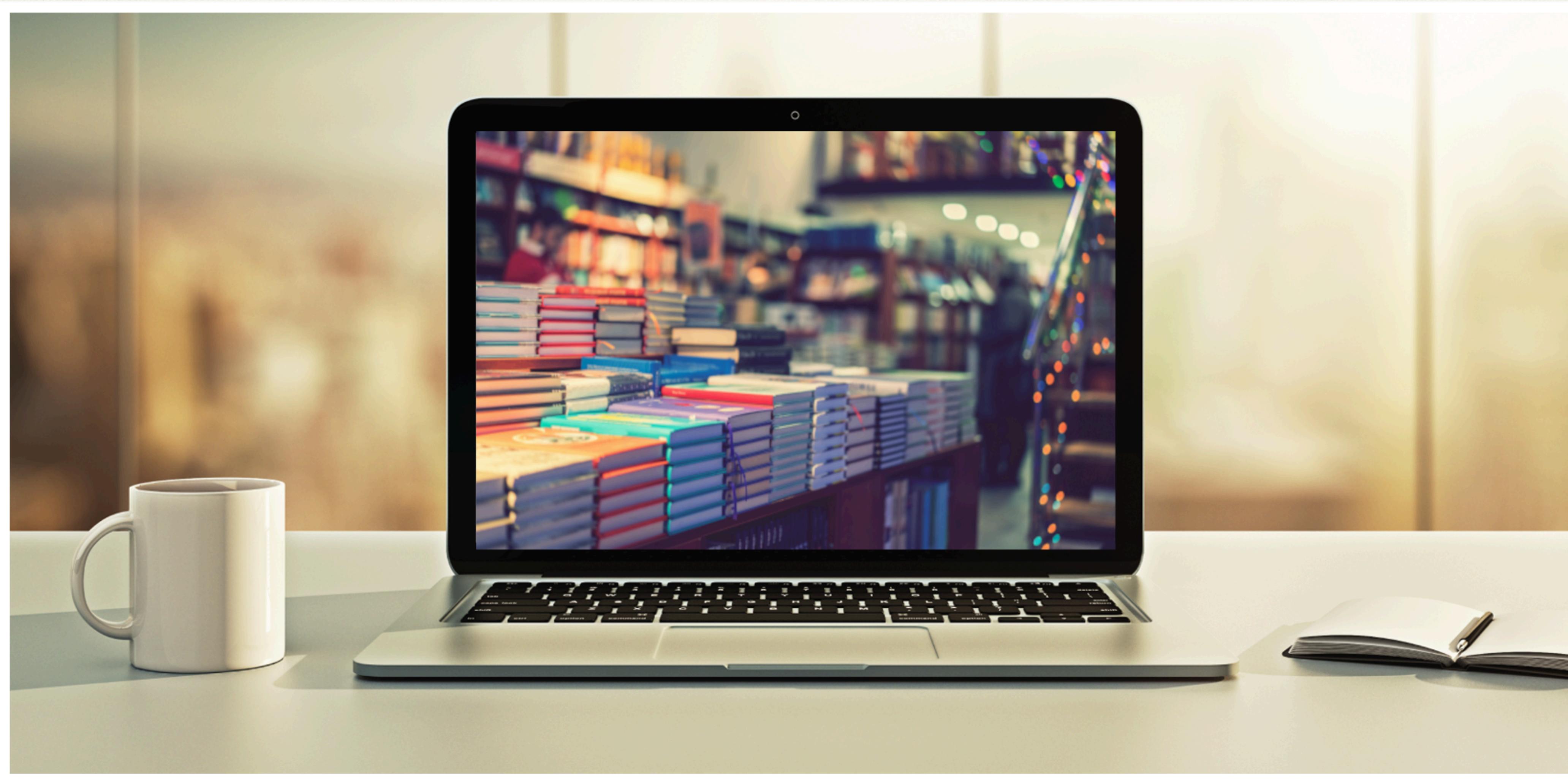


# Application Security Concepts



# Practical Results

- This is a basic introduction to Security Concepts
  - Focus on practical results
  - Not an A to Z reference
- We will cover following tasks related to our library project
  - **User login/logout security**
  - **Provide access to special pages only for authenticated customers**
  - **Keep track of book history for registered customers**

# The Problem

- We need to authenticate a user
- We need to know what actions a user / app is authorized to perform
- Delegate permissions to another app

# Key Terms

- Authentication
- Authorization
- OAuth 2
- OpenID Connect (OIDC)
- JSON Web Tokens (JWT)

# Authentication

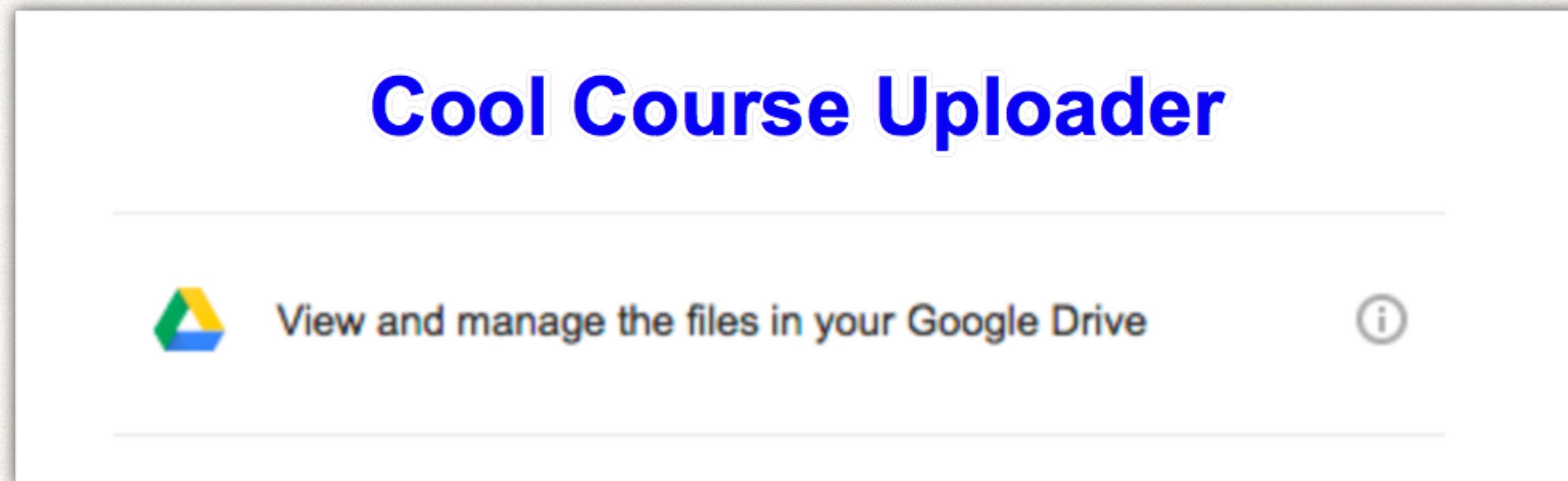
- The process of validating whether a user / app is who they claim to be
  - User name / password
  - Token / pin
  - Finger print / retina scan

# Authorization

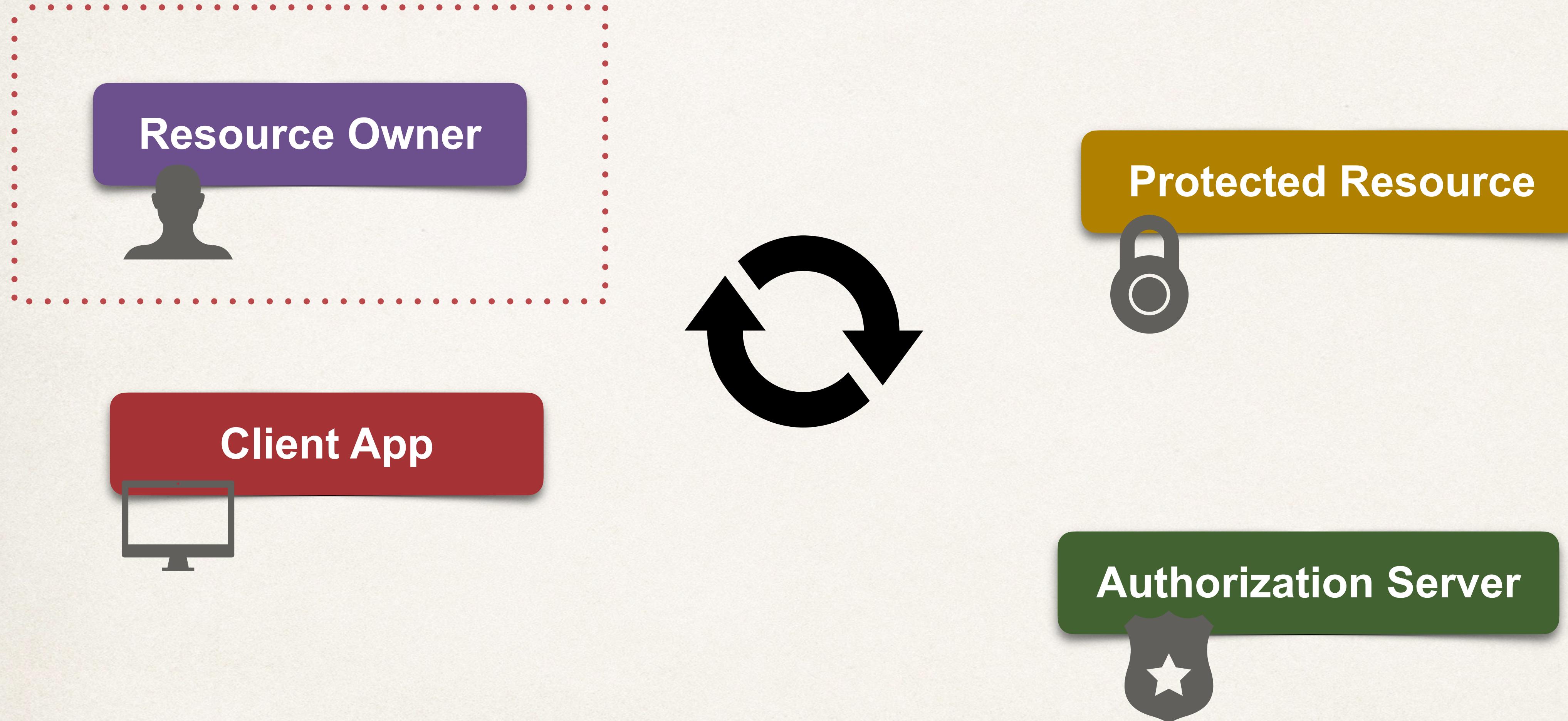
- Process of determining the actions a user / app can perform
- Commonly understood as roles
  - Guest user: minimal actions (read only)
  - Authorized user: read / write data in user account
  - Admin: full access to all accounts system wide

# OAuth 2

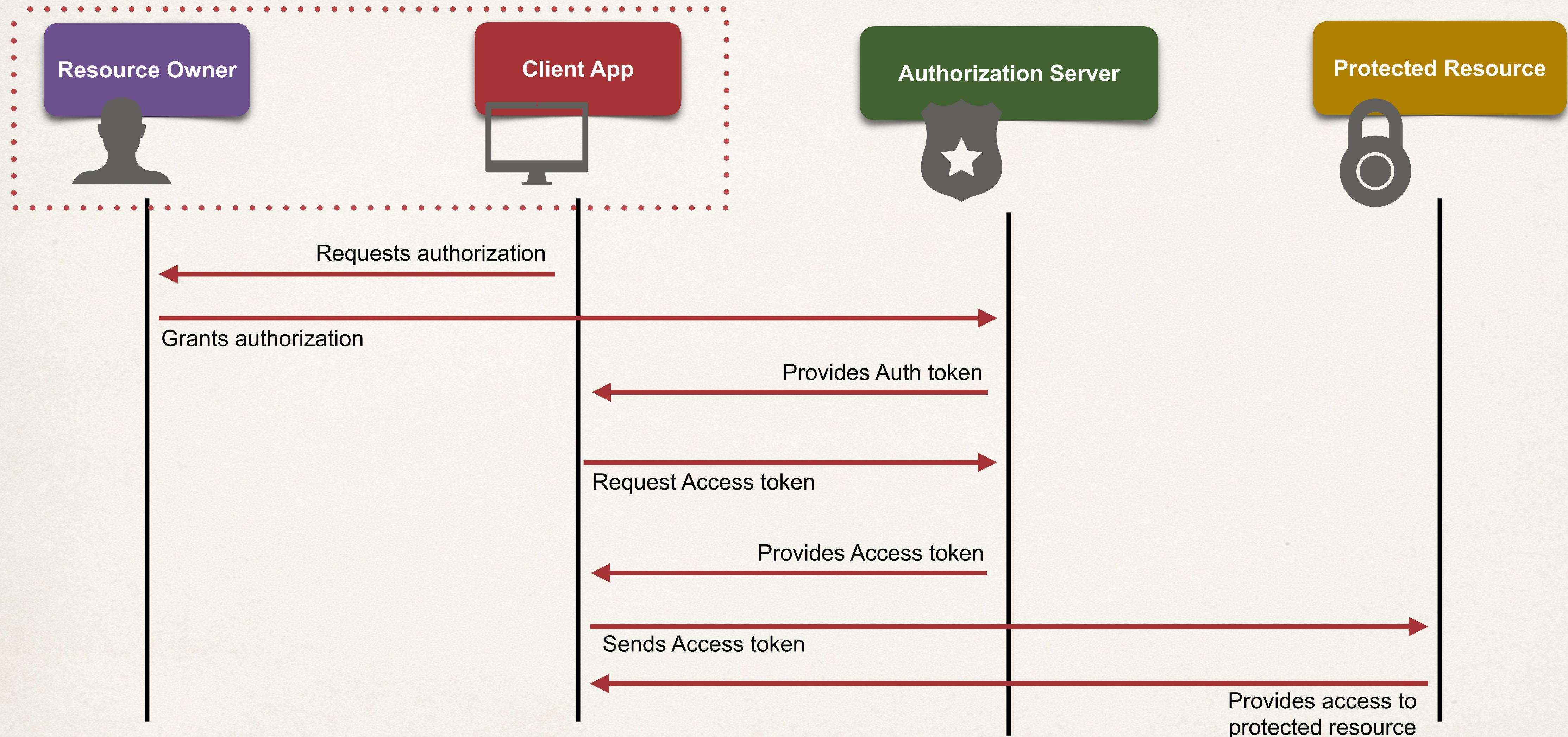
- Authorization framework that enables applications to have limited access to a resource on behalf of a resource owner (user)



# OAuth 2



# OAuth 2

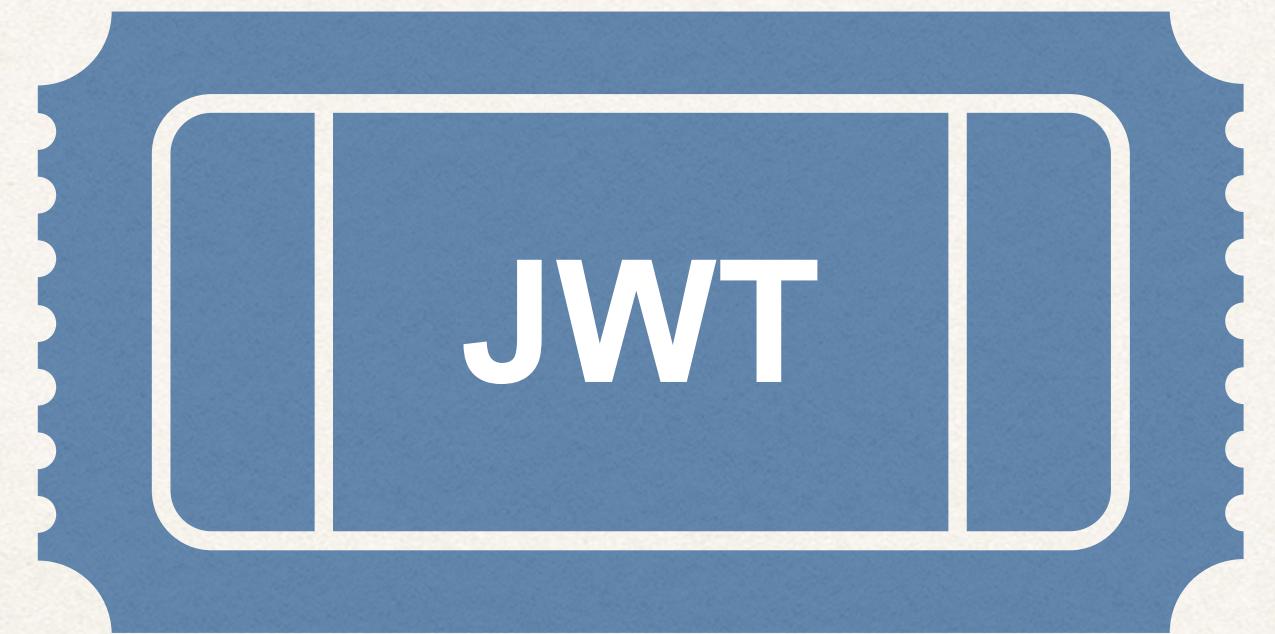


# OpenID Connect

- Identity layer on top of OAuth 2
- Allows clients to receive "identity" information about authenticated resource owners (users)
- Provided via an ID Token

# JSON Web Token (JWT)

- Open standard that defines self-contained way of describing tokens
- Secure and digitally signed to guarantee integrity
- Used by OAuth and OpenID Connect



# JSON Web Token (JWT)

**Header**

```
{  
  "alg": "HS256",  
  "typ": "JWT"  
}
```

Signing algorithm and type  
of token

**Payload**

```
{  
  "sub": "1111111111",  
  "name": "Susan Public",  
  ...  
}
```

Contains content for user  
data

**Signature**

```
{  
  ...  
}
```

Guarantee integrity of the  
token

# Authorization Servers

- Generate tokens and define security policies
- **Simple Solutions**
  - Create your own simple solution with code
  - A lot of low-level coding and vulnerable to security holes / flaws
- **Real-time Enterprise Solutions**
  - Off-the-shelf solutions from companies specializing in security
  - Cloud-based solutions and on-premise solutions

# Additional Resources

Technology	Website
OAuth 2	<b><a href="http://www.oauth.net">www.oauth.net</a></b>
OpenID Connect	<b><a href="http://wwwopenid.net/connect">www openid.net/connect</a></b>
JWT	<b><a href="http://www.jwt.io">www.jwt.io</a></b>