

# **OAuth**

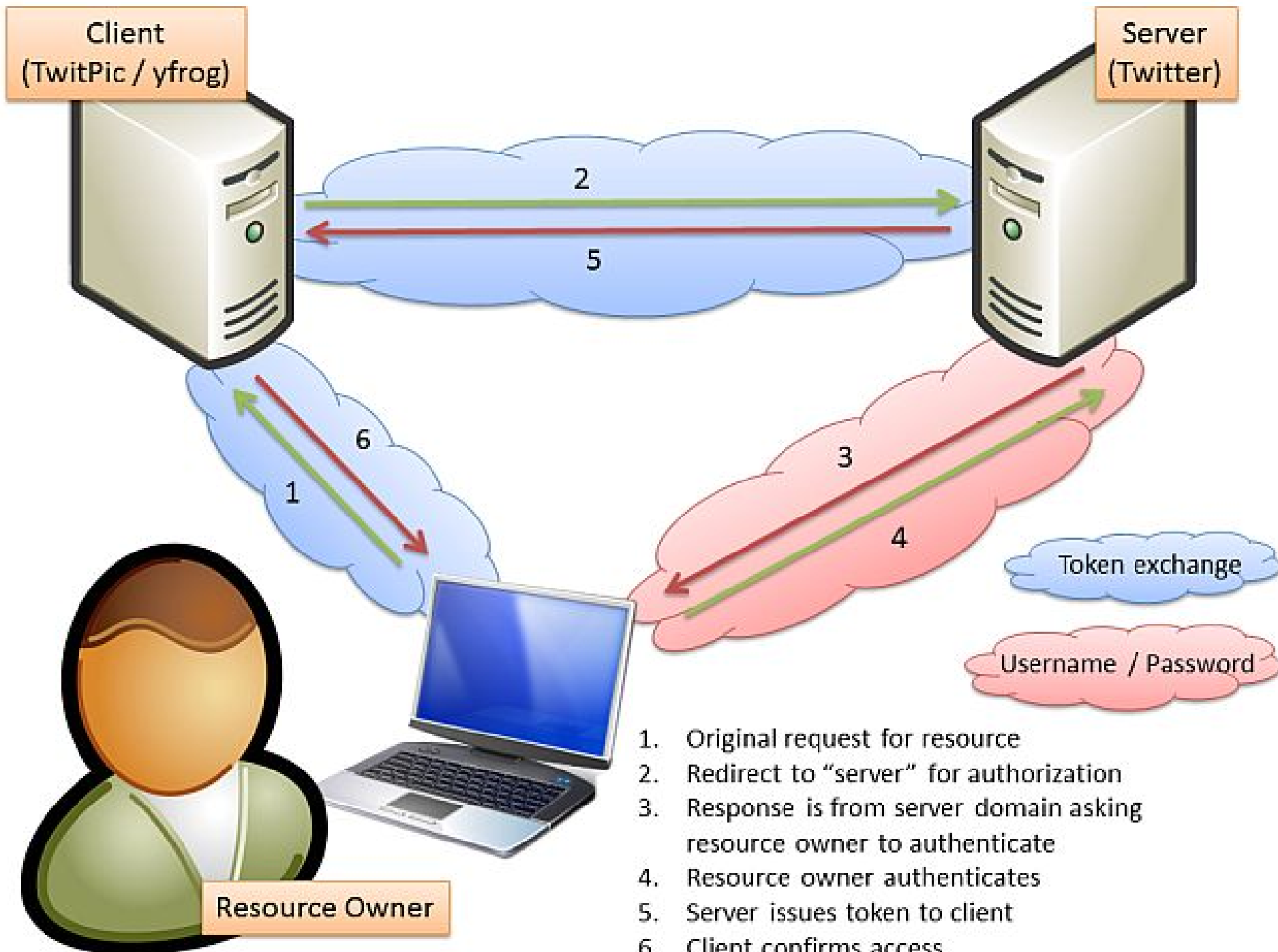
Martin Kuba, ÚVT MU

# Co je OAuth

- otevřený standard specifikující protokol pro **autorizaci** přístupu k vyjmenovaným operacím nějakého API, ale lze jej využít i pro autentizaci v případě, že dané API má operace pro získání informací o uživateli
- umožňuje povolit pro konkrétního **poskytovatele služby** jen určité operace (ze všech operací) na API určitého poskytovatele API
- podporují API Google, Facebook, Twitter, ...
- seznam implementací - <http://oauth.net/2/>

# Co umožňuje OAuth

- není omezeno jen na web, lze i pro mobilní aplikace (Android, iOS), desktopové, SmartTV, embedded v set-top-boxech
- spolupráce dvou různých web serverů
- např. uživatel Google Disk může povolit jinému webu od firmy X, případně jejich mobilní aplikaci, čtení dokumentů a zápis jejich upravených verzí
- aplikace může přistupovat k API i bez uživatele



1. Original request for resource
2. Redirect to "server" for authorization
3. Response is from server domain asking resource owner to authenticate
4. Resource owner authenticates
5. Server issues token to client
6. Client confirms access

# Jak to funguje

1. vývojář aplikace se zaregistruje u poskytovatele API
  - Google API console <https://code.google.com/apis/console/>
  - Facebook developers <https://developers.facebook.com/apps/>
2. zaregistruje aplikaci, získá *client\_secret*
3. při příchodu uživatele do aplikace přesměruje na OAuth server se žádostí o oprávnění k určitým operacím
4. aplikace získá od uživatele jednorázový kód
5. aplikace vymění kód a *client\_secret* za token
6. aplikace volá API a prokazuje se tokenem

# Registrace aplikace u Google

## Create Client ID ✕

**Client ID Settings**

**Application type**

- Web application  
Accessed by web browsers over a network.
- Service account  
Calls Google APIs on behalf of your application instead of an end-user. [Learn more](#)
- Installed application  
Runs on a desktop computer or handheld device (like Android or iPhone).

**Your site or hostname [\(more options\)](#)**  
For example: `www.example.com` or `localhost`

**Redirect URI**  
`https://www.example.com/oauth2callback`

---

[Learn more](#)

# Zaregistrovaná aplikace u Google

[Vyhledávání](#) [Obrázky](#) [Mapy](#) [Play](#) [YouTube](#) [Zprávy](#) [Gmail](#) [Disk](#) [Další](#) ▼

[martinkuba@gmail.com](#) ▼ | [Nastavení](#) ▼ | [Nápověda](#) | [Odhlásit se](#)



MUNI Photometric Arch... ▼

Overview

Services

Team

API Access

## API Access

To prevent abuse, Google places limits on API requests. Using a valid OAuth token or API key allows you to exceed anonymous limits by connecting requests back to your project.

### Authorized API Access

OAuth 2.0 allows users to share specific data with you (for example, contact lists) while keeping their usernames, passwords, and other information private. A single project may contain up to 20 client IDs. [Learn more](#)

### Branding information

The following information is shown to users whenever you request access to their private data.

Product name: MUNI Photometric Archive  
Google account: martinkuba@gmail.com  
Home page URL: <https://www.cerit-sc.cz/login/>

[Edit branding information...](#)

### Client ID for web applications

Client ID: 558708443072.apps.googleusercontent.com  
Email address: 558708443072@developer.gserviceaccount.com  
Client secret: 6\_woe [REDACTED] -LxMvG  
Redirect URIs: <https://www.cerit-sc.cz/login/google/auth>  
JavaScript origins: <https://www.cerit-sc.cz>

[Edit settings...](#)

[Reset client secret...](#)

[Download JSON](#)

[Delete...](#)

[Create another client ID...](#)

# Povolená API u Google

[Vyhledávání](#) [Obrázky](#) [Mapy](#) [Play](#) [YouTube](#) [Zprávy](#) [Gmail](#) [Disk](#) [Další](#) ▼

[martinkuba@gmail.com](#) ▼ | [Nastavení](#) ▼ | [Nápověda](#) | [Odhlásit se](#)

Google apis

MUNI Photometric Arch... ▼

All (56) Active (0) Inactive (56) Google Cloud Platform

Overview












Services

Team

API Access

## All services

Select services for the project.

Service	Status	Notes
 Ad Exchange Buyer API <a href="#">?</a>	<input type="checkbox"/> OFF	Courtesy limit: 1,000 requests/day
 Ad Exchange Seller API <a href="#">?</a>	<input type="checkbox"/> OFF	Courtesy limit: 10,000 requests/day
 AdSense Host API <a href="#">?</a>	<a href="#">Request access...</a>	Courtesy limit: 100,000 requests/day
 AdSense Management API <a href="#">?</a>	<input type="checkbox"/> OFF	Courtesy limit: 10,000 requests/day
 Analytics API <a href="#">?</a>	<input type="checkbox"/> OFF	Courtesy limit: 50,000 requests/day
 Audit API <a href="#">?</a>	<input type="checkbox"/> OFF	Courtesy limit: 10,000 requests/day
 BigQuery API <a href="#">?</a>	<input type="checkbox"/> OFF	Courtesy limit: 10,000 requests/day • <a href="#">Pricing</a>
 Blogger API v3 <a href="#">?</a>	<a href="#">Request access...</a>	Courtesy limit: 10,000 requests/day
 Books API <a href="#">?</a>	<input type="checkbox"/> OFF	Courtesy limit: 1,000 requests/day
 Calendar API <a href="#">?</a>	<input type="checkbox"/> OFF	Courtesy limit: 10,000 requests/day
 Custom Search API <a href="#">?</a>	<input type="checkbox"/> OFF	Courtesy limit: 100 requests/day • <a href="#">Pricing</a>



# Zaregistrovaná aplikace u Facebooku



Hledat aplikace



MUNI Photometri...

## Aplikace ▶ MUNI Photometric Archive

Upravit aplikaci

+ Vytvořit novou aplikaci

### Nastavení

Upravit nastavení

ID aplikace / API klíč

500753543283158

Tajný klíč aplikace

69ba4c6[redacted]74ba317

Název aplikace

muniastro

Režim pískoviště

Vypnuto

Platformy, na kterých aplikace běží

Přihlášení pomocí Facebooku

### Vývojářská upozornění

Zobrazit vše

Nemáte žádná vývojářská upozornění.

### Statistiky

Zobrazit vše

Uživatelé

0 Nových uživatelů denně

0 Aktivní uživatelé za den

Sdílení

0 Obsah sdílený za den

0,00 Ohlasy podle sdílení

### Úlohy

Upravit úlohy

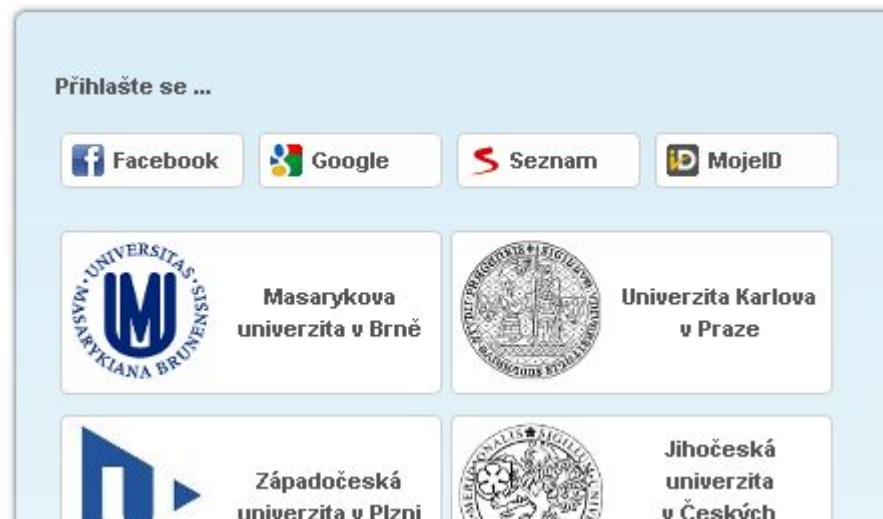
Úlohy

Správci:



# Příchod uživatele do aplikace

- uživatel si vybere poskytovatele účtu
- na screenshotu
  - Facebook a Google - OAuth
  - Seznam.cz a MojID - OpenID
  - MU, UK, ZČU, JU - SAML



# Odeslání uživatele na OAuth server

```
public class FacebookServlet extends HttpServlet {

    private static final String client_id = "500753543283158";
    private static final String client_secret = "69ba4c[REDACTED]ba317";
    private static final String redirect_uri = "https://www.cerit-sc.cz/login/facebook/auth";
    private static final String LOGIN_URL = "https://www.facebook.com/dialog/oauth";
    private static final String TOKEN_URL = "https://graph.facebook.com/oauth/access_token";
    private static final String SCOPE = "email";

    @Override
    protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
        // Step 1
        if ("/login".equals(req.getPathInfo())) {
            //initiate facebook authorization
            String state = Integer.toString(random.nextInt(Integer.MAX_VALUE)); //random value protecting against XSRF
            req.getSession(true).setAttribute("state", state);
            //redirect to Facebook to ask for permission on email
            String loginRedirectURL = LOGIN_URL
                + "?client_id=" + urlEncode(client_id)
                + "&redirect_uri=" + urlEncode(redirect_uri)
                + "&state=" + urlEncode(state)
                + "&scope=" + urlEncode(SCOPE);
            resp.sendRedirect(loginRedirectURL);
        } else if ("/auth".equals(req.getPathInfo())) {
```

# Uživatel se přihlásí k účtu ...

 Facebook

Přihlaste se pro používání vašeho účtu s aplikací MUNI Photometric Archive.

E-mail nebo  
telefon:

makub@ics.muni.cz

Heslo:

••••••••

Zůstat přihlášen(a)


**Přihlásit se** nebo **Zaregistrujte se na Facebook**

[Zapomněli jste své heslo?](#)

# ... a schválí povolení k operacím




# Obdobně u Google



The image shows a Google interface with a permission request. At the top left is the Google logo. At the top right, the user's name 'Martin Kuba' is displayed next to a small profile picture and a dropdown arrow. The main content area contains a message from 'Klient MUNI Photometric Archive' asking for permissions. There are two expandable sections: one for 'Zobrazení základních informací o účtu' (Account basic information) and another for 'Zobrazení vaší e-mailové adresy' (Your email addresses). At the bottom, there are two buttons: 'Povolit přístup' (Allow access) and 'Ne, děkuji' (No, thank you). On the right side, there is a sidebar for 'MUNI Photometric Archive' with a link to 'Další informace' (More information).

Google

Martin Kuba 

Klient **MUNI Photometric Archive** požaduje následující oprávnění:

- ▼ Zobrazení základních informací o účtu
  - Zobrazení jména, webové adresy veřejného profilu a fotky
  - Zobrazení pohlaví a data narození
  - Zobrazení země, jazyka a časového pásma
- ▼ Zobrazení vaší e-mailové adresy
  - Zobrazení e-mailové adresy přiřazené k vašemu účtu

**Povolit přístup** **Ne, děkuji**

**MUNI Photometric Archive**  
[Další informace](#)

# Aplikace vymění kód od uživatele a vlastní `client_secret` za token

```
} else if ("/auth".equals(req.getPathInfo())) {
//process google authorization
//check state for XSRF attack
String state = req.getParameter("state");
String state1 = (String) req.getSession(true).getAttribute("state");
if (state == null || !state.equals(state1)) {
    resp.sendError(HttpServletResponse.SC_FORBIDDEN, "state does not match, probably a XSRF attack");
    return;
}
//get code
String code = req.getParameter("code");
if (code == null) {
    resp.sendError(HttpServletResponse.SC_FORBIDDEN, "code not present");
    return;
}
//exchange code for token
RestTemplate restTemplate = new RestTemplate();
MultiValueMap<String, String> map = new LinkedMultiValueMap<>();
map.add("client_id", client_id);
map.add("client_secret", client_secret);
map.add("redirect_uri", redirect_uri);
map.add("code", code);
map.add("grant_type", "authorization_code");
JsonNode jsonNode = restTemplate.postForObject(TOKEN_URL, map, JsonNode.class);
String accessToken = jsonNode.path("access_token").asText();
String expires = jsonNode.path("expires_in").asText();
Log.debug("accessToken={ } expires={ }", accessToken, expires);
}
```

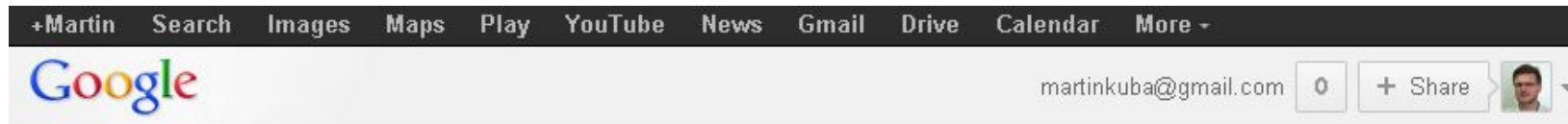
# Aplikace volá API

- aplikace volá určitá URL
- prokazuje se tokenem
- odpověď je obvykle JSON

```
//use token for getting user data  
JsonNode userData = restTemplate.getForObject(USER_INFO_URL+"?access_token={access_token}", JsonNode.class, accessToken);  
String userId = userData.path("id").asText();  
String userEmail = userData.path("email").asText();  
String userName = userData.path("name").asText();
```



# Uživatel může odebrat oprávnění



## Authorized Access to your Google Account

### Connected Sites, Apps, and Services

You have granted the following services access to your Google Account:

Android Login Service — Full Account Access [Revoke Access](#)

sfg.google.com — Google Calendar, Google Calendar [Revoke Access](#)

OAuth2 Login Demo — Profile Information [Revoke Access](#)

Google Developers — Google+ You [Revoke Access](#)

DevRates — Profile Information [Revoke Access](#)

ColorNote — [Revoke Access](#)

EasyPolls.net — Profile Information [Revoke Access](#)

MUNI Photometric Archive — Profile Information [Revoke Access](#)

MyTracks — Drive API [Revoke Access](#)

## Nastavení aplikací

Vaše jméno, profilová fotka, úvodní fotka, pohlaví, síť, uživatelské jméno a vaše ID číslo jsou vždy dostupné všem uživatelům Facebooku, a to včetně aplikací (důvody). Aplikace navíc mohou přistupovat k vašemu seznamu přátel a ostatním údajům, které jste na svém profilu nastavili jako veřejné.

Aplikace, které používáte	Chcete na Facebooku i jinde používat aplikace, plug-iny, hry a weby?	Zapnuto	Upravit
 <b>CiteULikeAuth</b>		Přátelé	<a href="#">Upravit</a> <span>✕</span>
 <b>Cities I've Visited</b>		Přátelé	<a href="#">Upravit</a> <span>✕</span>
 <b>Vimeo</b>		Přátelé	<a href="#">Upravit</a> <span>✕</span>
 <b>Heureka.cz</b>		Přátelé	<a href="#">Upravit</a> <span>✕</span>
 <b>Uplay</b>		Přátelé	<a href="#">Upravit</a> <span>✕</span>
 <b>Geocaching.com</b>		Přátelé	<a href="#">Upravit</a> <span>✕</span>

 **MUNI Photometric Archive** Poslední přihlášení: 21 únor [Zavřít](#)

Viditelnost aplikace:  **Přátelé** ▼

Tato aplikace vyžaduje:

- Vaše základní informace [?]
- Vaši e-mailovou adresu (makub@ics.muni.cz)

Poslední přístup k údajům: Základní informace Dnes  
[Zobrazit podrobnosti](#) · [Další informace](#)

Kdy si přejete být upozorněni? **Když vám aplikace pošle upozornění** ▼

[Odebrat aplikaci](#) · [Nahlásit aplikaci](#)

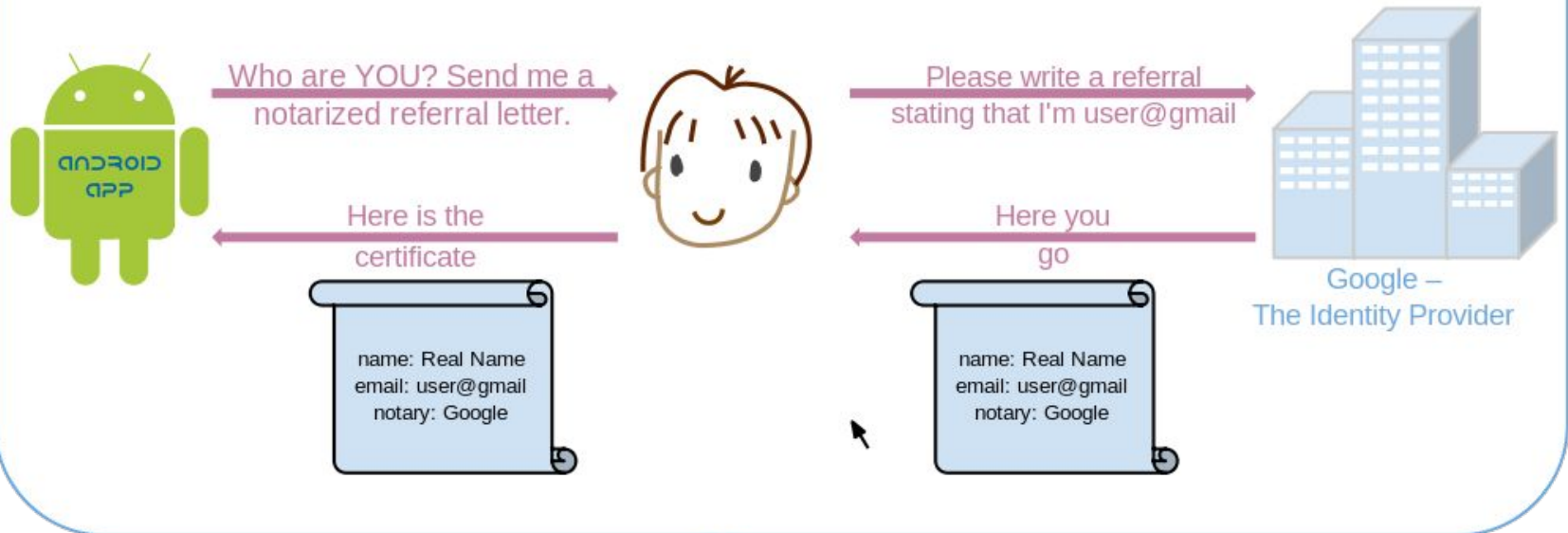
# OAuth shrnutí

- uživatel i aplikace jsou zaregistrovány
- oba mají své heslo (client\_secret u app)
- poskytovatel API v dokumentaci uvádí možná oprávnění (seznam operací)
- aplikace žádá uživatele o konkrétní oprávnění (povolení k určité množině operací)
- pokud uživatel schválí, aplikace získá token
- uživatel může kdykoliv token revokovat

# Srovnání s OpenID 1.0/2.0

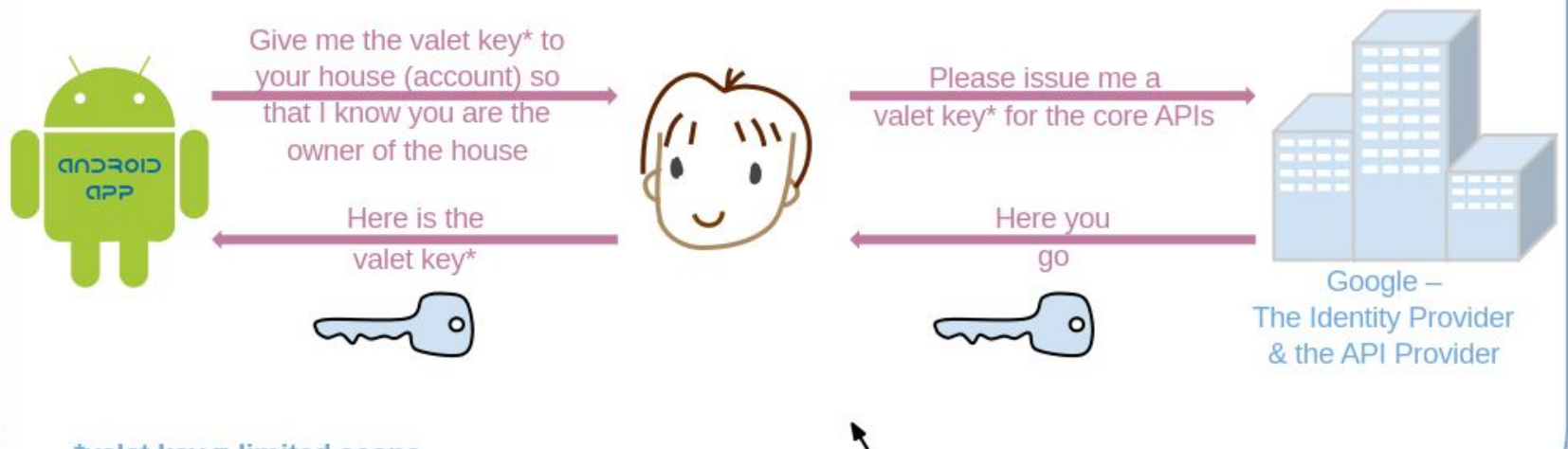
- otevřený standard
  - OpenID - pro autentizaci
  - OAuth - pro autorizaci
- aplikace se registrovat
  - OpenID - nemusí
  - OAuth - musí
- poskytovatel totožnosti
  - OpenID - kdokoliv, ale jak mu věřit ?
  - OAuth - konkrétní, API-specific

## OpenID Authentication



vs.

## Pseudo-Authentication using OAuth



\*valet key = limited scope  
OAuth Token

adapted from a drawing by @\_nat\_en

# Srovnání se SAML

- SAML

- autentizace, jako OpenID 1.0/2.0
- potřebuje Discovery Service/WAYF
- uživatel nemá kontrolu nad vydávanými údaji
- IdP a SP se musí vzájemně dohodnout
- SP nemůže požádat uživatele o více informací
- uživatel může schválit všechny nebo nic

- OAuth

- autorizace
- autentizace jako nulová autorizace
- uživatel má kontrolu nad vydávanými oprávněními
- může je i zpětně revokovat

# OpenID Connect (OIDC)

- nadstavba nad OAuth 2.0
- OAuth 2 - uživatel, vlastní zdroj/data na resource serveru, zplnomocňuje cizí aplikaci, aby jeho jménem se zdroji zacházela
- **authorization server** vydá **token** se specifickými **scopes** omezujícími možné akce na API **resource serveru**
- OIDC specifikuje jednotné **UserInfo API** autorizované pomocí OAuth 2 k vydání informací o uživateli

# **Konec**

Děkuji za pozornost