- Différences entre OpenID & OAuth
- OpenID Connect
- Avantages vs Inconvénients
- Frameworks & Librairies

## 1 Authentification locale

- Principe
- Architecture
- Avantages vs Inconvénients
- Bonnes pratiques
- Frameworks & Librairies

## 2 Authentification déléguée

- Principe
- Architecture
- Différences entre OpenID & OAuth
- OpenID Connect
- Avantages vs Inconvénients
- Frameworks & Librairies

# Authentification locale

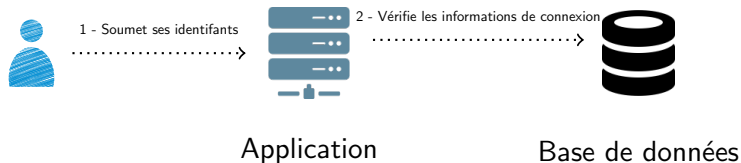
## Principe

- Architecture la plus *intuitive*
- Propose son propre système d'authentification
- Utilise généralement nom d'utilisateur/mot de passe
- Possible d'ajouter d'autres facteurs d'authentification
  - Code d'accès
  - Jeton d'authentification

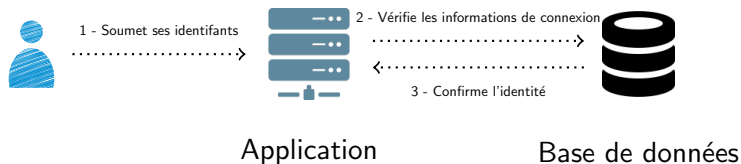
# Architecture



# Architecture



# Architecture



# Authentification locale

## Avantages vs Inconvénients

- Avantages
  - Solution relativement *simple*
  - Maîtrise de l'environnement
  - Confidentialité



# Authentification locale

## Avantages vs Inconvénients

- Avantages

- Solution relativement *simple*
- Maîtrise de l'environnement
- Confidentialité

- Inconvénients

- *Simple* de faire des erreurs
- Pas de distinction entre le service *métier* et le service *d'authentification*
- Un compte supplémentaire pour l'utilisateur...

# Authentification locale

## Bonnes pratiques

### HTTPS

- Chiffre les échanges entre le client et le serveur
- Prémunit des attaques *man-in-the-middle*

### Chiffrer le mot de passe

- Ne pas stocker le mot de passe en clair
- Utiliser des méthodes *lentes*
  - Bcrypt
  - PBKDF2

# Authentification locale

## Bonnes pratiques - suite

### Ajouter un sel

- Chaîne de caractères aléatoires
- Différente pour chaque utilisateur
- Protège de
  - Attaques par dictionnaire
  - Rainbow table

# Authentication locale

## Frameworks & Librairies

- Java/Scala

- Spring Security : <http://projects.spring.io/spring-security/>
- Silhouette : <http://silhouette.mohiva.com/>

- PHP

- Symfony FOSUserBundle :  
<http://symfony.com/doc/current/bundles/FOSUserBundle/index.html>

- NodeJS

- MEAN.JS : <http://meanjs.org/>

## 1 Authentification locale

- Principe
- Architecture
- Avantages vs Inconvénients
- Bonnes pratiques
- Frameworks & Librairies

## 2 Authentification déléguée

- Principe
- Architecture
- Différences entre OpenID & OAuth
- OpenID Connect
- Avantages vs Inconvénients
- Frameworks & Librairies

# Authentification déléguée

## Principe

- Sépare le rôle des différents composants :
  - **Fournisseur d'identité (Identity Provider)**
  - **Fournisseur de service (Service Provider)**
- Pour pouvoir utiliser le service, l'utilisateur doit fournir une preuve de son identité
- Un seul service peut être compatible avec plusieurs fournisseurs d'identité
- Un seul compte permet de s'authentifier auprès de plusieurs services (**Single Sign-On : SSO**)

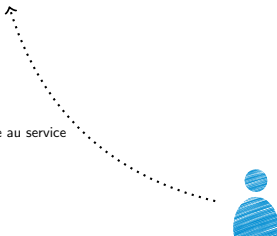
## Fournisseur de service



## Fournisseur d'identité



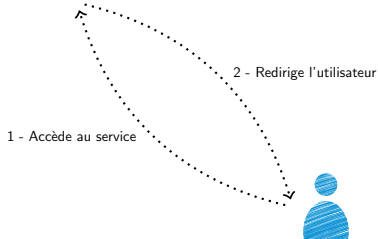
1 - Accède au service



## Fournisseur de service



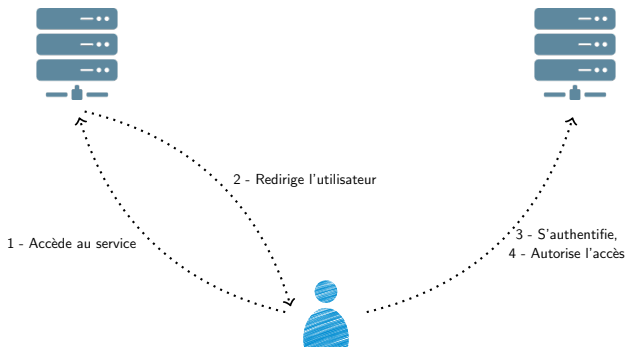
## Fournisseur d'identité





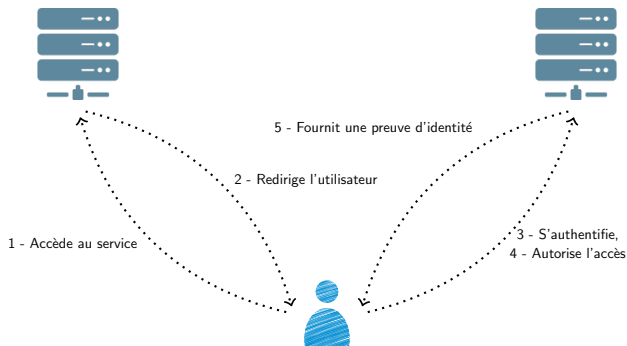
## Fournisseur de service

## Fournisseur d'identité



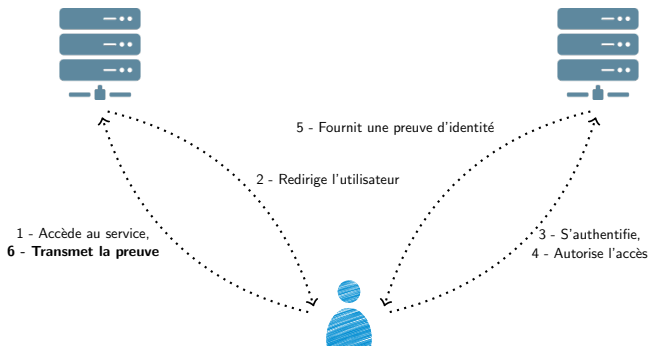
Fournisseur de service

Fournisseur d'identité



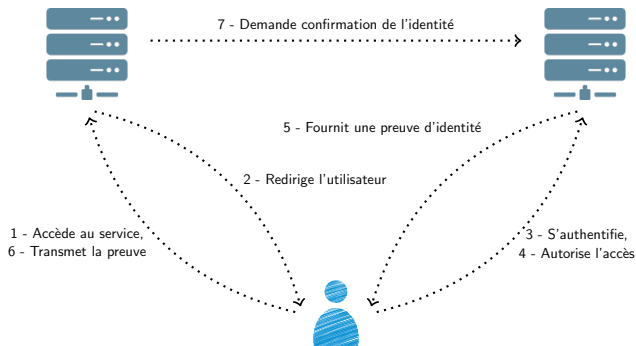
Fournisseur de service

Fournisseur d'identité



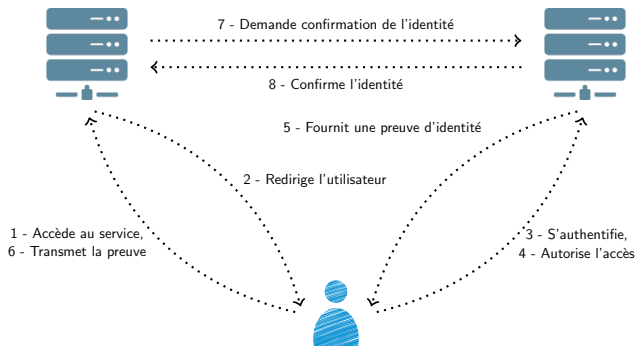
Fournisseur de service

Fournisseur d'identité



Fournisseur de service

Fournisseur d'identité



- Protocole d'*authentification*
  - Actuellement déprécié
  - Auparavant supporté par Google, Microsoft, Paypal...
- 
- Authentification avec un compte unique
  - Permet de récupérer les données du profil

- Protocole d'*autorisation*
- Actuellement en version 2.0
- Supporté par les géants du web
  - Google, Facebook, Slack...
- Ne parle plus d'*identités* mais de *ressources*
  - Propriétaire de ressources (Resource owner)
  - Serveur de ressources (Resource server)
  - Application cliente (Client)

# Protocole OAuth - explications

- Permet d'autoriser une application tierce à accéder à une ressource



- Permet d'autoriser une application tierce à accéder à une ressource

## OAuth avec Google

Permet de donner l'autorisation à un service de:

- Permet d'autoriser une application tierce à accéder à une ressource

## OAuth avec Google

Permet de donner l'autorisation à un service de:

- Récupérer les informations de son profil

- Permet d'autoriser une application tierce à accéder à une ressource

## OAuth avec Google

Permet de donner l'autorisation à un service de:

- Récupérer les informations de son profil
- Ajouter un évènement dans son calendrier

- Permet d'autoriser une application tierce à accéder à une ressource

## OAuth avec Google

Permet de donner l'autorisation à un service de:

- Récupérer les informations de son profil
- Ajouter un évènement dans son calendrier
- Envoyer des mails en son nom

- Permet d'autoriser une application tierce à accéder à une ressource

## OAuth avec Google

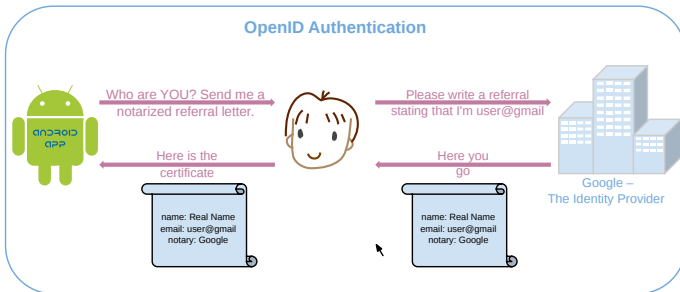
Permet de donner l'autorisation à un service de:

- Récupérer les informations de son profil
- Ajouter un évènement dans son calendrier
- Envoyer des mails en son nom

- Utilisé de façon détournée pour faire de la *pseudo-authentification*

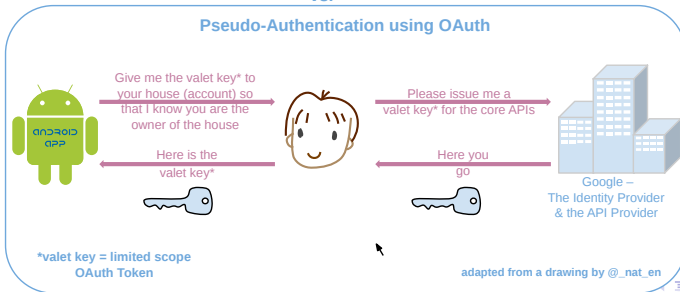
# Différences entre OpenID & OAuth

## OpenID Authentication



vs.

## Pseudo-Authentication using OAuth



- Nouvelle version du protocole OpenID
- Actuellement supporté par Google, Microsoft, Paypal...
- Basé sur OAuth 2.0
- Ajout d'une couche d'authentification

# Authentification déléguée

## Avantages vs Inconvénients

- Avantages
  - Interopérabilité
  - Scalabilité
  - Single Sign-On



# Authentification déléguée

## Avantages vs Inconvénients

- Avantages

- Interopérabilité
- Scalabilité
- Single Sign-On

- Inconvénients

- Plus *complexe* à bien implémenter
- Dépendant de services tiers
  - Disparition du fournisseur d'identité?
  - Fournisseur d'identité compromis?
- Repose sur l'utilisation d'un navigateur web

# Authentification déléguée

## Avantages vs Inconvénients

- Avantages

- Interopérabilité
- Scalabilité
- Single Sign-On

- Inconvénients

- Plus *complexe* à bien implémenter
- Dépendant de services tiers
  - Disparition du fournisseur d'identité?
  - Fournisseur d'identité compromis?
- Repose sur l'utilisation d'un navigateur web

- Mais rien n'empêche d'implémenter son propre fournisseur d'identité

# Authentification déléguée

## Frameworks & Bibliothèques

- Existe des bibliothèques pour implémenter :
  - L'application cliente
  - Le service d'authentification
- Bibliothèques pour OpenID Connect :
  - <http://openid.net/developers/libraries/>
- Bibliothèques pour OAuth 2.0 :
  - <https://oauth.net/code/>

Merci pour votre attention, avez-vous des questions?