Mécanismes d'authentification

Matthieu Nicolas

20/09/2016

Plan

Plan

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 [1]
  - Code d'accès
  - Jeton d'authentification

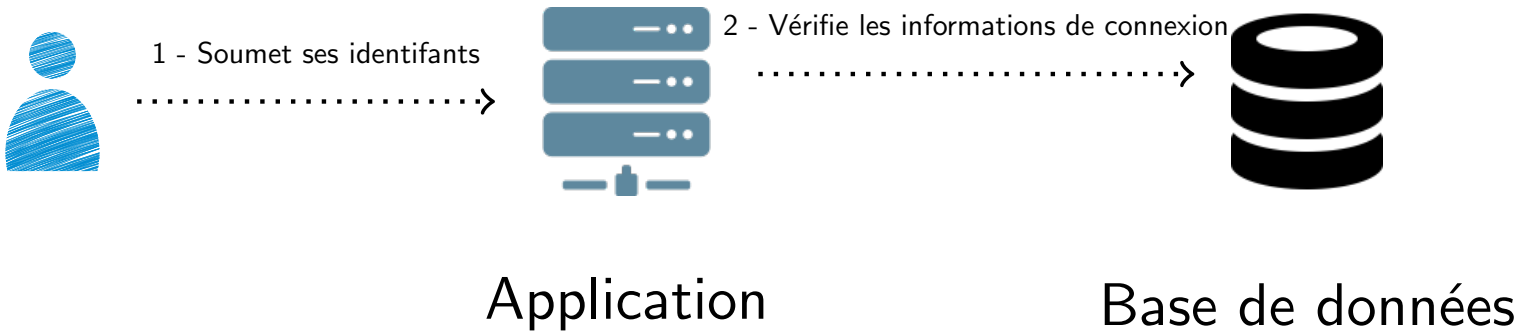
[1] *Different Ways to Authenticate Users with the Pros and Cons of each Method* : `https://pdfs.semanticscholar.org/3733/2607f7a7ac8284c514845957fd00583e5614.pdf`

Authentification locale

Architecture

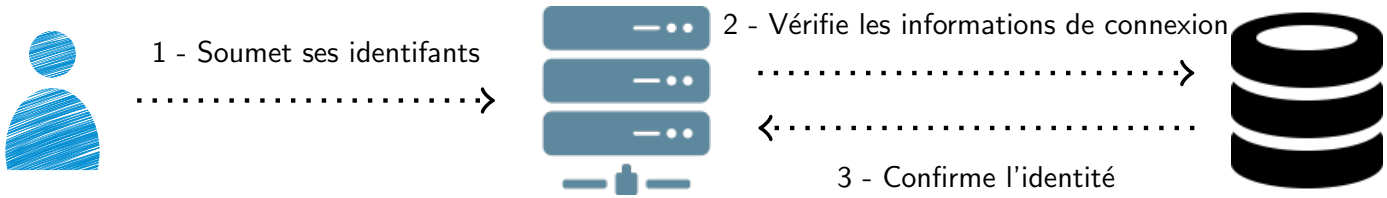
Authentification locale

Architecture



Authentification locale

Architecture



Application

Base de données

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 [2]

Ajouter un sel

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

[2] *Authentication Cheat Sheet* : `https://www.owasp.org/index.php/Authentication_Cheat_Sheet`

Authentification 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/`

Plan

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**) [3]

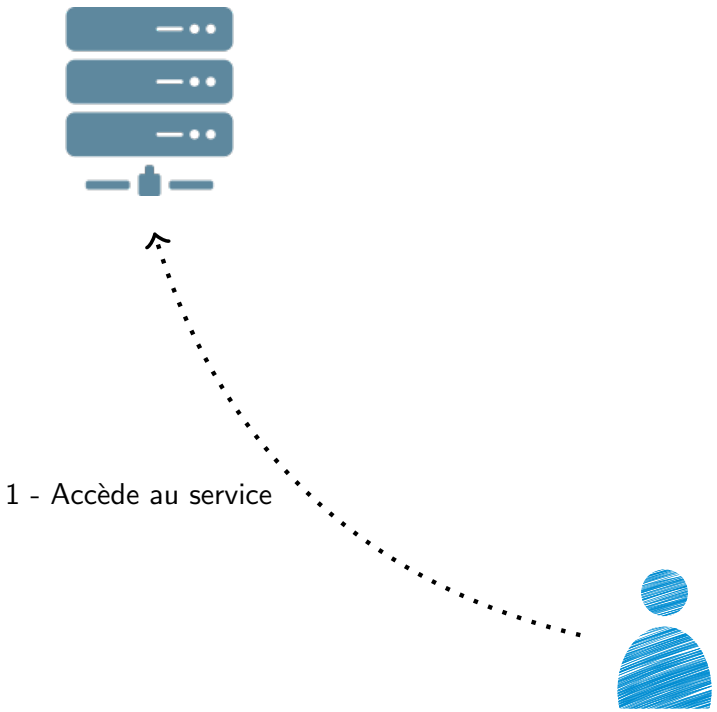
[3] *The Pros & Cons of Implementing Single Sign-On* : `https://www.neustar.biz/blog/what-is-single-sign-on-deployment-pros-cons`

Authentification déléguée

Architecture

Fournisseur de service

Fournisseur d'identité

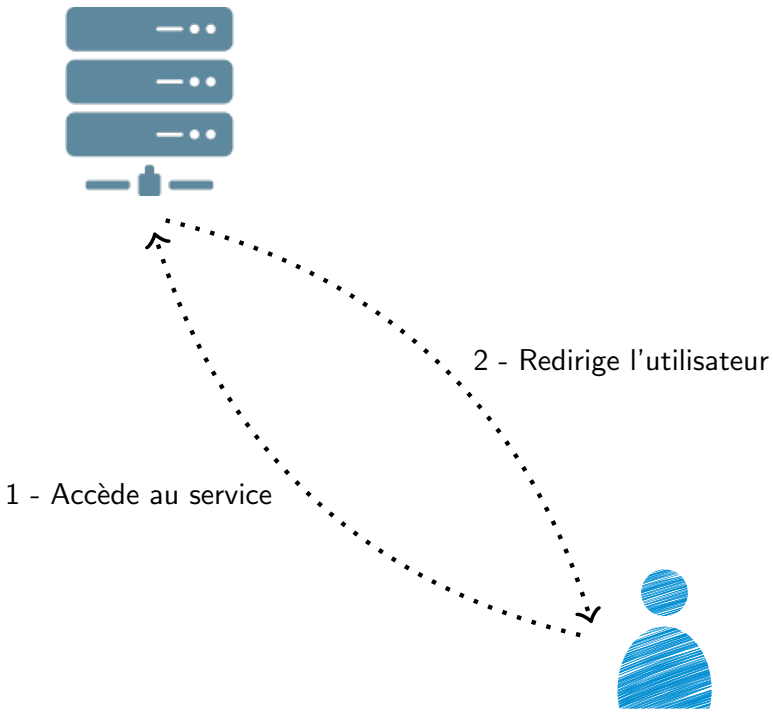


Authentification déléguée

Architecture

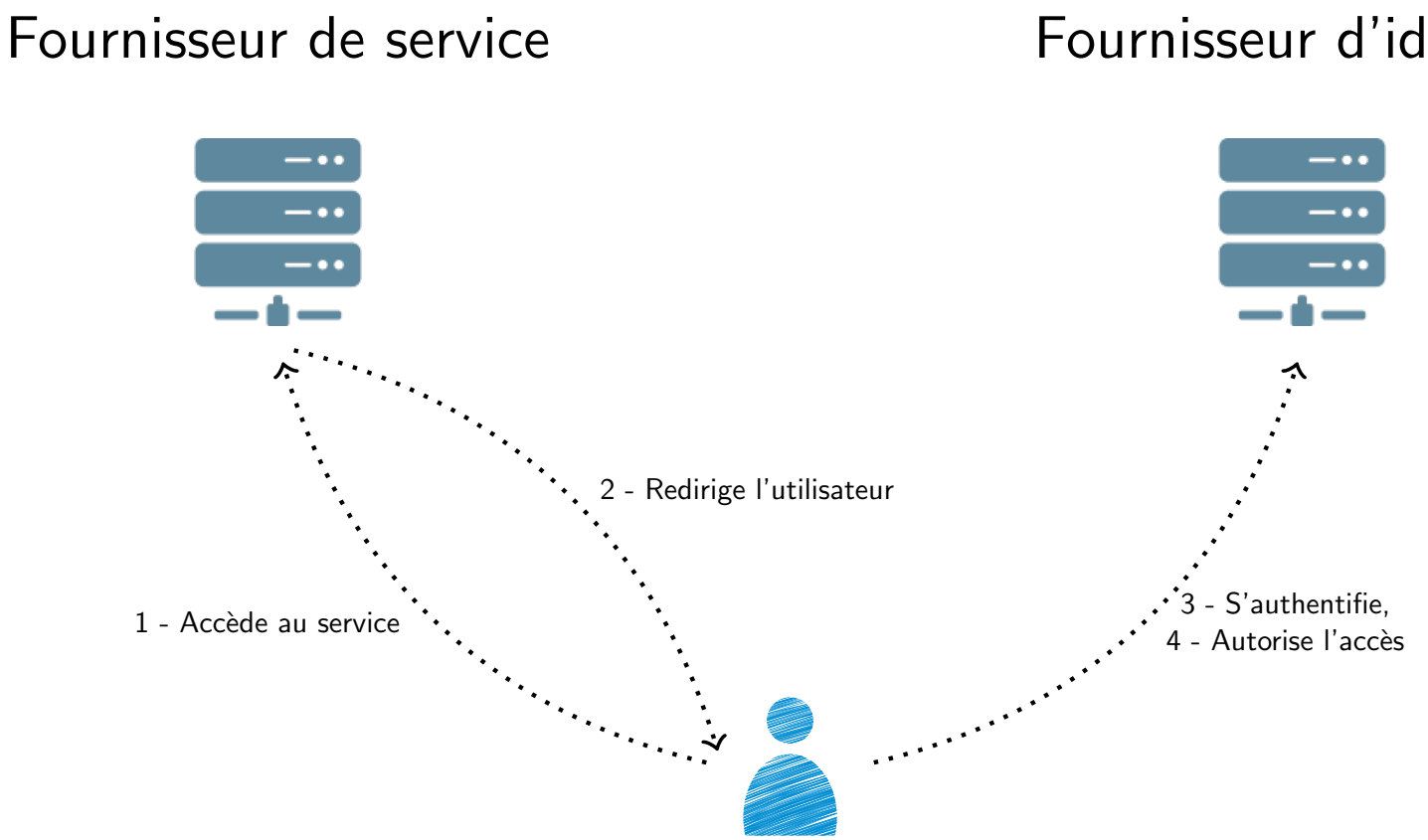
Fournisseur de service

Fournisseur d'identité



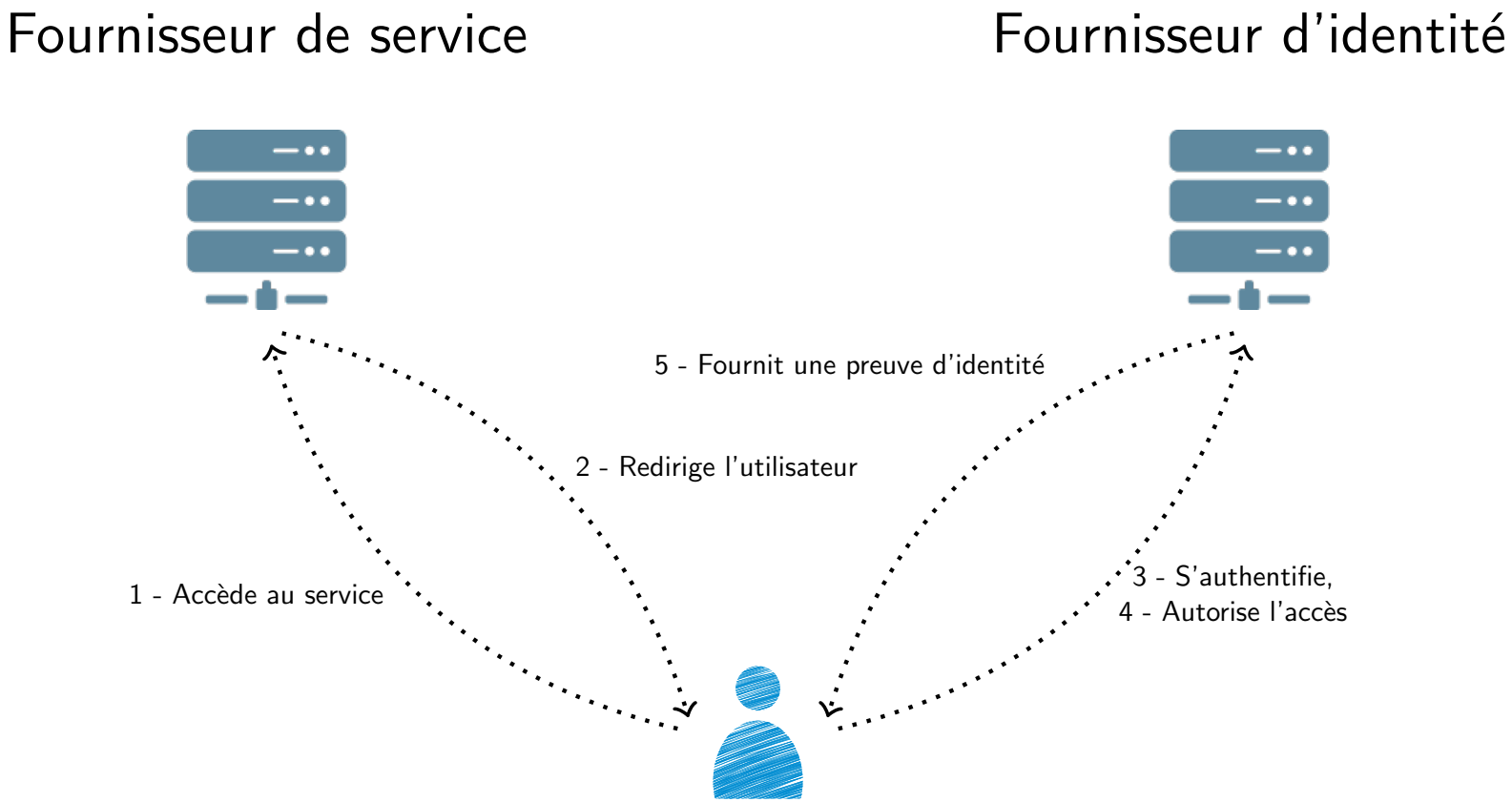
Authentification déléguée

Architecture



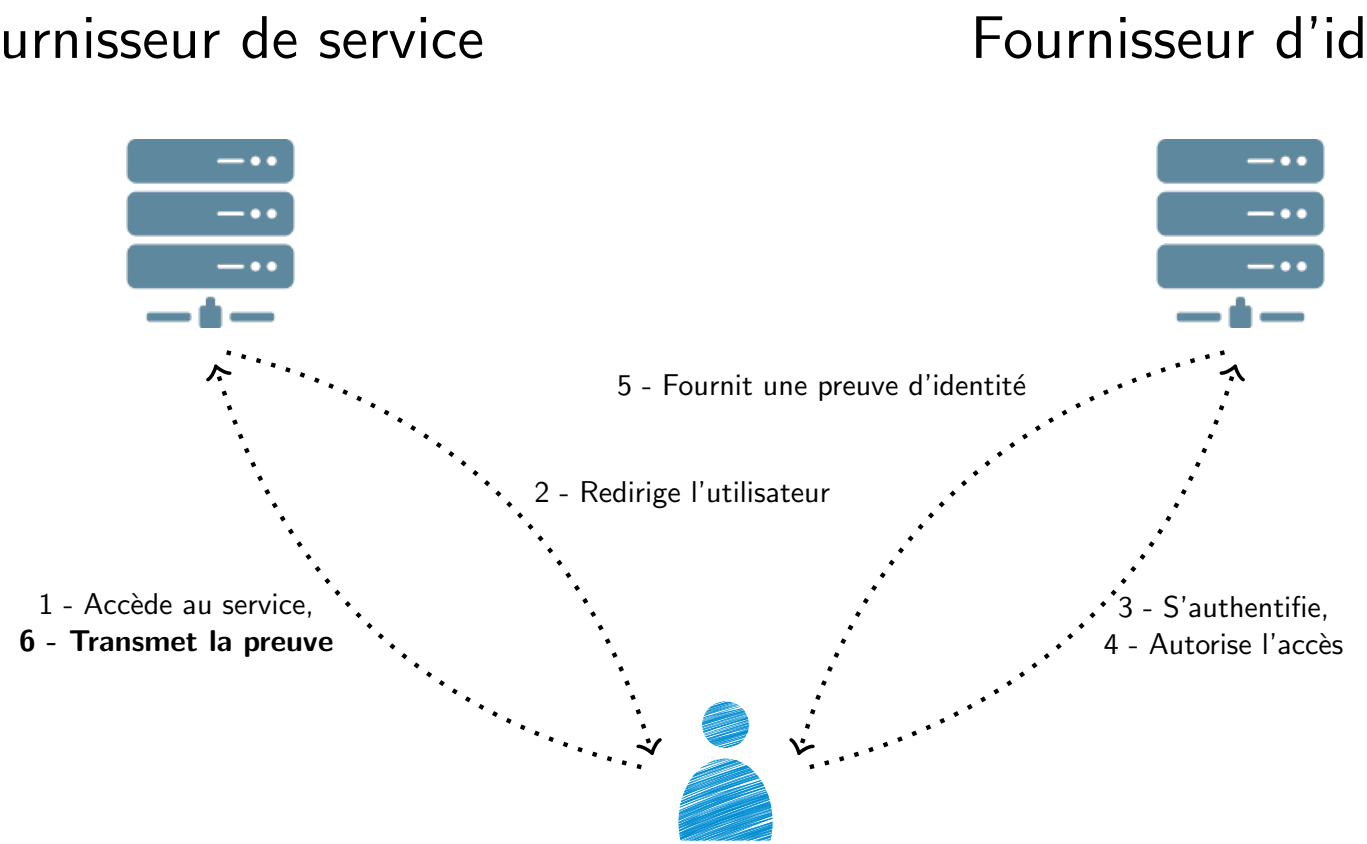
Authentification déléguée

Architecture



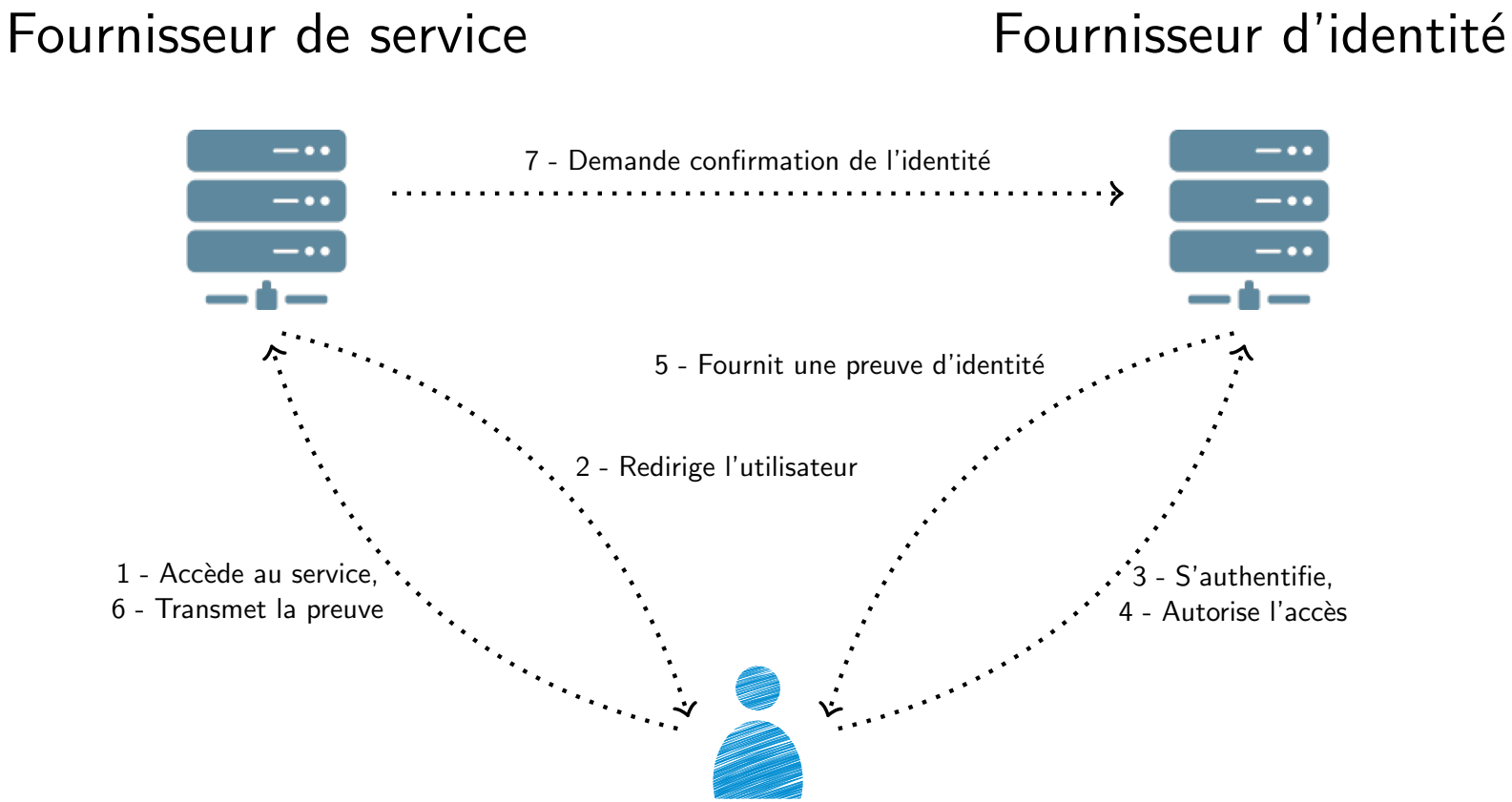
Authentification déléguée

Architecture



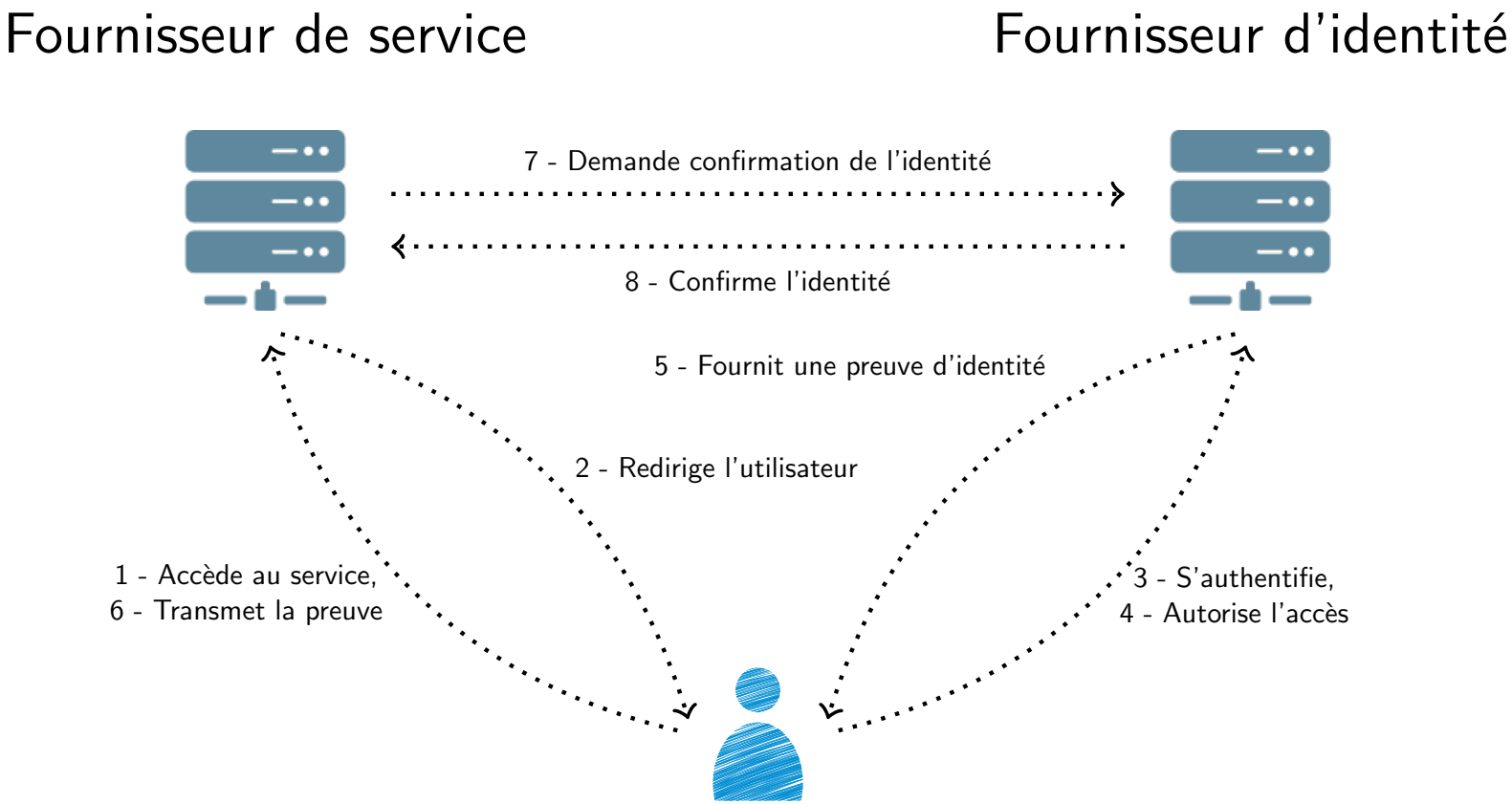
Authentification déléguée

Architecture



Authentification déléguée

Architecture



Authentification déléguée

Protocole OpenID

- Protocole d'*authentification* [4]
- Actuellement déprécié
- Auparavant supporté par Google, Microsoft, Paypal...

- Authentification avec un compte unique
- Permet de récupérer les données du profil

[4] *OpenID explaination* : `http://wiki.openid.net/w/page/12995171/Introduction`

Authentification déléguée

Protocole OAuth

- 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)

Authentification déléguée

Protocole OAuth - explications

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

Authentification déléguée

Protocole OAuth - explications

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

OAuth avec Google

Permet de donner l'autorisation à un service de:

Authentification déléguée

Protocole OAuth - explications

- 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

Authentification déléguée

Protocole OAuth - explications

- 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

Authentification déléguée

Protocole OAuth - explications

- 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

Authentification déléguée

Protocole OAuth - explications

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

OAuth avec Google

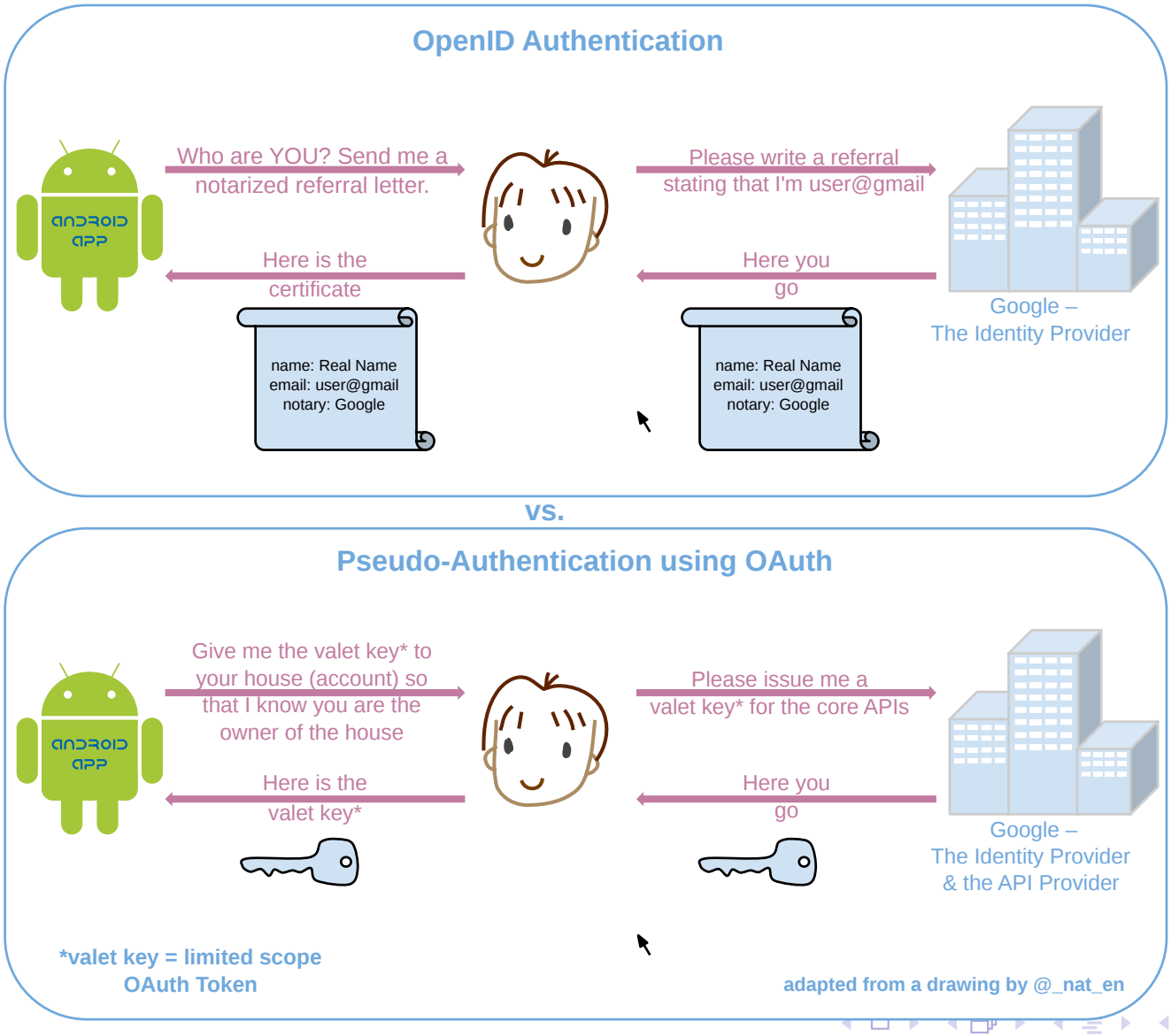
Permet de donner l'autorisation à un service de:

- 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*

Authentification déléguée

Différences entre OpenID & OAuth



Authentification déléguée

OpenID Connect

- Nouvelle version du protocole OpenID [5]
- Actuellement supporté par Google, Microsoft, Paypal...

- Basé sur OAuth 2.0
- Ajout d'une couche d'authentification

[5] *OpenID Connect in a nutshell* : `http://nat.sakimura.org/2012/01/20/openid-connect-nutshell/`

Authentification déléguée

Avantages vs Inconvénients

- Avantages
  - Interopérabilité
  - Scalabilité
  - Single Sign-On

[6] *OAuth 2 for Native Apps* :
`http://wiki.oauth.net/w/page/27249271/OAuthforNativeApps`

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 [6]

[6] *OAuth 2 for Native Apps* : `http://wiki.oauth.net/w/page/27249271/OAuthforNativeApps`

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 [6]
- Mais rien n'empêche d'implémenter son propre fournisseur d'identité

[6] *OAuth 2 for Native Apps* :
`http://wiki.oauth.net/w/page/27249271/OAuthforNativeApps`

Authentification déléguée

Frameworks & Librairies

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

Questions

Merci pour votre attention, avez-vous des questions?