OAuth2 for iOS mobile app

tweet: @corinnekrych

code: github corinnekrych

blog: corinnekrych.org

chat: irc aerogear

OAuth

OAuth2.0

OAuth2 on native app

Tokens

OAuth2 and beyond

OAuth2 server Keycloak

OAuth: problem to solve

Services

APIs like Google Drive, Twitter or your own..

Password-based security

provides credentials to gain access

Applications

that uses services

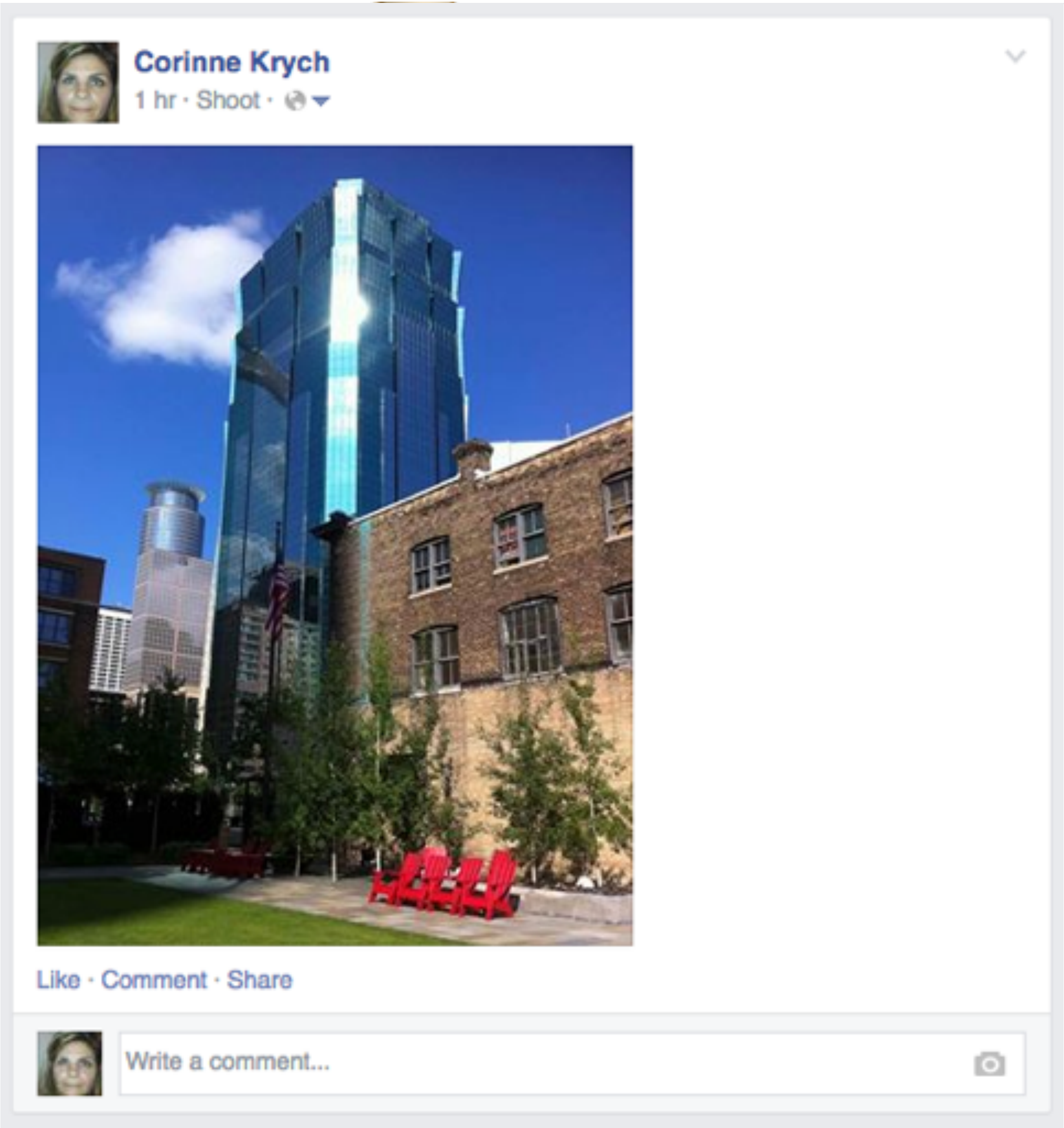
How many applications have your password?

Do you trust them all?

Are you sure?

Really really sure?

Demo: Shoot'nShare



Corinne Krych

1 hr · Shoot ·

Like · Comment · Share

Write a comment...

OAuth: what is it?

WIKIPEDIA

OAuth is an **open standard** to **authorization**. OAuth provides client applications a 'secure **delegated** access' to server resources on behalf of a resource owner.

OAuth != OpenID

WIKIPEDIA

OAuth is an open standard to authorization.

OpenID is an open specification for authentication and single sign-on (SSO).

OAuth → OAuth2.0

OpenID → OpenID Connect

OAuth Actors

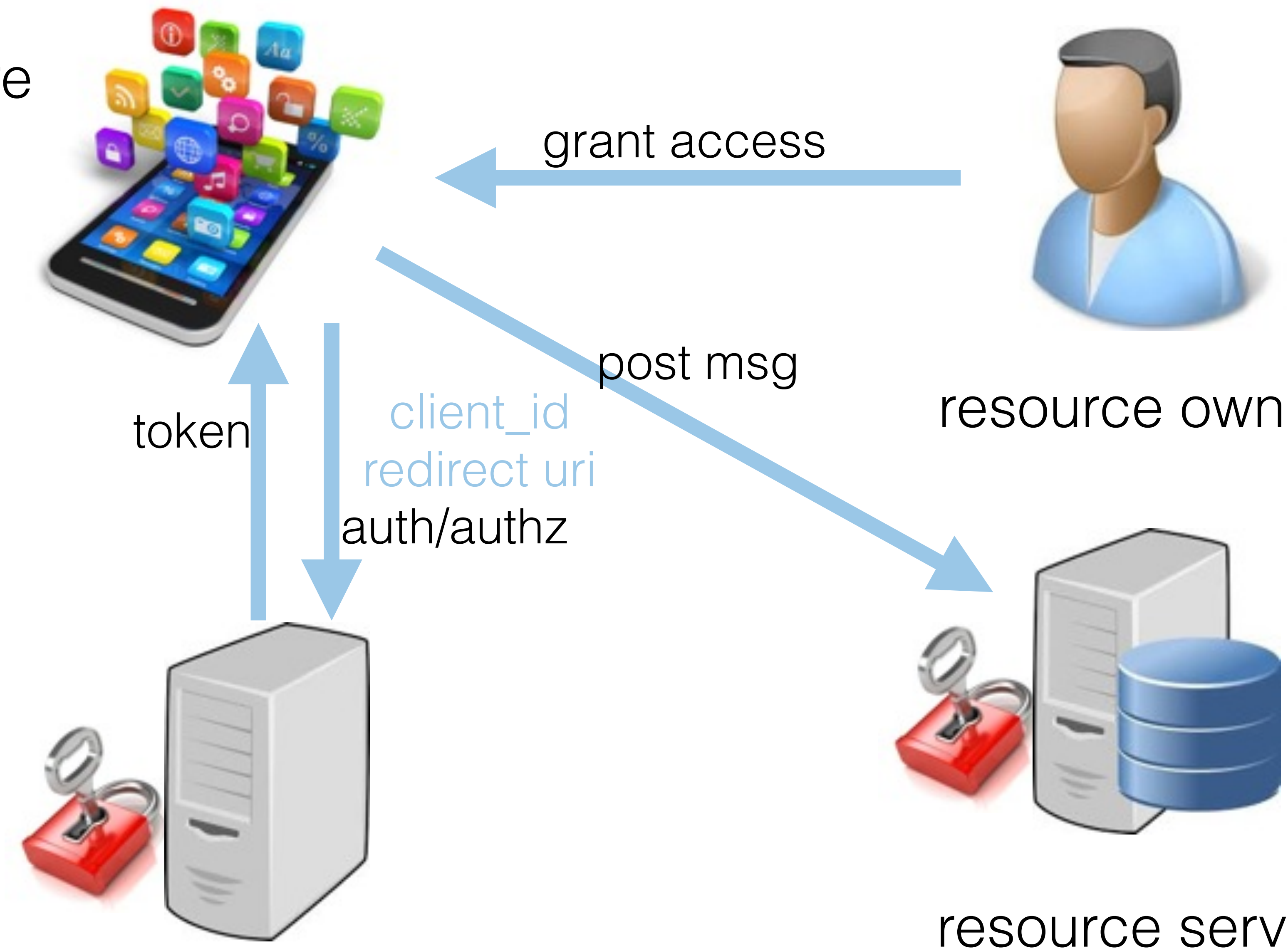
client app

resource owner

authorisation server

resource server

Example



Shoot'nShare client app

grant access

resource owner

post msg

token

client_id
redirect uri
auth/authz

resource server

authorisation server

OAuth

OAuth2.0

OAuth2 on native app

Tokens

OAuth2 and beyond

OAuth2 server Keycloak

OAuth2.0: Reloaded?

Introduction of **RFC6749**:

The OAuth 2.0 **authorization framework** enables a third-party application to obtain limited access to an HTTP service[…] **This specification replaces and obsoletes the OAuth 1.0 protocol described in RFC 5849.**

- no longer requires client app to have cryptography
- support more authz flows
- access tokens are "short-lived".

When to use 2.0?

Different Grant Types

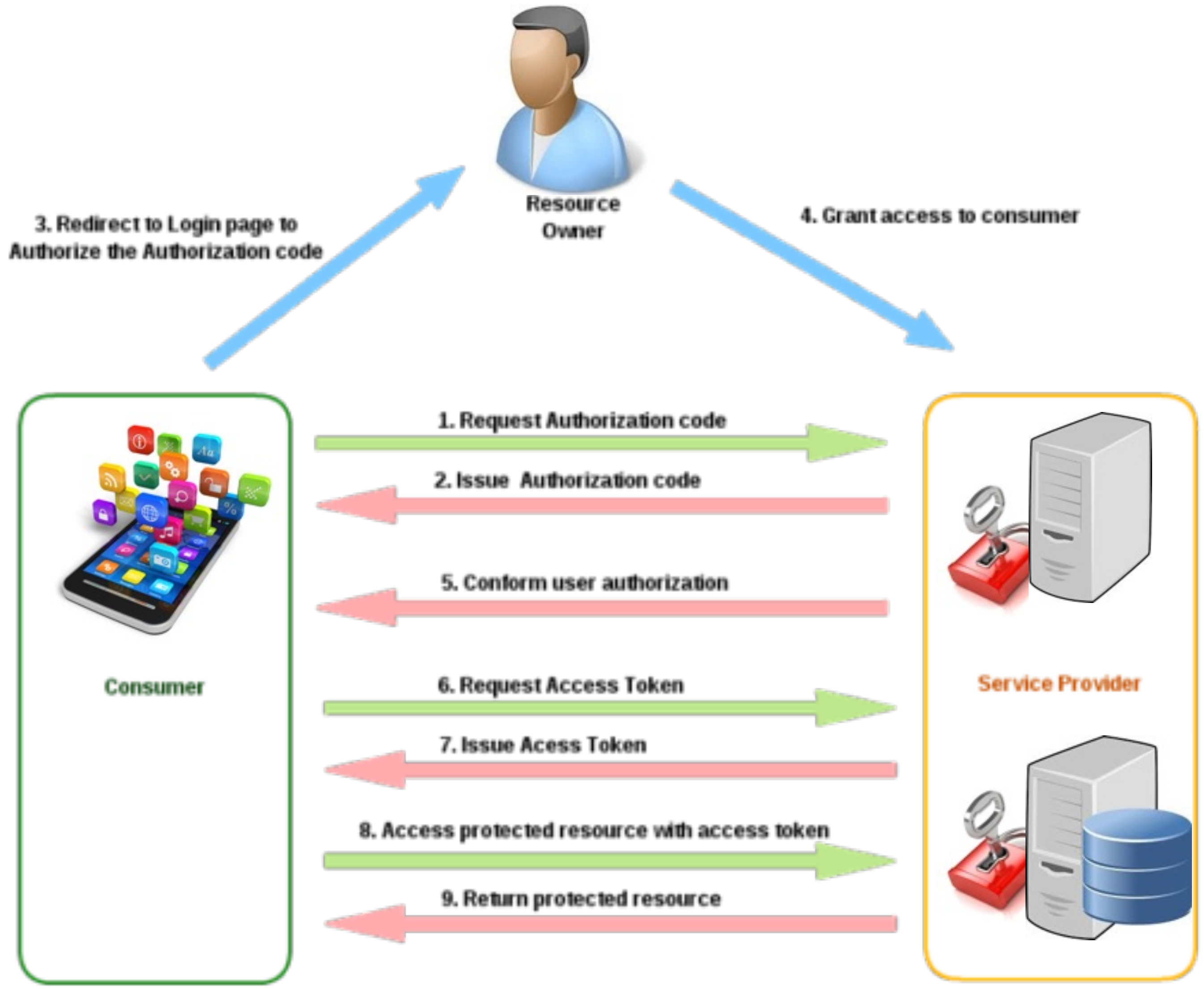
Authz code

Implicit (web)

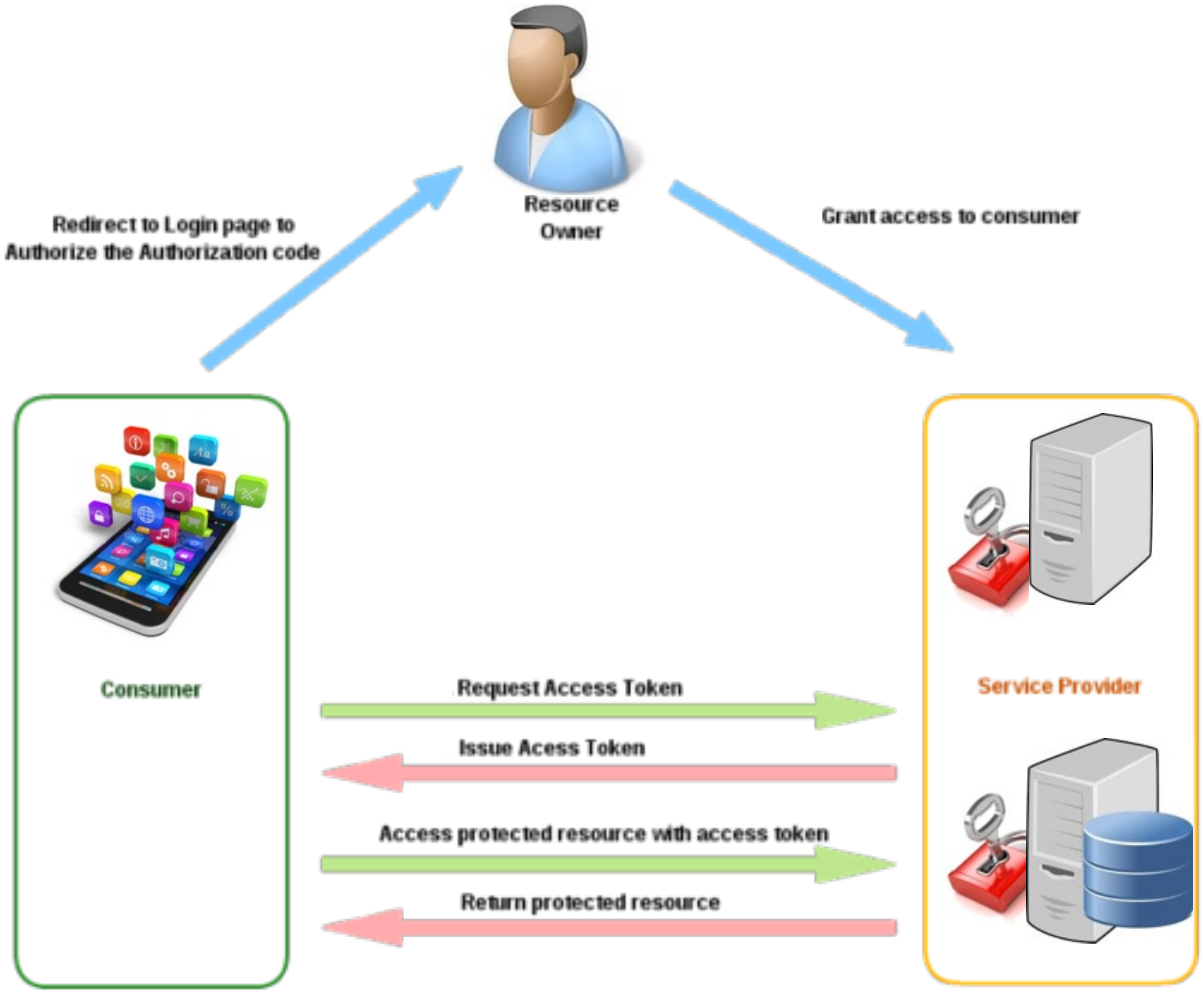
Client credentials

Resource owner credentials

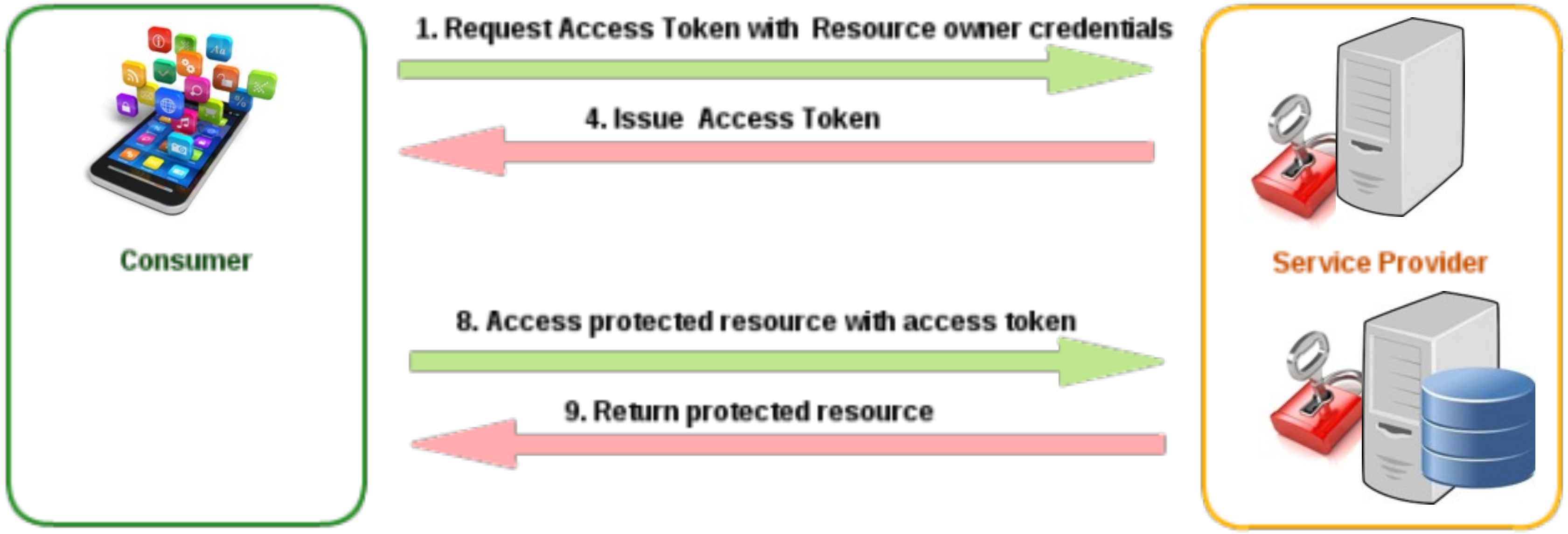
Authorization code grant


Resource Owner
3. Redirect to Login page to Authorize the Authorization code
4. Grant access to consumer
1. Request Authorization code
2. Issue Authorization code
5. Conform user authorization
6. Request Access Token
7. Issue Acess Token
8. Access protected resource with access token
9. Return protected resource
Consumer
Service Provider

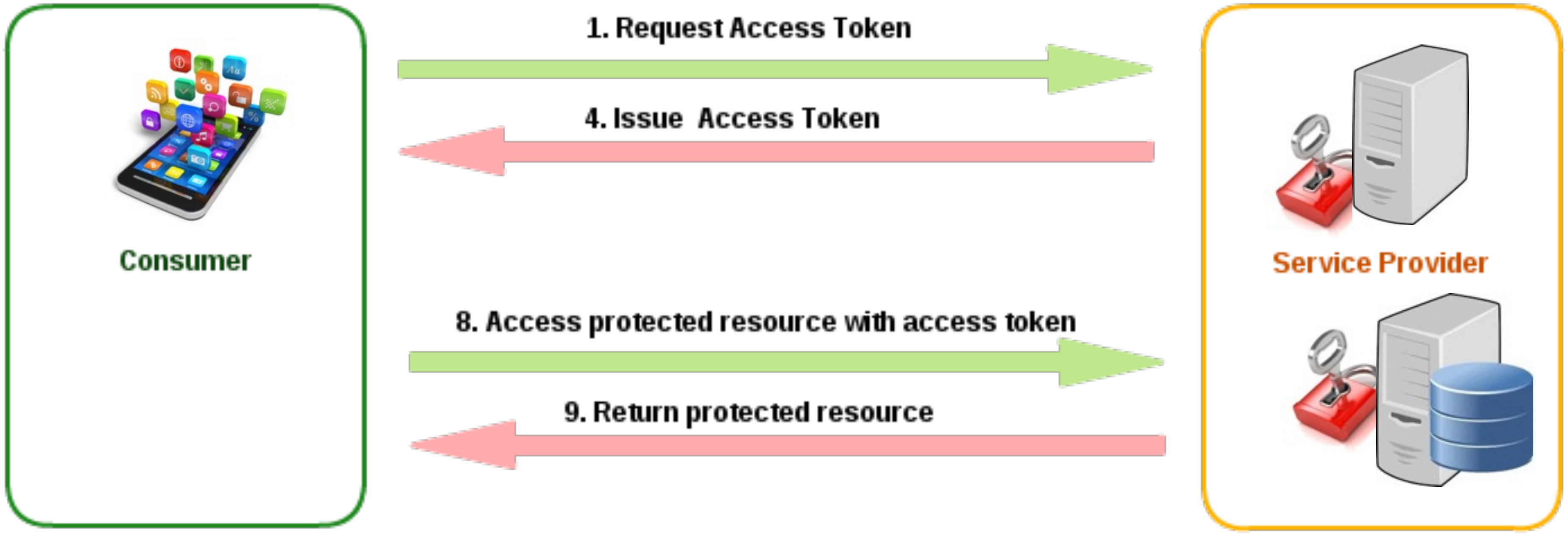
Implicit grant


Resource Owner
Redirect to Login page to Authorize the Authorization code
Grant access to consumer
Consumer
Service Provider
Request Access Token
Issue Acess Token
Access protected resource with access token
Return protected resource

Resource owner credentials grant


Consumer
1. Request Access Token with Resource owner credentials
4. Issue Access Token
Service Provider
8. Access protected resource with access token
9. Return protected resource

Client credentials grant


1. Request Access Token
4. Issue Access Token
Consumer
Service Provider
8. Access protected resource with access token
9. Return protected resource

OAuth

OAuth2.0

OAuth2 on native app

Authz code flow

Tokens

OAuth2 server Keycloak

As a iOS mobile user…

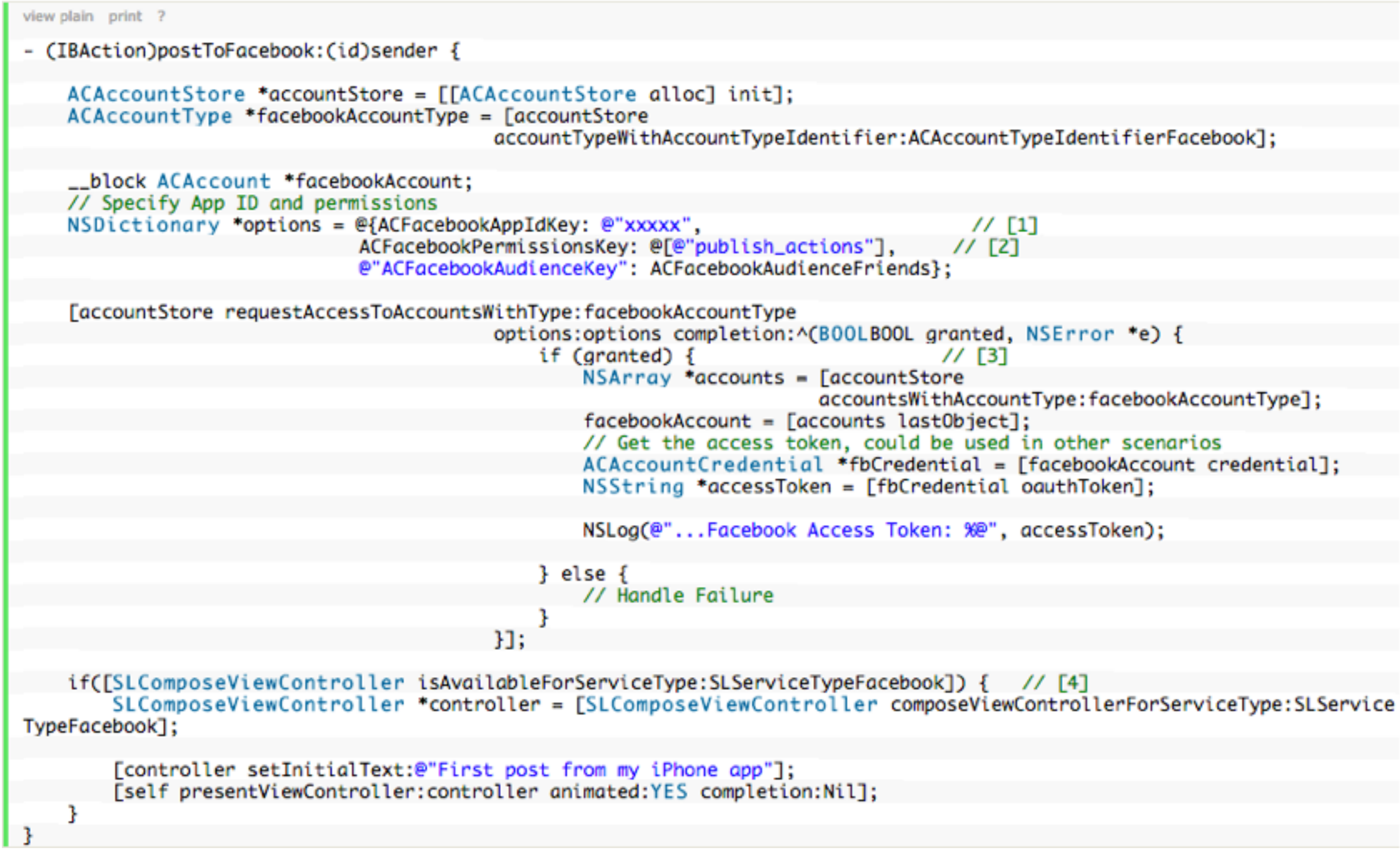
I want to share photos on Facebook so that my family and friends know all about my life.

Simple Use Case

Different ways…

- Social.framework
- Facebook iOS sdk
- (open source) libraries

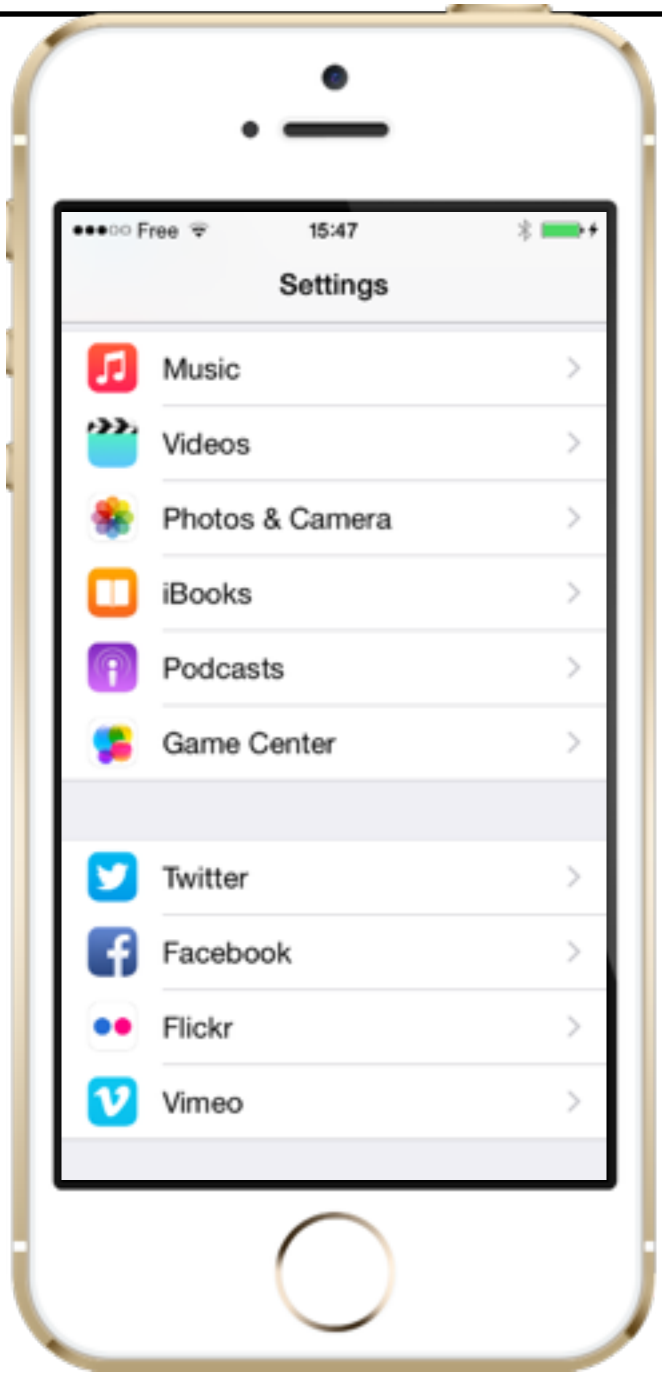
Using Social.framework



view plain print ?

```
- (IBAction)postToFacebook:(id)sender {

    ACAccountStore *accountStore = [[ACAccountStore alloc] init];
    ACAccountType *facebookAccountType = [accountStore
                                   accountTypeWithAccountTypeIdentifier:ACAccountTypeIdentifierFacebook];

    __block ACAccount *facebookAccount;
    // Specify App ID and permissions
    NSDictionary *options = @{ACFacebookAppIdKey: @"xxxxx",                    // [1]
                             ACFacebookPermissionsKey: @[@"publish_actions"],      // [2]
                             @"ACFacebookAudienceKey": ACFacebookAudienceFriends};

    [accountStore requestAccessToAccountsWithType:facebookAccountType
                                   options:options completion:^(BOOLBOOL granted, NSError *e) {
                                       if (granted) {                        // [3]
                                           NSArray *accounts = [accountStore
                                                                accountsWithAccountType:facebookAccountType];
                                           facebookAccount = [accounts lastObject];
                                           // Get the access token, could be used in other scenarios
                                           ACAccountCredential *fbCredential = [facebookAccount credential];
                                           NSString *accessToken = [fbCredential oauthToken];

                                           NSLog(@"...Facebook Access Token: %@", accessToken);

                                       } else {
                                           // Handle Failure
                                       }
                                   }];

    if([SLComposeViewController isAvailableForServiceType:SLServiceTypeFacebook]) {   // [4]
        SLComposeViewController *controller = [SLComposeViewController composeViewControllerForServiceType:SLService
TypeFacebook];

        [controller setInitialText:@"First post from my iPhone app"];
        [self presentViewController:controller animated:YES completion:Nil];
    }
}
```

… but limited



●●●○○ Free 15:47

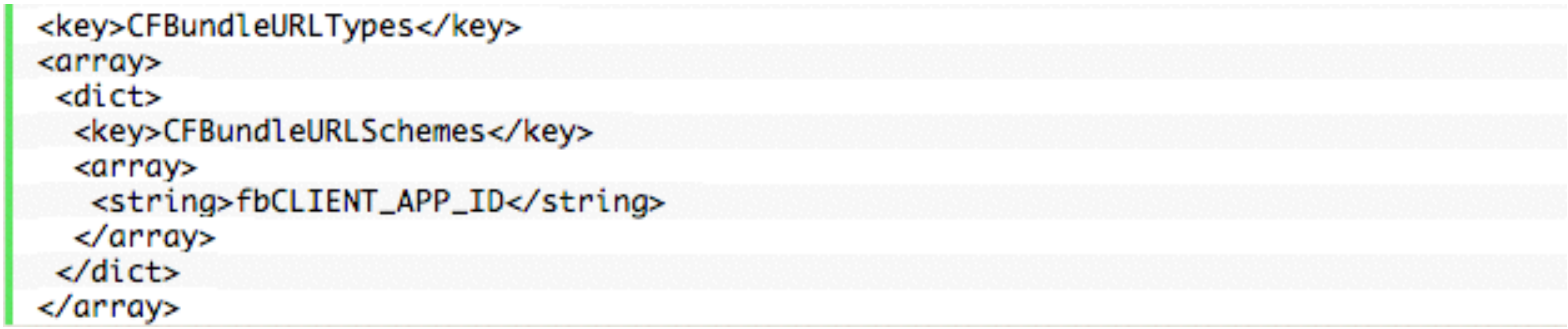
Settings

- Music
- Videos
- Photos & Camera
- iBooks
- Podcasts
- Game Center

- Twitter
- Facebook
- Flickr
- Vimeo

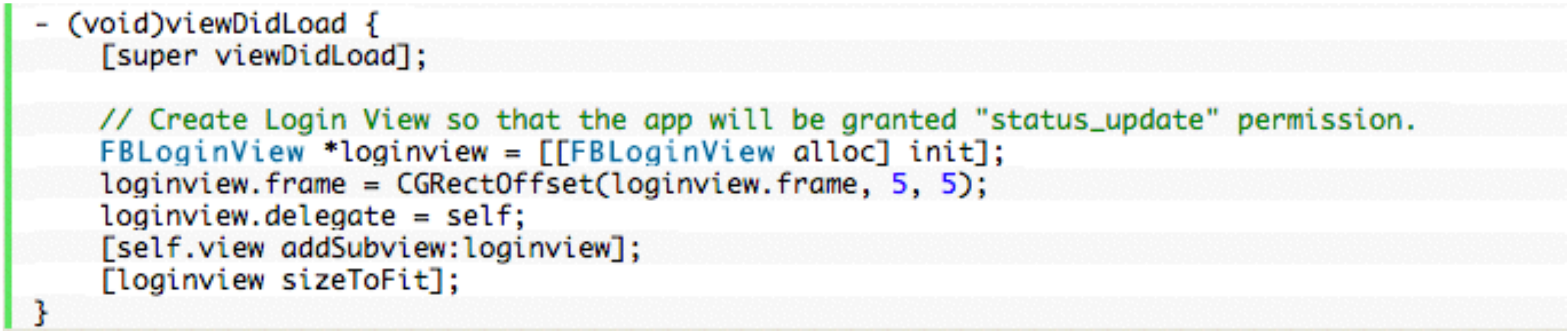
Using Facebook SDK

1. Configure URI schema



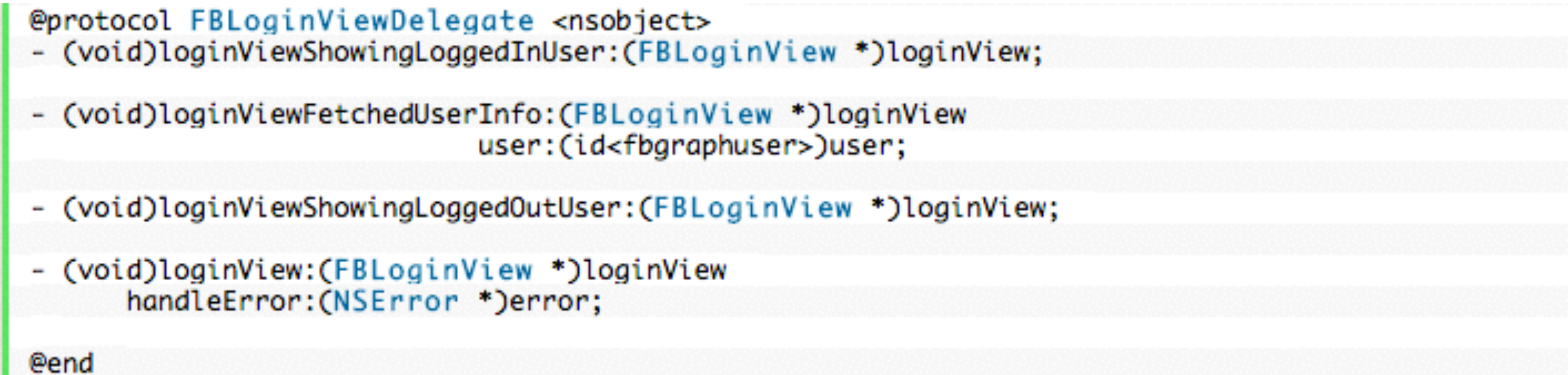
```
<key>CFBundleURLTypes</key>
<array>
 <dict>
  <key>CFBundleURLSchemes</key>
  <array>
   <string>fbCLIENT_APP_ID</string>
  </array>
 </dict>
</array>
```

2. Create Login view



```
- (void)viewDidLoad {
    [super viewDidLoad];

    // Create Login View so that the app will be granted "status_update" permission.
    FBLoginView *loginview = [[FBLoginView alloc] init];
    loginview.frame = CGRectOffset(loginview.frame, 5, 5);
    loginview.delegate = self;
    [self.view addSubview:loginview];
    [loginview sizeToFit];
}
```

3. Implement delegate methods



```
@protocol FBLoginViewDelegate <nsobject>
- (void)loginViewShowingLoggedInUser:(FBLoginView *)loginView;

- (void)loginViewFetchedUserInfo:(FBLoginView *)loginView
                            user:(id<fbgraphuser>)user;

- (void)loginViewShowingLoggedOutUser:(FBLoginView *)loginView;

- (void)loginView:(FBLoginView *)loginView
      handleError:(NSError *)error;

@end
```

… but not open

LinkedIn
skype
LIVEJOURNAL
foursquare
vimeo
YouTube
facebook
g+
Blogger
my
twitter
flickr
Behance
tumblr.
Like
yelp

Using aerogear-ios-oauth2

OPENSHIFT

AeroGear

UnifiedPush Server

keycloak

Keycloak Server

… and

100% open source

extensible

iOS sdk

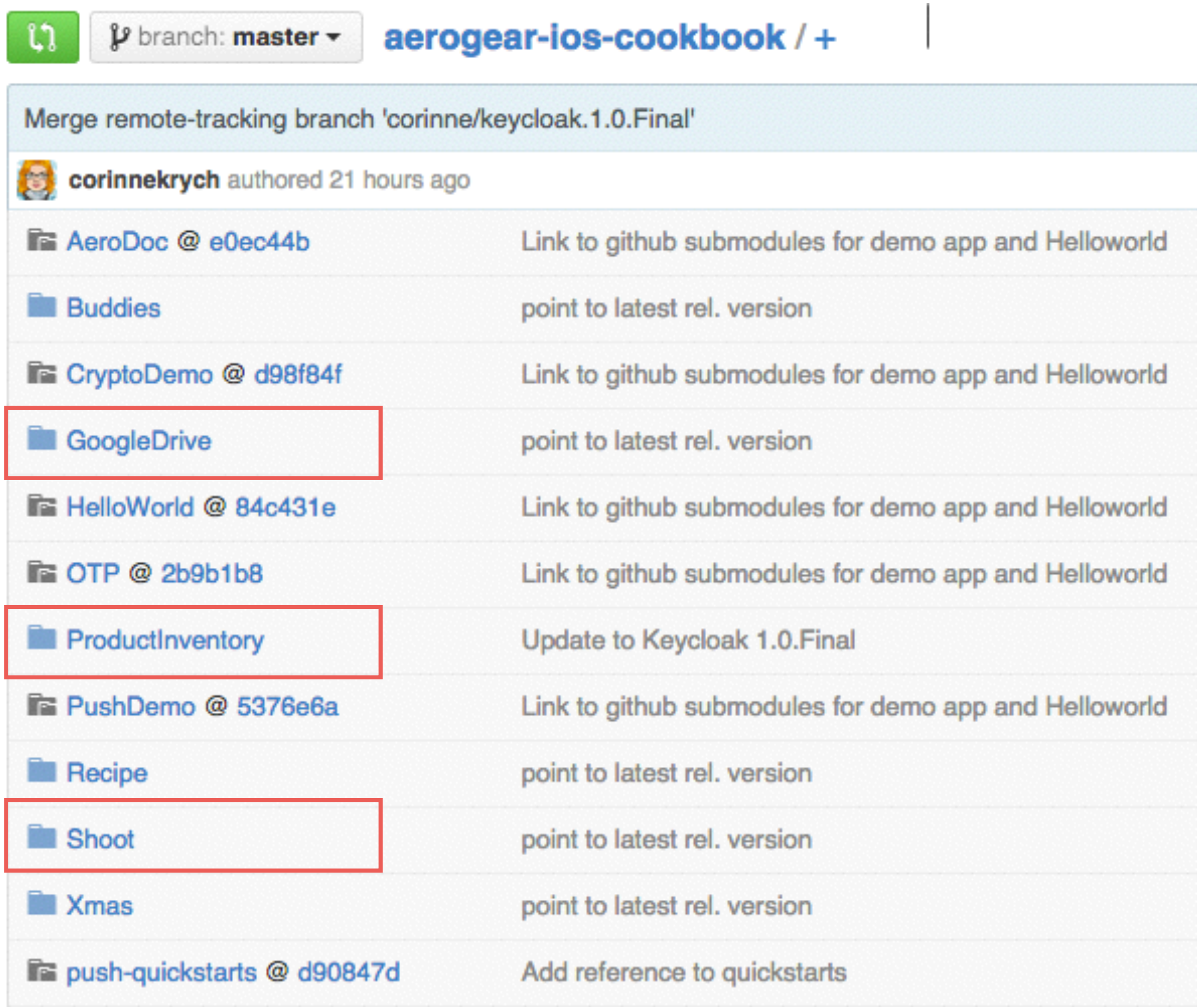
Android sdk

support: Facebook, Google, Keycloak

cookbook demos

aerogear-ios-cookbook

aerogear-backend-cookbook



aerogear-ios-cookbook / +

Merge remote-tracking branch 'corinne/keycloak.1.0.Final'

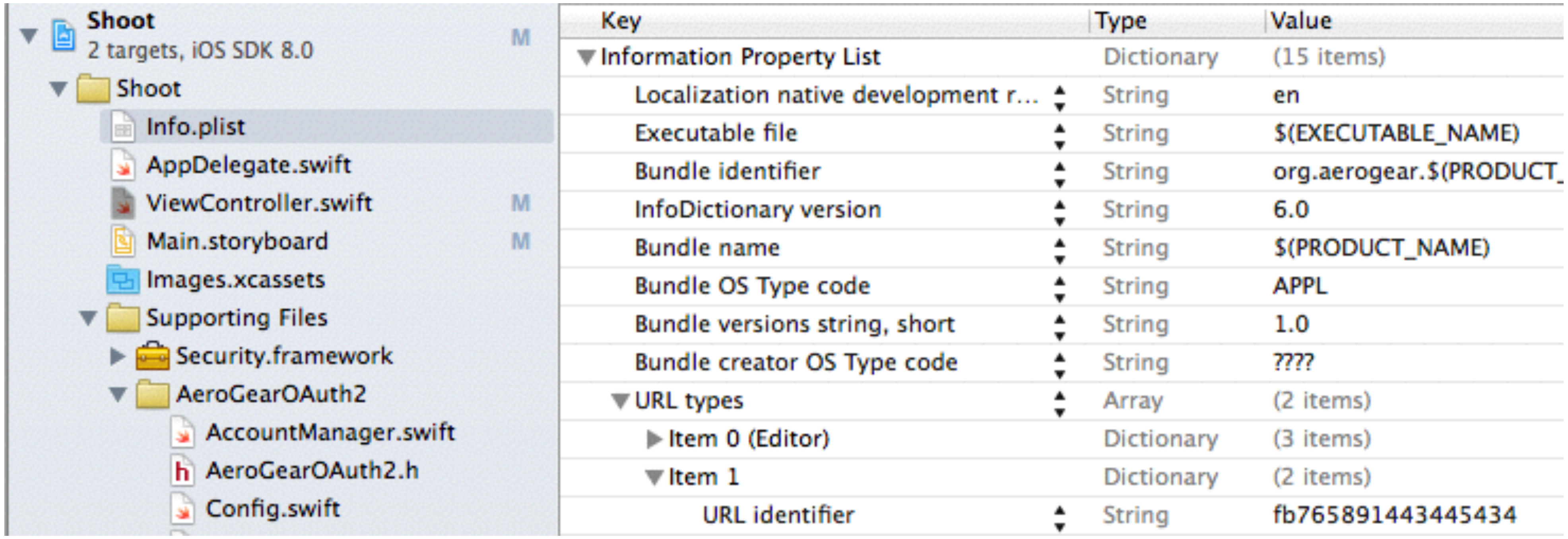
corinnekrych authored 21 hours ago

AeroDoc @ e0ec44b	Link to github submodules for demo app and Helloworld
Buddies	point to latest rel. version
CryptoDemo @ d98f84f	Link to github submodules for demo app and Helloworld
GoogleDrive	point to latest rel. version
HelloWorld @ 84c431e	Link to github submodules for demo app and Helloworld
OTP @ 2b9b1b8	Link to github submodules for demo app and Helloworld
ProductInventory	Update to Keycloak 1.0.Final
PushDemo @ 5376e6a	Link to github submodules for demo app and Helloworld
Recipe	point to latest rel. version
Shoot	point to latest rel. version
Xmas	point to latest rel. version
push-quickstarts @ d90847d	Add reference to quickstarts

How to come back?

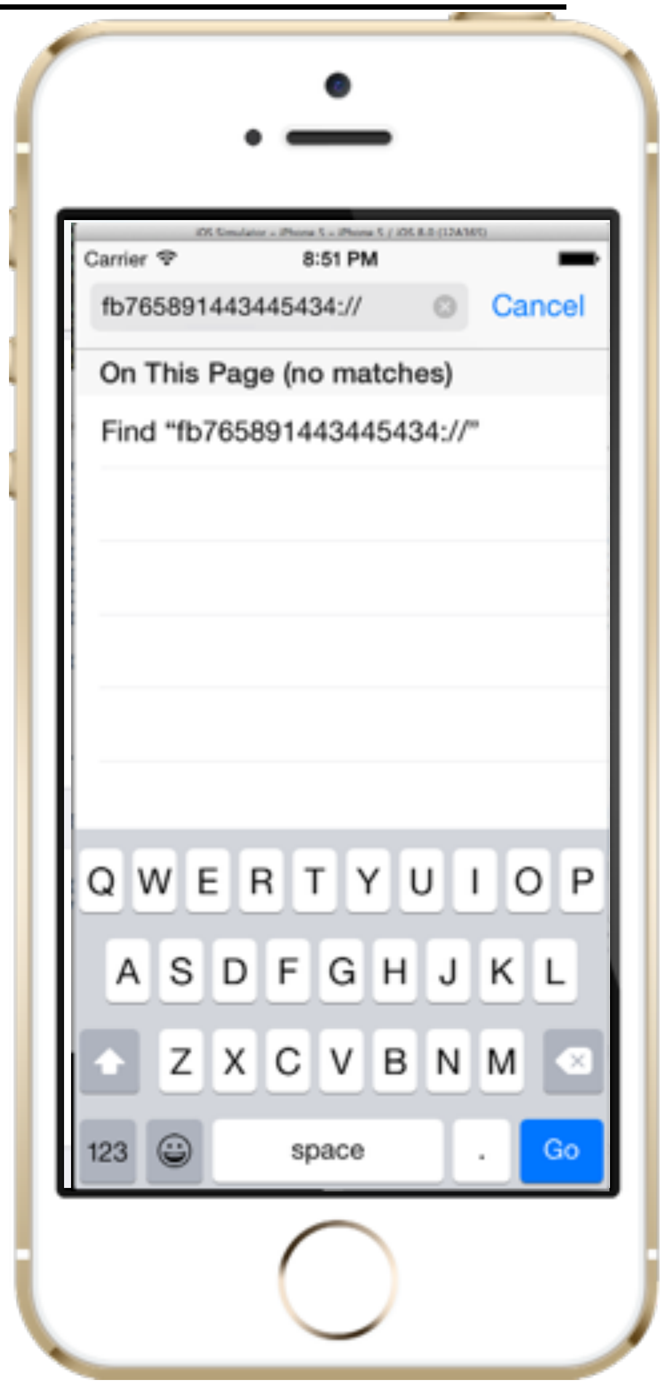
iOS **custom URL**

- define in project plist
- install app
- open safari and type in URL key



- Shoot
  2 targets, iOS SDK 8.0 M
  - Shoot
    - Info.plist
    - AppDelegate.swift
    - ViewController.swift M
    - Main.storyboard M
    - Images.xcassets
    - Supporting Files
      - Security.framework
      - AeroGearOAuth2
        - AccountManager.swift
        - AeroGearOAuth2.h
        - Config.swift

Key	Type	Value
Information Property List	Dictionary	(15 items)
Localization native development r...	String	en
Executable file	String	$(EXECUTABLE_NAME)
Bundle identifier	String	org.aerogear.$(PRODUCT_
InfoDictionary version	String	6.0
Bundle name	String	$(PRODUCT_NAME)
Bundle OS Type code	String	APPL
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
URL types	Array	(2 items)
Item 0 (Editor)	Dictionary	(3 items)
Item 1	Dictionary	(2 items)
URL identifier	String	fb765891443445434



Carrier 8:51 PM

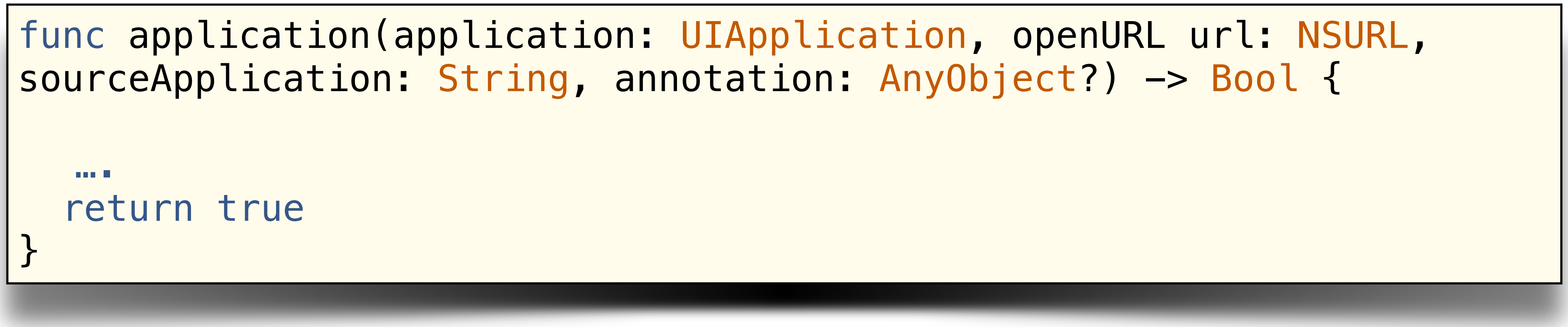
fb765891443445434:// Cancel

On This Page (no matches)

Find "fb765891443445434://"

Where do you come back?

AppDelegate.swift off course

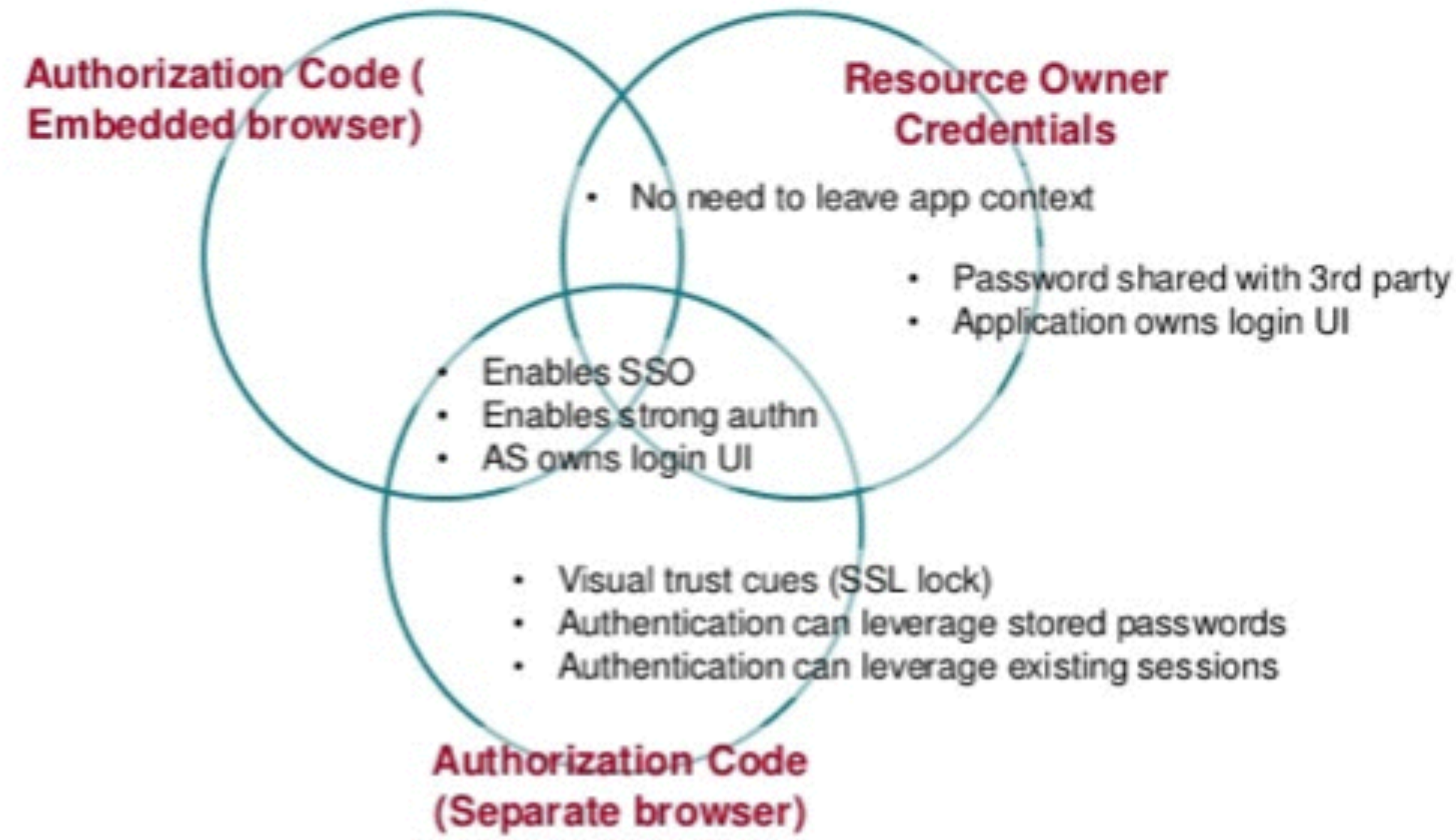


```
func application(application: UIApplication, openURL url: NSURL,
sourceApplication: String, annotation: AnyObject?) -> Bool {


   ….
  return true
}
```

Implicit access token request

```
let facebookConfig = FacebookConfig(
       clientId: "YYY",
       clientSecret: "XXX",
       scopes:["photo_upload, publish_actions"])

let fbModule =  AccountManager.addFacebookAccount(facebookConfig)
let http = Http()
http.authzModule = fbModule
let filename = self.imageView.accessibilityIdentifier;
let multiPartData = MultiPartData(data:
UIImageJPEGRepresentation(self.imageView.image, 0.2),
                                          name: "image",
                                      filename: filename,
                                      mimeType: "image/jpg")
http.POST("https://graph.facebook.com/me/photos", parameters:["data": multiPartData],
completionHandler:{(response, error) in
            if (error != nil) {
                println("Error uploading file: \(error)")
            } else {
                println("Successfully uploaded: " + response!.description)
            }
})
```

How to dance on native?



Authorization Code (Embedded browser)

Resource Owner Credentials

- No need to leave app context

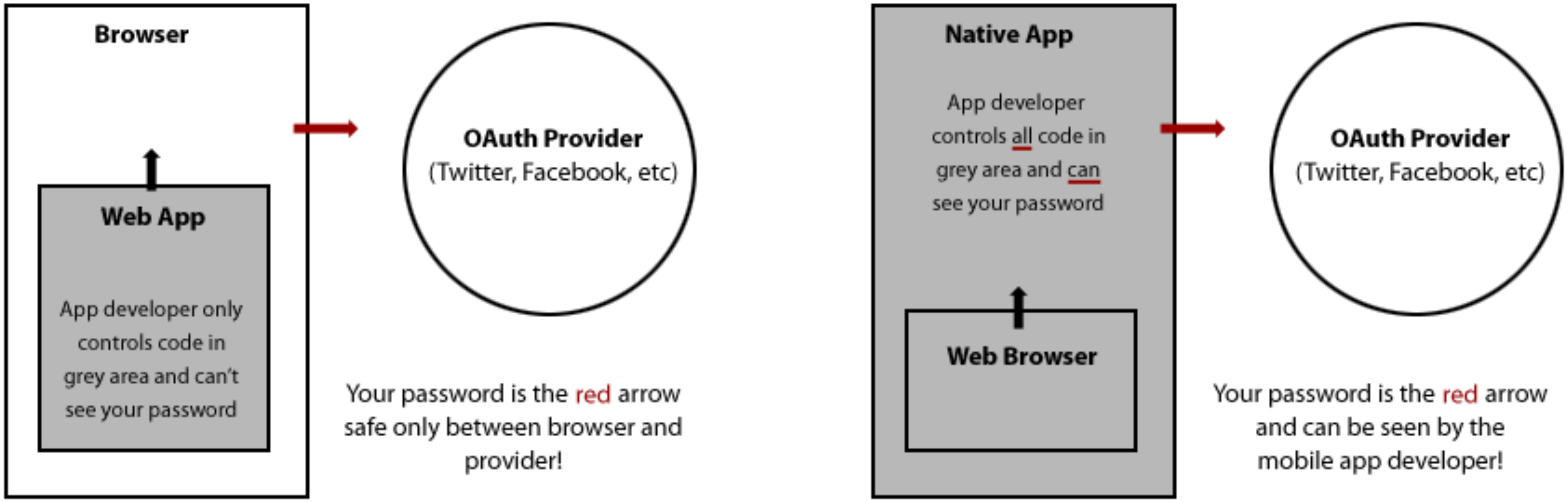
- Password shared with 3rd party
- Application owns login UI

- Enables SSO
- Enables strong authn
- AS owns login UI

- Visual trust cues (SSL lock)
- Authentication can leverage stored passwords
- Authentication can leverage existing sessions

Authorization Code (Separate browser)

Embedded or external?



Browser

Web App

App developer only controls code in grey area and can't see your password

OAuth Provider
(Twitter, Facebook, etc)

Your password is the red arrow safe only between browser and provider!

Native App

App developer controls all code in grey area and can see your password

Web Browser

OAuth Provider
(Twitter, Facebook, etc)

Your password is the red arrow and can be seen by the mobile app developer!

OAuth

OAuth2

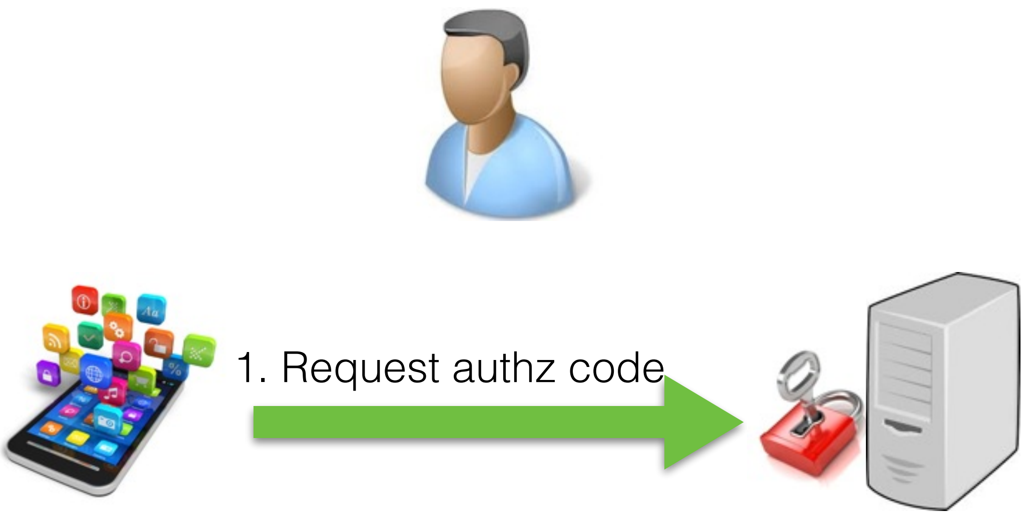
OAuth2 on native app

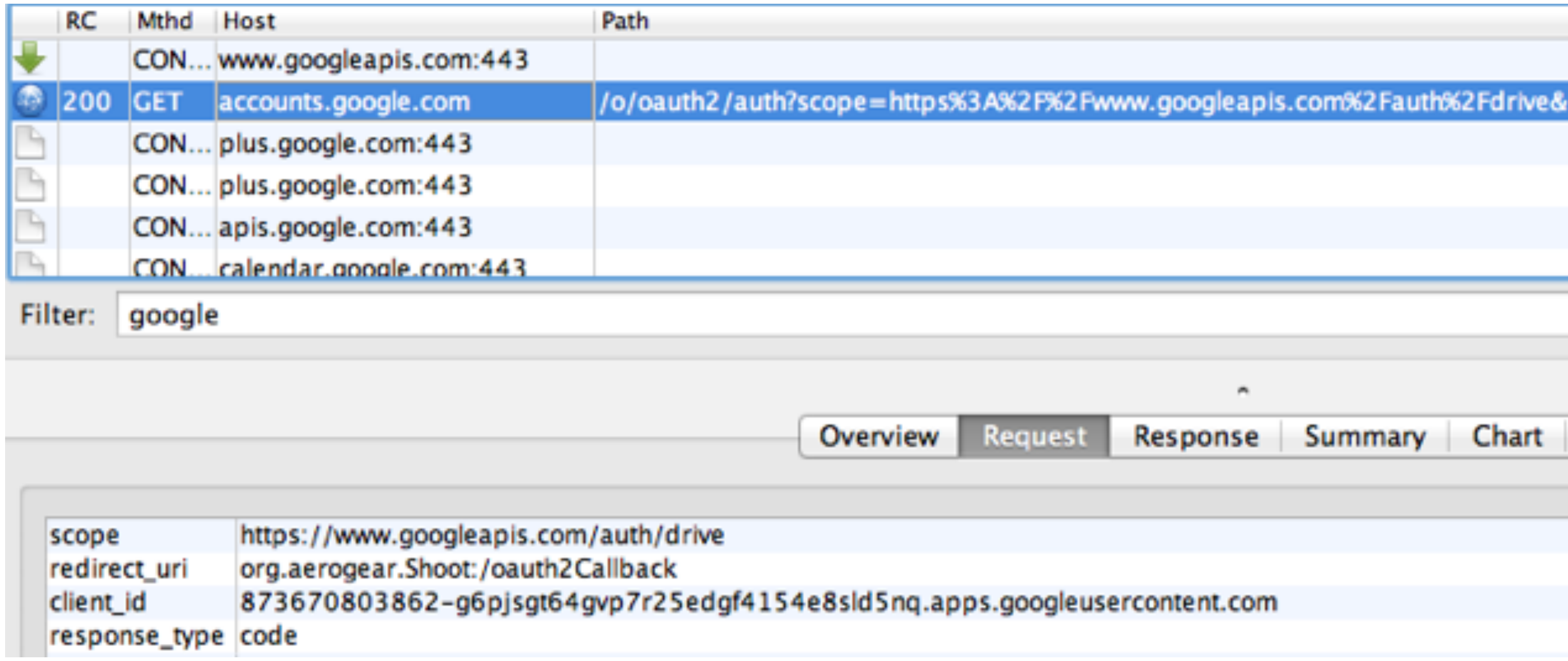
Authz code Flow

Tokens

OAuth2 server Keycloak

1: Ask for authz code





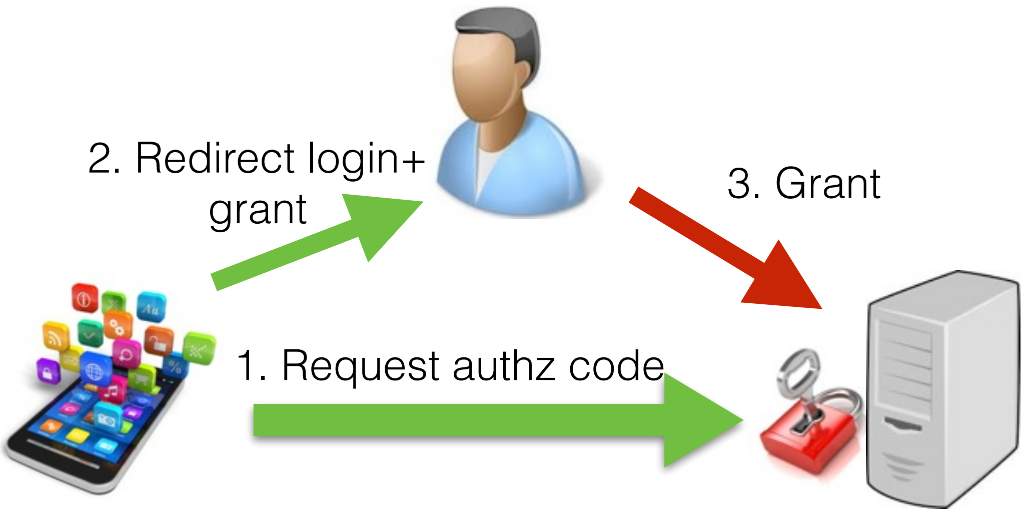
	RC	Mthd	Host	Path
		CON...	www.googleapis.com:443	
	200	GET	accounts.google.com	/o/oauth2/auth?scope=https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fdrive&
		CON...	plus.google.com:443	
		CON...	plus.google.com:443	
		CON...	apis.google.com:443	
		CON...	calendar.google.com:443	

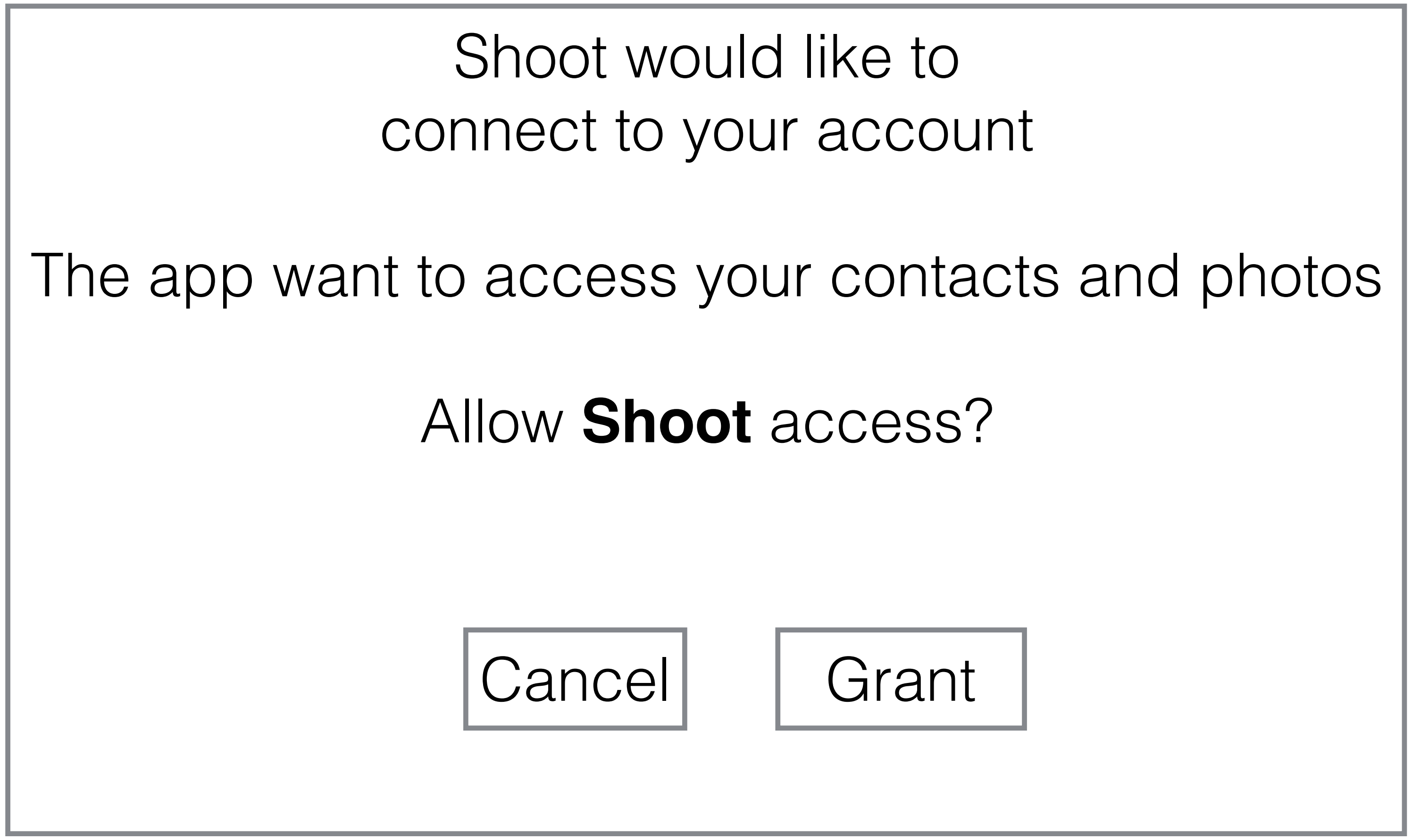
Filter: google

scope	https://www.googleapis.com/auth/drive
redirect_uri	org.aerogear.Shoot:/oauth2Callback
client_id	873670803862-g6pjsgt64gvp7r25edgf4154e8sld5nq.apps.googleusercontent.com
response_type	code

```
GET /o/oauth2/auth?scope=https%3A%2F%2Fwww.googleapis.com%2Fauth
%2Fdrive&redirect_uri=org.aerogear.Shoot%3A%2Foauth2Callback&client_id=873670803862-
g6pjsgt64gvp7r25edgf4154e8sld5nq.apps.googleusercontent.com&response_type=code
```

2, 3: Ask grant

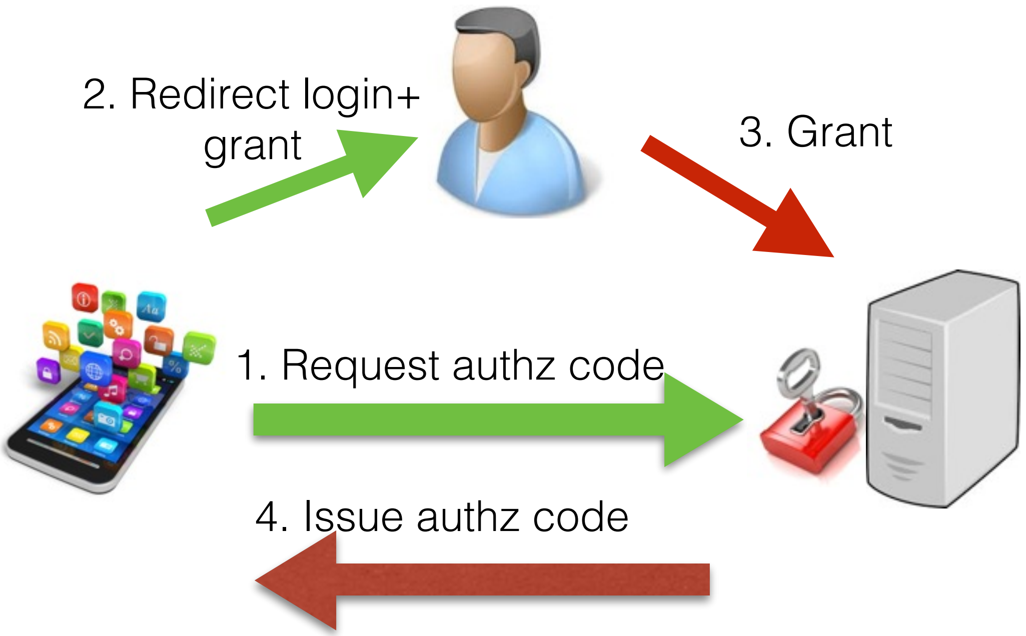



Shoot would like to
connect to your account

The app want to access your contacts and photos

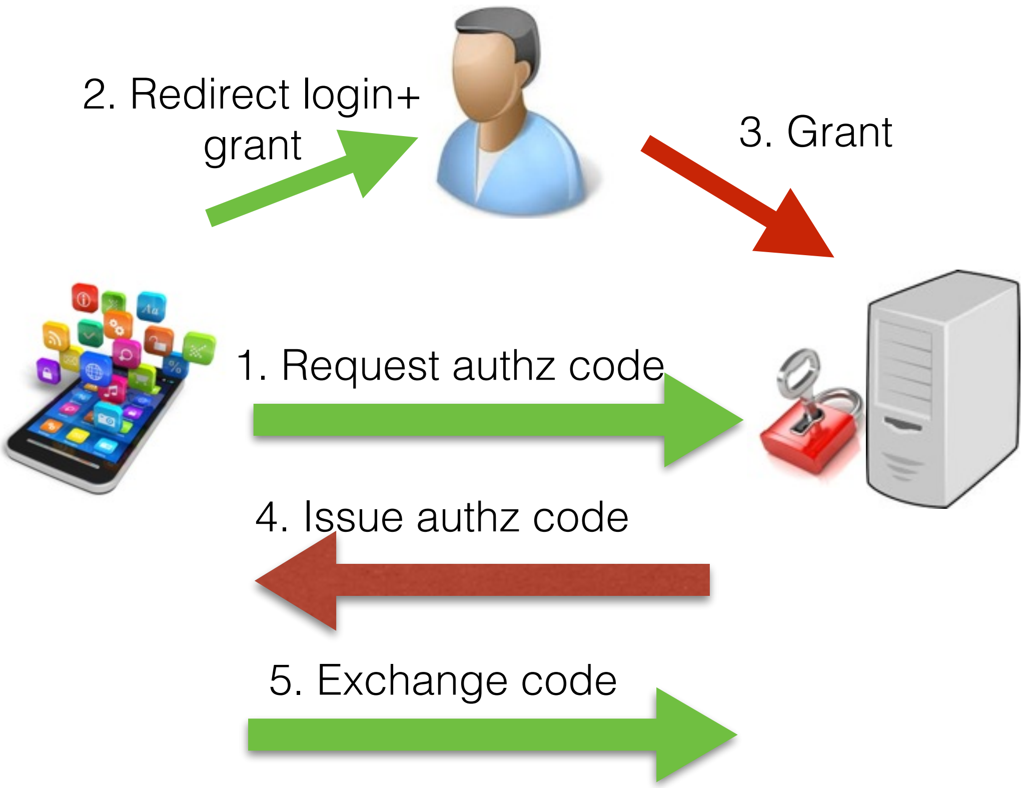
Allow **Shoot** access?

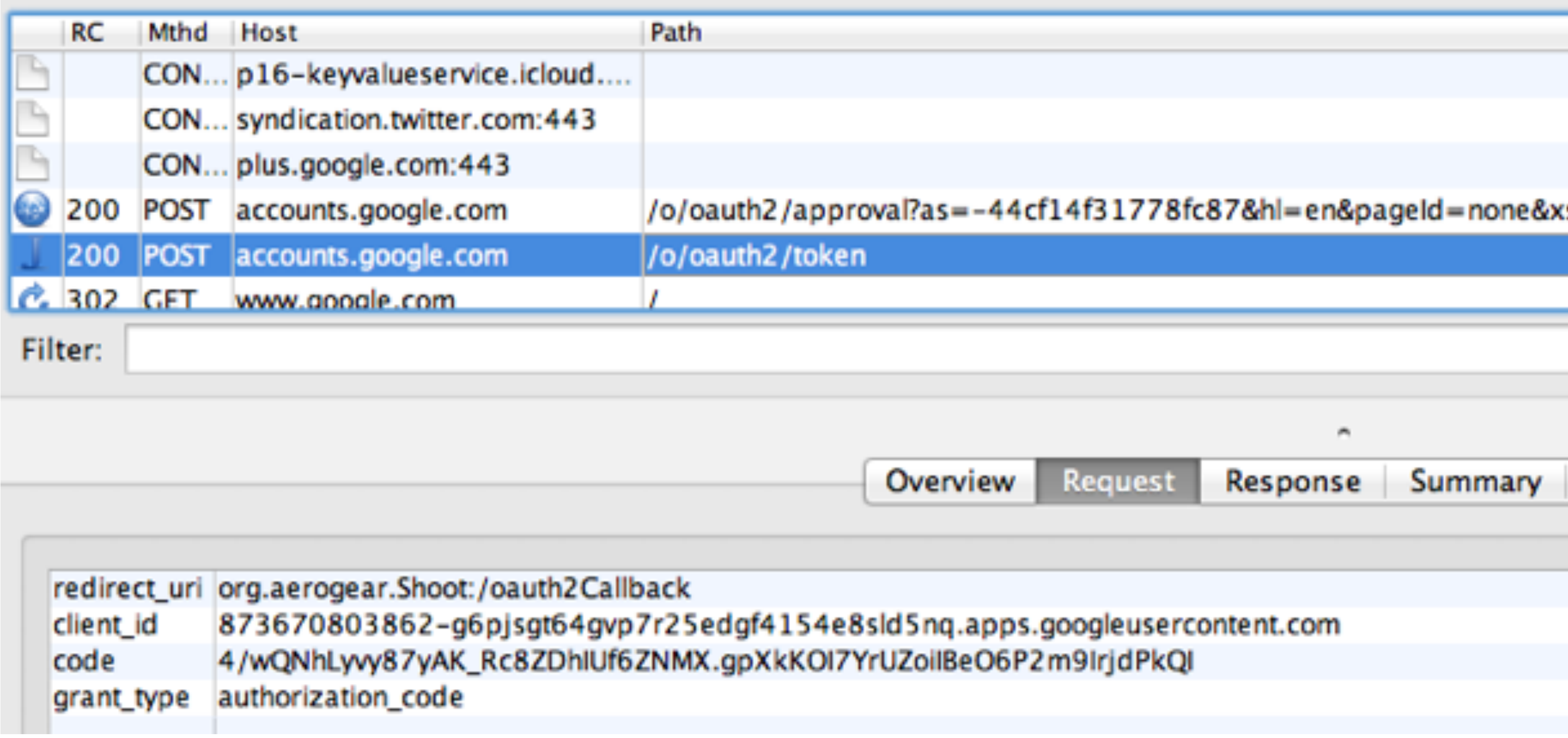
4:Authz code



- redirect you to the URI specified in the redirect_uri query string parameter,
- passing the authorization code in the query string.
- in AppDelegate implement openURL callback `application:openURL:sourceApplication:annotation:`
- extract code from query

5:Exchange code for token



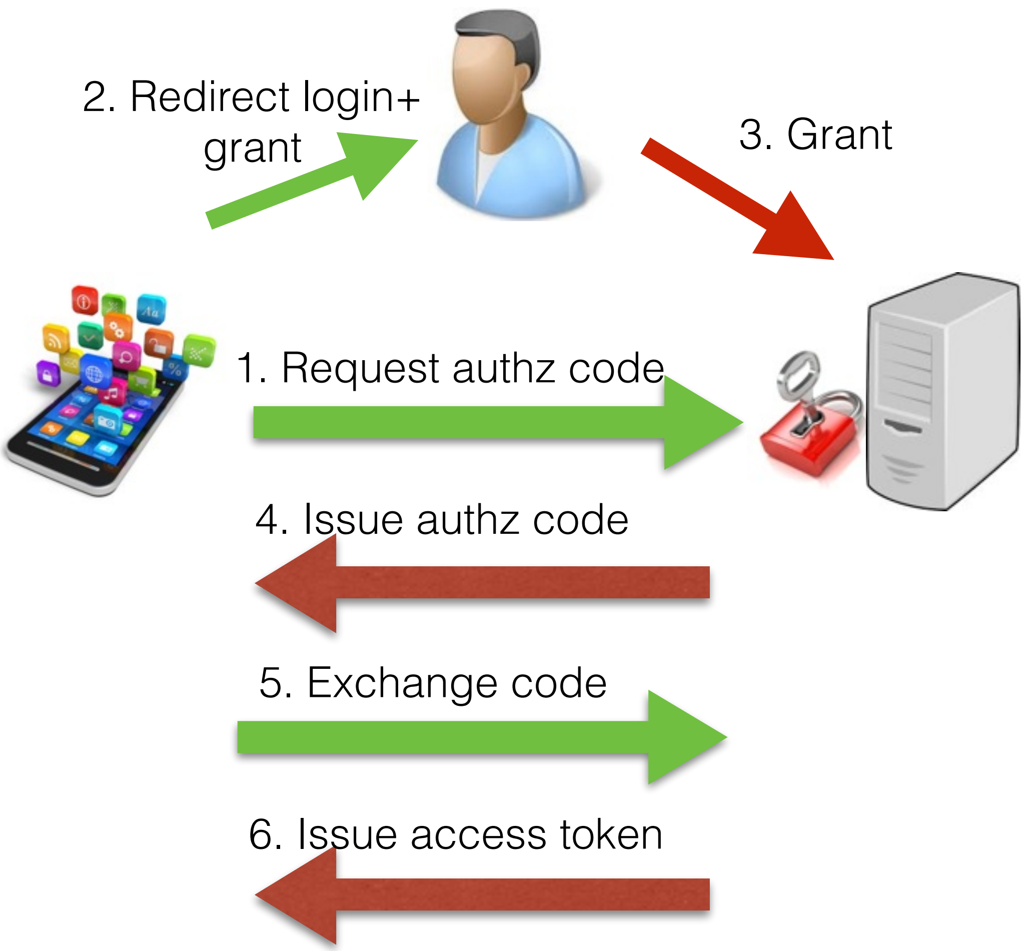


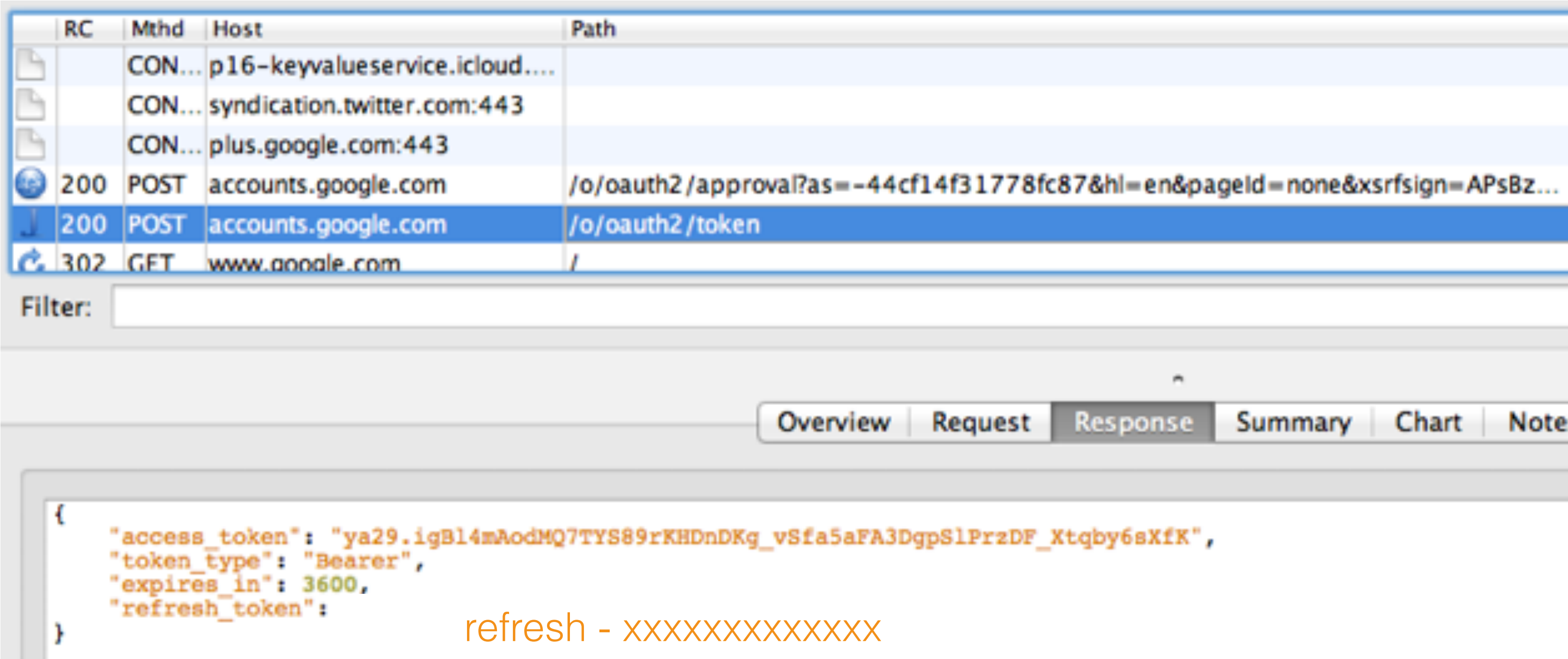
RC	Mthd	Host	Path
	CON...	p16-keyvalueservice.icloud....	
	CON...	syndication.twitter.com:443	
	CON...	plus.google.com:443	
200	POST	accounts.google.com	/o/oauth2/approval?as=-44cf14f31778fc87&hl=en&pageId=none&x
200	POST	accounts.google.com	/o/oauth2/token
302	GET	www.google.com	/

Filter:

redirect_uri	org.aerogear.Shoot:/oauth2Callback
client_id	873670803862-g6pjsgt64gvp7r25edgf4154e8sld5nq.apps.googleusercontent.com
code	4/wQNhLyvy87yAK_Rc8ZDhlUf6ZNMX.gpXkKOI7YrUZoilBeO6P2m9IrjdPkQI
grant_type	authorization_code

6: Get access token



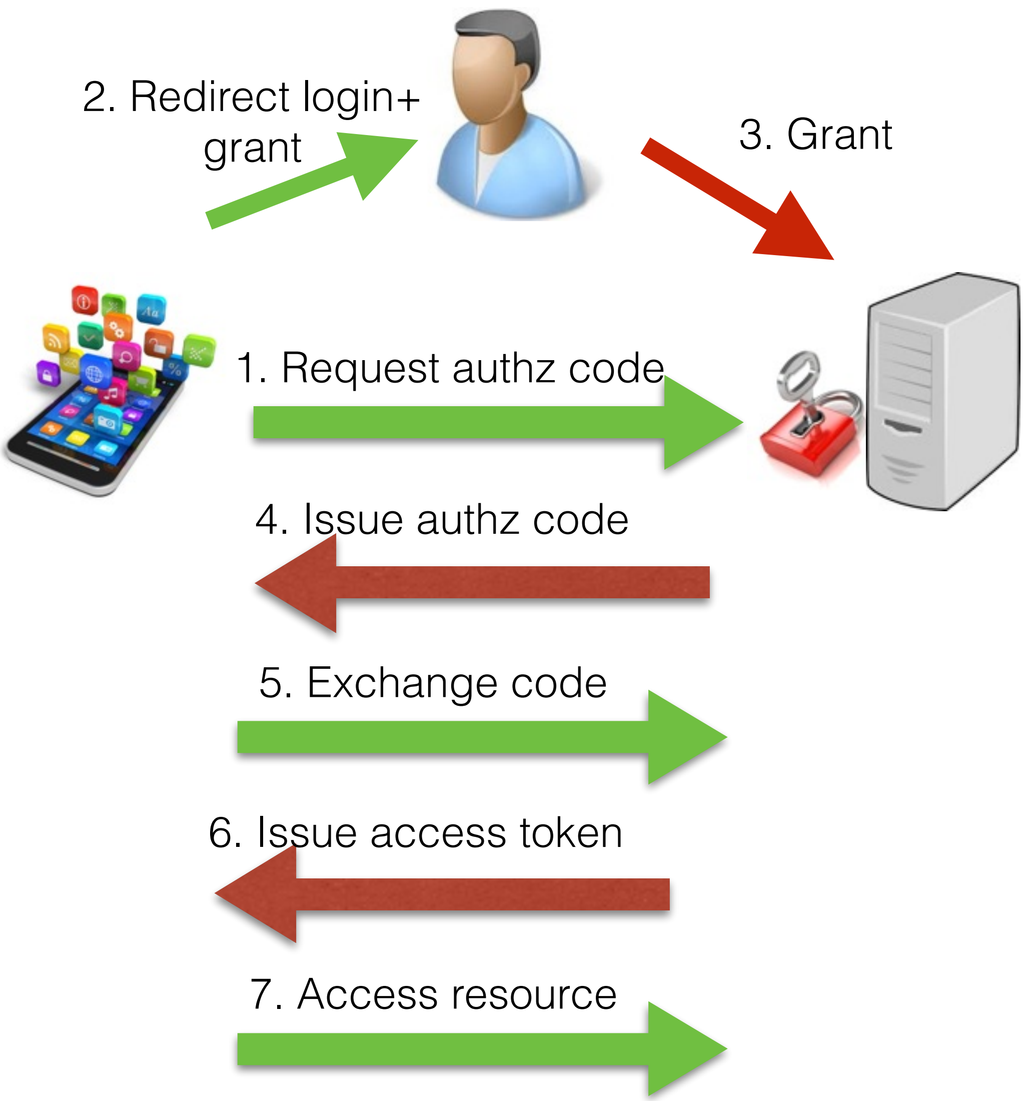


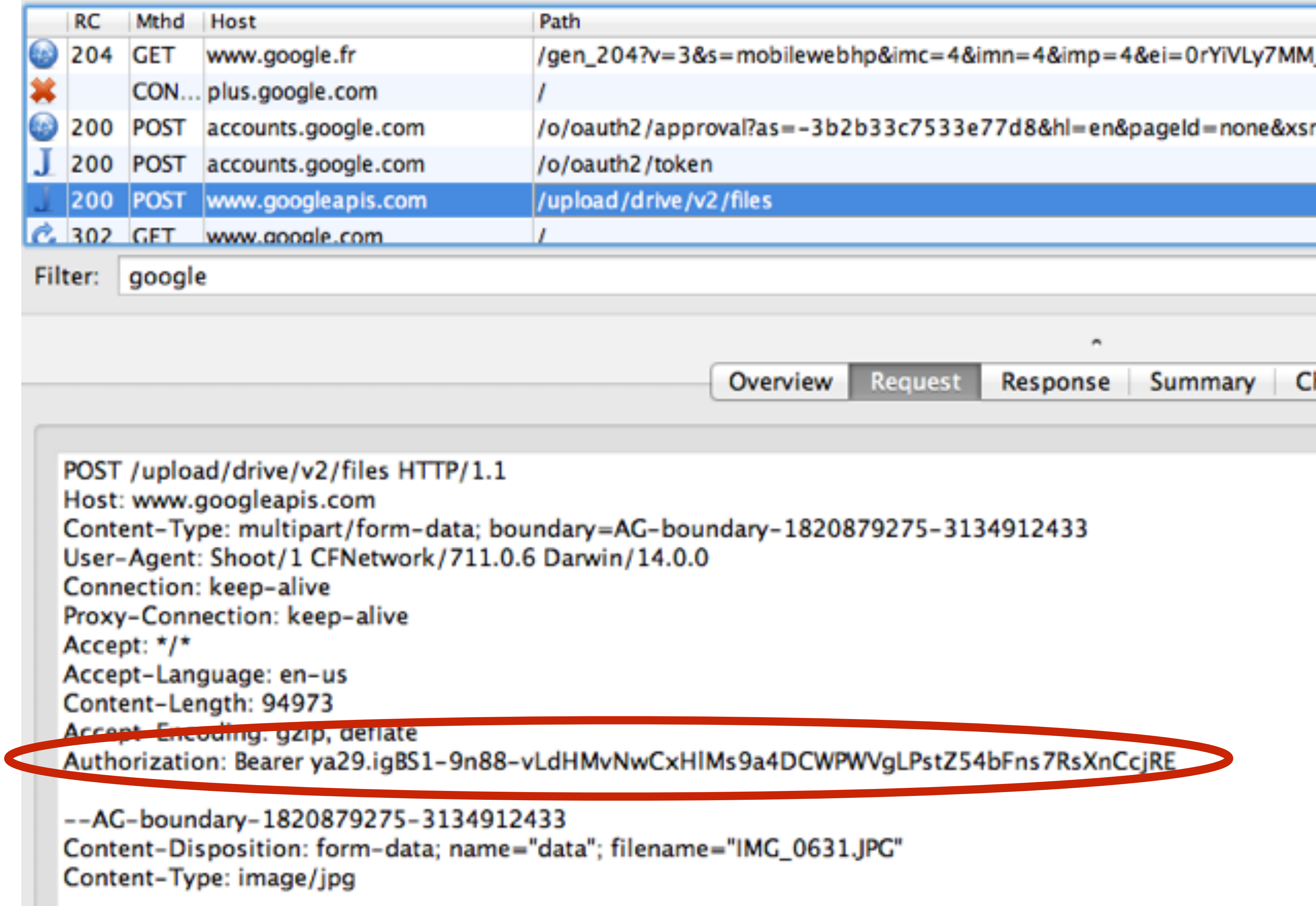
RC	Mthd	Host	Path
	CON...	p16-keyvalueservice.icloud....	
	CON...	syndication.twitter.com:443	
	CON...	plus.google.com:443	
200	POST	accounts.google.com	/o/oauth2/approval?as=-44cf14f31778fc87&hl=en&pageId=none&xsrfsign=APsBz...
200	POST	accounts.google.com	/o/oauth2/token
302	GET	www.google.com	/

```
{
    "access_token": "ya29.igBl4mAodMQ7TYS89rKHDnDKg_vSfa5aFA3DgpSlPrzDF_Xtqby6sXfK",
    "token_type": "Bearer",
    "expires_in": 3600,
    "refresh_token":
}
```

refresh - xxxxxxxxxxxxx

7: Access resource



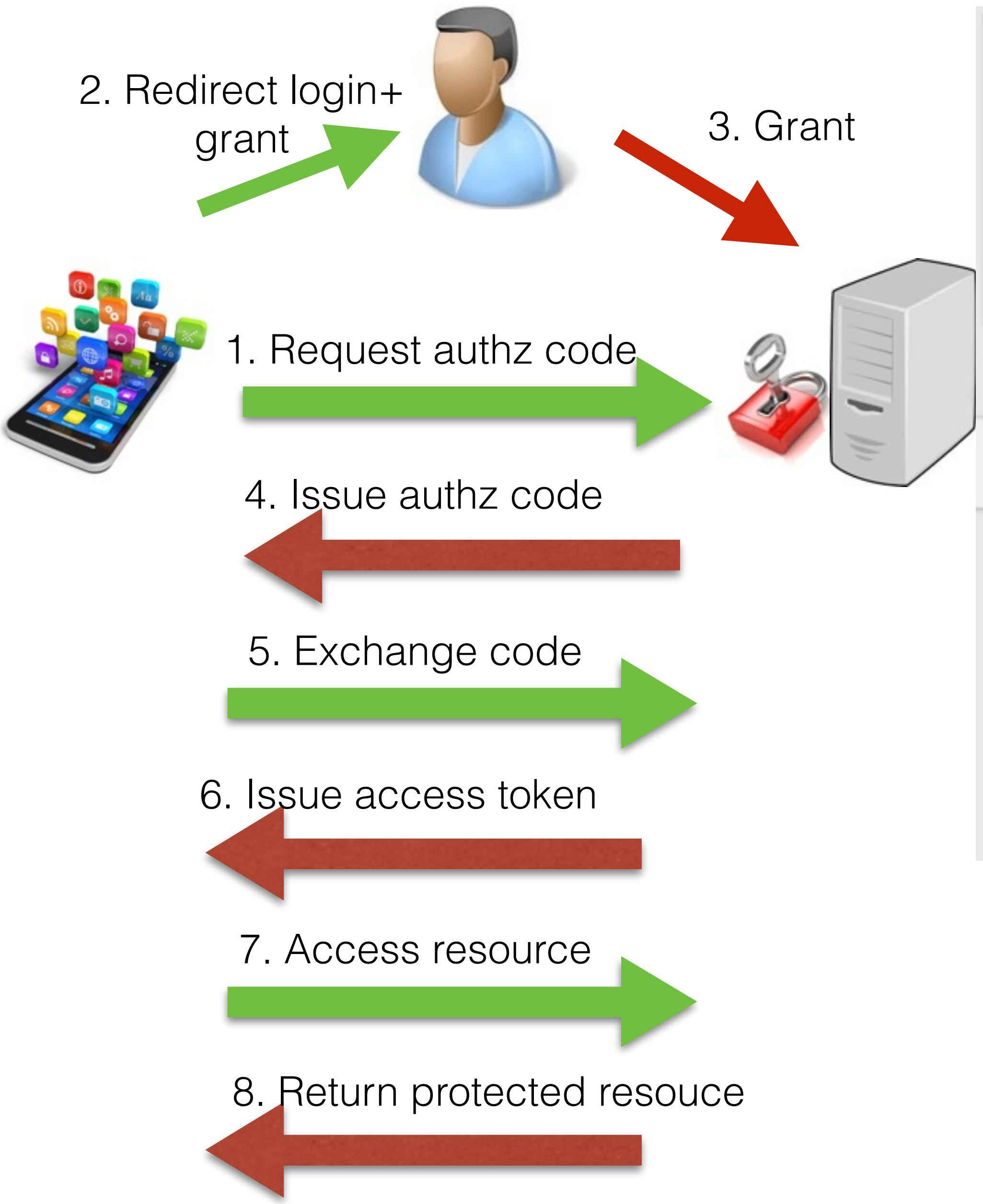


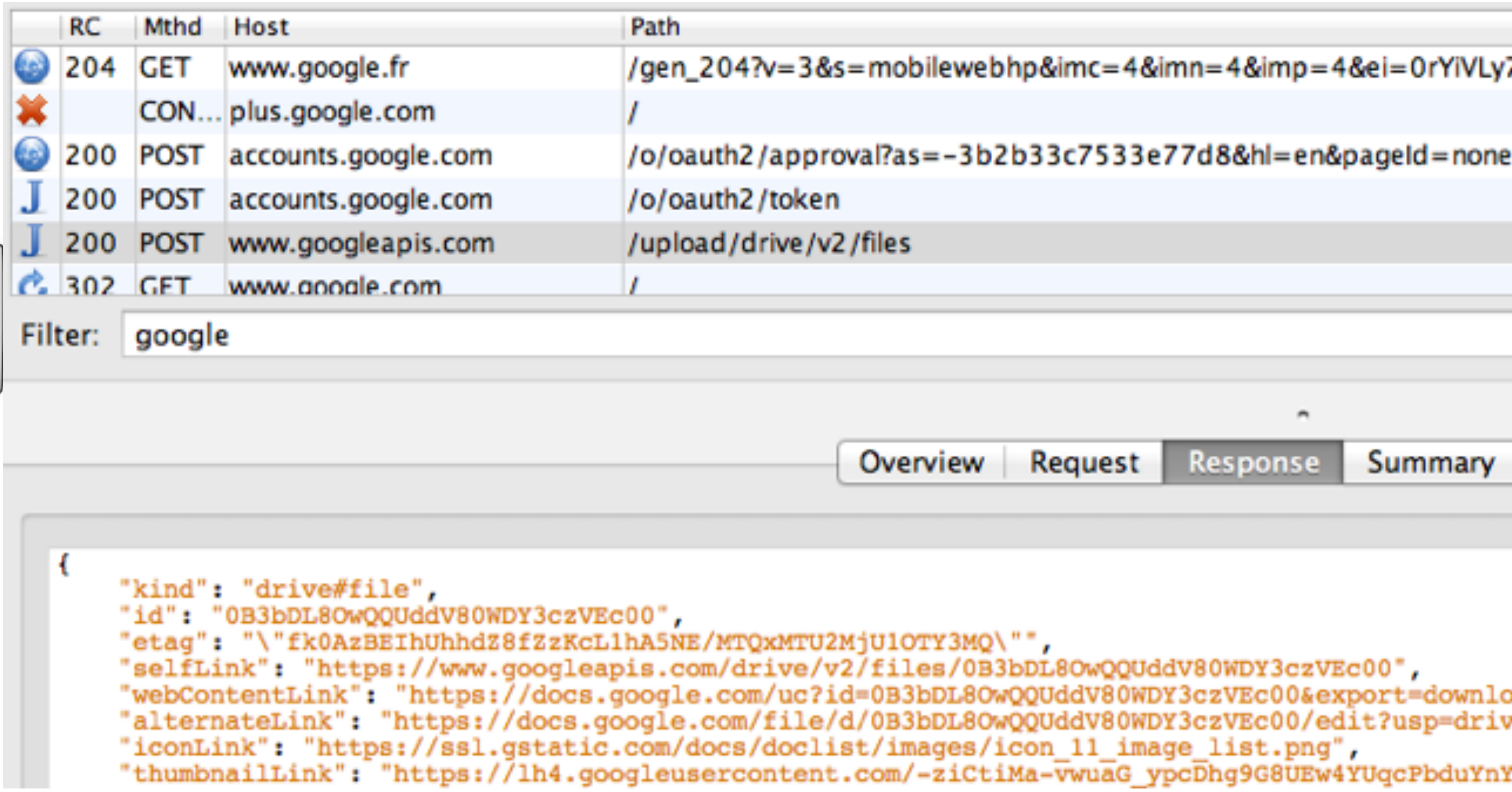
	RC	Mthd	Host	Path
	204	GET	www.google.fr	/gen_204?v=3&s=mobilewebhp&imc=4&imn=4&imp=4&ei=0rYiVLy7MM
✖		CON...	plus.google.com	/
	200	POST	accounts.google.com	/o/oauth2/approval?as=-3b2b33c7533e77d8&hl=en&pageId=none&xsr
J	200	POST	accounts.google.com	/o/oauth2/token
J	200	POST	www.googleapis.com	/upload/drive/v2/files
	302	GET	www.google.com	/

Filter: google

```
POST /upload/drive/v2/files HTTP/1.1
Host: www.googleapis.com
Content-Type: multipart/form-data; boundary=AG-boundary-1820879275-3134912433
User-Agent: Shoot/1 CFNetwork/711.0.6 Darwin/14.0.0
Connection: keep-alive
Proxy-Connection: keep-alive
Accept: */*
Accept-Language: en-us
Content-Length: 94973
Accept-Encoding: gzip, deflate
Authorization: Bearer ya29.igBS1-9n88-vLdHMvNwCxHlMs9a4DCWPWVgLPstZ54bFns7RsXnCcjRE

--AG-boundary-1820879275-3134912433
Content-Disposition: form-data; name="data"; filename="IMG_0631.JPG"
Content-Type: image/jpg
```

8: Return protected resource





	RC	Mthd	Host	Path
●	204	GET	www.google.fr	/gen_204?v=3&s=mobilewebhp&imc=4&imn=4&imp=4&ei=0rYiVLy
✖		CON...	plus.google.com	/
●	200	POST	accounts.google.com	/o/oauth2/approval?as=-3b2b33c7533e77d8&hl=en&pageId=none
J	200	POST	accounts.google.com	/o/oauth2/token
J	200	POST	www.googleapis.com	/upload/drive/v2/files
↻	302	GET	www.google.com	/

Filter: google

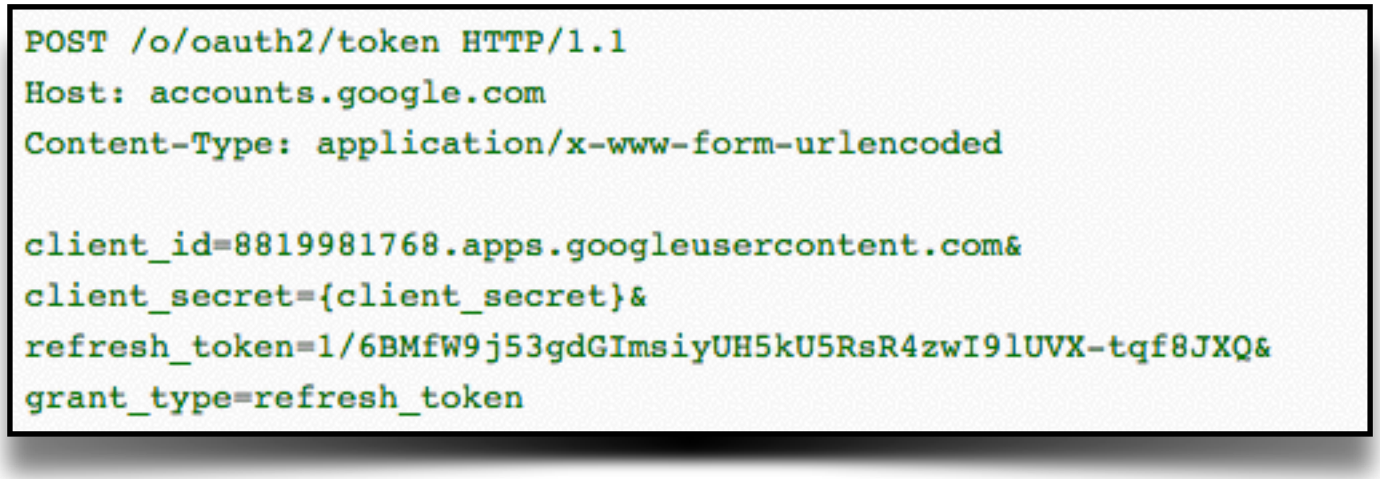
```
{
    "kind": "drive#file",
    "id": "0B3bDL8OwQQUddV80WDY3czVEc00",
    "etag": "\"fk0AzBEIhUhhdZ8fZzKcL1hA5NE/MTQxMTU2MjU1OTY3MQ\"",
    "selfLink": "https://www.googleapis.com/drive/v2/files/0B3bDL8OwQQUddV80WDY3czVEc00",
    "webContentLink": "https://docs.google.com/uc?id=0B3bDL8OwQQUddV80WDY3czVEc00&export=downlo
    "alternateLink": "https://docs.google.com/file/d/0B3bDL8OwQQUddV80WDY3czVEc00/edit?usp=driv
    "iconLink": "https://ssl.gstatic.com/docs/doclist/images/icon_11_image_list.png",
    "thumbnailLink": "https://lh4.googleusercontent.com/-ziCtiMa-vwuaG_ypcDhg9G8UEw4YUqcPbduYnY
```

As a iOS mobile user…

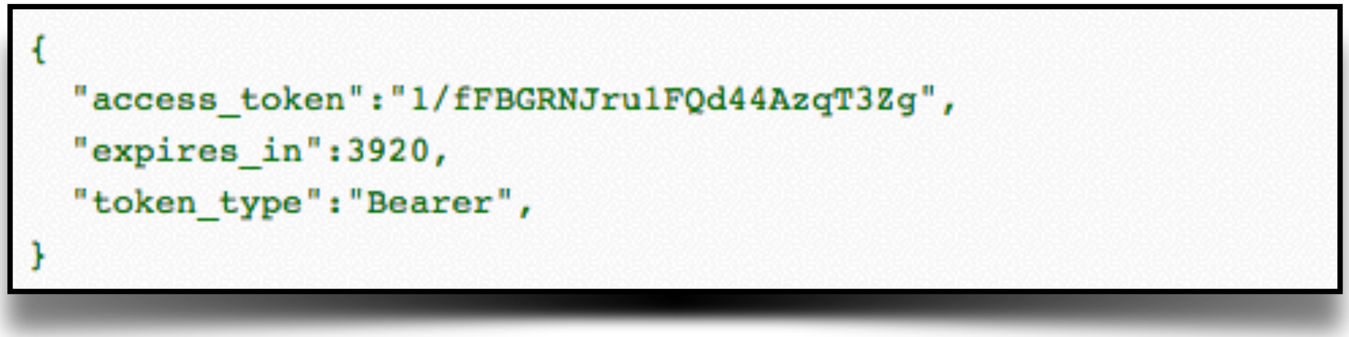
I want to be upgraded on any server side role changes so that I am up to date for my allowed grant.

Refresh token

Google Developers

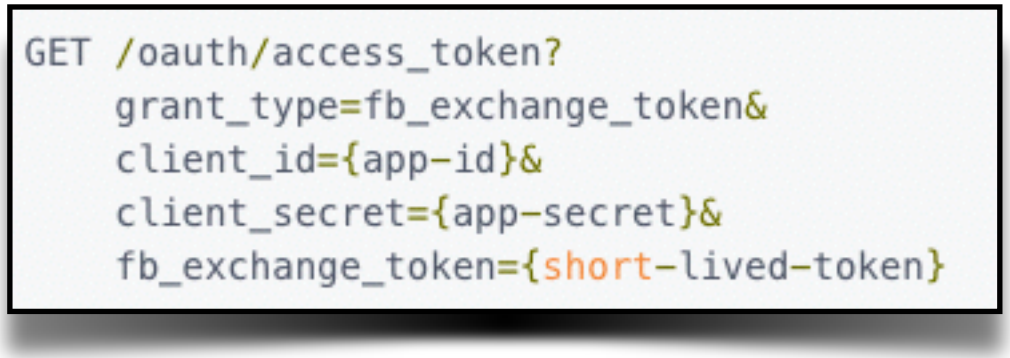


```
POST /o/oauth2/token HTTP/1.1
Host: accounts.google.com
Content-Type: application/x-www-form-urlencoded

client_id=8819981768.apps.googleusercontent.com&
client_secret={client_secret}&
refresh_token=1/6BMfW9j53gdGImsiyUH5kU5RsR4zwI9lUVX-tqf8JXQ&
grant_type=refresh_token
```

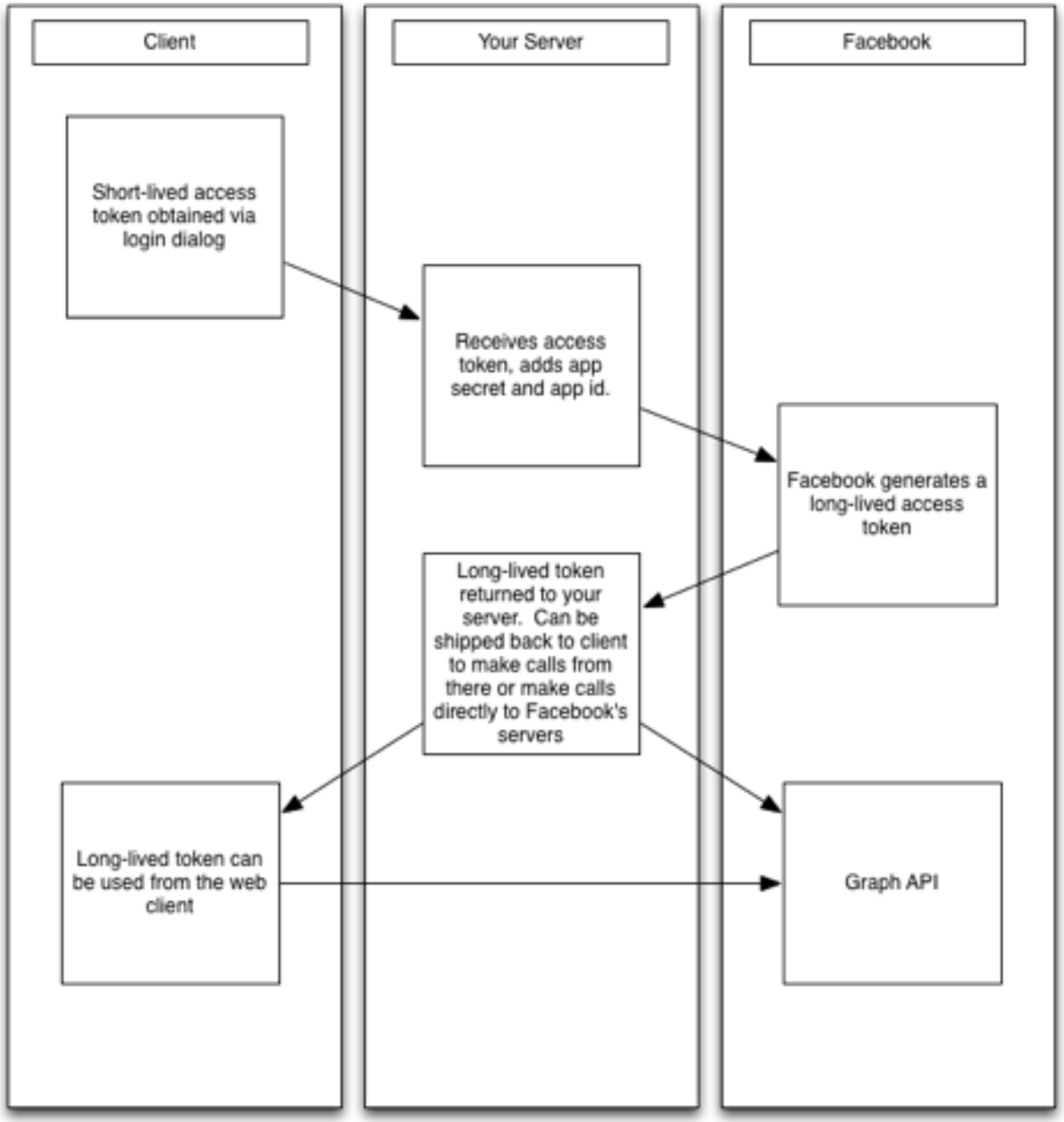


```
{
  "access_token":"1/fFBGRNJru1FQd44AzqT3Zg",
  "expires_in":3920,
  "token_type":"Bearer",
}
```

Short-lived / long-lived token



```
GET /oauth/access_token?
    grant_type=fb_exchange_token&
    client_id={app-id}&
    client_secret={app-secret}&
    fb_exchange_token={short-lived-token}
```



Client

Your Server

Facebook

Short-lived access token obtained via login dialog

Receives access token, adds app secret and app id.

Facebook generates a long-lived access token

Long-lived token returned to your server. Can be shipped back to client to make calls from there or make calls directly to Facebook's servers

Long-lived token can be used from the web client

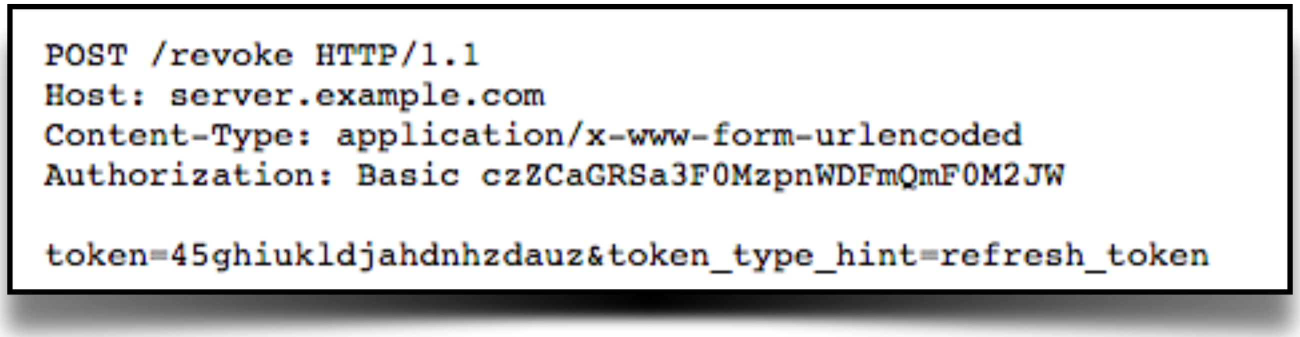
Graph API

As a iOS mobile user…

I want to keep my personal photos secure so that in the event I loose my iPhone personal photos won’t leak on social networks.

Revoke token

RFC7009: OAuth2.0 Token Revocation



```
POST /revoke HTTP/1.1
Host: server.example.com
Content-Type: application/x-www-form-urlencoded
Authorization: Basic czZCaGRSa3F0MzpnWDFmQmF0M2JW

token=45ghiukldjahdnhzdauz&token_type_hint=refresh_token
```

- Implementations MUST support the revocation of refresh tokens
- and SHOULD support the revocation of access tokens
- revocation of a particular token may cause the revocation of related tokens and the underlying authorization grant

OAuth

OAuth2

OAuth2 on native app

Authz code flow

Tokens

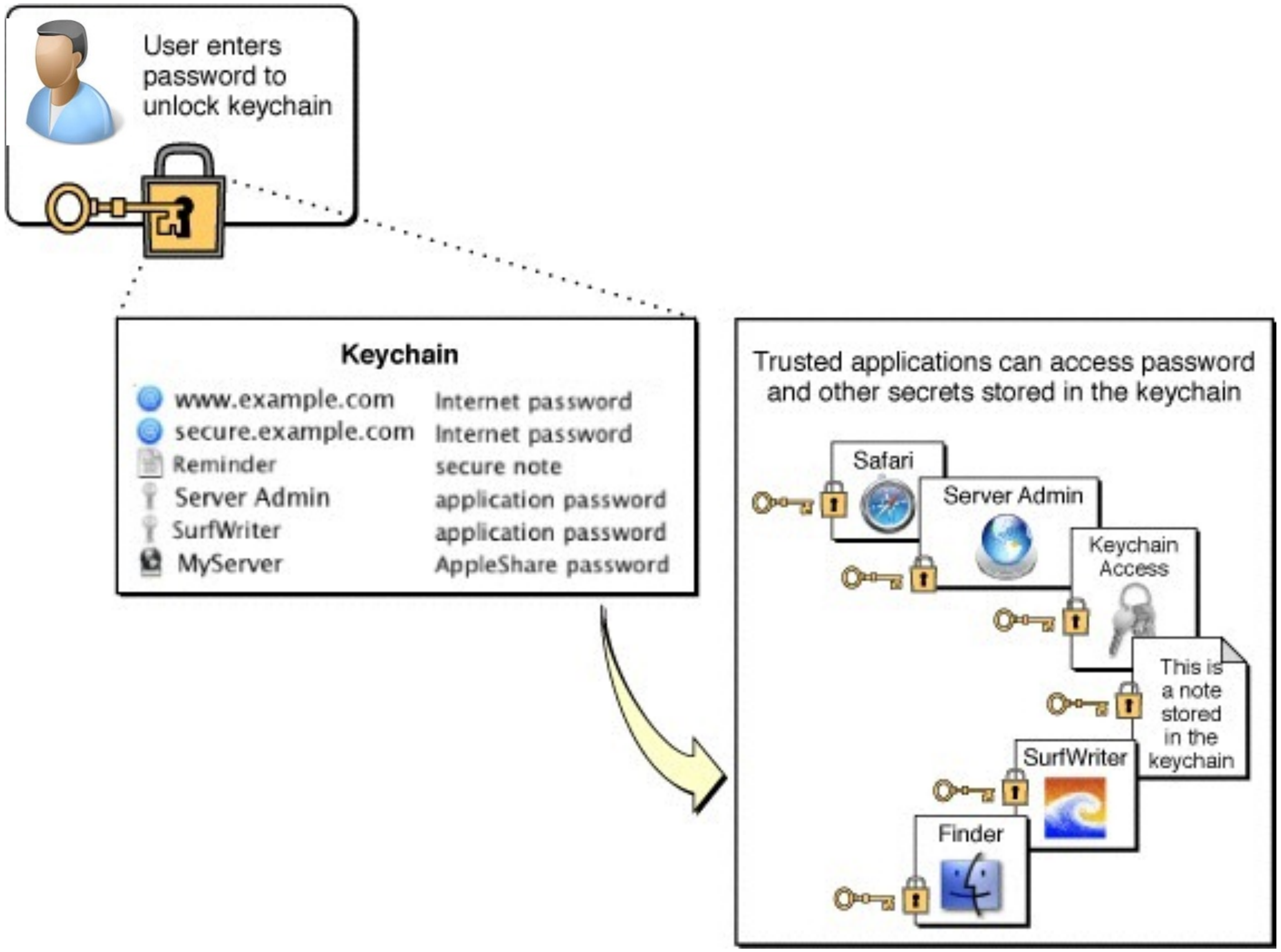
OAuth2 server Keycloak

As a iOS mobile user…

I want to grant access once at start-up
and not on successive login so
that I can easily share photos with one click.

Where to store our tokens?

iOS Keychain



User enters password to unlock keychain

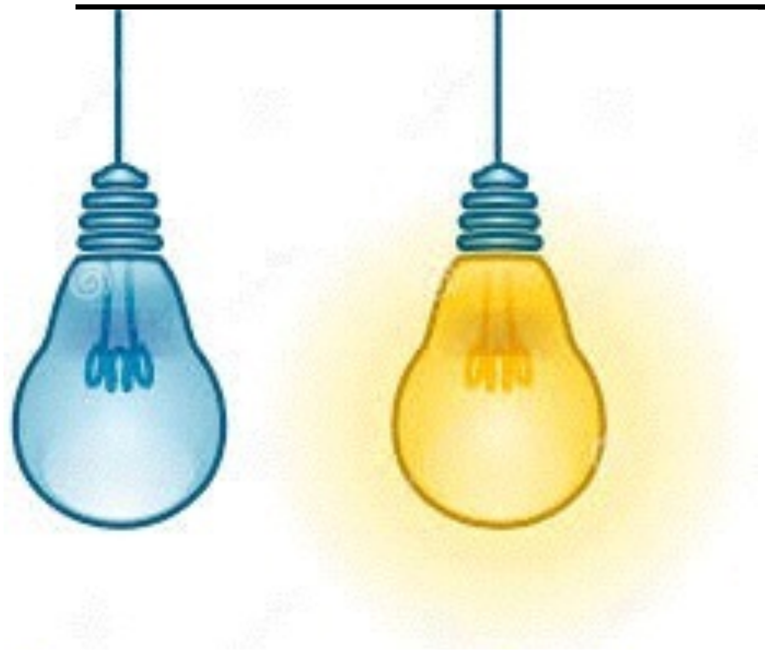
Keychain

www.example.com	Internet password
secure.example.com	Internet password
Reminder	secure note
Server Admin	application password
SurfWriter	application password
MyServer	AppleShare password

Trusted applications can access password and other secrets stored in the keychain

ACL with iOS 7/8

WhenPasscodeSet blog

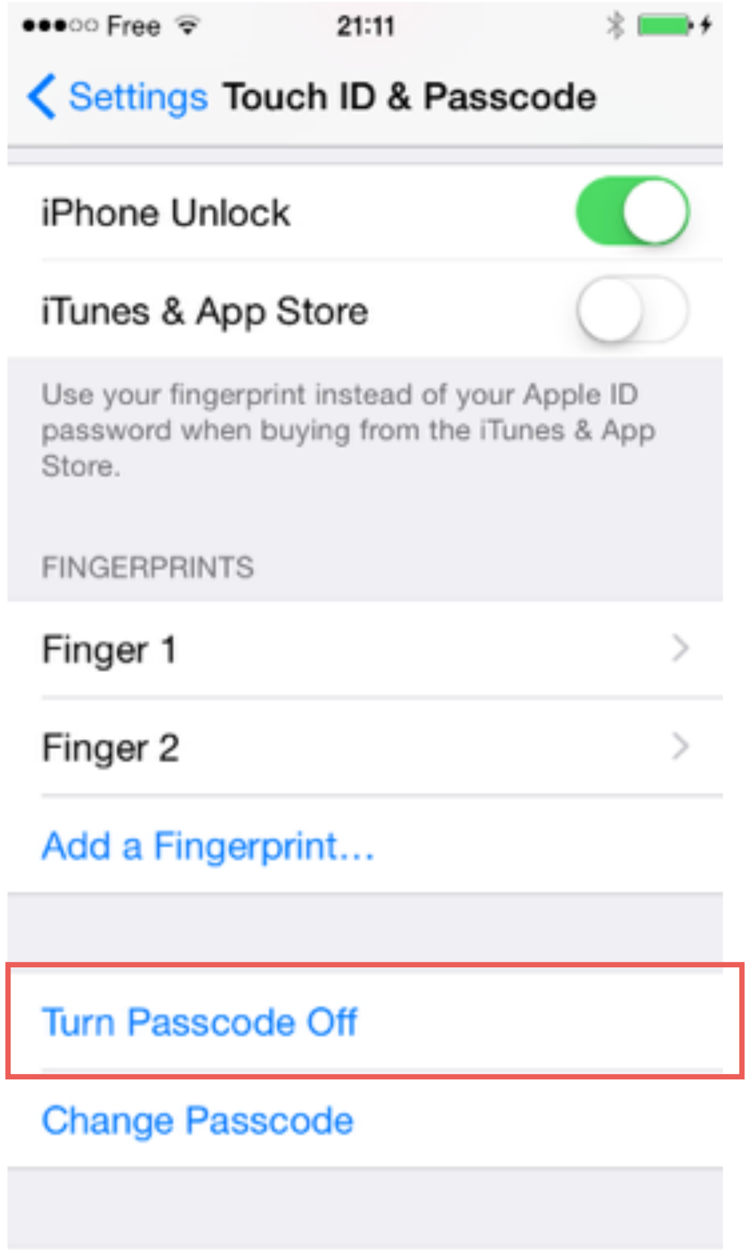


iOS 8 - Keychain with ACL
“WhenPasscodeSet”:

- 💡 when passcode is on
  - add element A
  - read element A
- 💡 when passcode is off
  - read element A => not found
  - add element B => not allowed

Secure your tokens

=



iPhone Unlock ●

iTunes & App Store ○

Use your fingerprint instead of your Apple ID password when buying from the iTunes & App Store.

FINGERPRINTS

OAuth

OAuth2

OAuth2 on native app

Authz code flow

Tokens

OAuth2 server Keycloak

As a iOS mobile user…

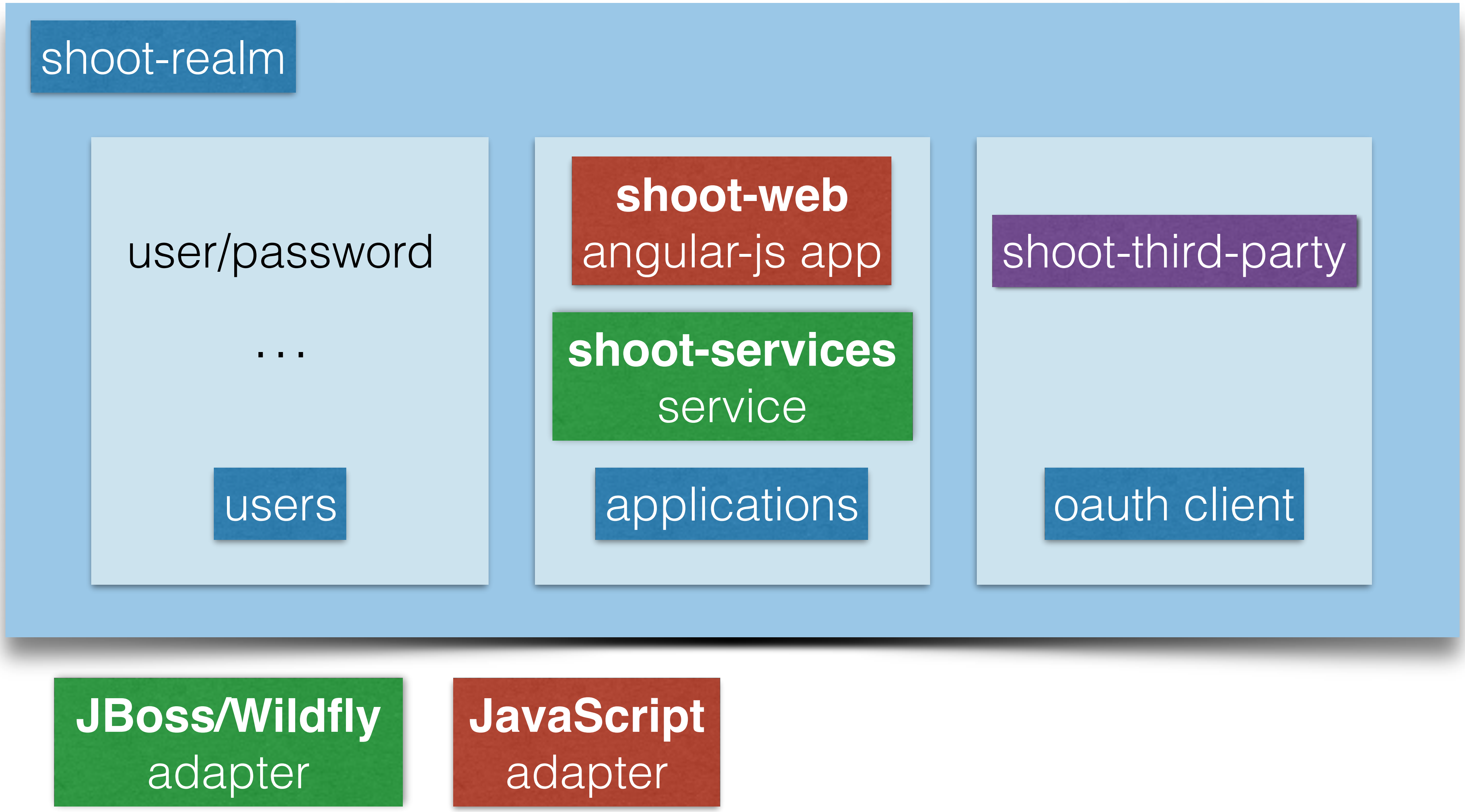
I want to share photos on ~~Facebook and upload them on Google Drive~~ my own social network so that my family, friends, coworkers, cats and dogs know all about my life.

Your own OAuth2.0 server

keycloak

Keycloak concepts

Keycloak documentation



That's all Folks!