UNIVERSITÀ DELLA CALABRIA

il Campus per eccellenza

OAuth authentication

Mario Alviano

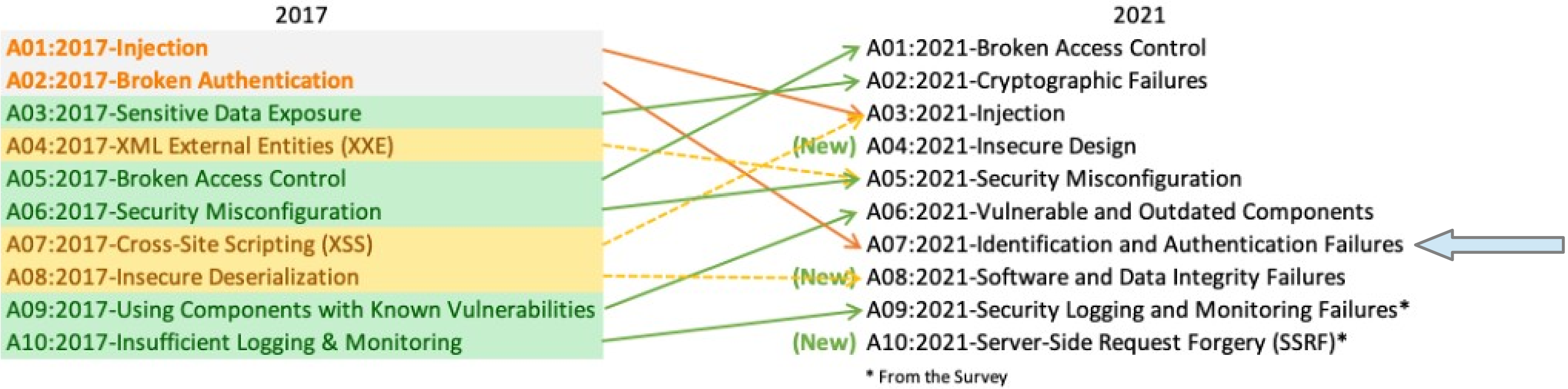
Main References

Bug Bounty Bootcamp – Chapter 20 (something)

https://portswigger.net/web-security/oauth

OWASP Top Ten

A broad consensus about the most critical security risks to web applications



2017

- A01:2017-Injection
- A02:2017-Broken Authentication
- A03:2017-Sensitive Data Exposure
- A04:2017-XML External Entities (XXE)
- A05:2017-Broken Access Control
- A06:2017-Security Misconfiguration
- A07:2017-Cross-Site Scripting (XSS)
- A08:2017-Insecure Deserialization
- A09:2017-Using Components with Known Vulnerabilities
- A10:2017-Insufficient Logging & Monitoring

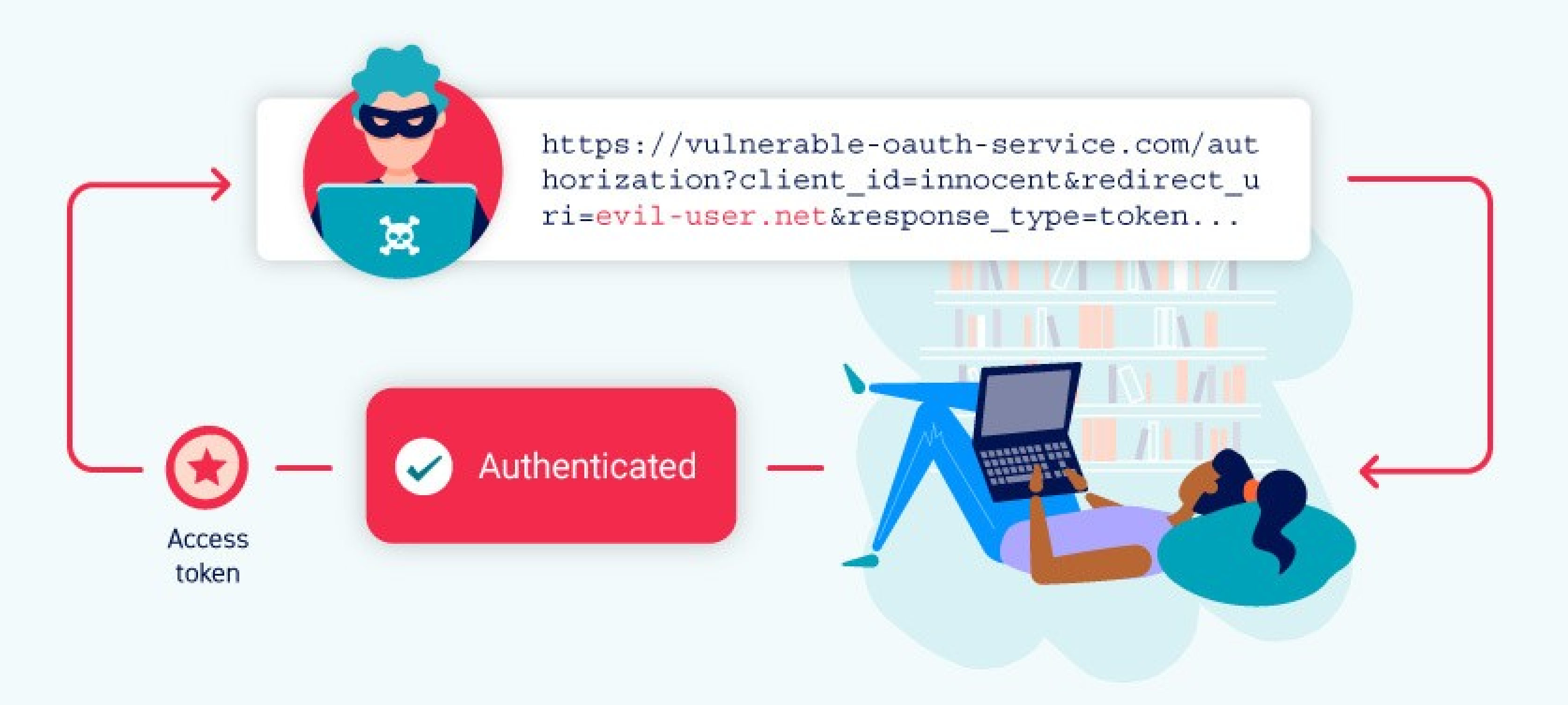
2021

- A01:2021-Broken Access Control
- A02:2021-Cryptographic Failures
- A03:2021-Injection
- (New) A04:2021-Insecure Design
- A05:2021-Security Misconfiguration
- A06:2021-Vulnerable and Outdated Components
- A07:2021-Identification and Authentication Failures
- (New) A08:2021-Software and Data Integrity Failures
- A09:2021-Security Logging and Monitoring Failures*
- (New) A10:2021-Server-Side Request Forgery (SSRF)*

* From the Survey

Single Sign-On (SSO)

Allows users to access multiple services without logging in multiple times.



https://vulnerable-oauth-service.com/authorization?client_id=innocent&redirect_uri=evil-user.net&response_type=token...

It's a token-based protocol, often built using OAuth 2.0

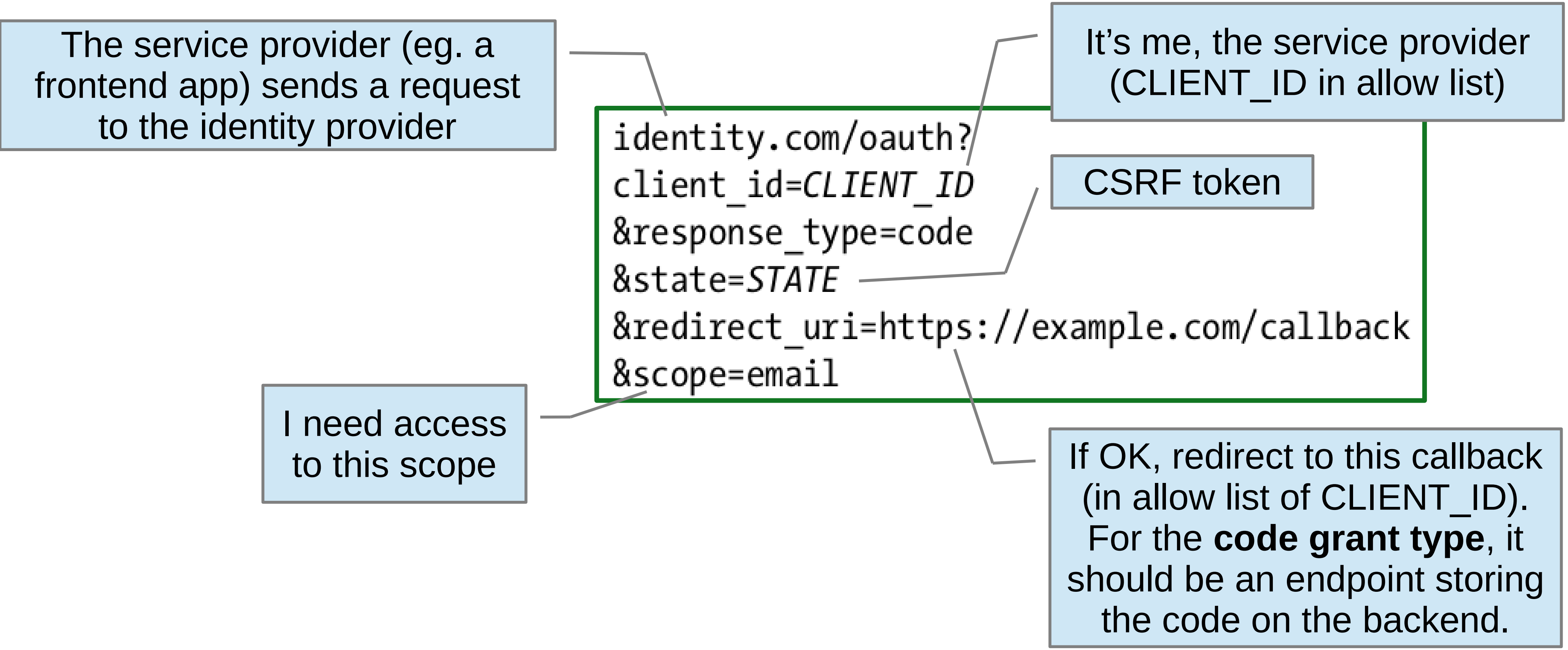
How Oauth works

User want to access service provider

Credentials stored in a different server

How can the service provider recognize users without asking their credentials?

The service provider requests access to user information from the identity provider.
Requested permissions and pieces of data will go under the name of scope.
The identity provider creates a unique access token to grant access to the scope.



```
identity.com/oauth?
client_id=CLIENT_ID
&response_type=code
&state=STATE
&redirect_uri=https://example.com/callback
&scope=email
```

The service provider (eg. a frontend app) sends a request to the identity provider

It’s me, the service provider (CLIENT_ID in allow list)

CSRF token

I need access to this scope

If OK, redirect to this callback (in allow list of CLIENT_ID). For the **code grant type**, it should be an endpoint storing the code on the backend.

```
identity.com/oauth?
client_id=CLIENT_ID
&response_type=code
&state=STATE
&redirect_uri=https://example.com/callback
&scope=email
```

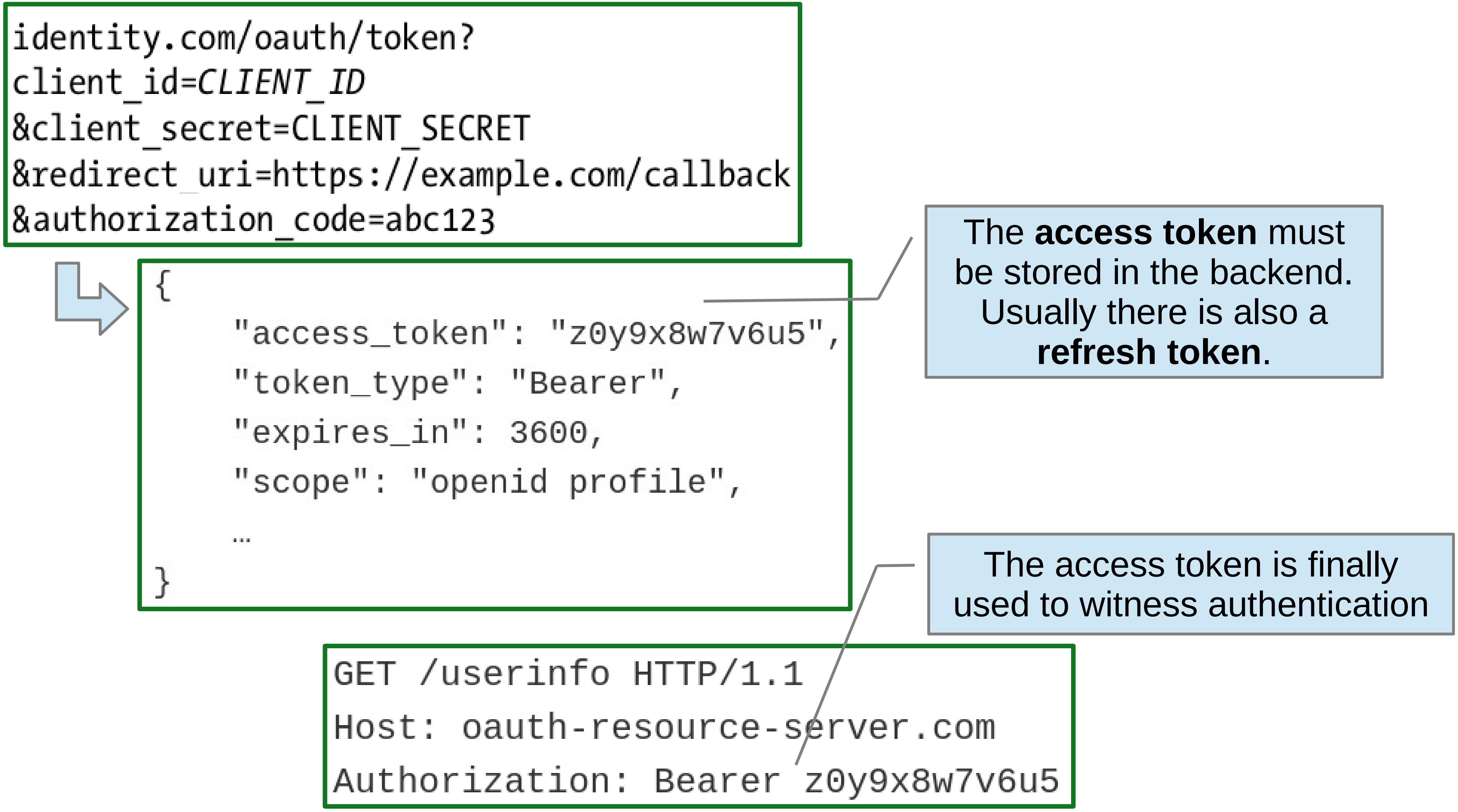
```
https://example.com/callback?code=abc123&state=STATE
```

Store this **authorization code** in the backend

At this point all communication is backend-to-backend.

```
identity.com/oauth/token?
client_id=CLIENT_ID
&client_secret=CLIENT_SECRET
&redirect_uri=https://example.com/callback
&authorization_code=abc123
```

The service provider can exchange the authorization code for an **access token**



```
identity.com/oauth/token?
client_id=CLIENT_ID
&client_secret=CLIENT_SECRET
&redirect_uri=https://example.com/callback
&authorization_code=abc123
```

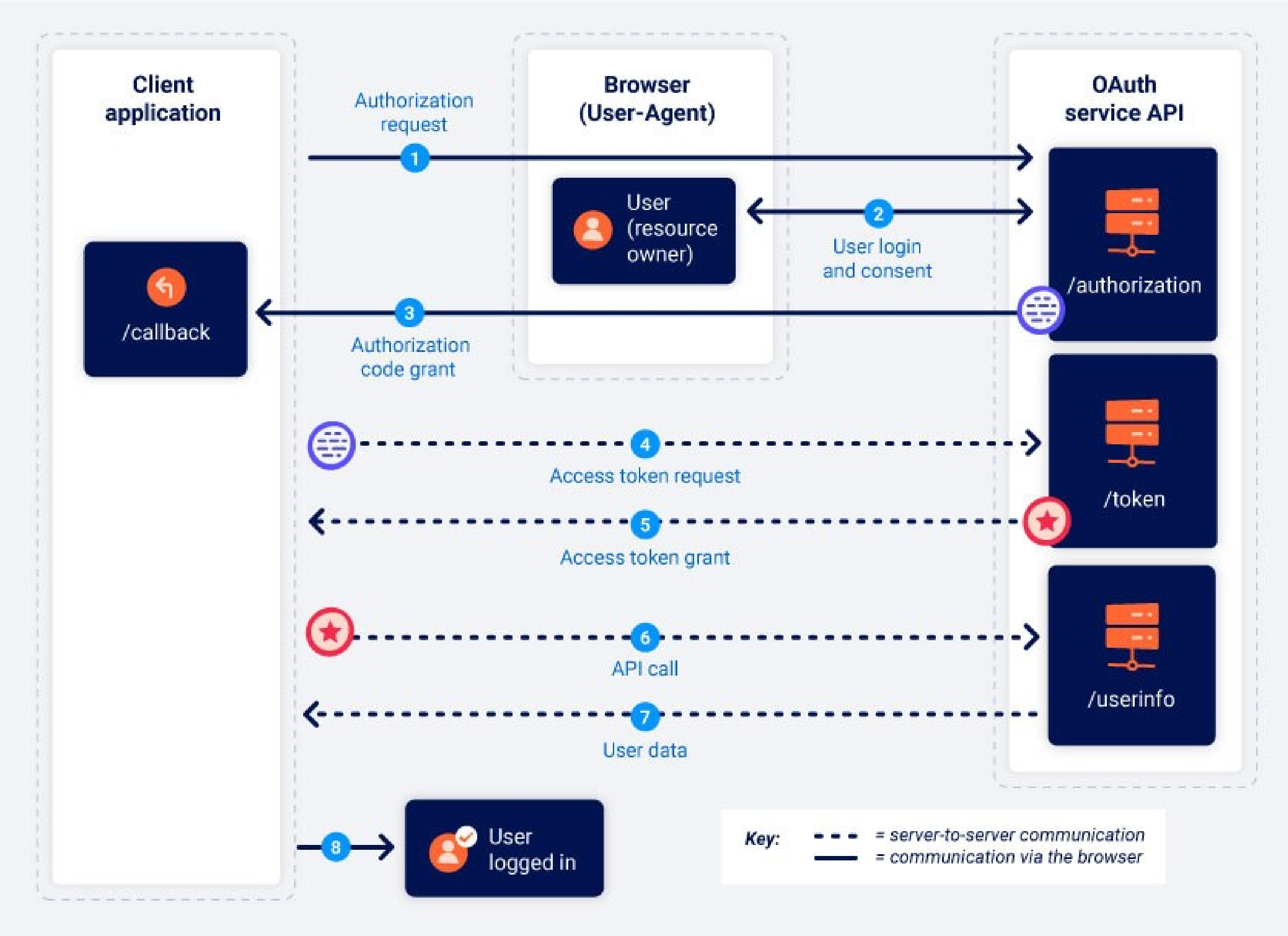
```
{
    "access_token": "z0y9x8w7v6u5",
    "token_type": "Bearer",
    "expires_in": 3600,
    "scope": "openid profile",
    …
}
```

The **access token** must be stored in the backend. Usually there is also a **refresh token**.

The access token is finally used to witness authentication

```
GET /userinfo HTTP/1.1
Host: oauth-resource-server.com
Authorization: Bearer z0y9x8w7v6u5
```

Summary of code grant type (or flow)



It's the most secure because all sensitive communication happens backend-to-backend.

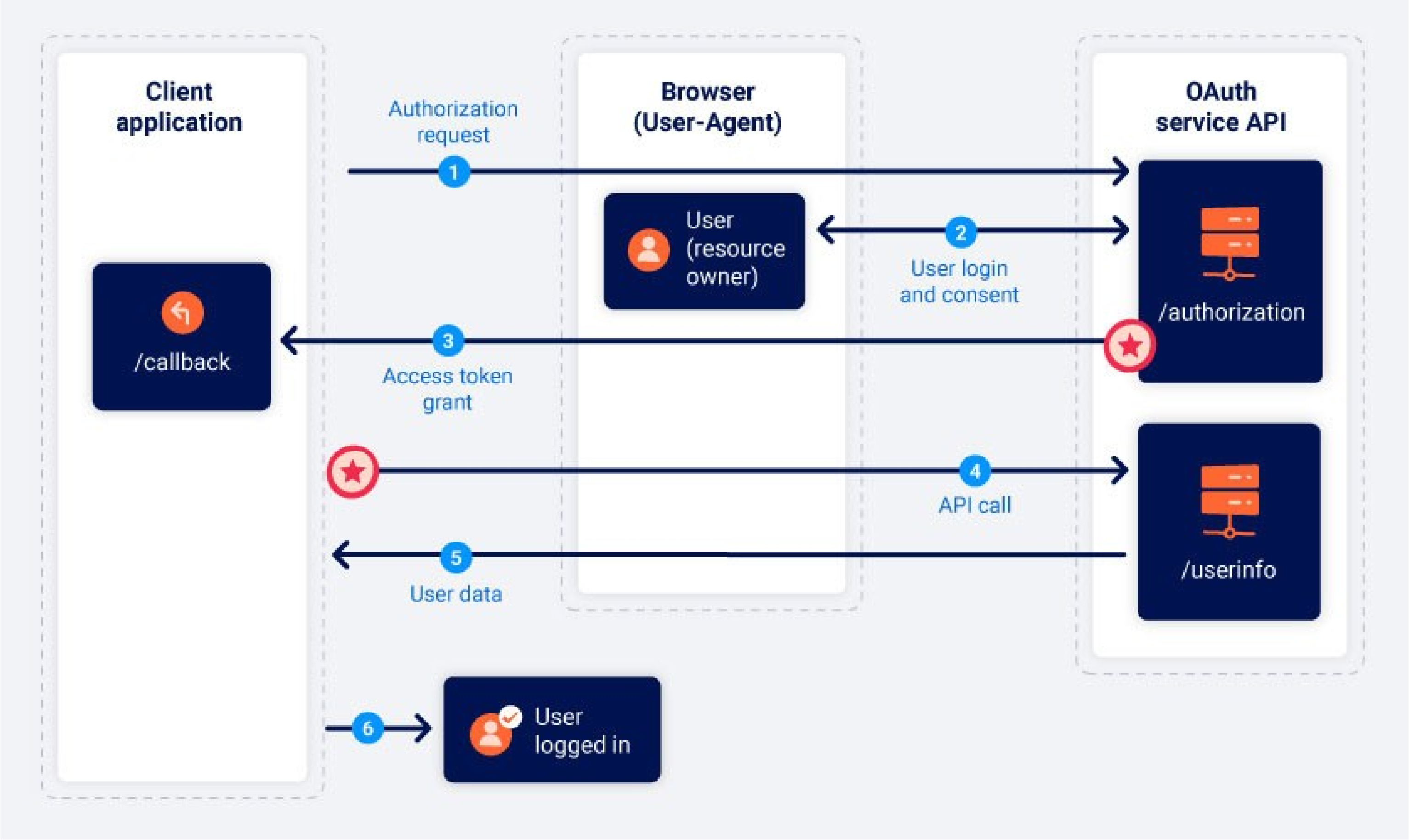
What if you don't have a backend (eg. single-page apps)?!?

Use the weaker **implicit grant type**.

No authorization token.

Keep in mind that the access token is exposend in the browser.

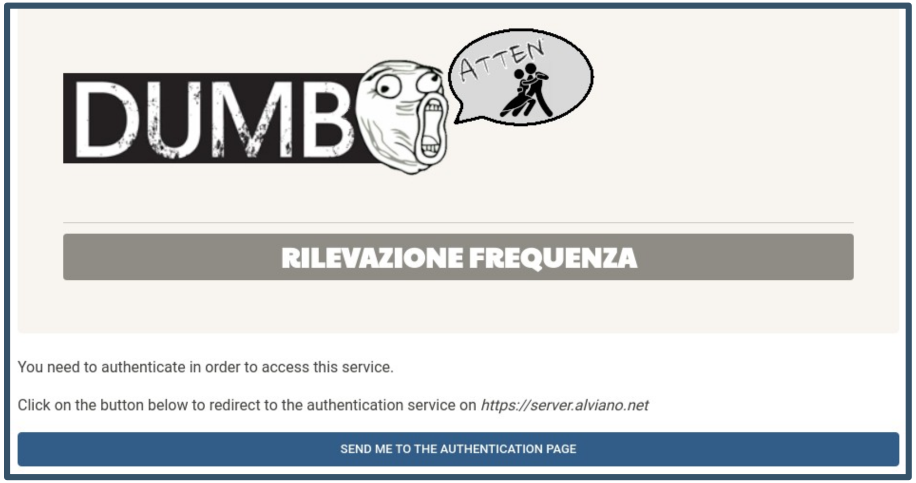
Summary of implicit grant type (or flow)



If the single-page app is hosted in the same domain of the identity provider, use session-based authentication.
It’s safer!

dumbo.alviano.net

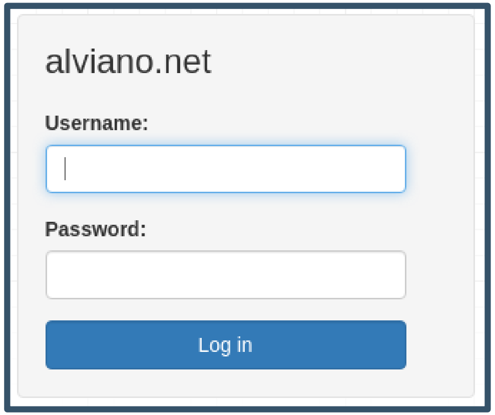
server.alviano.net



RILEVAZIONE FREQUENZA

You need to authenticate in order to access this service.

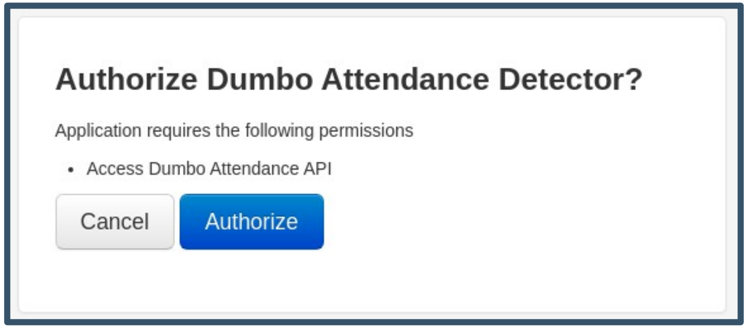
Click on the button below to redirect to the authentication service on *https://server.alviano.net*



alviano.net

Username:

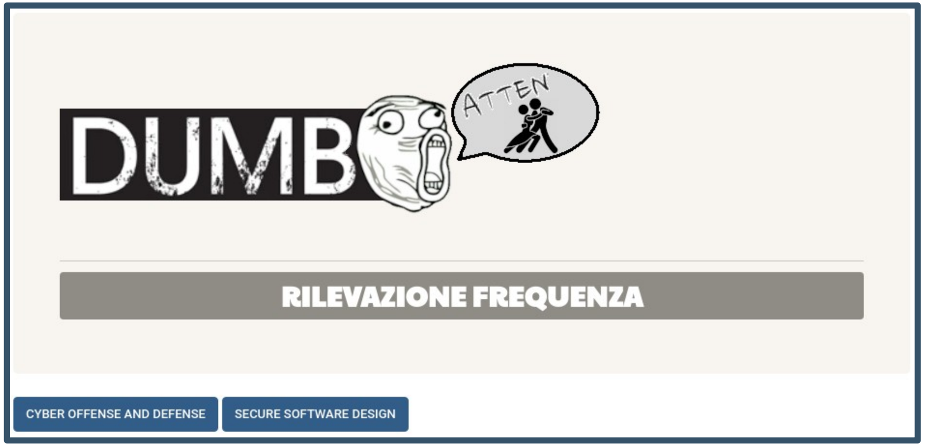
Password:



Authorize Dumbo Attendance Detector?

Application requires the following permissions

- Access Dumbo Attendance API



RILEVAZIONE FREQUENZA

I just use this to achieve session-based authentication.

Vulnerabilities

- Open redirects lead to token theft
  - Use allow lists
- Improper validation
  - Use allow lists
- Custom implementations
  - Use well established libraries
- Excessive trust on confidentiality of tokens stored in the browser
  - There is no way to make the implicit grant type 100% safe
    - If there is XSS, tokens can be stolen
  - Application storage is better than cookies
    - At least data is not transmitted with every request
  - Session storage is better than application storage
    - Different tabs, different data
  - Browser memory is better than session storage
    - Must study the source code to find data

Questions