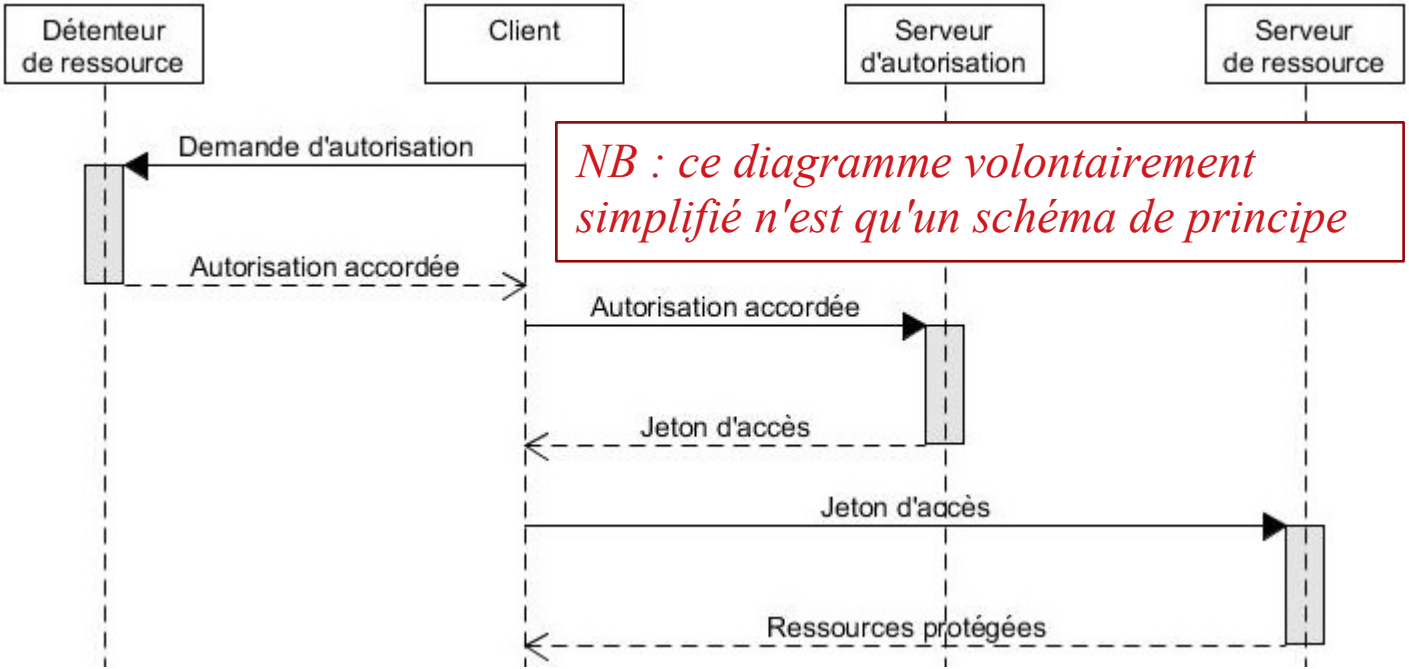
Norme/Protocole "OAuth2"

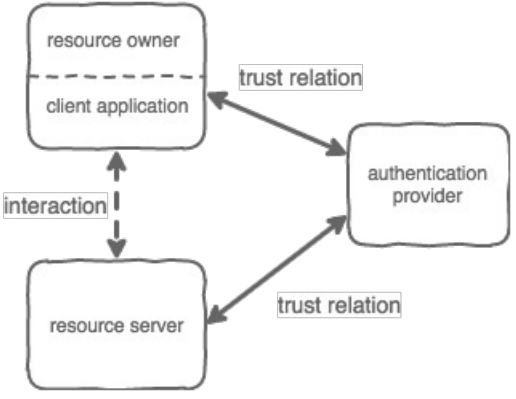
OAuth (**O**pen **Authorization**) existant en versions "1" et "2" , est une norme (RFC 6749 et 6750) qui correspond à un **protocole de "délégation d'autorisation" .** Ceci permet par exemple d'autoriser une application cliente à accéder à une API d'une autre application (ex : FaceBook , Twitter , …) de façon à accéder à des données protégées.



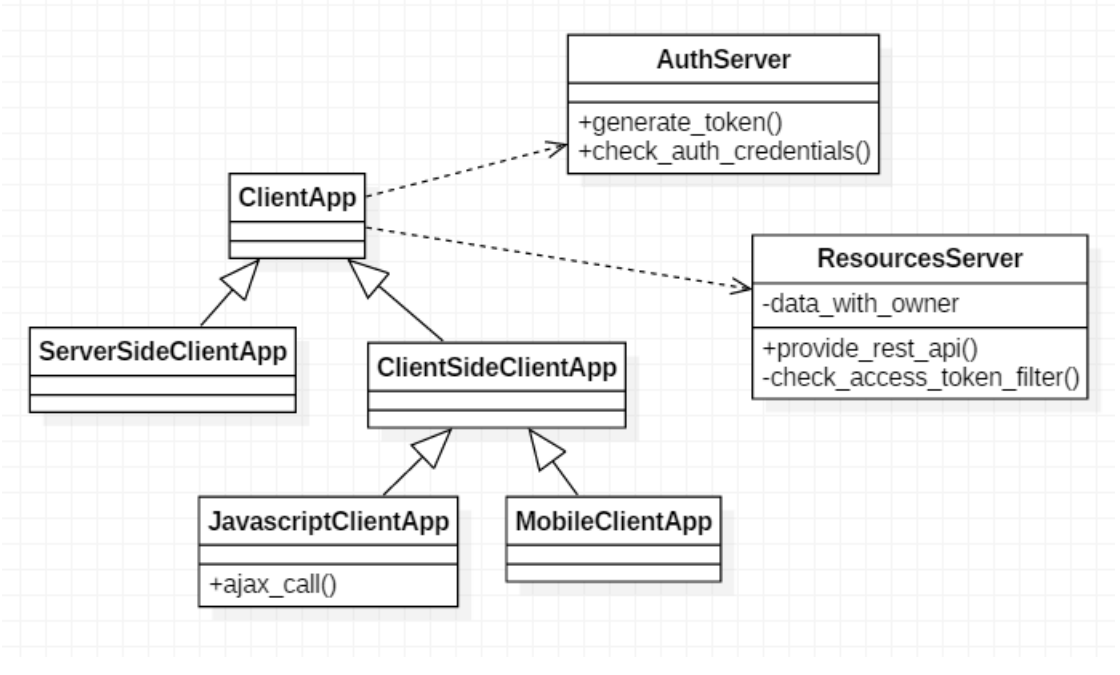
Attention, **OAuth** doit être accompagné de ***HTTPS*** pour être un minimum sécurisé et les autorisations doivent dans la plupart des cas être associées à une authentification basée sur un compte utilisateur pour que le jeton d'accès construit puisse véhiculer une information précise du type :

"*L'utilisateur Uxyz authentifié par OrgXy est via ce jeton d'accès autorisé à accéder aux ressources accessibles via certaine(s) Api(s) "*

Un jeton d'accès (souvent au format JWT) construit par OAuth2 aura un délai d'expiration court ou moyen . Dans certains cas , un autre "*refresh token*" permettra d'obtenir un nouveau jeton d'accès .



third party authentication



Les 4 modes d'autorisation (variantes) de OAuth2

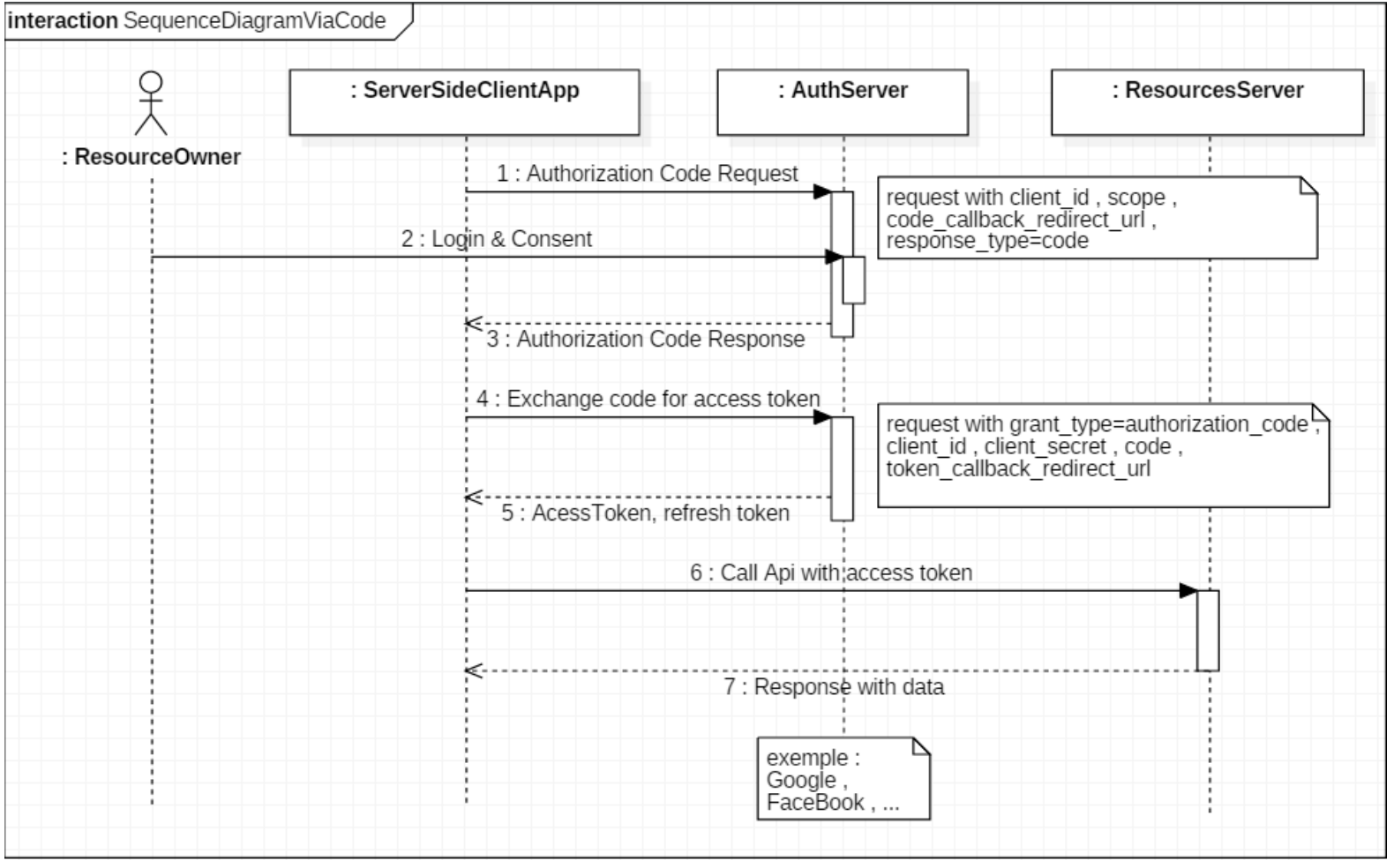
* via un **code** temporaire à échanger contre un jeton d'accès : l'application cliente est un site web (avec une technologie serveur fiable) qui peut gérer des redirections d'URLs .

* **implicite** (jeton direct) : l'application cliente est simple et exposée (ex : javascript , mobile) *[attention : ce mode doit etre accompagné de précautions pour une bonne sécurité]*

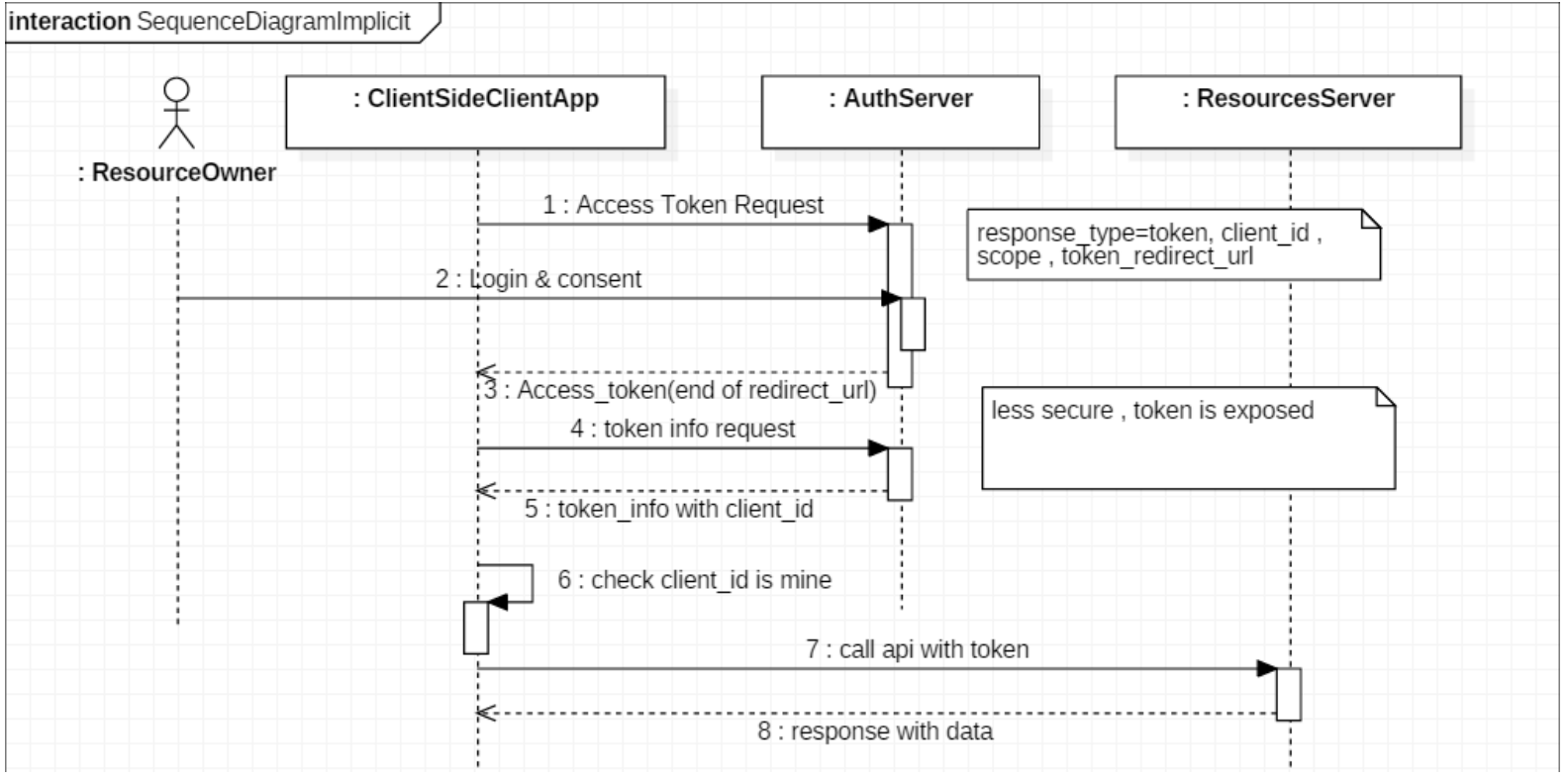
* via **mot de passe** (retransmis) : seulement applicable si l'application cliente et le serveur d'autorisation sont gérés par la même organisation (entreprise ou ...) .

* ***client App credential*** : au lieu d'authentifier un utilisateur précis , c'est appli "client" qui demande une autorisation d'accès auprès d'une autre .

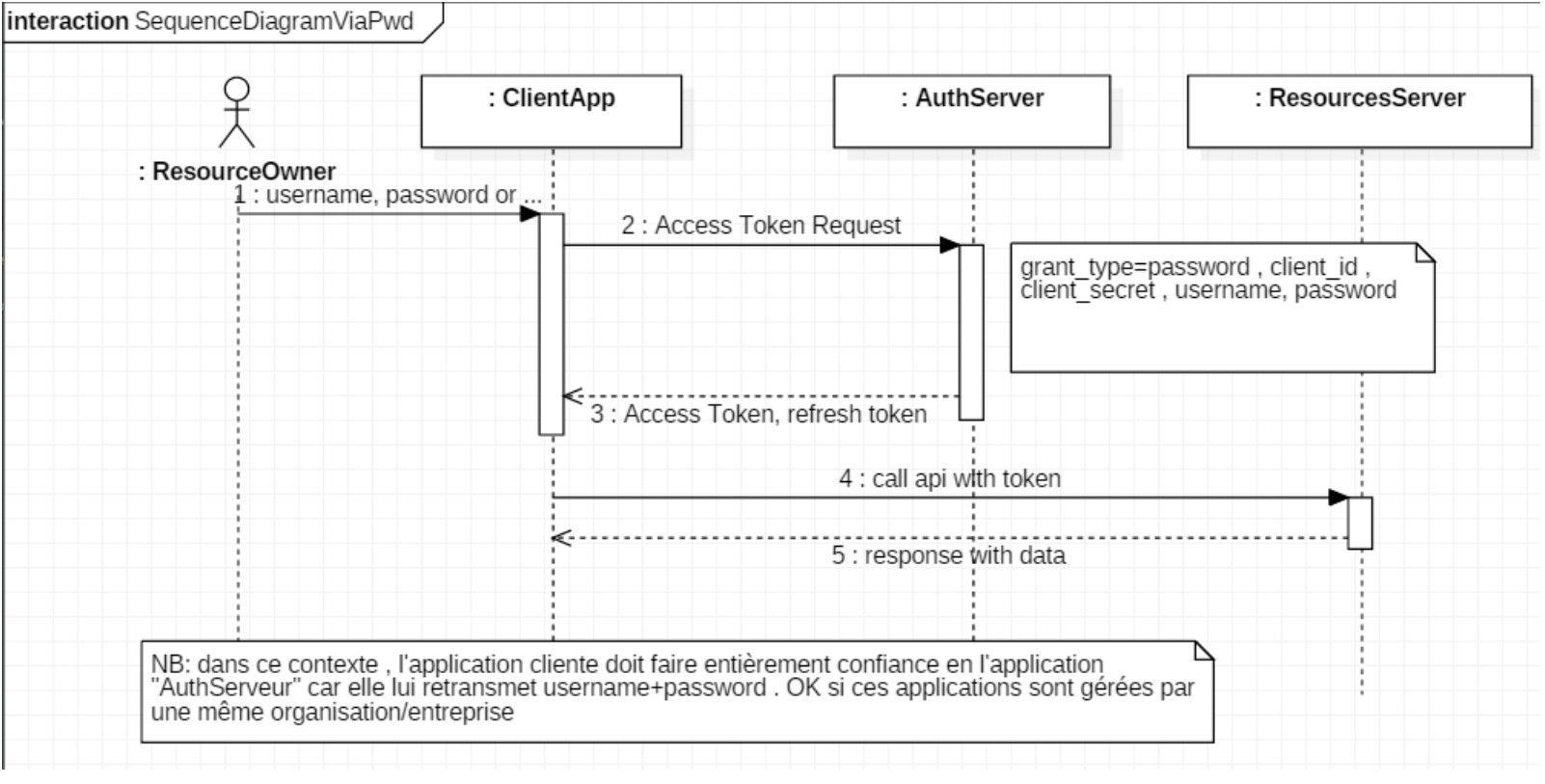
Autorisation OAuth2 par code (inter-organisations)



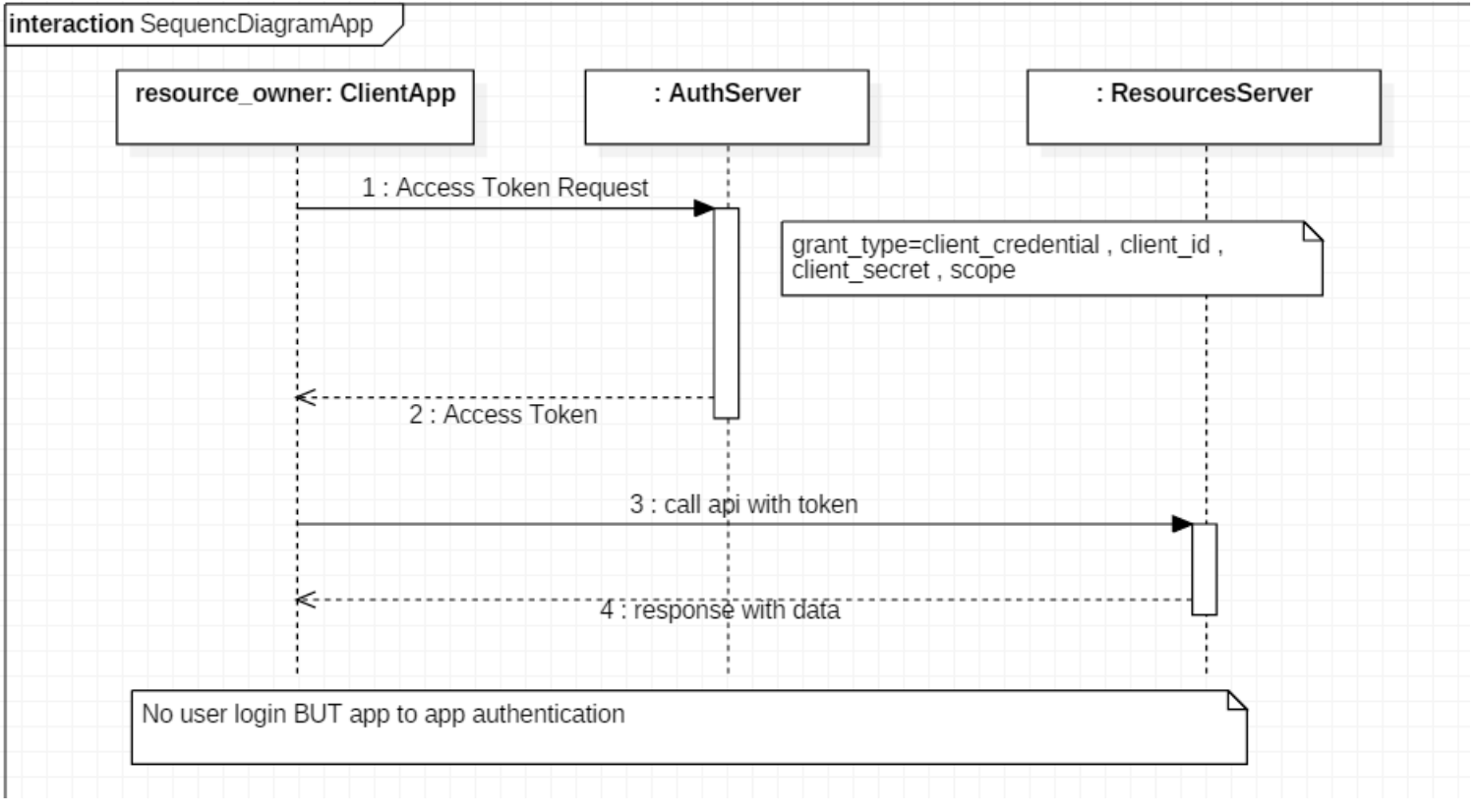
Autorisation OAuth2 implicite (jeton direct)



Autorisation OAuth2 par mot de passe (intra)



Autorisation OAuth2 pour application utilisateur

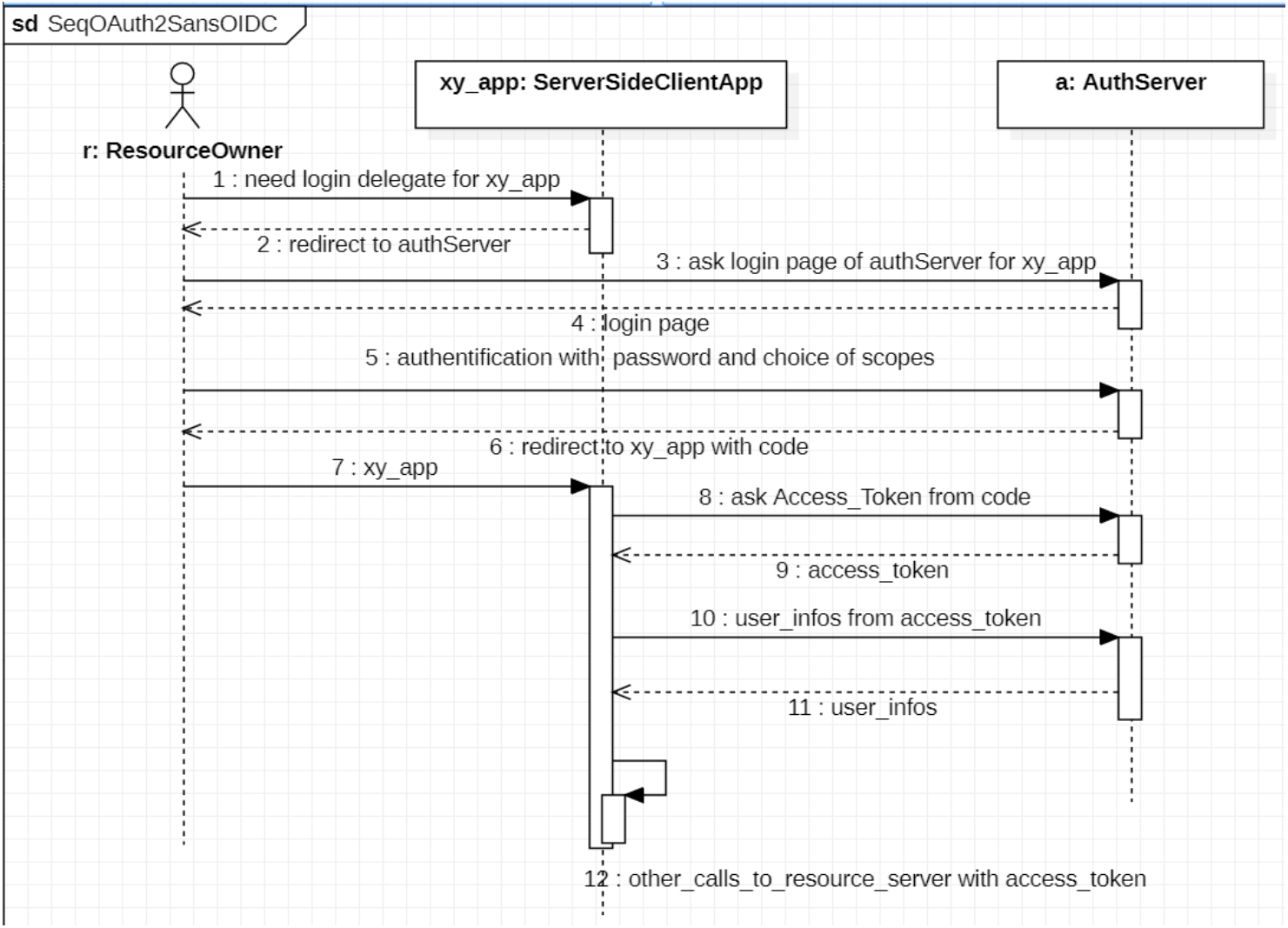


OpenId Connect

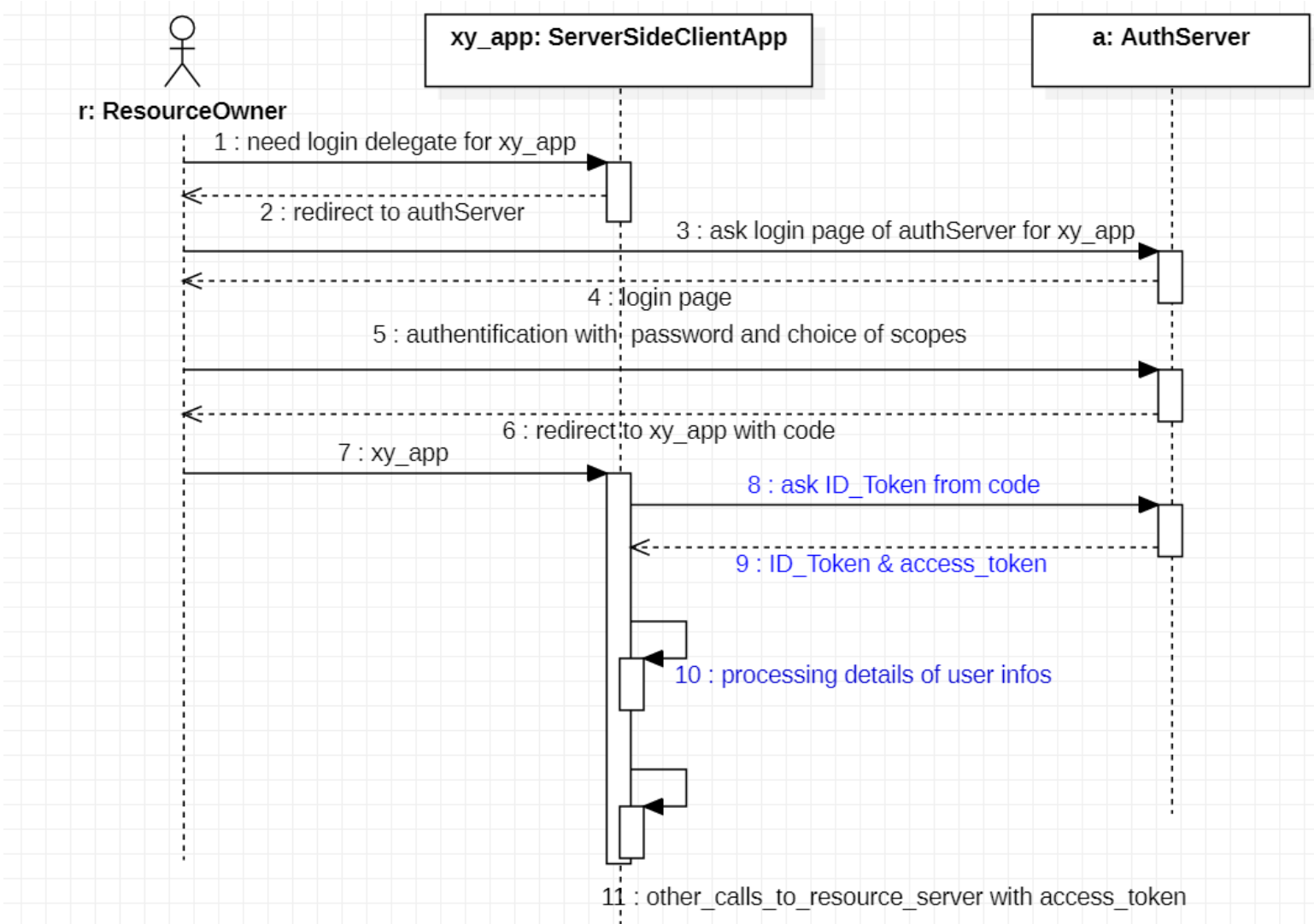
OpenID Connect (OIDC) est un protocole d'authentification forte basé sur une transmission standardisée ("***ID Token***" au format JWT) des informations sur l'utilisateur identifié .

OpenID Connect s'appuie en interne sur OAuth2 et permet d'**obtenir** simplement **des informations plus précises sur l'utilisateur à authentifier** .

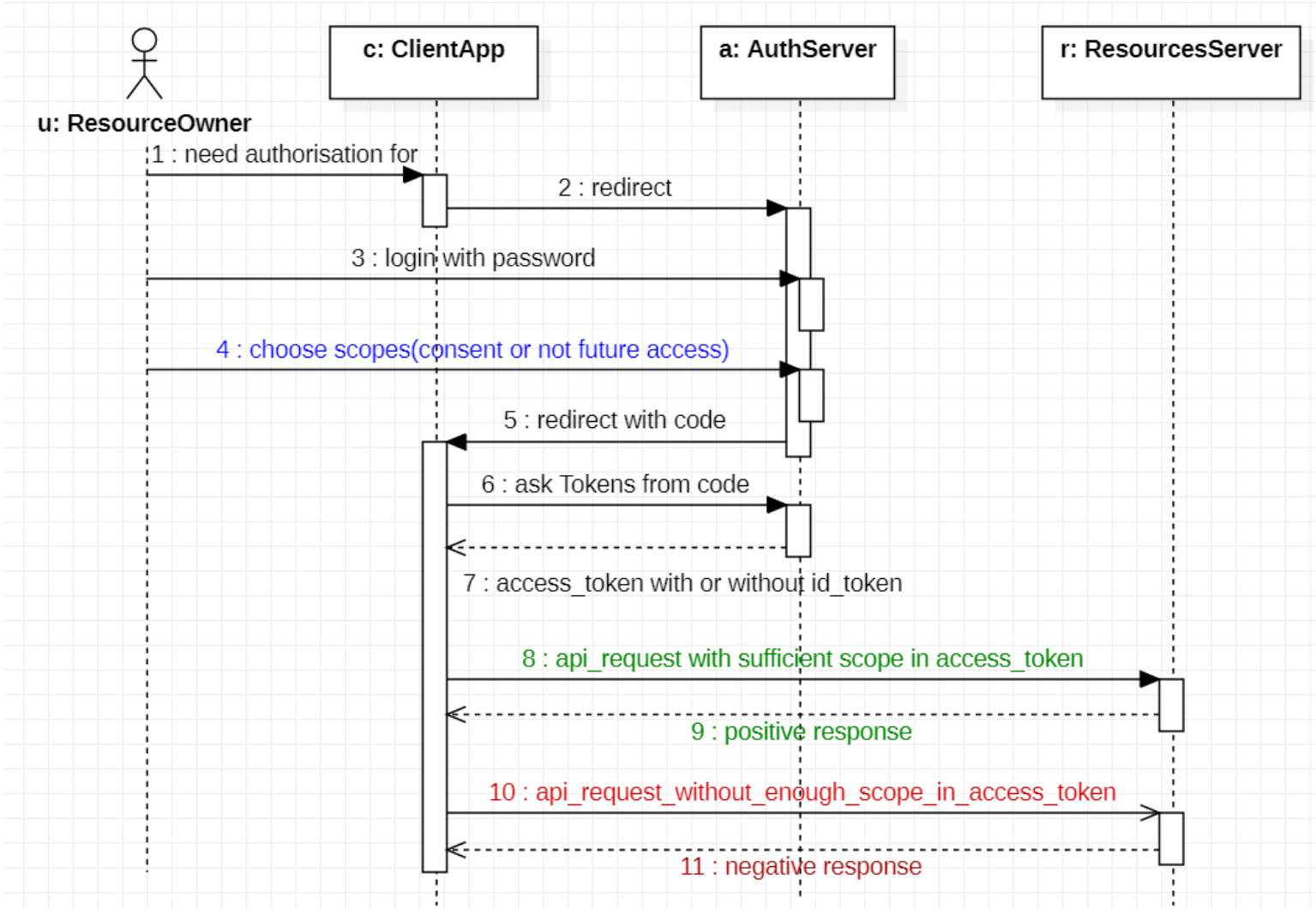
OAuth2 sans openID



avec OpenID Connect



OAuth2 or OIDC with **Scopes**



Quelques serveurs d'authentification OAuh2/OIDC

- Google **Hydra** (très perfectionné et très performant mais complexe car besoin de compléments à personnaliser)

- **keycloak** est un serveur d'autorisation oauth2/oidc basé sur jboss wildfly et java >=8 (relativement simple à installer). Il est par défaut basé sur une base H2 et dispose d'une ihm intégrée pour configurer des utilisateurs

- **okta** est une entreprise spécialisée dans l'identité numérique et offre des services de type "*authorisation oauth2/oidc as as service*"

- Les grandes plate-formes "cloud" (***Azure*** , ***AWS***, …) intégrent un service OAuth2/OIDC (ex : ***Azure Active Directory***, ***Amazon-Cognito*** , …)

- quelques autres