REST

Stateless Web Services

Web Services

- Data (The Guardian API)
- Social Activity (Post to Facebook)
- Banking (Paypal transfer)
- Texting (Twilio SMS service)
- Shipping (Fedex, UPS)
- Creating/destroying computers (Amazon EC2)
- Controlling Humans (Amazon Mechanical Turk)

Web Services

- RPC/RMI
- SOAP
- REST

RPC

- Request-Reply protocol
- Assumes a common pre-negotiated interface
  - Corba; Java RMI; XML-RPC; Google Protocol Buffers
- Used within the same company/network

SOAP

- Evolution of XML-RPC
- XML payload is encapsulated within a XML request/reply
- Supports HTTP, SMTP, and other protocols
- WSDL files describe the methods, parameters and response types (RMI Interface-like), and service endpoints

Design philosophy behind Rest

- REST: Representational State Transfer
  - "Collection of architecture principles to implement web applications"
- SOAP is XML within XML; too much overhead
- KISS, Keep It Simple and Stupid
- Re-use the web as webservices

HTTP GET

GET http://todo.ideias3.com/todos
Accept: application/xml

---

200 OK
Connection: keep-alive
Content-Length: 55
Content-Type: application/xml;charset=utf-8
Date: Fri, 18 Nov 2016 10:09:00 GMT
BODY

```
<?xml version="1.0" encoding="UTF-8"?>
<todos></todos>
```

HTTP PUT

PUT http://todo.ideias3.com/todos/1
Accept: application/xml
Content-Length: 15

name=ProjectoSD
———

201 Created
Connection
Connection: keep-alive
Content-Length: 7
Content-Type: application/xml;charset=utf-8
Date: Fri, 18 Nov 2016 10:09:00 GMT
BODY
Created

HTTP GET

GET **http://todo.ideias3.com/todos**
Accept: application/xml

---

200 OK
Connection: keep-alive
Content-Length: 105
Content-Type: application/xml;charset=utf-8
Date: Fri, 18 Nov 2016 10:09:00 GMT
BODY

```
<?xml version="1.0" encoding="UTF-8"?>
<todos>
  <todo id="1" completed="false">ProjectoSD</todo>
</todos>
```

HTTP POST

POST http://todo.ideias3.com/todos/1
Accept: application/xml
Content-Length: 15

```
completed=true
```

---

200 OK
Connection
Connection: keep-alive
Content-Length: 7
Content-Type: application/xml;charset=utf-8
Date: Fri, 18 Nov 2016 10:09:00 GMT
BODY
Updated

HTTP GET

GET **http://todo.ideias3.com/todos**
Accept: application/xml

---

200 OK
Connection: keep-alive
Content-Length: 104
Content-Type: application/xml;charset=utf-8
Date: Fri, 18 Nov 2016 10:09:00 GMT
BODY

```
<?xml version="1.0" encoding="UTF-8"?>
<todos>
  <todo id="1" completed="true">ProjectoSD</todo>
</todos>
```

HTTP DELETE

DELETE http://todo.ideias3.com/todos/**1**
Accept: application/xml

---

200 OK
Connection
Connection: keep-alive
Content-Length: 7
Content-Type: application/xml;charset=utf-8
Date: Fri, 18 Nov 2016 10:09:00 GMT
BODY
`Deleted`

HTTP GET

GET http://todo.ideias3.com/todos
Accept: application/xml

---

200 OK
Connection: keep-alive
Content-Length: 55
Content-Type: application/xml;charset=utf-8
Date: Fri, 18 Nov 2016 10:09:00 GMT
BODY

```
<?xml version="1.0" encoding="UTF-8"?>
<todos></todos>
```

HTTP GET

GET **http://todo.ideias3.com/todos**
Accept: application/json

---

200 OK
Connection: keep-alive
Content-Length: 135
Content-Type: application/json;charset=utf-8
Date: Fri, 18 Nov 2016 10:09:00 GMT
BODY


```
{"todos": [
     {
        "id":"1",
        "name":"Hello",
        "completed":"false"
    },  {
        "id":"2",
        "name":"Bye",
        "completed":"false"
    },
]}
```

REST

- Uses HTTP as the transport mechanism
- Resource-oriented: URLs define resources, VERBs describe operations on the resources
- Agnostic to the payload format (XML, JSON, CSV, HTML, YAML, PDF, DOCX, JPG, MP3)

REST / HTTP

- Client-Server model (similar to RPC)
- Stateless
  - Each request should keep its own state
  - Shift load from the server to the client
  - Scalable
- Cacheable (all HTTP optimizations apply)

REST VERB

- GET HTTP://LOCATION/RESOURCE - read resource
- PUT HTTP://LOCATION/RESOURCE - create resource at that address, overriding existing data
- POST HTTP://LOCATION/RESOURCE - modifies resource
- DELETE HTTP://LOCATION/RESOURCE - destroys resource

Q: REST vs SOAP

- REST when stateless (memory usage limited, more scalable)
- REST has low serialization overheads (less bandwidth)
- SOAP has WSDL descriptions, used to automate web services programming (enterprise)
- SOAP supports asynchronous processing

Q: Callbacks in REST

- Web Hooks
  - Client tells server to REST him at the resource http://client_ip/callback_resource
  - Requires client port to be open to the internet

OAuth

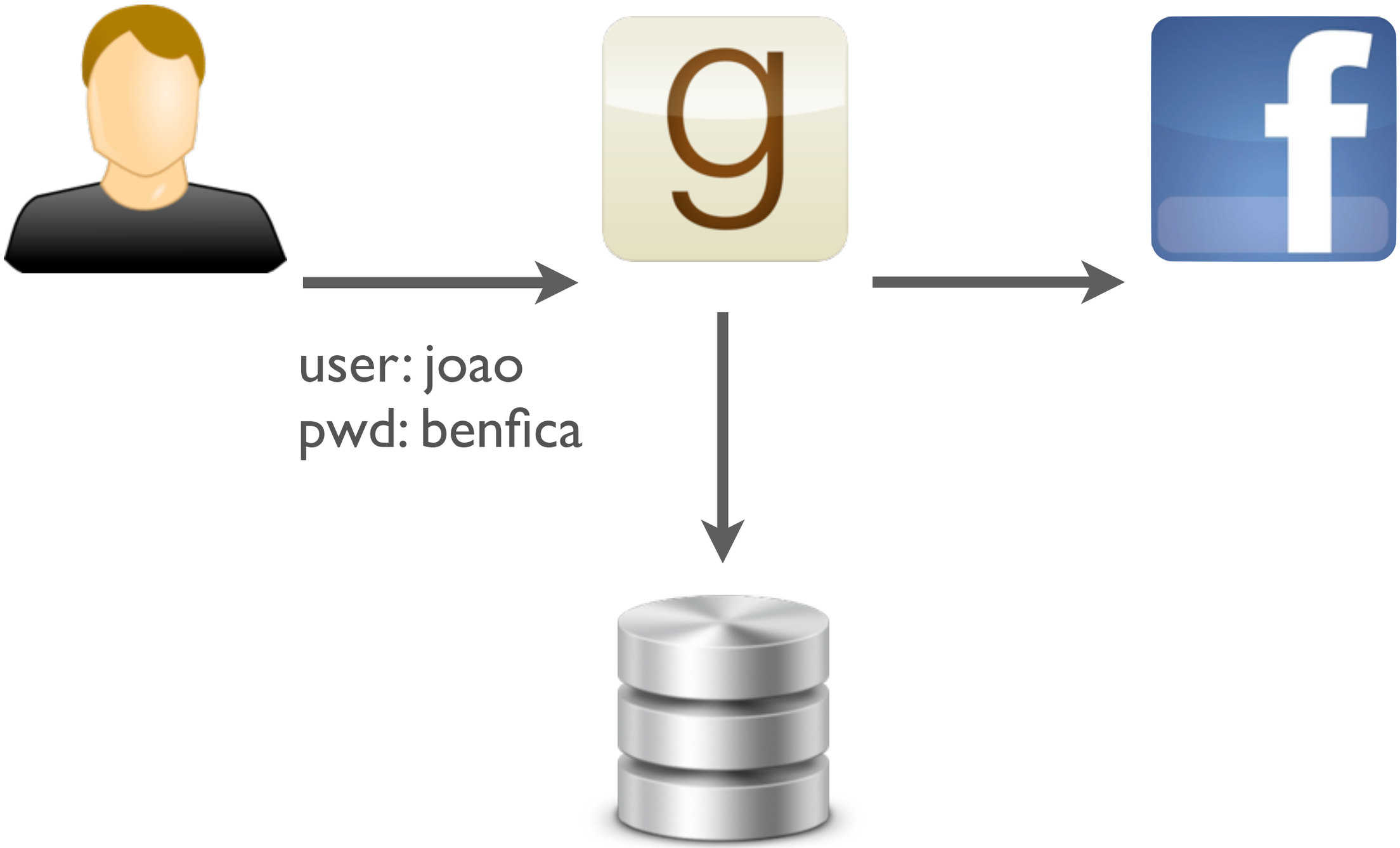
Authorization for Web Services

João

GoodReads

Facebook

João acabou de ler 1984, e atribuiu-lhe uma classificação de 5 estrelas.


user: joao
pwd: benfica

Scenarios

- Goodreads is sold to a Nigerian prince, starts spamming your friends, and they block João. João has no friends now :(
- Goodreads is sold to blackmailers that use private information and photos.
- Goodreads starts charging $1 for each book you are recommended, automatically.
- Goodreads is hacked. They do all of the above, and they change the password. João had his identity stolen, and cannot do anything about it.

Real-world Scenarios

In fact, the passwords and usernames *had* been wiggled out, but not from Dropbox. Rather, they were stolen from third-party services in previous attacks that happened "some time" ago, the company said.

Dropbox told users to change their passwords after detecting suspicious activity and told The Next Web that this all went down months ago.

Here's the statement form Dropbox:

> Dropbox has not been hacked. These usernames and passwords were unfortunately stolen from other services and used in attempts to log in to Dropbox accounts.
>
> We'd previously detected these attacks and the vast majority of the passwords posted have been expired for some time now. All other remaining passwords have been expired as well.

In fact, an attack against a SnapChat third-party service, SnapSaved (which promised to do exactly that: save the supposedly disappearing images before they disappeared), was how hundreds of thousands of SnapChat images wound up bobbing up on the internet last week.

https://nakedsecurity.sophos.com/2014/10/14/dropbox-passwords-leaked-third-party-services-blamed/

Bonus Scenario

- João worries about all of this, and he updates his password frequently. Every time he does so, he has to go to Goodreads and all other apps that use Facebook to update his password there.

OAuth

- Password are only entered at http://facebook.com/ (OAuth Provider)
  - HTTP Redirects are used to streamline the process
- Goodreads (OAuth Consumer) has an unique APP_ID, provided by Facebook (can be revoked)
- João gives Goodreads an ACCESS_TOKEN that is valid for use in Facebook (can be revoked)
- João can select partial permissions (Post to Timeline, No friends list, No credit card)

Step 0

GoodReads Developer registers

for a App Key and App Secret at

developers.facebook.com

(RunKeeper would receive a different pair

App Key/Secret)

Apps ▸ RestSD ▸ Permissions

Configuring Permissions

Use these settings to enter the set of permissions your app requires when displayed in App Center

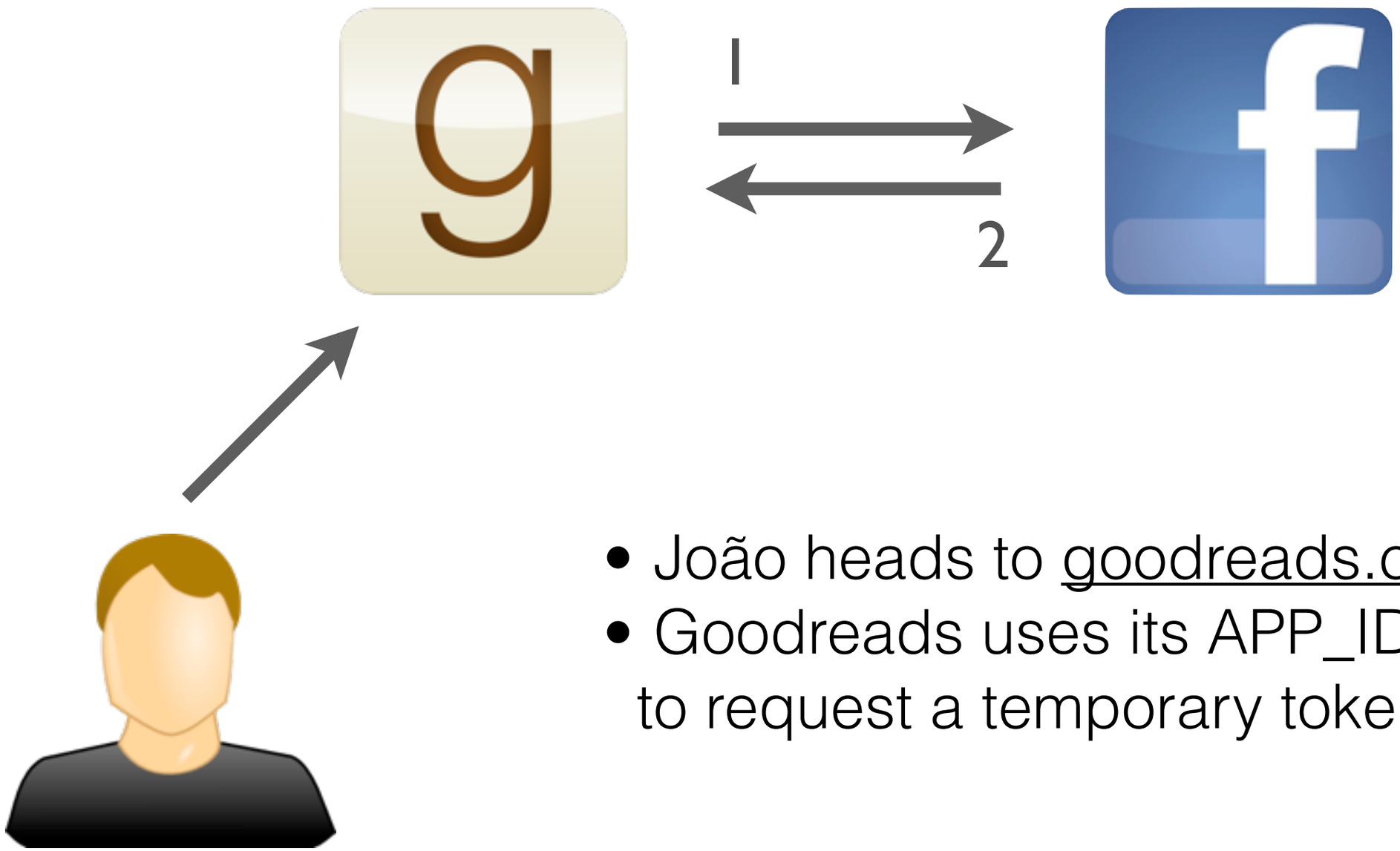
Learn more about Configuring Permissions

Configure your permissions

Default Activity Privacy: [?]	None (User Default)
User & Friend Permissions: [?]	read_friendlists × read_stream ×
Extended Permissions: [?]	publish_actions ×
Auth Token Parameter: [?]	Query String (?code=...)

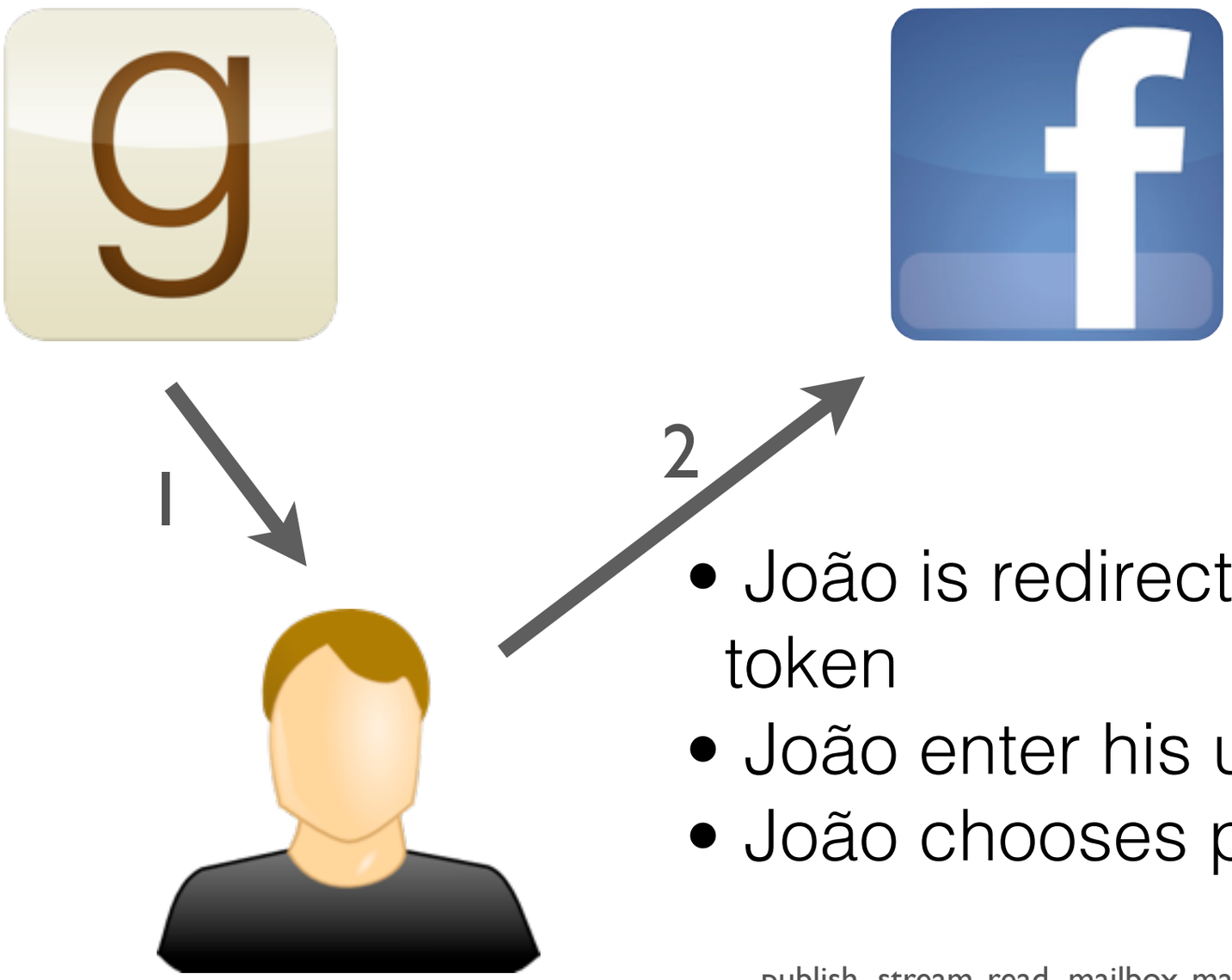
Save Changes

Step 1



- João heads to goodreads.com
- Goodreads uses its APP_ID/SECRET to request a temporary token from FB

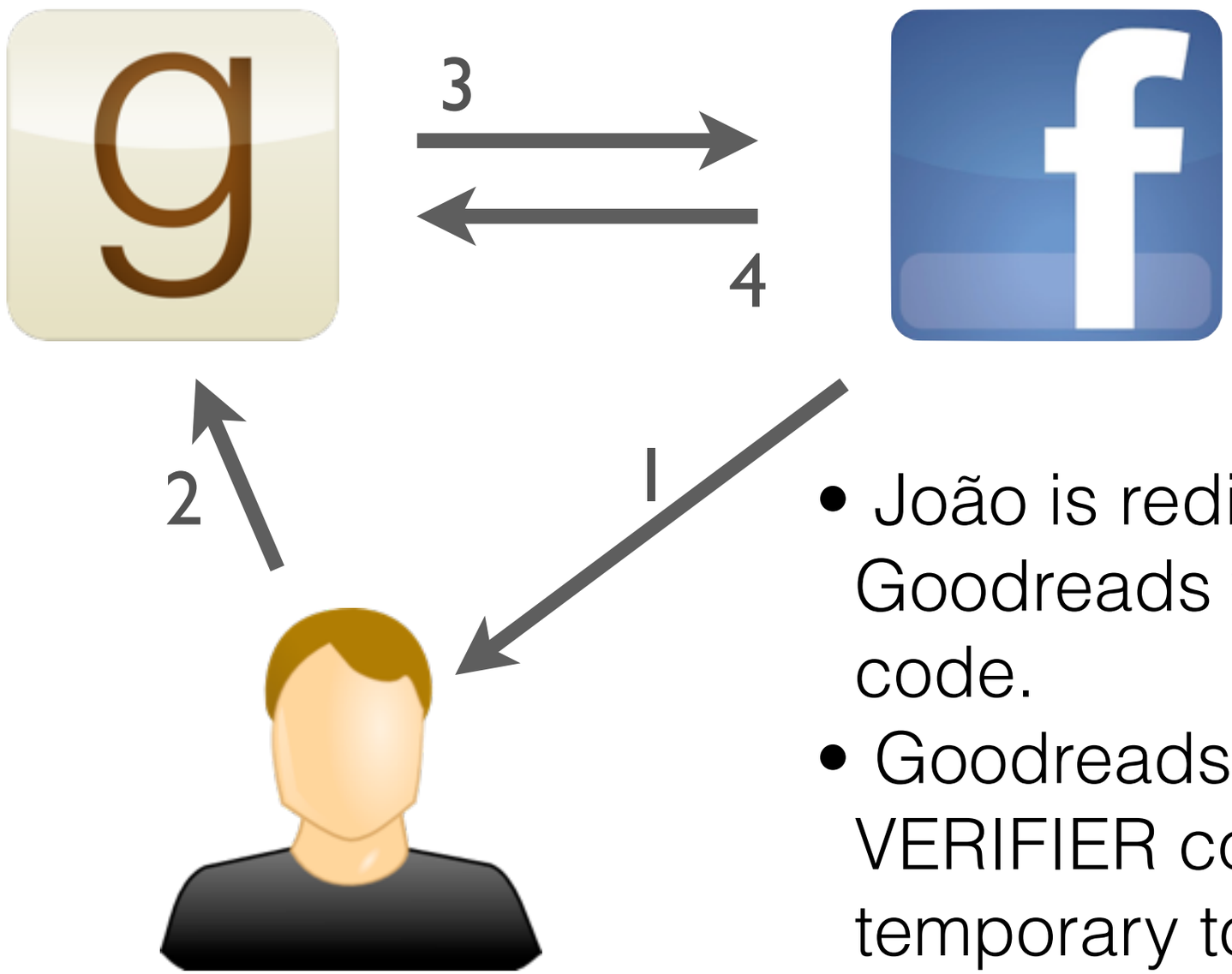
Step 2



- João is redirected to fb.com, with the token
- João enter his user/pass in fb.com
- João chooses permissions

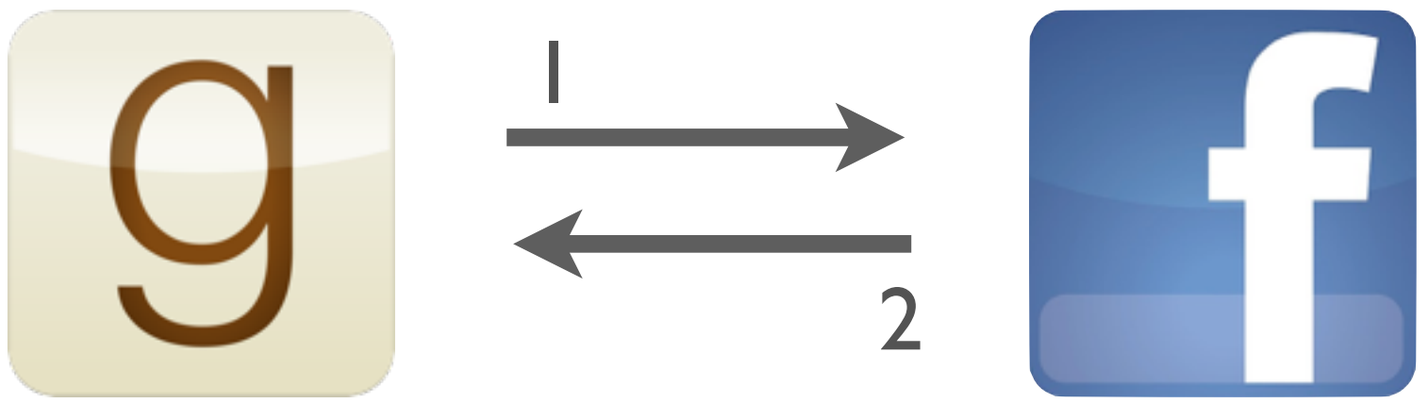
publish_stream, read_mailbox, manage_friendlists, etc...

Step 3



- João is redirected to Goodreads with a VERIFIER code.
- Goodreads uses the VERIFIER code with the temporary token to obtain an ACCESS_TOKEN.

Step 4, 5, ...



- All Goodreads requests carry the ACESS_TOKEN to prove they were authorized

OAuth advantages

- Passwords cannot be stolen
- Consumers cannot lock you out
- Consumers can be blocked by the Provider (revoking APP_ID/SECRET)
- Consumers can be blocked by the User (revoking ACCESS_TOKEN)

Search for people, places and things

Home 9 Alcides

BillPin	Only Me	Edit ×
ZZKKO.COM	Only Me	Edit ×
Doodle Numbers	Only Me	Edit ×
Goodreads	Last logged in: Less than 24 hours ago	Close

Visibility of app: [?] Custom

This app needs:
- Your basic info [?]
- Your email address (alcidesfonseca@gmail.com)
- Your likes
- Friends' likes

This app can also: **Post on your behalf** ×
This app may post on your behalf, including books you rated, books you wanted to read and more.

Last data access: Basic Information Nov 12
See details · Learn more

When to notify you? The app sends you a notification

Legal: Privacy Policy · Terms of Service

Remove app · Report app

SlideShare	Custom	Edit ×
TasteKid	Custom	Edit ×
Udemy	Custom	Edit ×

OAuth advantages (II)

- Scopes allow partial access (needs good design and implementation!)
- Agnostic to the cryptographic protocols

OAuth in the wild

- Dropbox
- Facebook
- Github
- Google
- Foursquare
- SalesForce
- Citrix
- Twitter
- Windows Live
- Many others

Q: What can you do with REST?