Token-based Authentication – Part 2

Dominick Baier
http://leastprivilege.com
@leastprivilege

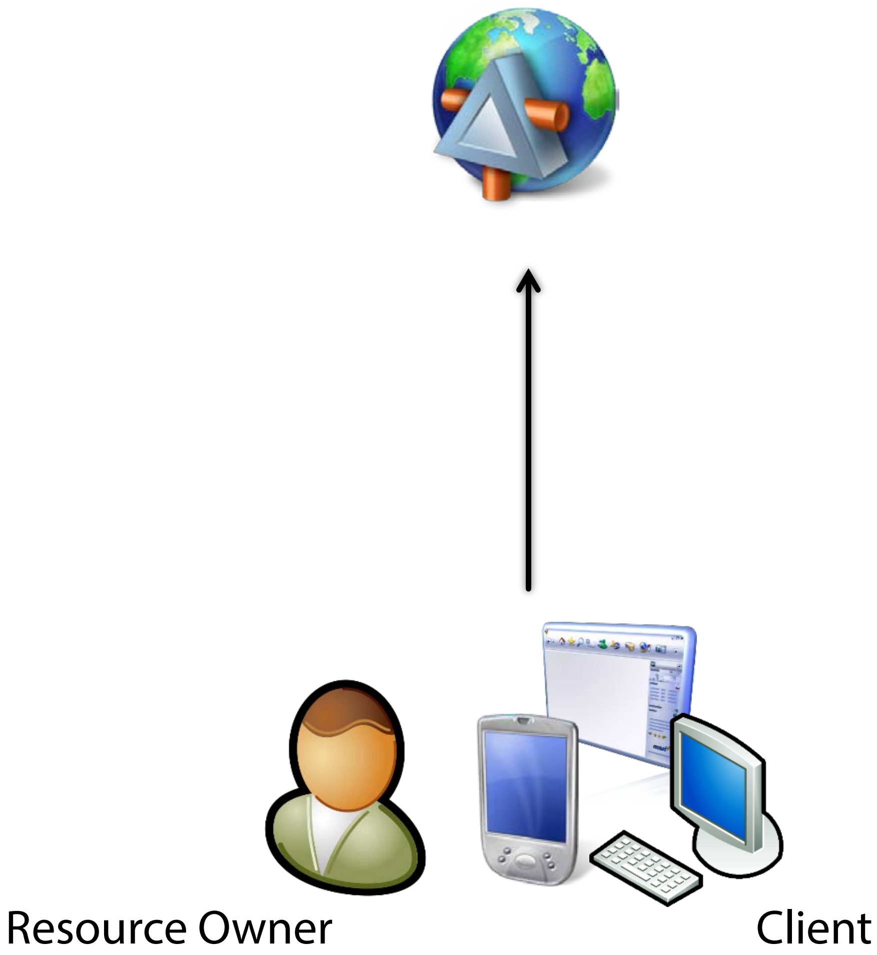
pluralsight
hardcore dev and IT training

Separating User Credentials From the Client…

- **Local / mobile / user-agent based clients**
  - Implicit Flow

- **Server-based / confidential clients**
  - Authorization Code Flow

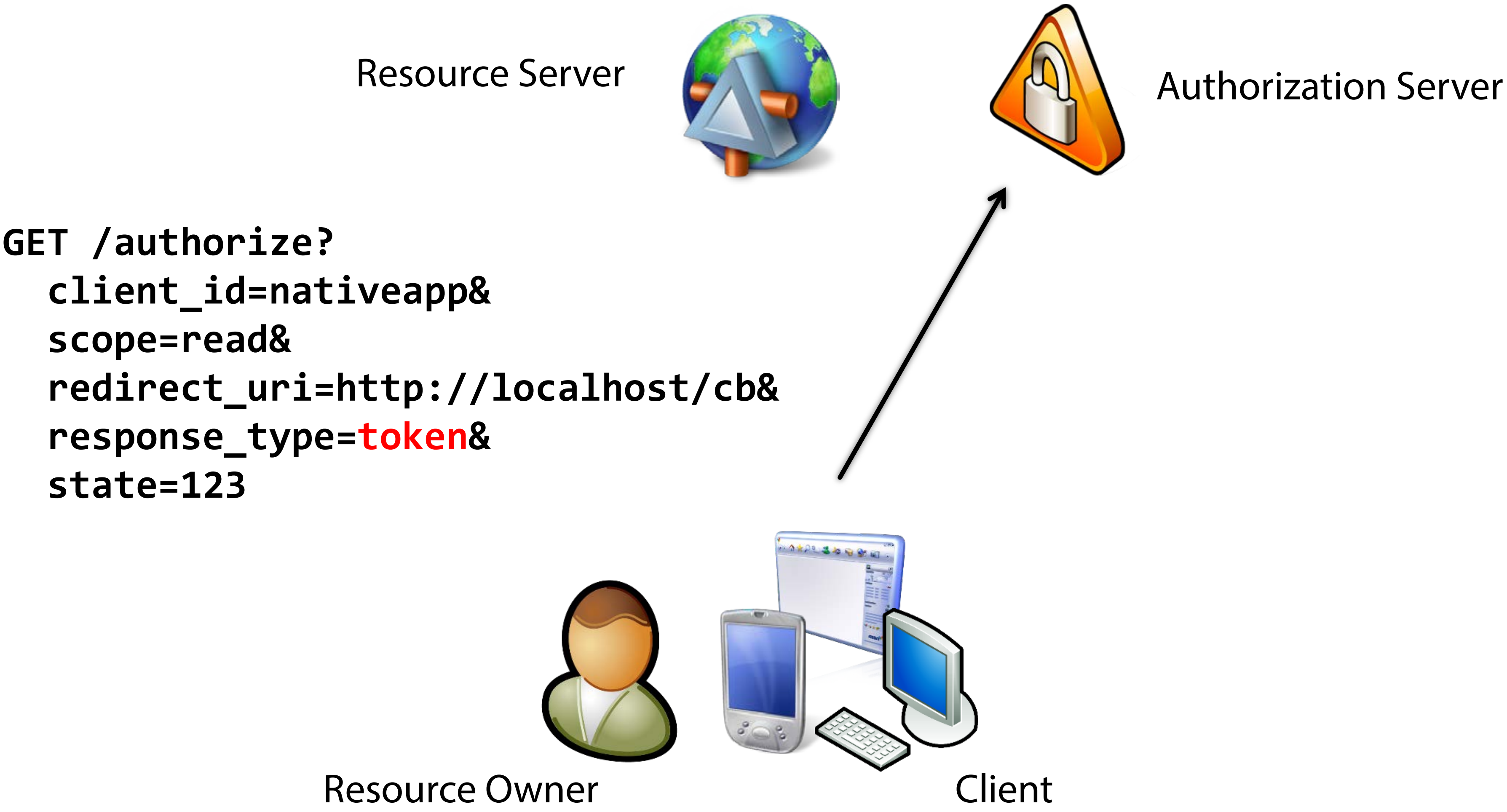
Implicit Flow (Native / Local Clients)



Resource Owner

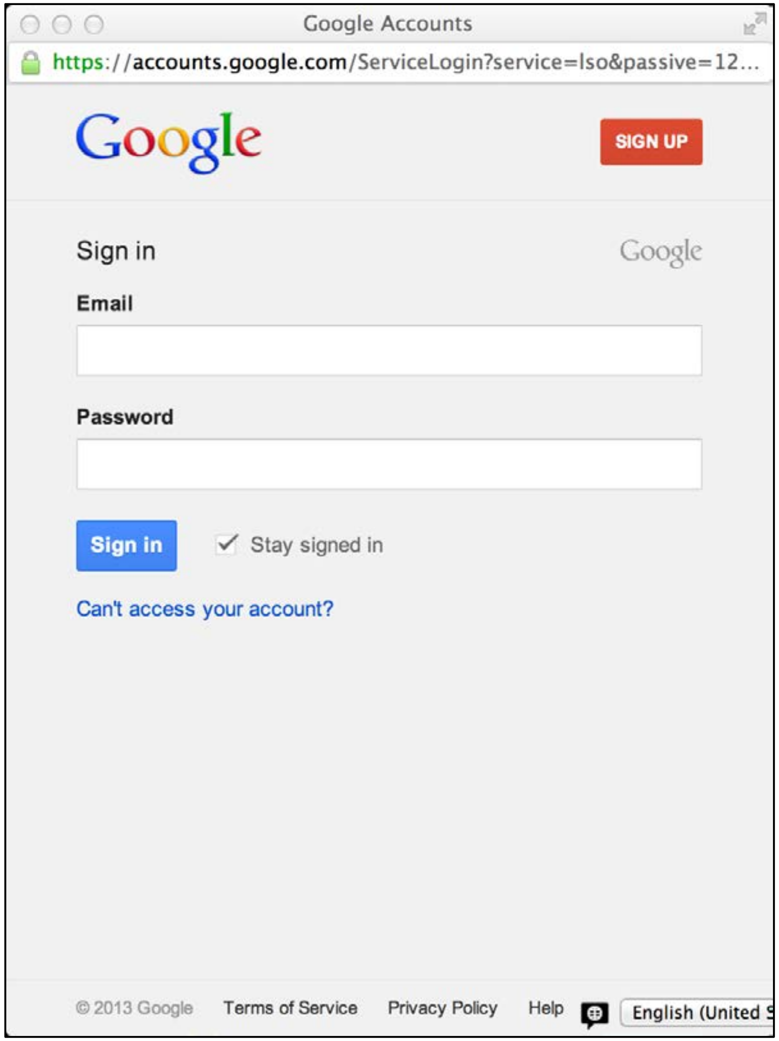
Client

Step 1a: Authorization Request



```
GET /authorize?
  client_id=nativeapp&
  scope=read&
  redirect_uri=http://localhost/cb&
  response_type=token&
  state=123
```

Step 1b: Authentication



Google Accounts

https://accounts.google.com/ServiceLogin?service=lso&passive=12...

Sign in

Email

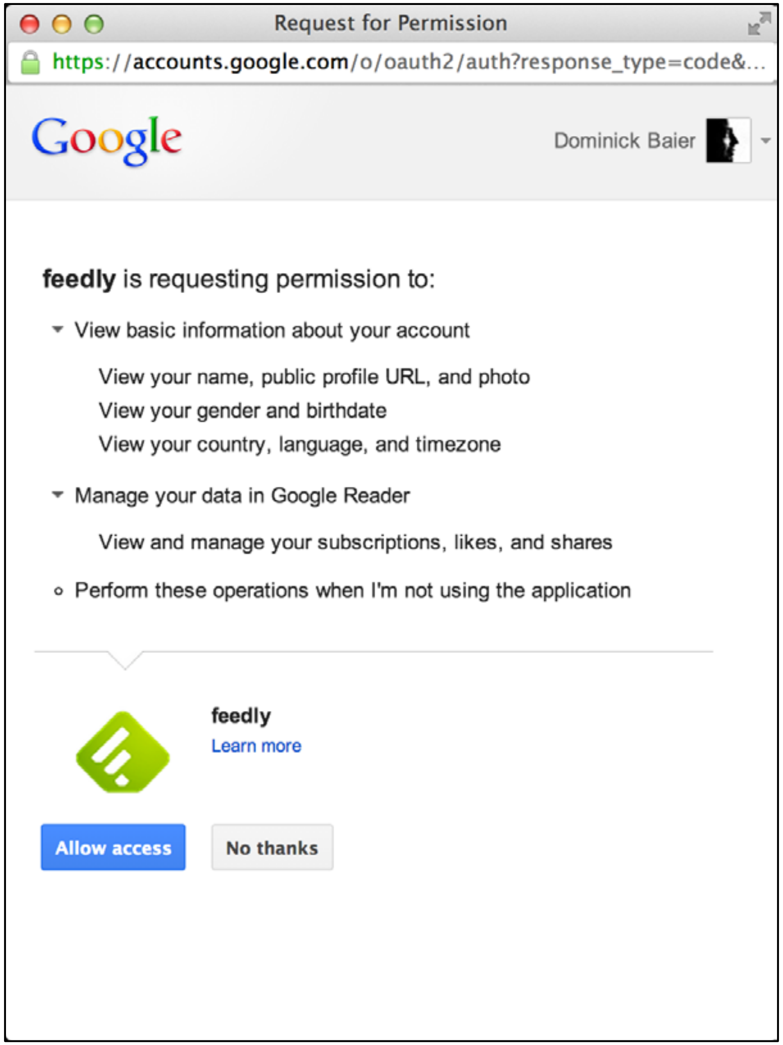
Password

☑ Stay signed in

Can't access your account?

© 2013 Google

Step 1c: Consent



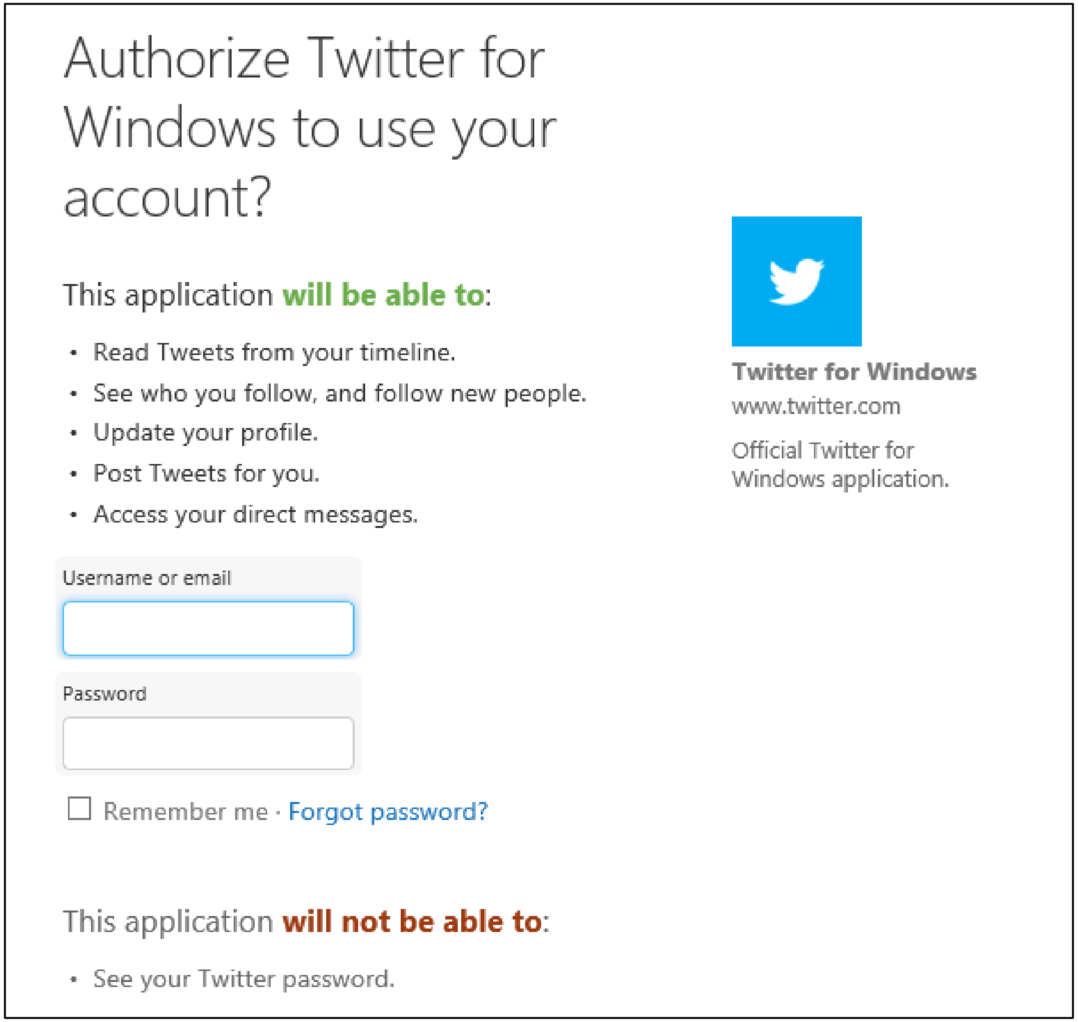
Request for Permission
https://accounts.google.com/o/oauth2/auth?response_type=code&...
Google
Dominick Baier

feedly is requesting permission to:

- View basic information about your account
  - View your name, public profile URL, and photo
  - View your gender and birthdate
  - View your country, language, and timezone
- Manage your data in Google Reader
  - View and manage your subscriptions, likes, and shares
- Perform these operations when I'm not using the application

feedly

Twitter Consent



Authorize Twitter for Windows to use your account?

This application **will be able to**:

- Read Tweets from your timeline.
- See who you follow, and follow new people.
- Update your profile.
- Post Tweets for you.
- Access your direct messages.

Username or email

Password

□ Remember me · Forgot password?

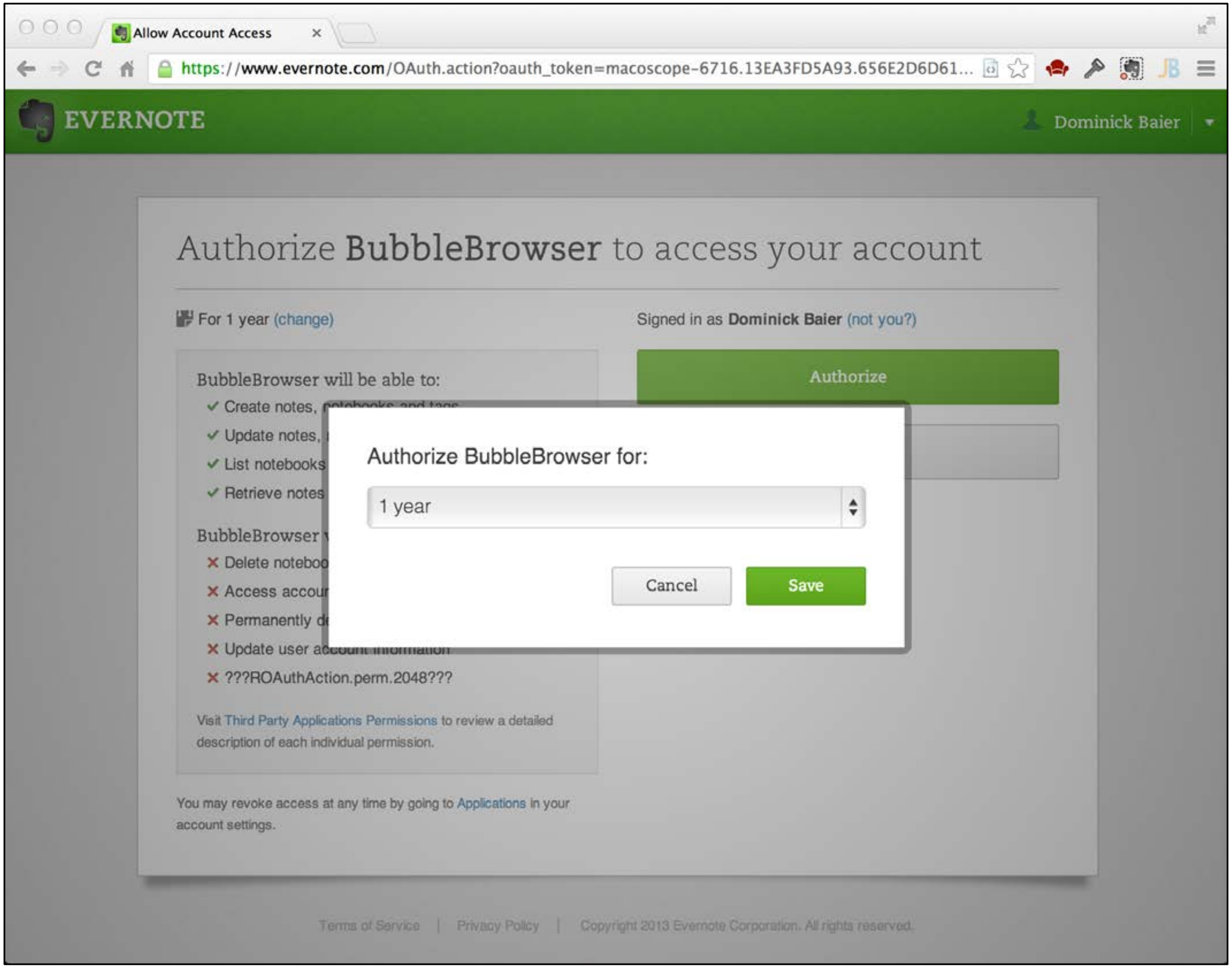
This application **will not be able to**:

- See your Twitter password.

Twitter for Windows
www.twitter.com

Official Twitter for Windows application.

Evernote Consent



Allow Account Access
https://www.evernote.com/OAuth.action?oauth_token=macoscope-6716.13EA3FD5A93.656E2D6D61...
EVERNOTE
Dominick Baier
Authorize
Cancel
Save

Authorize **BubbleBrowser** to access your account

For 1 year (change)

Signed in as **Dominick Baier** (not you?)

BubbleBrowser will be able to:

- ✓ Create notes, notebooks and tags
- ✓ Update notes,
- ✓ List notebooks
- ✓ Retrieve notes

BubbleBrowser

- ✗ Delete notebo
- ✗ Access accou
- ✗ Permanently d
- ✗ Update user account information
- ✗ ???ROAuthAction.perm.2048???

Visit Third Party Applications Permissions to review a detailed description of each individual permission.

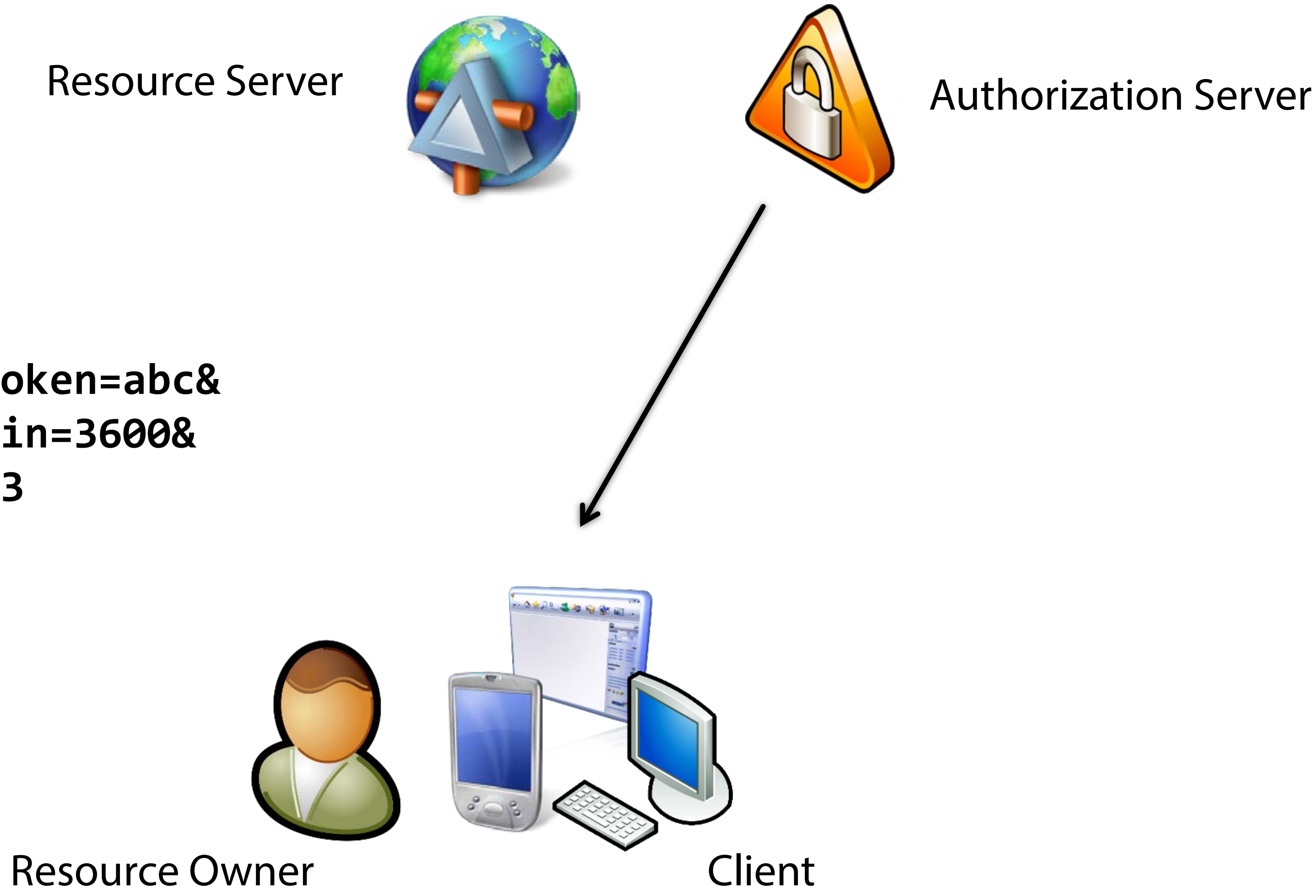
You may revoke access at any time by going to Applications in your account settings.

Authorize BubbleBrowser for:

1 year

Terms of Service | Privacy Policy | Copyright 2013 Evernote Corporation. All rights reserved.

Step 1d: Token Response

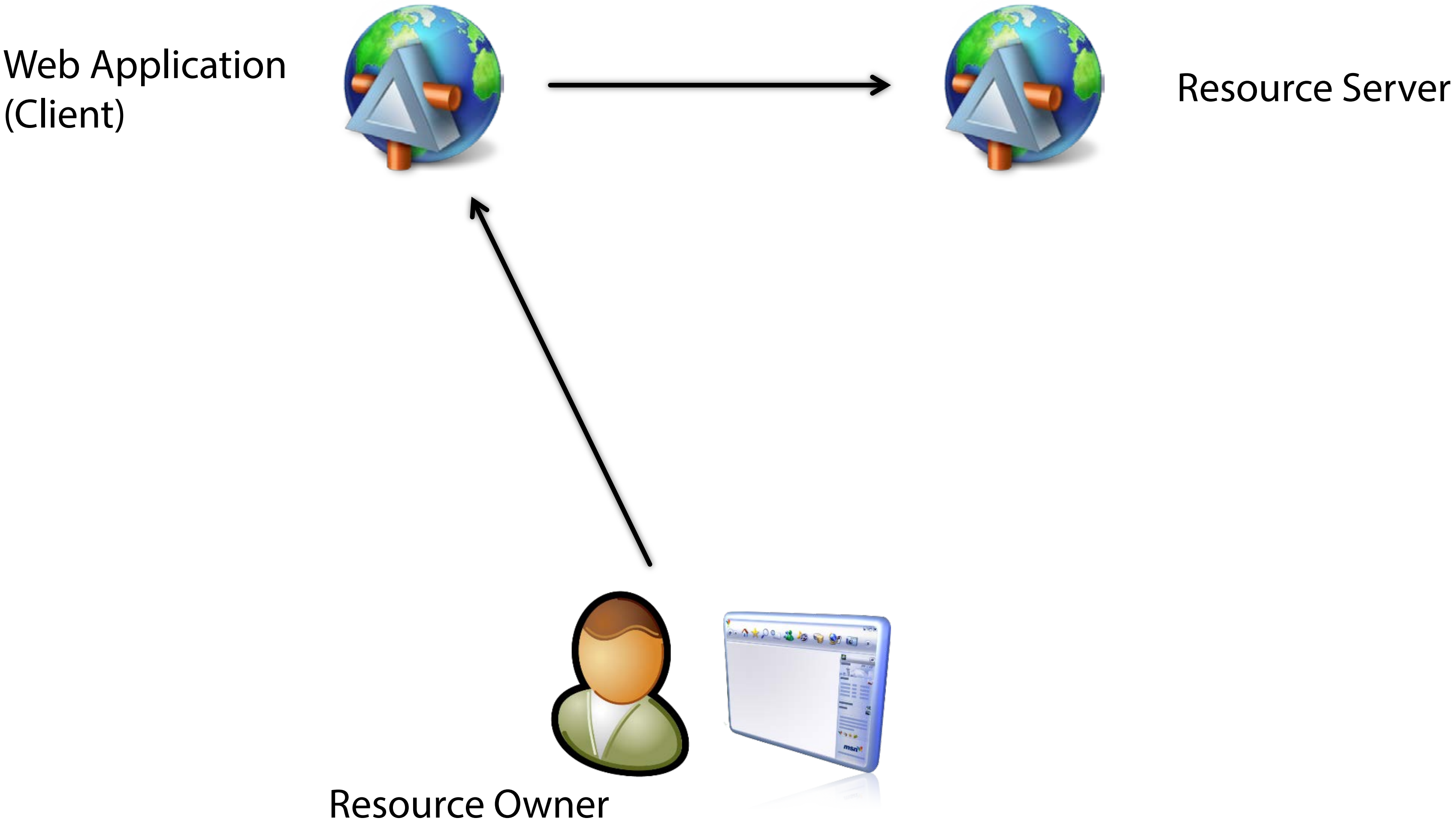


```
GET /cb#
  access_token=abc&
  expires_in=3600&
  state=123
```

Summary – Implicit Flow

- **User enters credentials at the authorization server**
  - not at the client
- **authorization server returns (short lived) access token**
  - to reduce exposure of token
- **Often combined with OS helper mechanisms**
  - cookie container
  - native APIs

Authorization Code Flow (Server-based Clients)

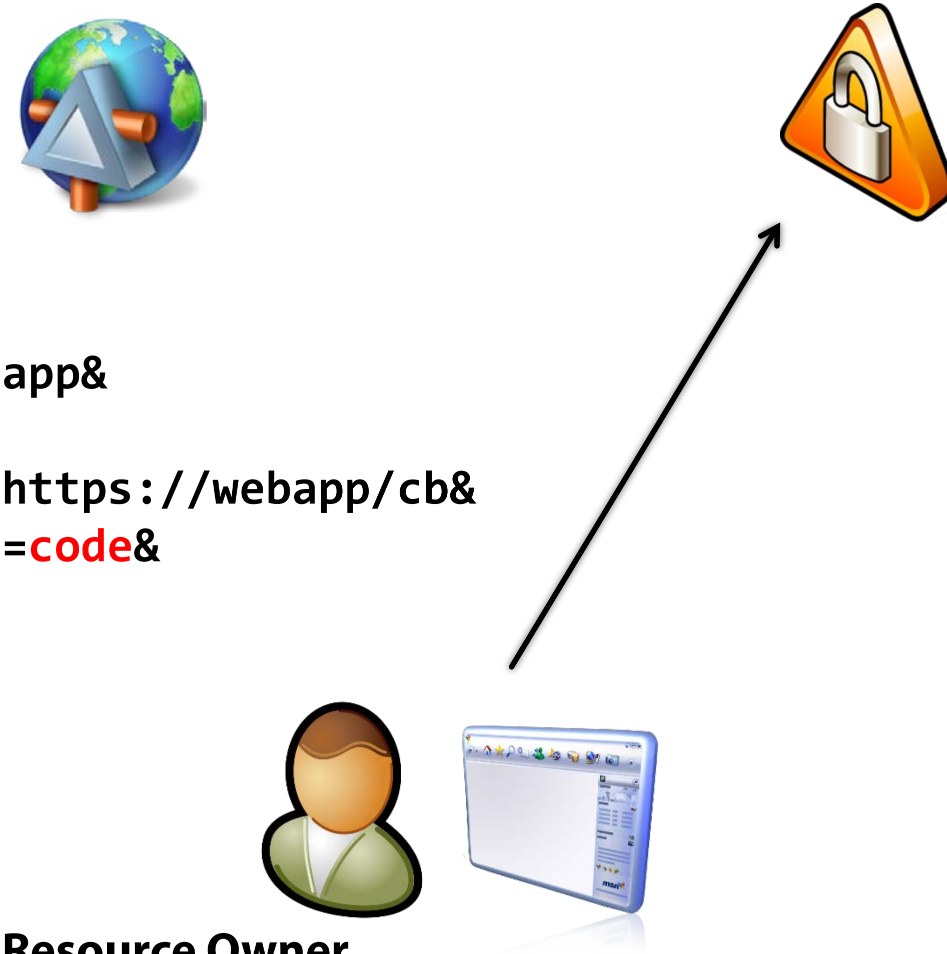


Web Application (Client)

Resource Server

Resource Owner

Step 1a: Authorization Request

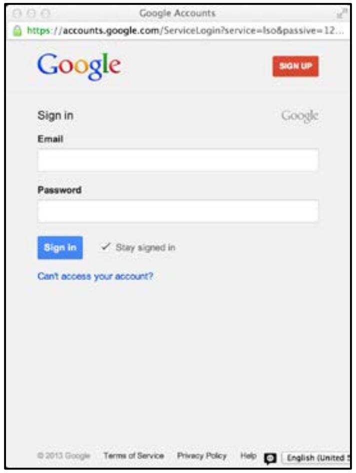


Web Application
(Client)

Authorization Server

```
GET /authorize?
  client_id=webapp&
  scope=read&
  redirect_uri=https://webapp/cb&
  response_type=code&
  state=123
```

Resource Owner



Google Accounts

https://accounts.google.com/ServiceLogin?service=lso&passive=12...

Sign in

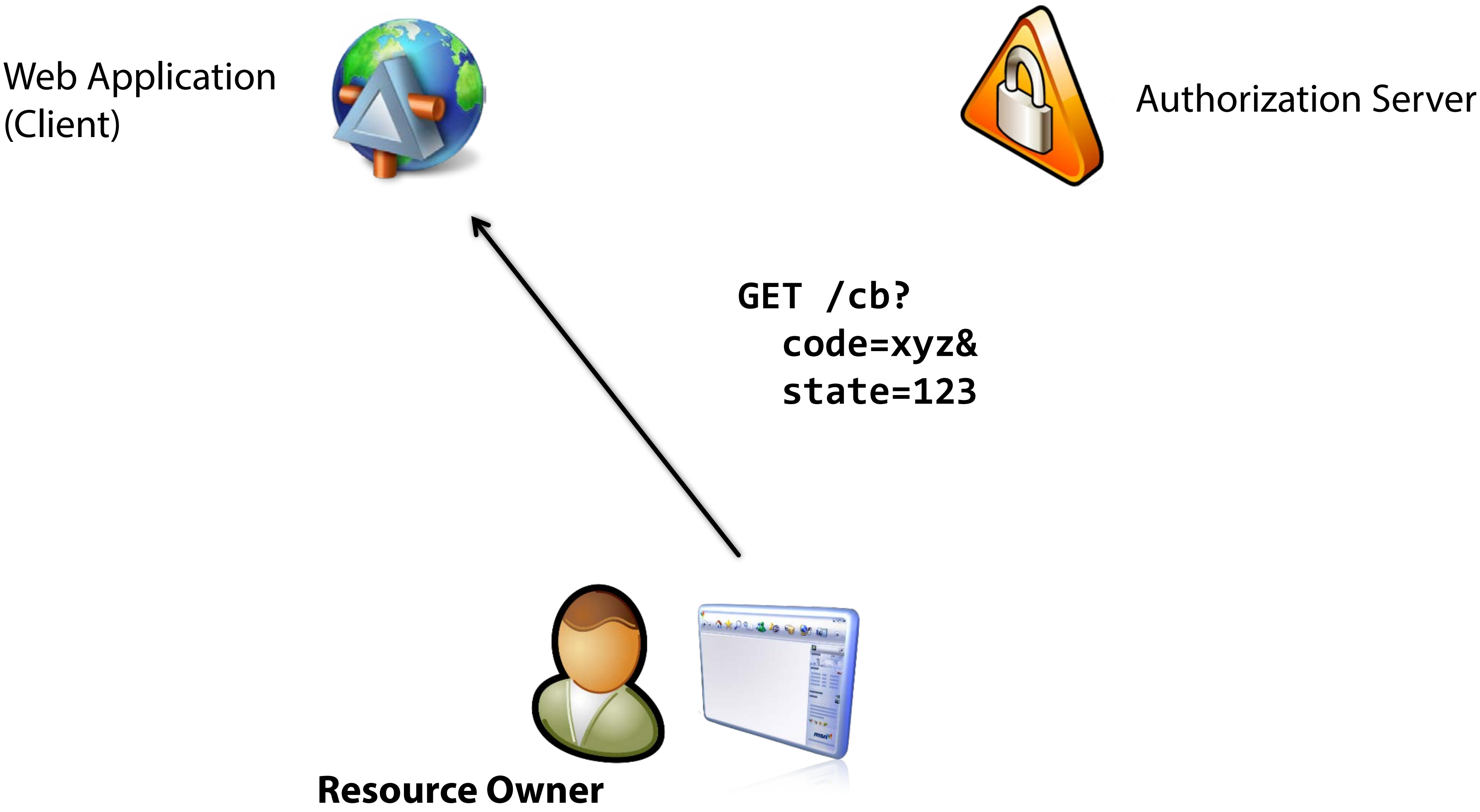
Email

Password

✓ Stay signed in

Can't access your account?

Step 1d: Authorization Response



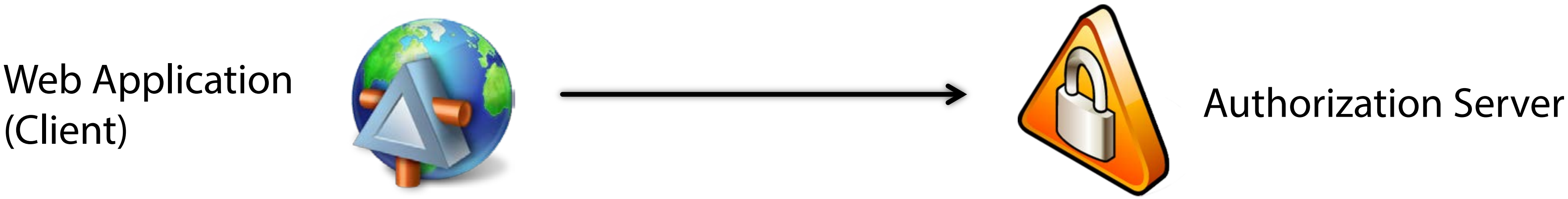
Web Application
(Client)

Authorization Server

```
GET /cb?
  code=xyz&
  state=123
```

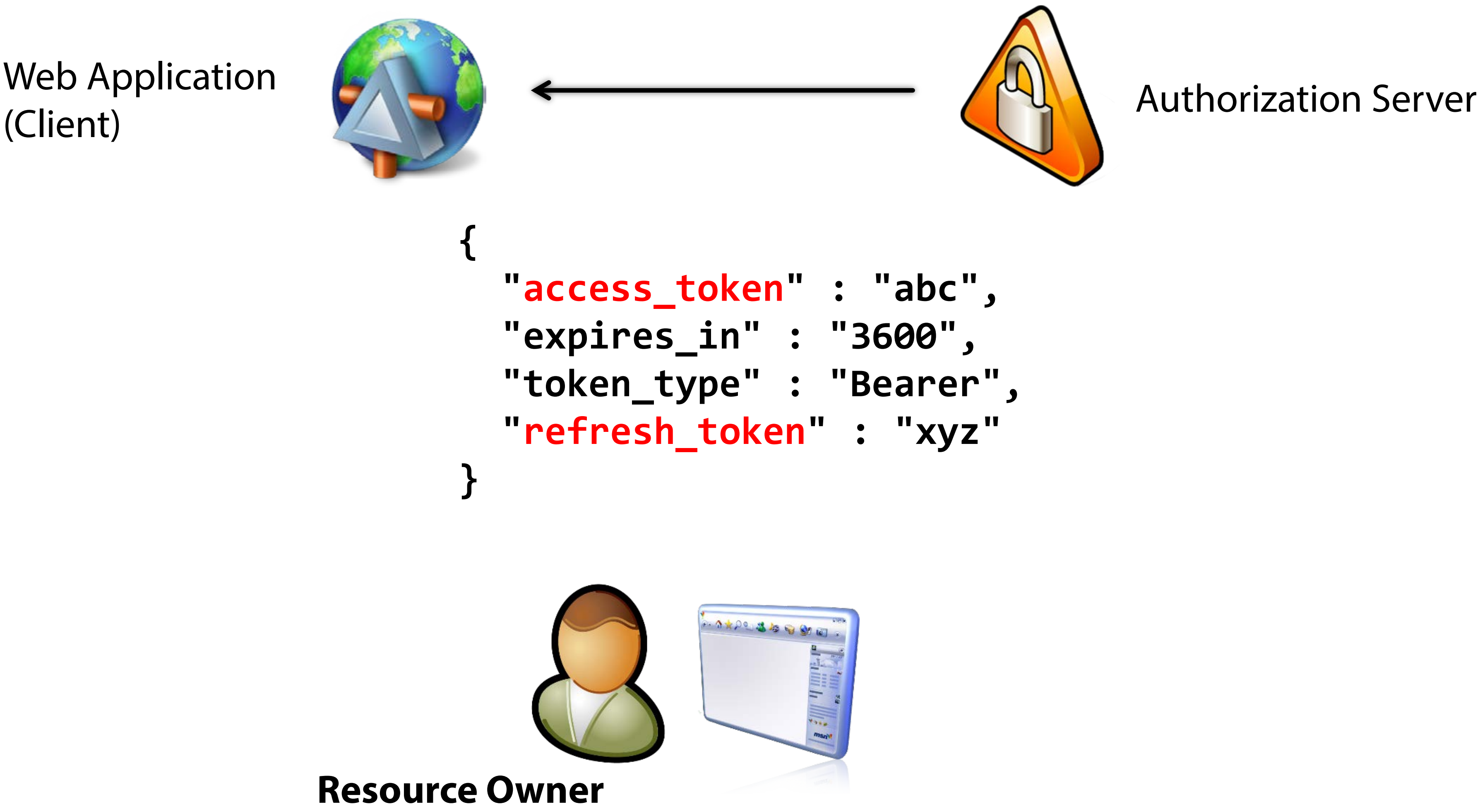
Resource Owner

Step 2a: Token Request



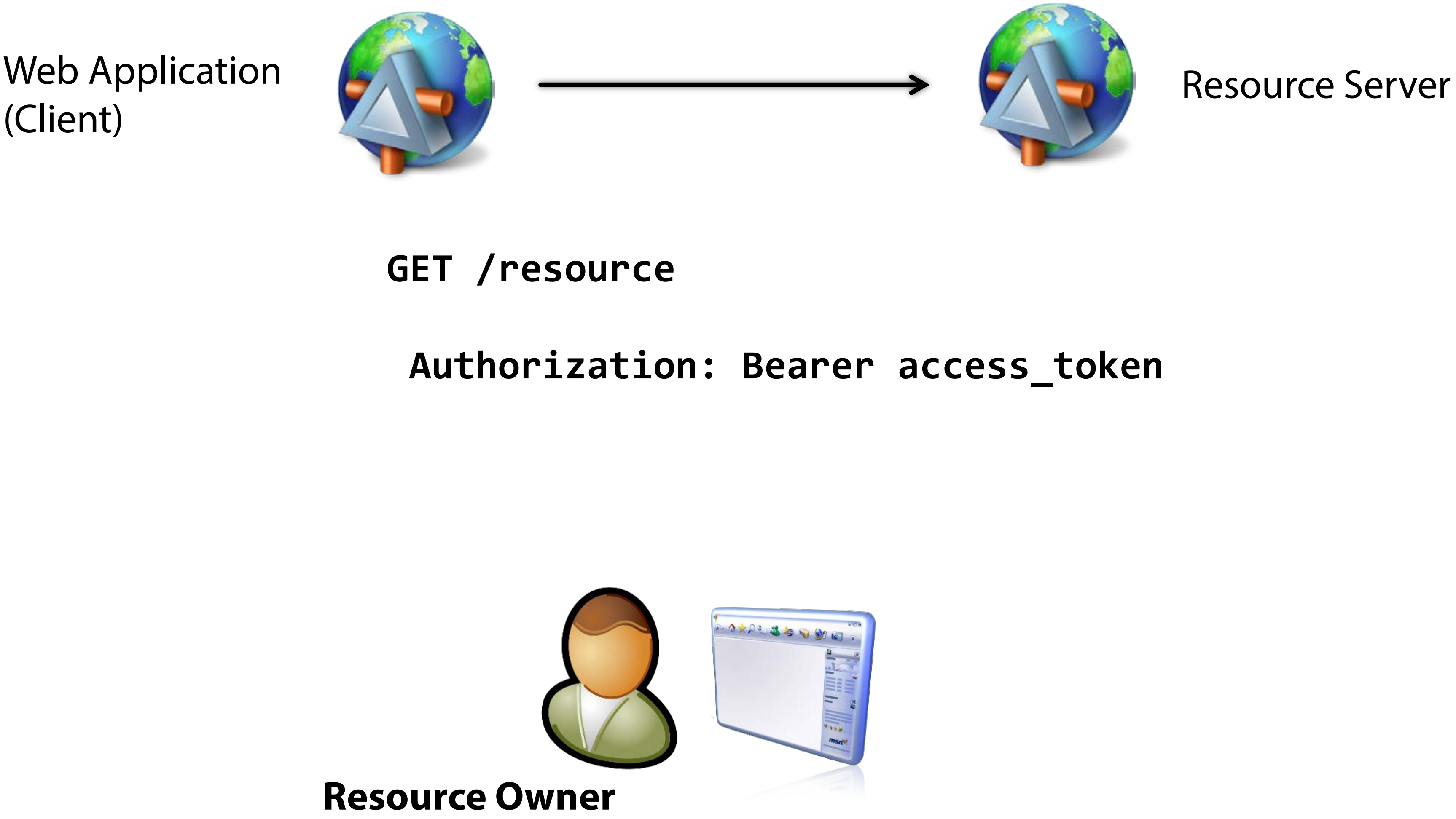
```
POST /token
 Authorization: Basic (client_id:secret)

grant_type=authorization_code&
authorization_code=xyz
```

Step 2b: Token Response



```
{
  "access_token" : "abc",
  "expires_in" : "3600",
  "token_type" : "Bearer",
  "refresh_token" : "xyz"
}
```

Step 3: Resource Access



Web Application
(Client)

Resource Server

```
GET /resource

 Authorization: Bearer access_token
```

Resource Owner

Summary – Code Flow

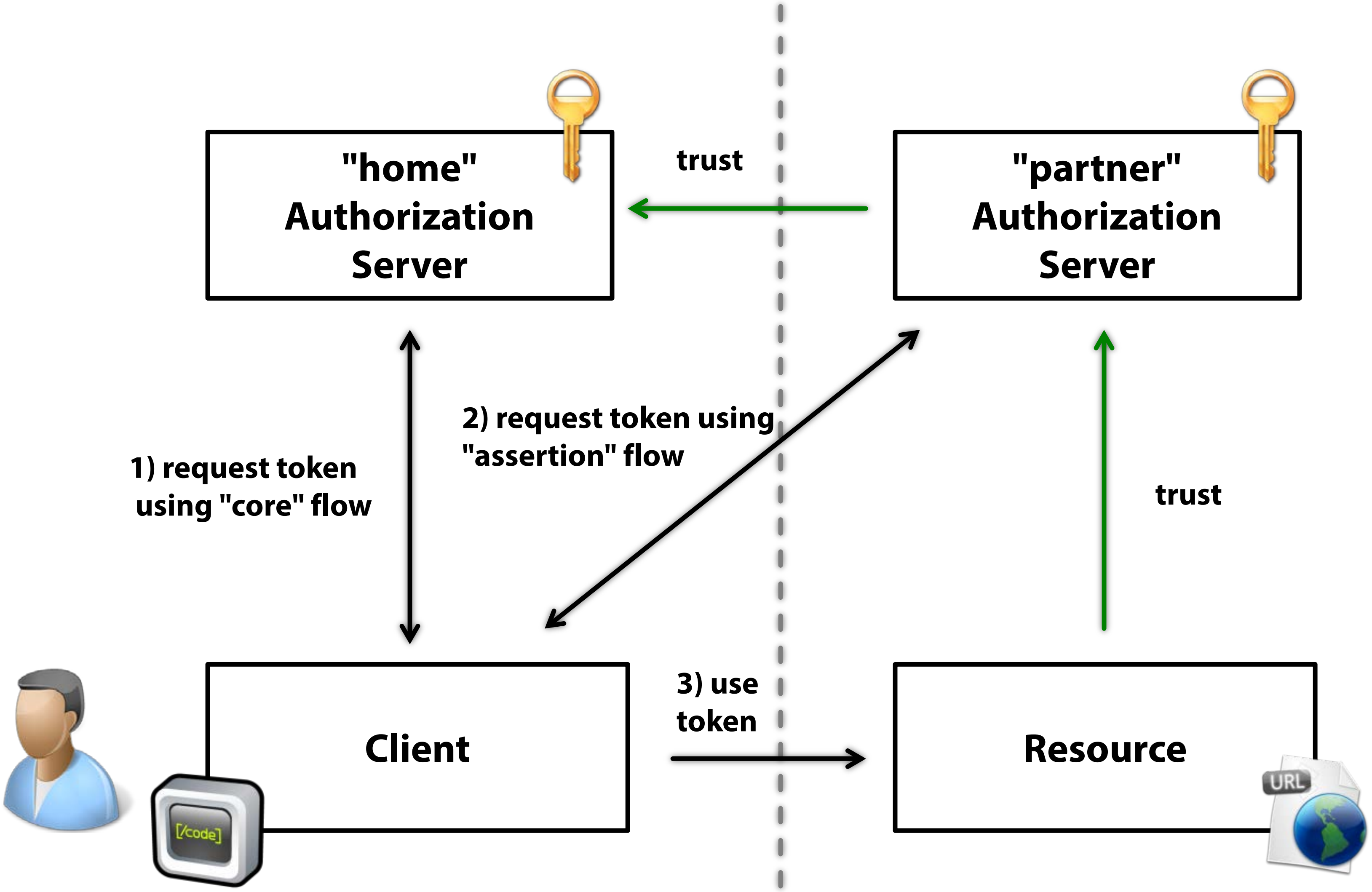
- **Designed for "confidential" clients**
  - client can store secret securely
  - client authentication and authorization based on client identity possible
  - typically server-based applications
- **Accountability is provided**
  - access token never leaked to the browser
- **Long-lived access can be implemented**

Crossing Trust Boundaries…

- **So far authorization server and resource server are always in the same trusted subsystem**
  - your client accessing your back-end
  - facebook client accessing facebook back-end
  - translate between identity management systems

- **What if you want to cross the line?**
  - Assertion Flow

Assertion Flow



"home" Authorization Server

"partner" Authorization Server

trust

1) request token using "core" flow

2) request token using "assertion" flow

trust

Client

3) use token

Resource

Summary

- **The notion of an authorization server simplifies the security scenarios**
  - □ passwords as credential don‘t work anymore
  - □ many users, clients, APIs, scopes
  - □ think of flows as patterns
- **Web API v2 OAuth2 middleware make gettting started easier**
  - □ Thinktecture AuthorizationServer is a ready to use full featured implementation