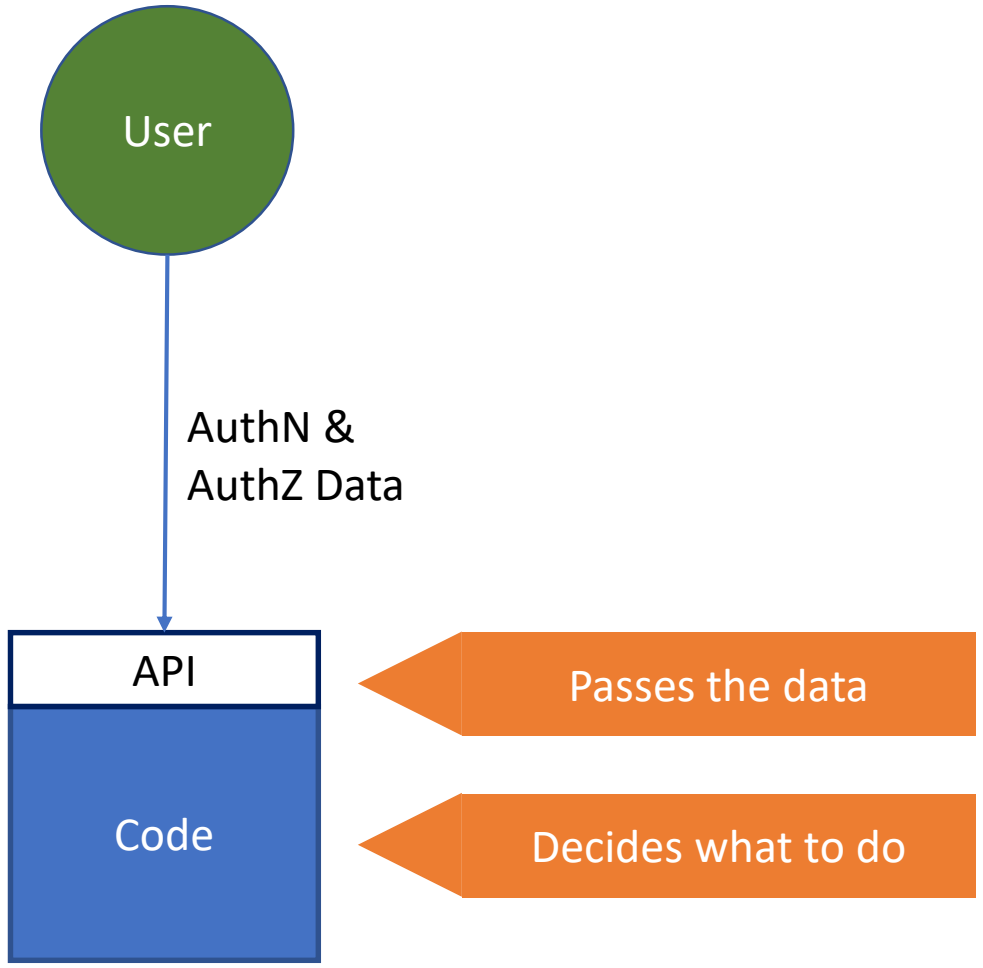
Authentication & Authorization

Memi Lavi
www.memilavi.com

AuthN & AuthZ

- Your API will not always be publicly accessible
- Only authorized users may access it
- Authentication (AuthN) – Who is the user
- Authorization (AuthZ) – What is he allowed to do

AuthN & AuthZ In The API



AuthN & AuthZ

- Many authn & authz mechanisms
- We will focus on OAuth2

OAuth2

- Standard protocol for authentication & authorization
- Widely used, mainly in web apps
- We'll discuss only high-level details

OAuth2 Components

User	The user who wants to access protected resources in the API

OAuth2 Components

User	The user who wants to access protected resources in the API
Client App	The client application accessing the API

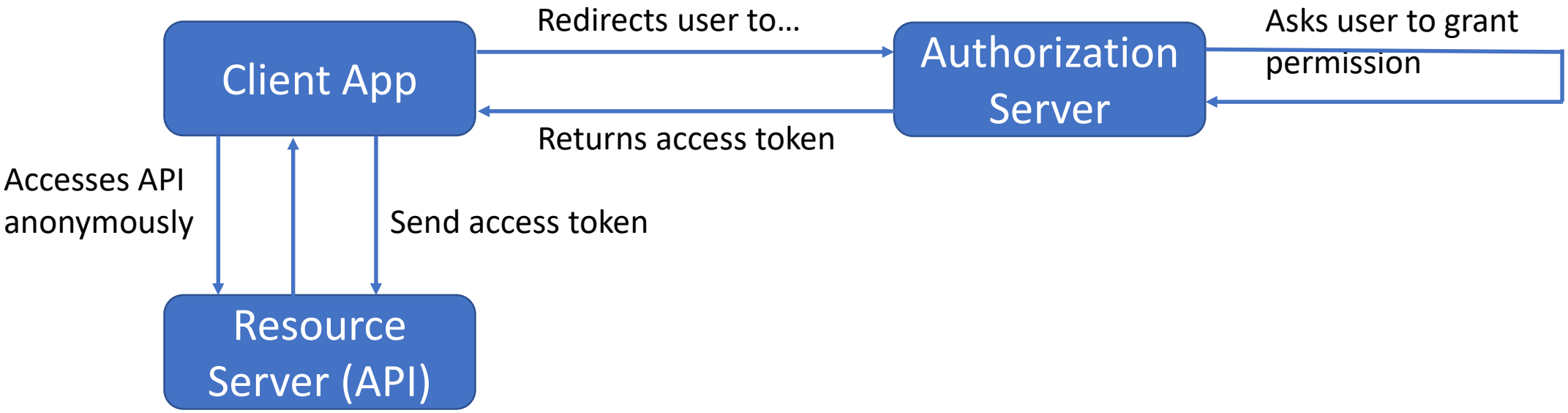
OAuth2 Components

User	The user who wants to access protected resources in the API
Client App	The client application accessing the API
Authorization Server	Authorizes the user for the client application

OAuth2 Components

User	The user who wants to access protected resources in the API
Client App	The client application accessing the API
Authorization Server	Authorizes the user for the client application
Resource Server	The API being accessed

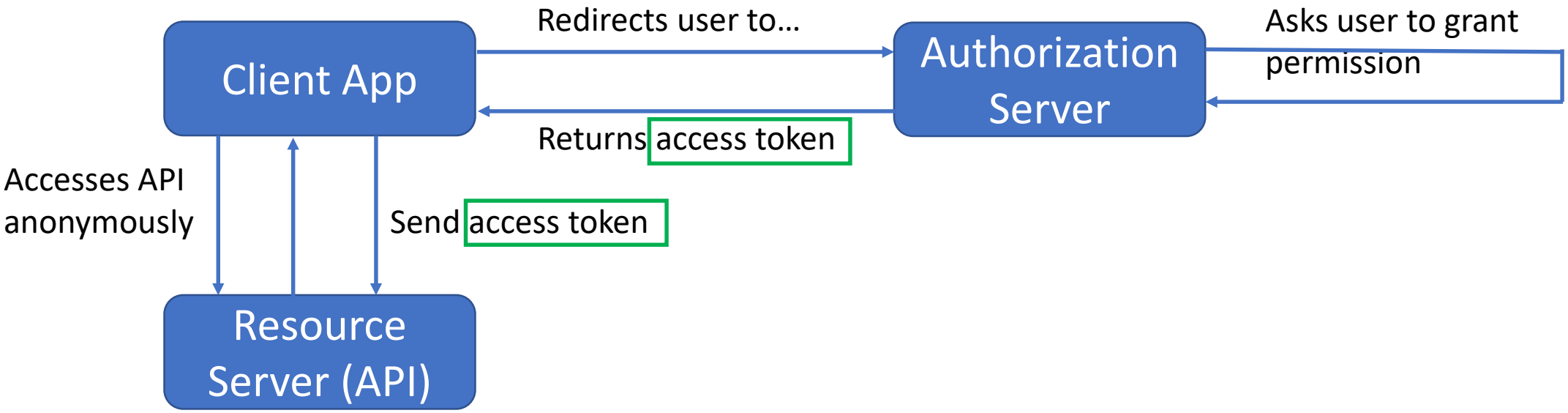
OAuth2 Flow


Client App
Redirects user to…
Authorization Server
Asks user to grant permission
Returns access token
Accesses API anonymously
Send access token
Resource Server (API)

App Registration

- Authorization Server should be familiar with the Resource Server (API)
- Resource Server must register itself with the Authorization Server

OAuth2 Flow

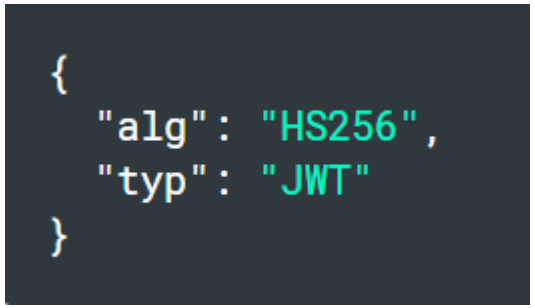

Client App
Redirects user to…
Authorization Server
Asks user to grant permission
Returns access token
Accesses API anonymously
Send access token
Resource Server (API)

JWT

- JSON Web Token
- Contains the data the server needs in order to authenticate the user

JWT

- Has three sections:
  - Header – type of token (JWT) and signing algorithm



Source: https://jwt.io/introduction/

JWT

- Has three sections:
  - Payload – Data on the user

```
{
  "sub": "1234567890",
  "name": "John Doe",
  "admin": true
}
```

Source: https://jwt.io/introduction/

JWT

- Has three sections:
  - Signature

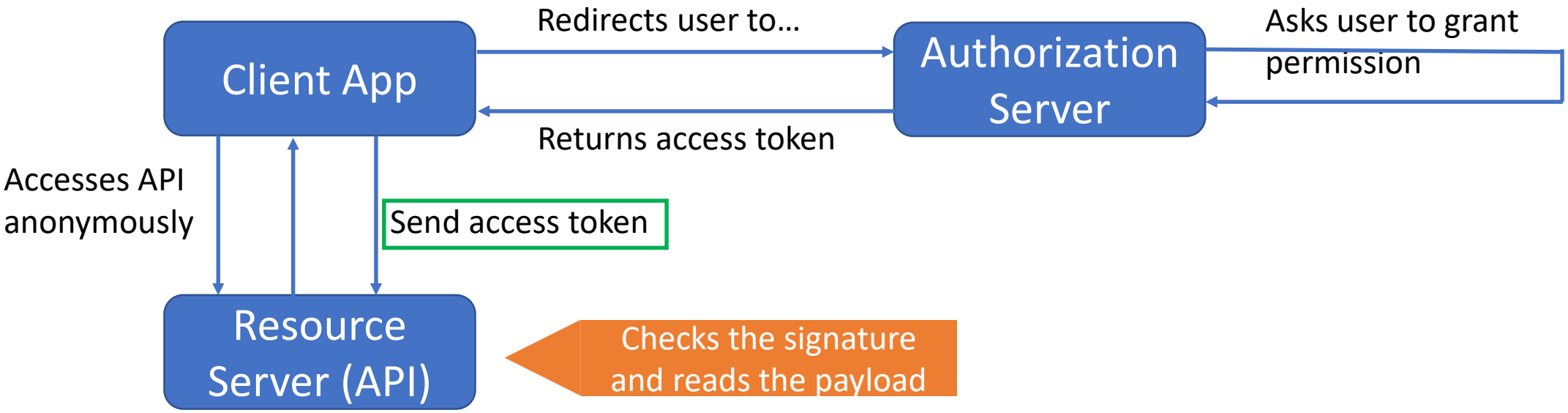
JWT

- The three parts are:
  - Base64 encoded
  - Concatenated with .

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4
gRG9lIiwiaXNTb2NpYWwiOnRydWV9.
4pcPyMD09olPSyXnrXCjTwXyr4BsezdI1AVTmud2fU4
```

Source: https://jwt.io/introduction/

OAuth2 Flow


Client App
Redirects user to…
Authorization Server
Asks user to grant permission
Returns access token
Accesses API anonymously
Send access token
Resource Server (API)
Checks the signature and reads the payload

JWT & REST API

- JWT Should be sent with the *Authorization: bearer* header
- Can be sent also in body or request parameter
  - Not recommended

```
GET /resource HTTP/1.1
Host: server.example.com
Authorization: Bearer mF_9.B5f-4.1JqM
```

Source: https://tools.ietf.org/html/rfc6750