ESCAPE

European Science Cluster of Astronomy & Particle physics ESFRI research Infrastructures

ESCAPE WP5 AAI Integration Workshop

Andrea Ceccanti - INFN

andrea.ceccanti@cnaf.infn.it

ESCAPE -The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n°824064

Outline

- Introduction to the ESCAPE AAI
  - ~~Basic AAI concepts: authentication & authorization~~
  - ~~INDIGO IAM: key features~~
  - OAuth and OpenID Connect basics
  - Web application integration demonstration
  - AAI in the ESCAPE data lake demo
    - VOMS authn/z
    - Token-based authn/z

← **Already covered in the <u>ESCAPE AAI Webinar</u>**

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Shared Google doc for feedback/questions

- A shared Google doc is linked to the agenda
  - Open in write access to anybody

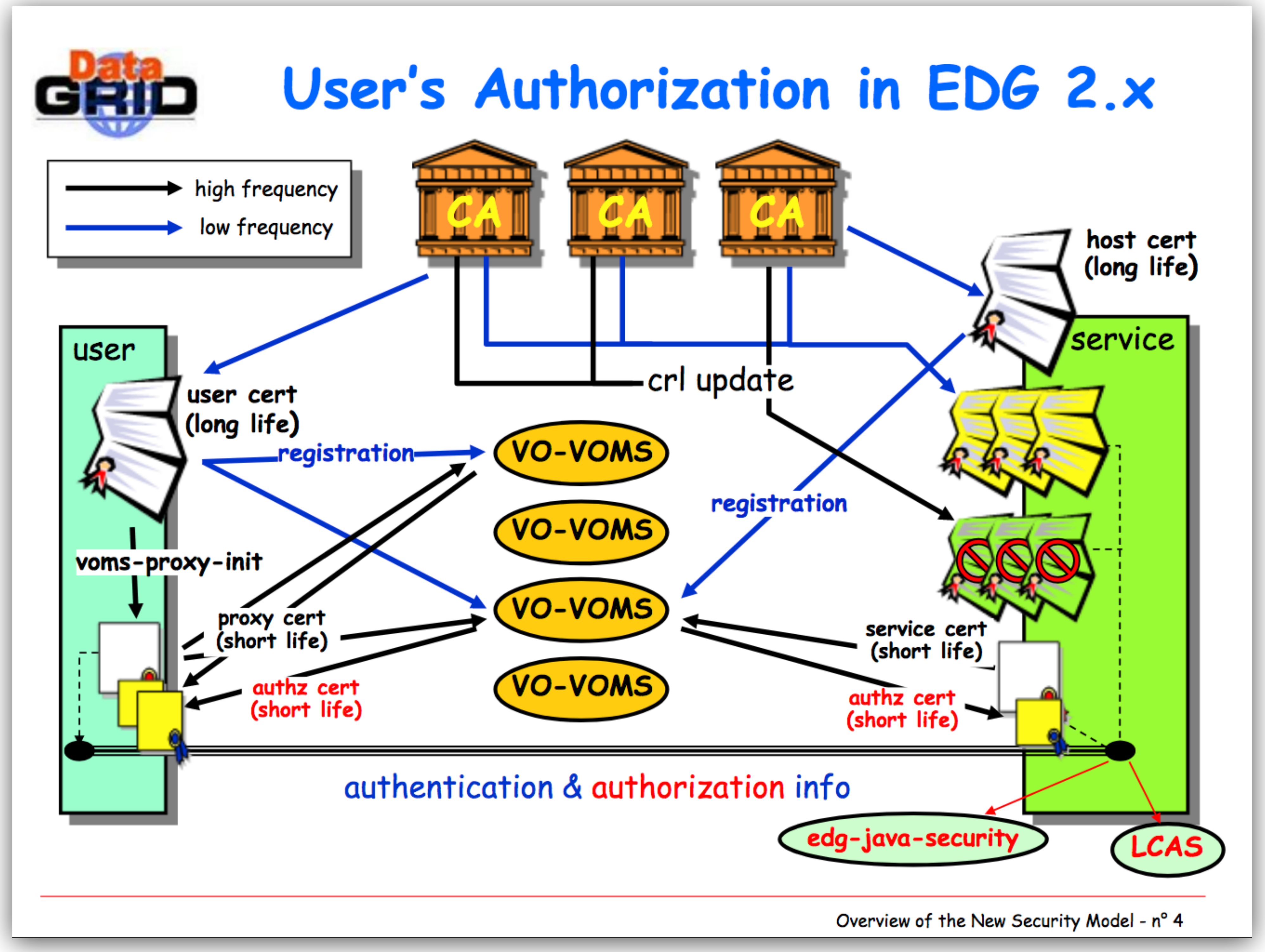
- https://docs.google.com/document/d/1S1AsnQPGHSH3W68Le6VBvbZ2uDAXy-nMHSPC6A0MjZQ/edit#

- Please use it to provide feedback/questions/comments on the Webinar

Introduction to the ESCAPE AAI

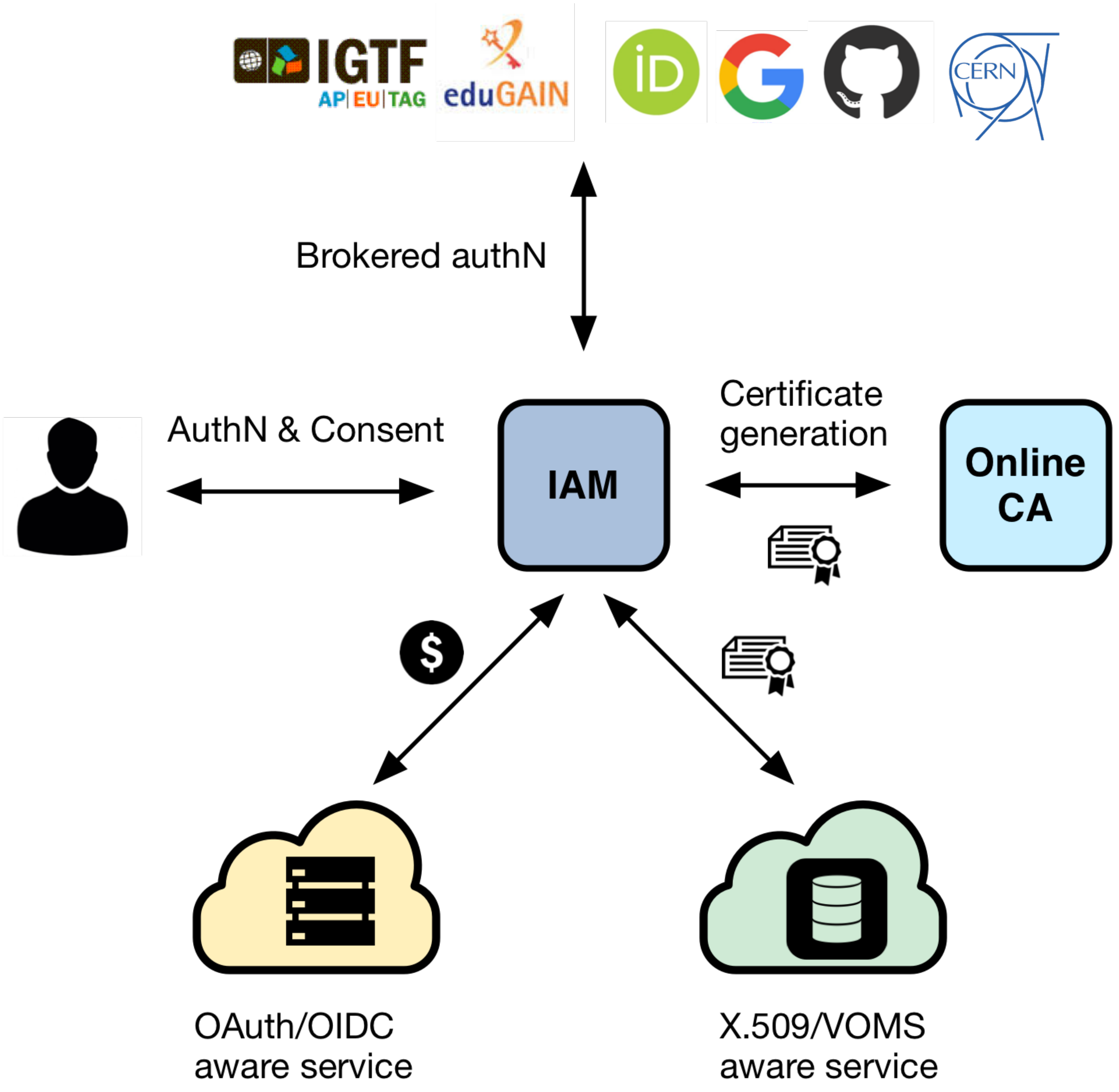
ESCAPE Data Lake AAI and WLCG

Current, X.509 based AAI



Move beyond X.509

Future, token-based AAI



Approach: leverage and build upon the WLCG experience

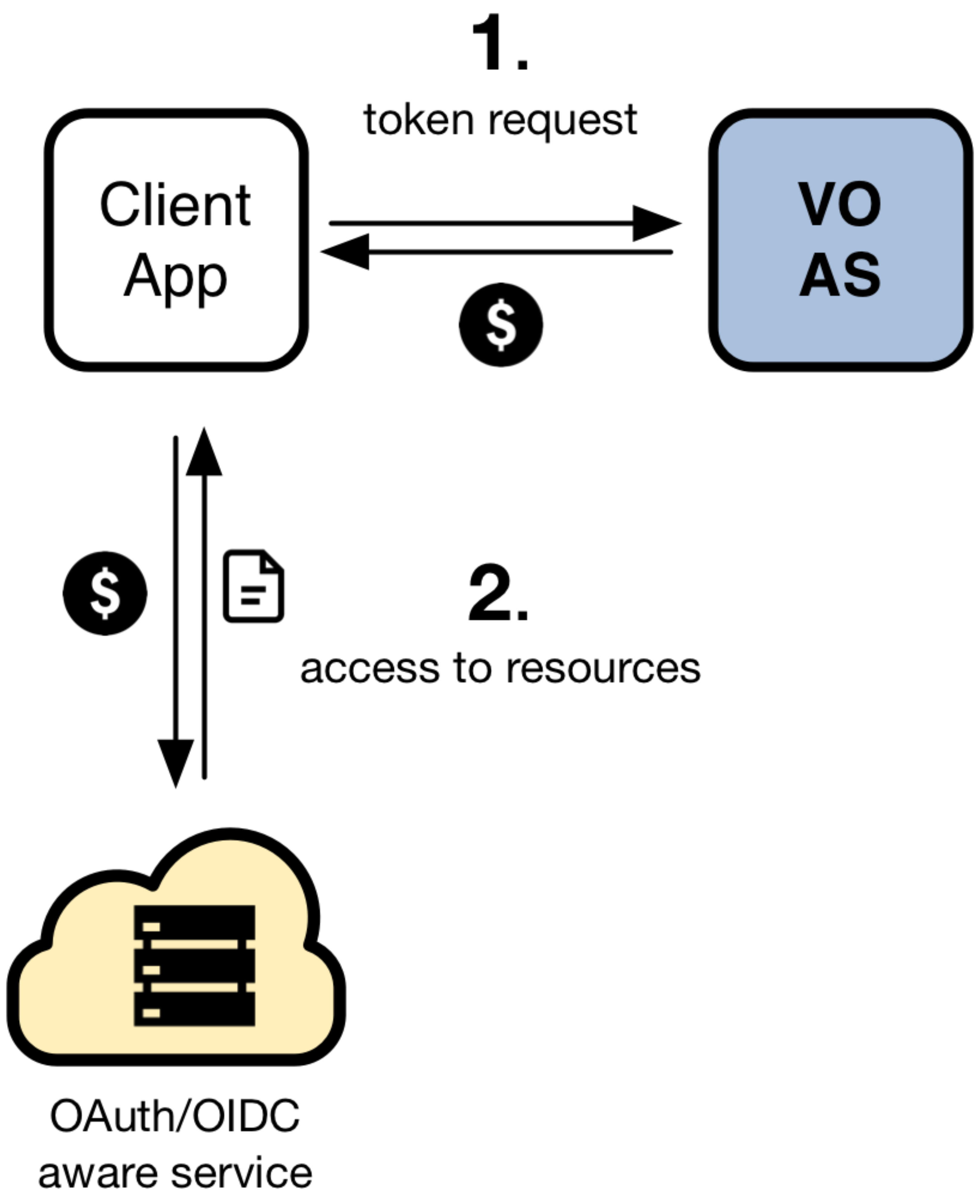
Funded by the European Union's Horizon 2020 - Grant N° 824064

Token-based AuthN/Z from 10000 mt

- In order to access resources/services, a **client application** needs an **access token**

- The token is obtained from **a Virtual Organization** (which acts as an OAuth Authorization Server) using standard **OAuth/OpenID Connect** flows

- **Authorization** is then **performed at the services** leveraging info extracted from the token:
  - **Identity attributes**: e.g., **groups**
  - **OAuth scopes**: capabilities linked to access tokens at token creation time



In practice…

- The central authorization servers provides **attributes** that can be used for authorization at services, e.g.:

- groups/roles, e.g.: **cms**, **lofar**, **production-manager**
  - capabilities, e.g.: **storage.read:/cms, submit-job**

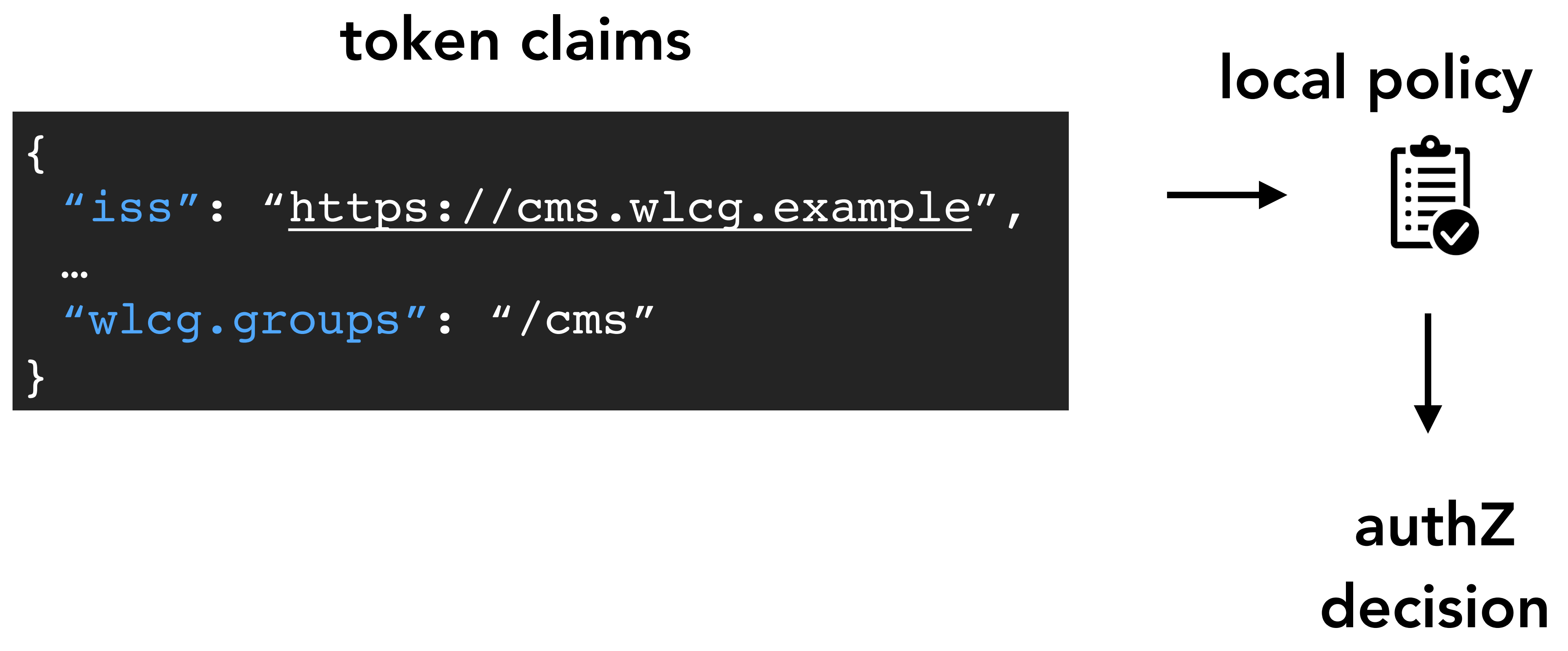
- This information is exposed to services via **signed JWT tokens** and **via OAuth/OpenID Connect protocol message exchanges** (aka flows)

- **Services** can then **grant or deny access** to functionality based on this information. Examples:
  - allow read access on the **/cms** to all members of the **cms** group
  - allow read access on the **/lofar** namespace to anyone with the capability **storage.read:/lofar**

Funded by the European Union's Horizon 2020 - Grant N° 824064

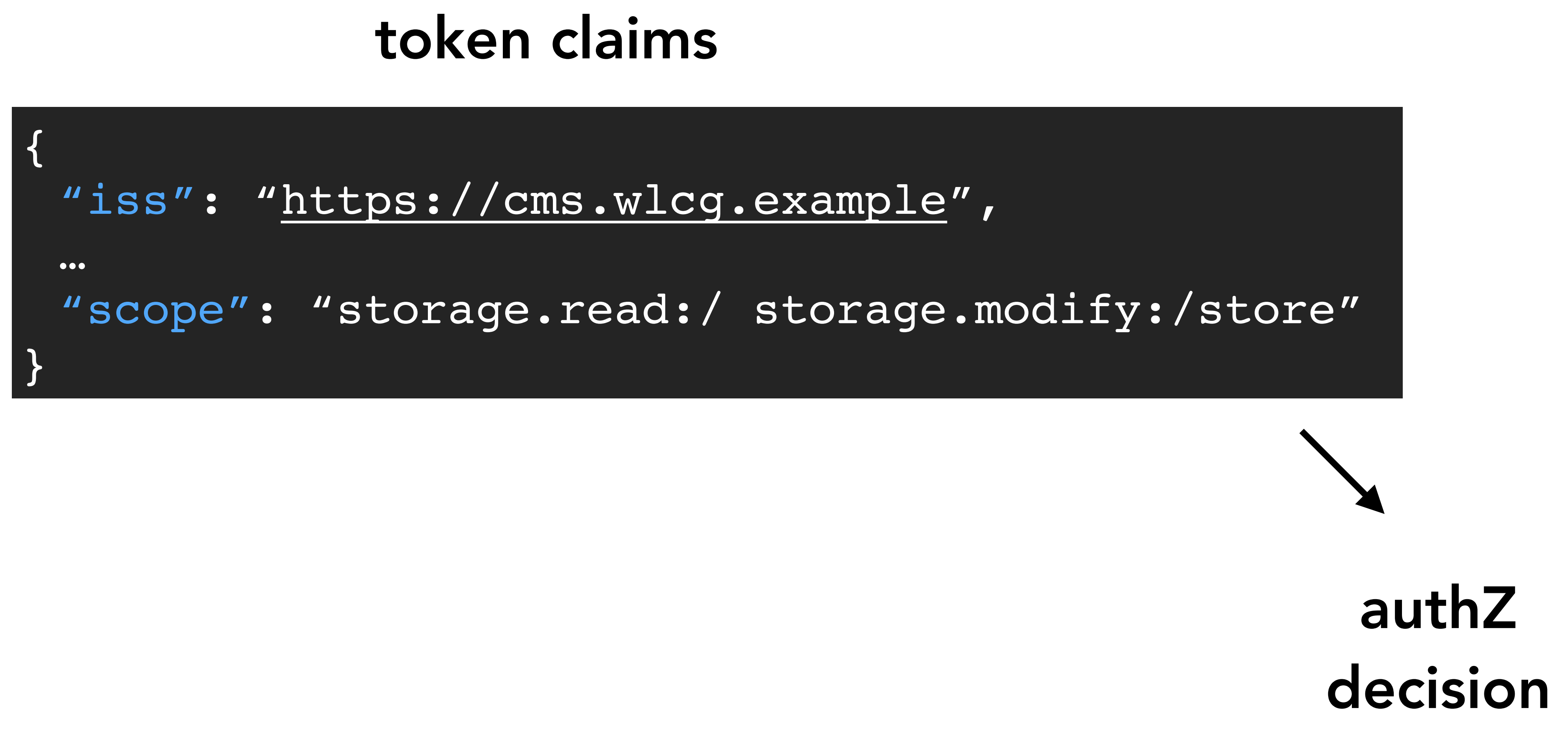
Identity-based vs Scope-based Authorization

- **Identity-based authorization:** the token brings information about attribute ownership (e.g., groups/role membership), the service maps these attributes to a local authorization policy



```
{
  "iss": "https://cms.wlcg.example",
  …
  "wlcg.groups": "/cms"
}
```

- **Scope-based authorization:** the token brings information about which actions should be authorized at a service, the service needs to understand these capabilities and honor them. The authorization policy is managed at the VO level



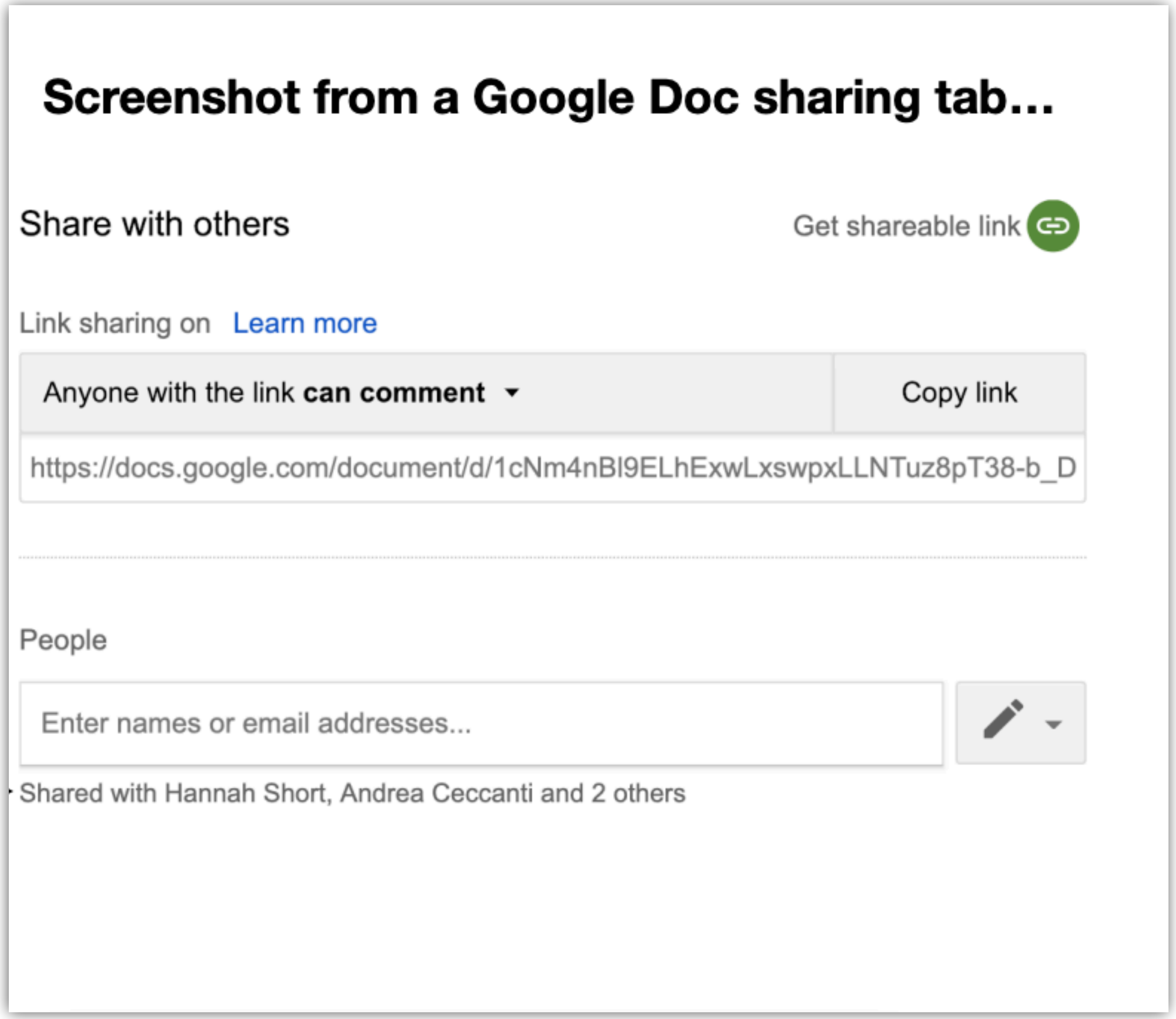
```
{
  "iss": "https://cms.wlcg.example",
  …
  "scope": "storage.read:/ storage.modify:/store"
}
```

Identity-based vs Scope-based Authorization

The two models can coexist, even in the context of the same application!

scope-based authZ →

identity-based authZ →



Screenshot from a Google Doc sharing tab…

Share with others

Get shareable link

Link sharing on Learn more

Anyone with the link **can comment** ▾ | Copy link

https://docs.google.com/document/d/1cNm4nBI9ELhExwLxswpxLLNTuz8pT38-b_D

People

Enter names or email addresses...

Shared with Hannah Short, Andrea Ceccanti and 2 others

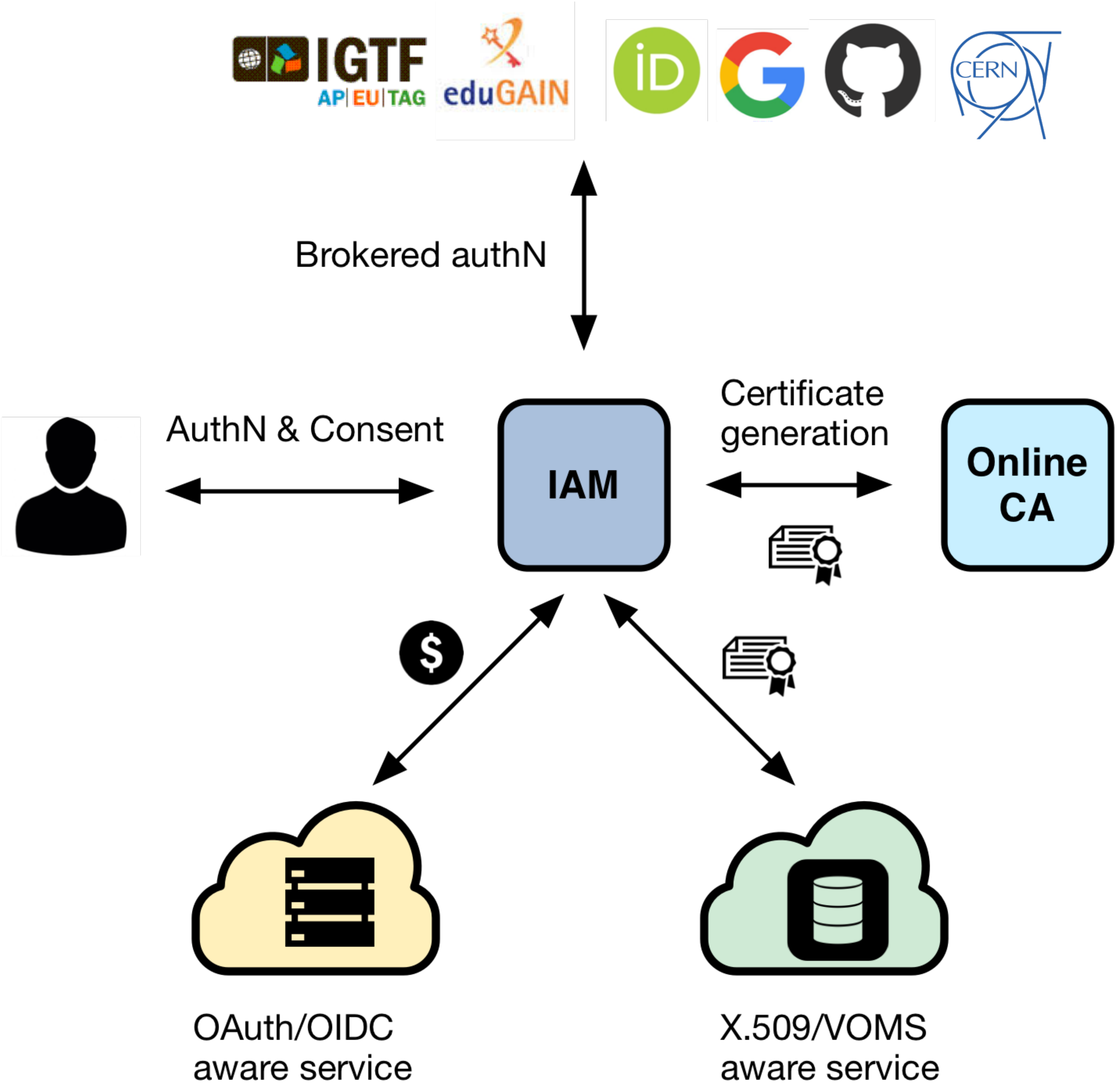
*** Slide courtesy of B. Bockelman**

Funded by the European Union's
Horizon 2020 - Grant N° 824064

INDIGO Identity and Access Management Service

- A **VO*-scoped** authentication and authorization service that
  - supports **multiple authentication mechanisms**
  - provides users with a **persistent, VO-scoped** identifier
  - exposes **identity information**, **attributes** and **capabilities** to services via **JWT** tokens and standard **OAuth & OpenID Connect** protocols
  - can integrate existing **VOMS**-aware services
  - supports **Web** and **non-Web access**, **delegation** and **token renewal**

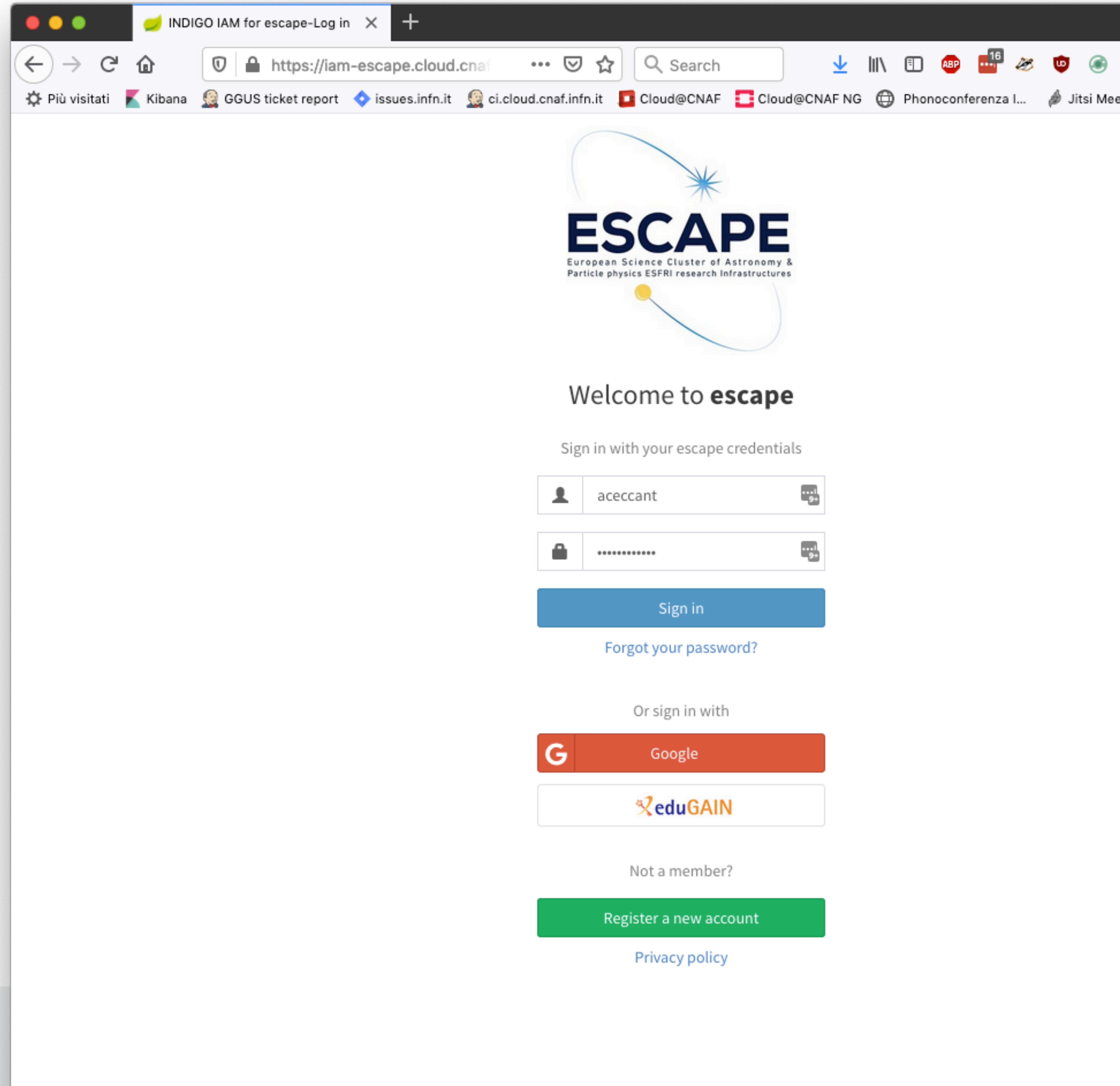
*VO = Virtual Organization



Funded by the European Union's Horizon 2020 - Grant N° 824064

The ESCAPE IAM instance

- Escape IAM instance available
  - Root of trust for the ESCAPE Data Lake
  - 87 registered users
  - 12 groups
  - AuthN with EduGAIN, X.509 certificates, Google, username/password
  - VOMS endpoint available
  - Registration open
    - Administrator-vetted registration flow
  - Documentation available here


INDIGO IAM for escape-Log in
https://iam-escape.cloud.cna
Search
Più visitati
Kibana
GGUS ticket report
issues.infn.it
ci.cloud.cnaf.infn.it
Cloud@CNAF
Cloud@CNAF NG
Phonoconferenza I...
Jitsi Mee
ESCAPE
European Science Cluster of Astronomy & Particle physics ESFRI research Infrastructures

Welcome to **escape**

Sign in with your escape credentials

aceccant

••••••••••••

Enabling technologies

IAM enabling technologies in one slide

- **OAuth 2.0**
  - a standard framework for **delegated authorization**
  - widely adopted in industry

- **OpenID Connect**
  - an **identity layer** built on top of OAuth 2
  - "OAuth-based authentication done right"

BUNDESREPUBLIK DEUTSCHLAND T22000129
FEDERAL REPUBLIC OF GERMANY / REPUBLIQUE FEDERALE D'ALLEMAGNE
PERSONALAUSWEIS
IDENTITY CARD / CARTE D'IDENTITE

Name/Surname/Nom
MUSTERMANN
GEB. GABLER

Vornamen/Given names/Prénoms
ERIKA

Geburtstag/Date of birth/Date de naissance
12.08.1964

Staatsangehörigkeit/Nationality/Nationalité
DEUTSCH

Geburtsort/Place of birth/Lieu de naissance
BERLIN

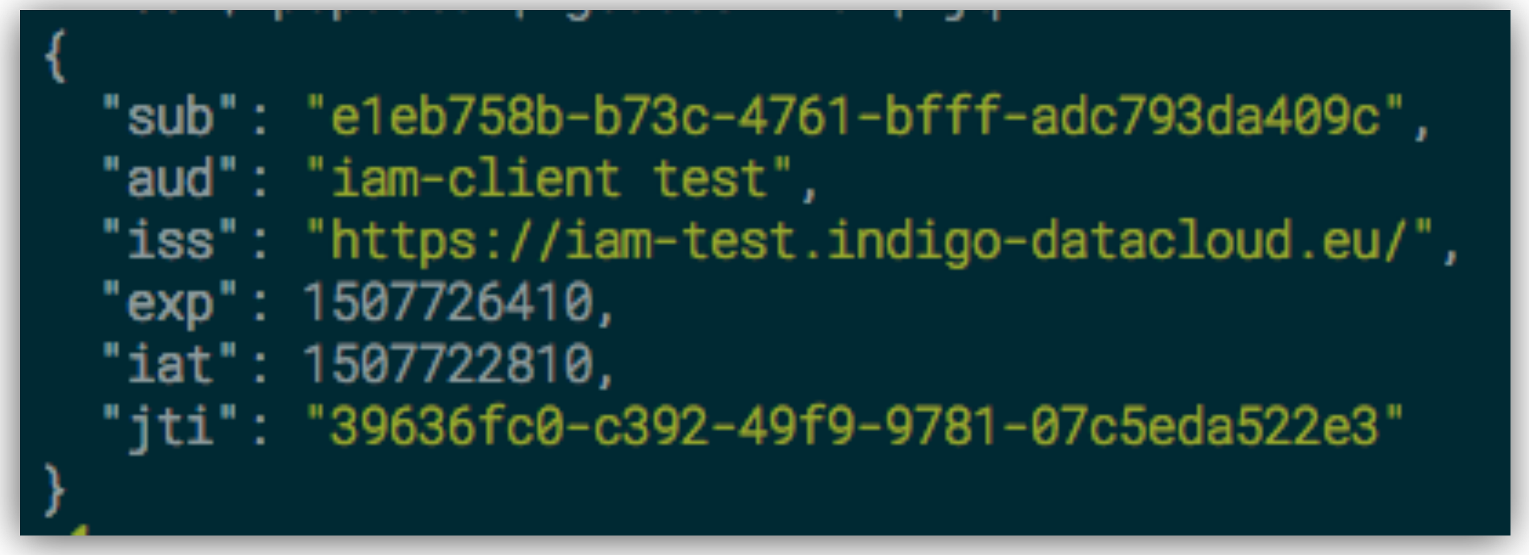
Gültig bis/Date of expiry/Date d'expiration
31.10.2020

938568

Unterschrift der Inhaberin/des Inhabers - Signature of bearer - Signature de la titulaire/du titulaire

MUSTER

- **JSON Web Tokens (JWTs)**
  - a **compact**, **URL-safe** means of representing **claims** to be transferred between two (or more) parties



```
{
  "sub": "e1eb758b-b73c-4761-bfff-adc793da409c",
  "aud": "iam-client test",
  "iss": "https://iam-test.indigo-datacloud.eu/",
  "exp": 1507726410,
  "iat": 1507722810,
  "jti": "39636fc0-c392-49f9-9781-07c5eda522e3"
}
```

Funded by the European Union's
Horizon 2020 - Grant N° 824064

OAuth: a delegated authorization framework

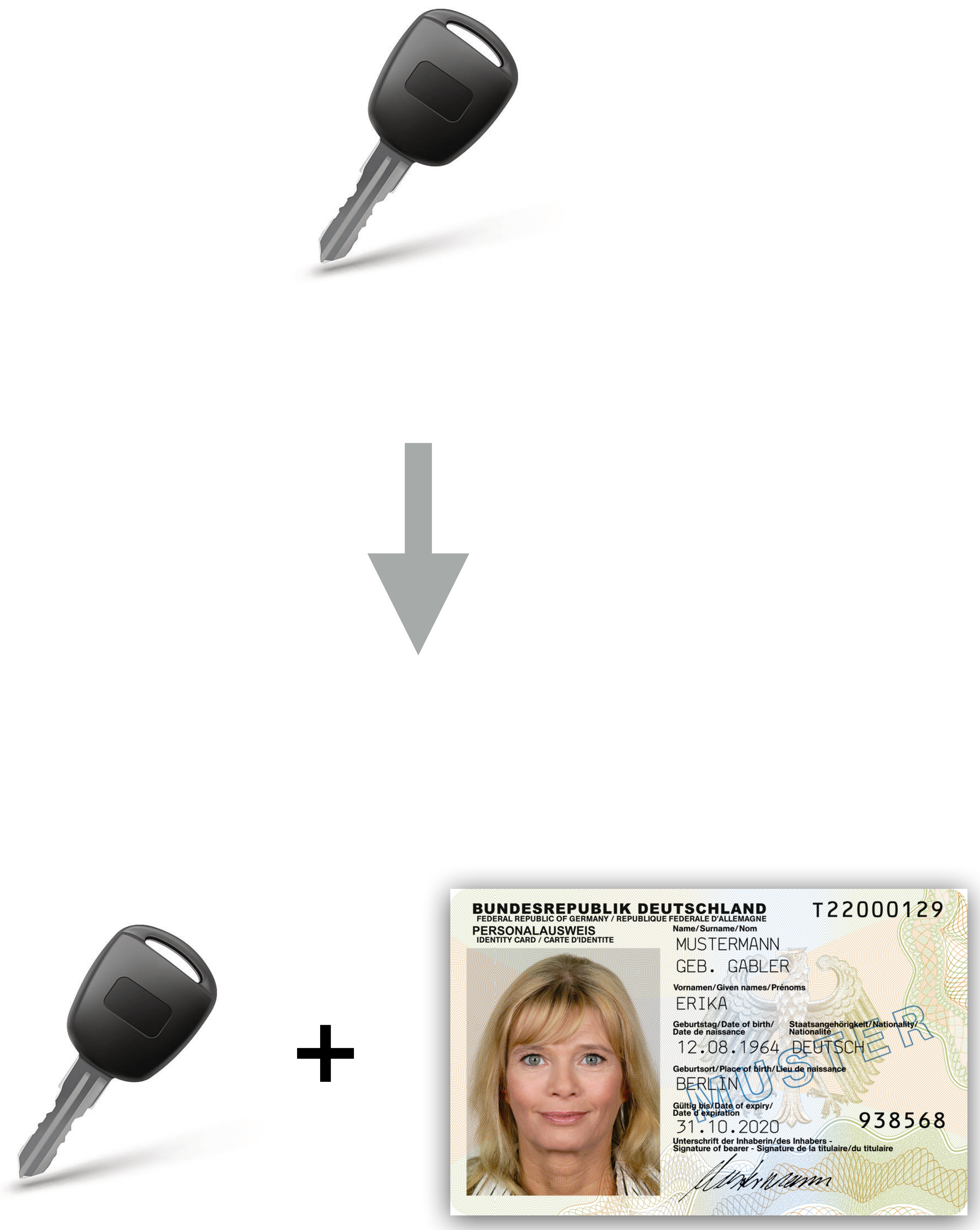
- OAuth defines how **controlled delegation of privileges** can happen among collaborating services

- Provides answers to questions like:
  - How can an application request access to protected resources?
    - How can I obtain **an access token**?
  - How is authorization information exchanged across parties?
    - How is the **access token** presented to **protected resources**? (i.e. API)

Funded by the European Union's
Horizon 2020 - Grant N° 824064

OpenID Connect: an identity layer for OAuth

- OAuth is a **delegated authorization** protocol
  - an **access token** states the **authorization rights** of the client application presenting the token to access some resources
- OpenID Connect extends OAuth to provide a standard **identity layer**
  - i.e. information about **who the user is** and **how it was authenticated** via an additional **ID token (JWT)** and a dedicated **user information query endpoint** at the OpenID Connect Identity provider
  - provides ability to establish **login sessions** (SSO)



JSON Web Tokens (JWT)

- **JSON Web Token** (JWT) is an open standard that defines a compact, self-contained way of securely transmitting information between parties as a JSON object

- JWTs are typically **signed** and, if confidentiality is a requirement, can be **encrypted**.

- JWTs integrity and signatures can be verified **independently** in a **distributed fashion** by relying parties

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Why OAuth, OpenID Connect and JWT?

- Standard, widely adopted in industry
  - Do not reinvent the wheel, reuse existing knowledge and tools, extend when needed

- Reduced integration complexity at relying services
  - Off-the-shelf libraries and components

- Authentication-mechanism agnostic
  - The AAI is not bound to a specific authentication mechanism

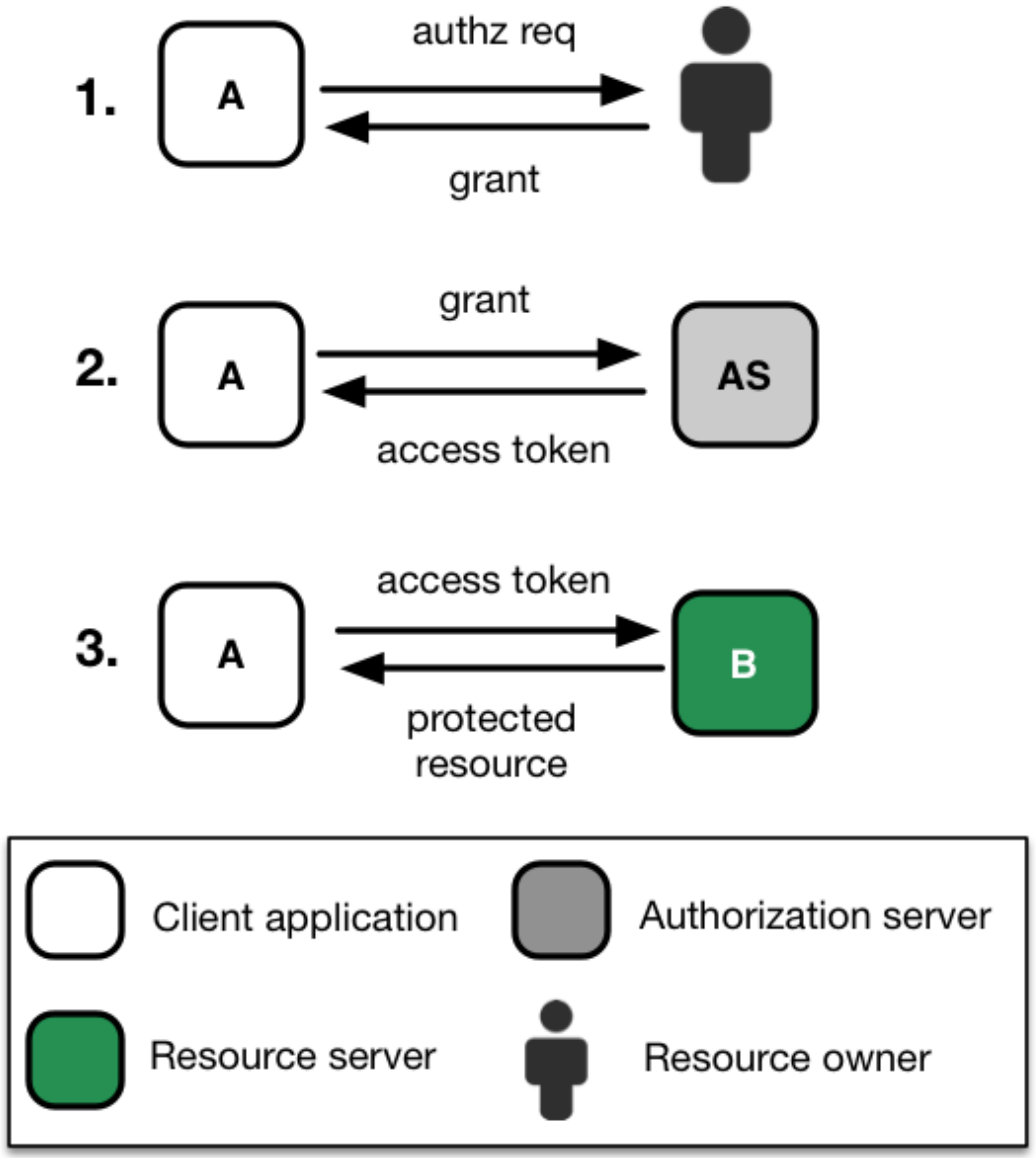
- Distributed verification of access and identity tokens
  - It scales

Funded by the European Union's
Horizon 2020 - Grant N° 824064

A brief introduction to OAuth, OpenID Connect and JWTs

OAuth roles

- **Resource owner**
  - A user that owns resources hosted at a service
- **Client**
  - An application that wants to have access to user resources
- **Authorization server**
  - A service that authenticates users and client applications and issues access tokens according to some policy
- **Resource server**
  - A service that holds protected resources and grants access based on access tokens issued by the authorization server



Funded by the European Union's
Horizon 2020 - Grant N° 824064

OAuth/OpenID Connect actors and roles

Actor	Role	Example
Authorization Server (AS)	Asserting party	ESCAPE IAM instance
Resource Server (RS)	Relying party	The RUCIO REST API
Client	Relying party	The RUCIO Web Dashboard
Resource Owner	Subject	A registered IAM ESCAPE user

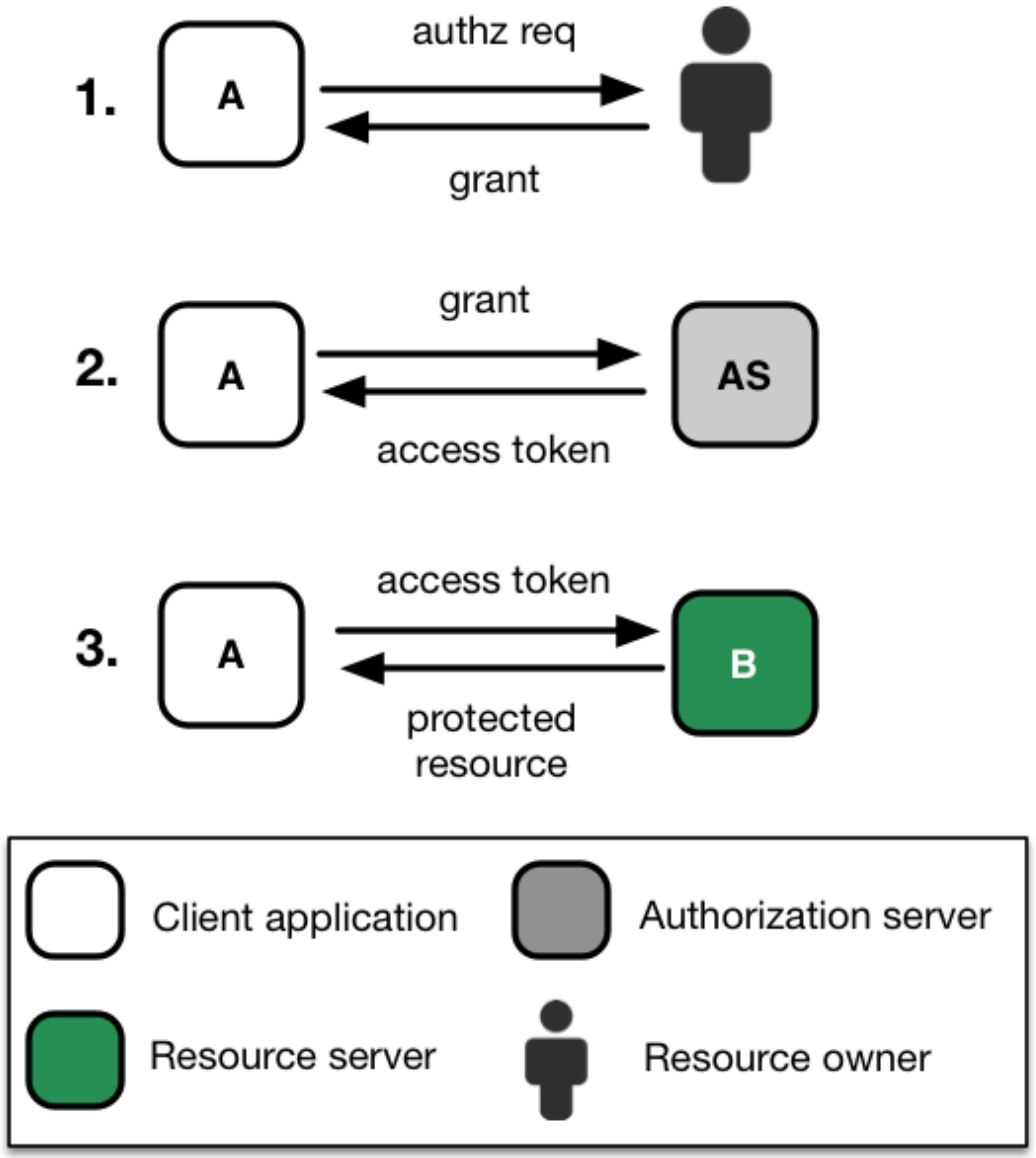
Funded by the European Union's
Horizon 2020 - Grant N° 824064

OAuth client registration

- In OAuth clients that interact with an Authorization Server (AS) need to be **registered**

- When a client is registered, it typically receives the client **credentials**
  - **client_id:** the client "username"
  - **client_secret:** the client "password"

- Credentials are required in some OAuth/OpenID Connect flows or to access specific endpoints, where different privileges may be assigned to different clients



Funded by the European Union's Horizon 2020 - Grant N° 824064

OAuth client types

https://tools.ietf.org/html/rfc6749#section-2.1

- **confidential:** Clients capable of maintaining the confidentiality of their credentials (e.g., client implemented on a secure server with restricted access to the client credentials), or capable of secure client authentication using other means

- **public:** Clients incapable of maintaining the confidentiality of their credentials (e.g., clients executing on the device used by the resource owner, such as an installed native application or a web browser-based application), and incapable of secure client authentication via any other means.

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Handling client credentials

- Client credentials must be maintained confidential
  - **not** stored in Docker images or source code
    - use ENV variables or other secret management mechanisms to pass secrets to your application

- Follow recommendations in the client app security section of the OAuth security recommendations
  - https://tools.ietf.org/html/rfc6819#section-5.3

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Client registration in practice

- To register a new client in IAM, follow the instructions in the documentation:
  - https://indigo-iam.github.io/docs/v/current/user-guide/client-registration.html

- Client registration is necessary to integrate any application that needs to “drive” an authorization flow
  - i.e., if your app needs to show a “Login with ESCAPE” button, you need to register a client

- For protected resources (APIs) integration, registration is **NOT** needed

Funded by the European Union’s
Horizon 2020 - Grant N° 824064

OAuth/OpenID Connect grant types

Authorization grant types

=

Authorization Flows

=

Ways for an application to get tokens

OAuth/OpenID Connect grant types

Grant Type	Context	Client type
Authorization code	Server-side apps	Confidential
Implicit	Client-side, Javascript apps	Public
Device code	Limited-input devices, CLIs	Confidential
Resource owner password credentials	Trusted apps, CLIs	Confidential
Client credentials	Server-side apps	Confidential
Refresh token	Server-side apps	Confidential
Token exchange	Server-side apps	Confidential

Funded by the European Union's
Horizon 2020 - Grant N° 824064

OAuth/OpenID Connect provider metadata

- OAuth & OpenID Connect provide a standard way to expose the authorization server/OpenID provider configuration to clients

- Information is published at **a well-known endpoint** for the server, e.g.:
  - https://dodas-iam.cloud.cnaf.infn.it/**.well-known/openid-configuration**

- Clients can use this information to know about
  - location of key material used to sign/encrypt tokens
  - supported grant types/authorization flows
  - endpoint locations
  - supported claims
  - …

- and implement **automatic client configuration**

Funded by the European Union's
Horizon 2020 - Grant N° 824064

OAuth/OpenID Connect provider metadata

Example metadata document:

<u>https://iam-escape.cloud.cnaf.infn.it/.well-known/openid-configuration</u>

Funded by the European Union's
Horizon 2020 - Grant N° 824064

OAuth scopes

- OAuth provides **scopes** as a standard mechanism to express authorization permissions granted to client applications

- In practice, scopes are a set of strings included in an access token that limit what are the operations that can be authorized by clients presenting such access token
  - User consent is based on scopes requested

- OAuth scopes are commonly used in industry to define the authorization on service APIs. Examples:
  - https://api.slack.com/docs/oauth-scopes
  - https://developer.github.com/apps/building-oauth-apps/understanding-scopes-for-oauth-apps/#available-scopes
  - https://developers.google.com/identity/protocols/googlescopes

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Commonly used OAuth/OIDC scopes

- `openid`: signals that the client wants to receive authentication information about the user

- `profile`: used to request profile information (name, address and other information)

- `email`: used to request access to the user's email (name, address)

- `offline_access`: used to request refresh tokens

- `wlcg.groups`: used to request the inclusion of group information in tokens (when the WLCG JWT profile is used in IAM)

Funded by the European Union's
Horizon 2020 - Grant N° 824064

OAuth bearer token usage

- There's a standard that defines how to send tokens to resource servers
- Typically, tokens are sent in the **`Authorization`** HTTP header, following the rules defined in RFC 6750, as in the following example HTTP request

```
GET / HTTP/1.1
Host: apache.test.example
Authorization: Bearer eyJraWQiOiJy…rYI
User-Agent: curl/7.65.3
Accept: */*
```

The token!

Funded by the European Union's
Horizon 2020 - Grant N° 824064

JSON Web Tokens: definition

Citing RFC 7519:

- **JSON Web Token** (JWT) is a compact, URL-safe means of representing claims to be transferred between two parties.

- The **claims** in a JWT are **encoded as a JSON object** that is used as the payload of a JSON Web Signature (JWS) structure OR as the plaintext of a JSON Web Encryption (JWE) structure, **enabling the claims to be digitally signed or integrity protected with a Message Authentication Code (MAC) and/or encrypted**.

Funded by the European Union's
Horizon 2020 - Grant N° 824064

JWT

Citing RFC 7519:

- A JWT is represented as a sequence of **URL-safe parts** separated by period ('.') characters. Each part contains a **base64url-encoded value**.

- The number of parts in the JWT is dependent upon the representation of the resulting JSON Web Signature (JWS) using the JWS Compact Serialization or JSON Web Encryption (JWE) using the JWE Compact Serialization.

Funded by the European Union's
Horizon 2020 - Grant N° 824064

JWT: Header.Body.Signature

```
eyJraWQiOiJyc2ExIiwiYWxnIjoiUlMyNTYifQ.eyJ3bGNnLnZlciI6IjEuMCIsInN
1YiI6IjI0MTY4N2U4LTUzNzQtNDU0OS1iOWY2LWEzODY2ZjBiZjZkYSIsImF1ZCI6I
mh0dHBzOlwvXC93bGNnLmNlcm4uY2hcL2p3dFwvdjFcL2FueSIsIm5iZiI6MTYxMDk
4MzAzOCwic2NvcGUiOiJvcGVuaWQgcHJvZmlsZSB3bGNnLmdyb3VwcyIsImlzcyI6I
mh0dHBzOlwvXC9pYW0tZXNjYXBlLmNsb3VkLmNuYWYuaW5mbi5pdFwvIiwiZXhwIjo
xNjEwOTg2NjM4LCJpYXQiOjE2MTA5ODMwMzgsImp0aSI6IjA5NjIwZTQ3LWE5NTQtN
GZjNS1hMzMxLTE1NDBiMmU0MjYzYyIsImNsaWVudF9pZCI6IjEyMDIwYjM1LTQ0ZTI
tNDljYS1hODU2LWQwNzE2OTUyNzkwZCIsIndsY2cuZ3JvdXBzIjpbIlwvZXNjYXBlI
iwiXC9lc2NhcGVcL2NtcyIsIlwvZXNjYXBlXC9waWxvdHMiLCJcL2VzY2FwZVwveGZ
lcnMiXX0.b64QOAjMoQfcJtin6hTLxtUepqjbbZ9pmb4xp5MoXeM3d4TyY1OIyQtcg
eZl4_mAzc22thTLbtu675xM7LswfrqFdc9eNPqi2VQzpdYae4S-
bK_3r9Dev-8o7PKiHNLtytNTK6Djre8WQF2TUX-oHsDqP2EJDskuqu-GAdhjLVI
```

Funded by the European Union's
Horizon 2020 - Grant N° 824064

JWS compact serialization form

- From https://tools.ietf.org/html/rfc7515#section-3.1

In the JWS Compact Serialization, a JWS is represented as the concatenation:

BASE64URL(UTF8(JWS Protected Header)) || '.' ||

BASE64URL(JWS Payload) || '.' ||

BASE64URL(JWS Signature)

Funded by the European Union's Horizon 2020 - Grant N° 824064

JWT: Header.Body.Signature

Header

```
{
  "kid": "rsa1",
  "alg": "RS256"
}
```

Body

```
{
  "wlcg.ver": "1.0",
  "sub": "241687e8-5374-4549-b9f6-a3866f0bf6da",
  "aud": "https://wlcg.cern.ch/jwt/v1/any",
  "nbf": 1610983038,
  "scope": "openid profile wlcg.groups",
  "iss": "https://iam-escape.cloud.cnaf.infn.it/",
  "exp": 1610986638,
  "iat": 1610983038,
  "jti": "09620e47-a954-4fc5-a331-1540b2e4263c",
  "client_id": "12020b35-44e2-49ca-a856-d0716952790d",
  "wlcg.groups": [
    "/escape",
    "/escape/cms",
    "/escape/pilots",
    "/escape/xfers"
  ]
}
```

Signature

```
b64QOAjMoQfcJtin6hTLxtUep
qjbbZ9pmb4xp5MoXeM3d4TyY1
OIyQtcgeZl4_mAzc22thTLbtu
675xM7LswfrqFdc9eNPqi2VQz
pdYae4S-
bK_3r9Dev-8o7PKiHNLtytNTK
6Djre8WQF2TUX-
oHsDqP2EJDskuqu-GAdhjLVI
```

Funded by the European Union's
Horizon 2020 - Grant N° 824064

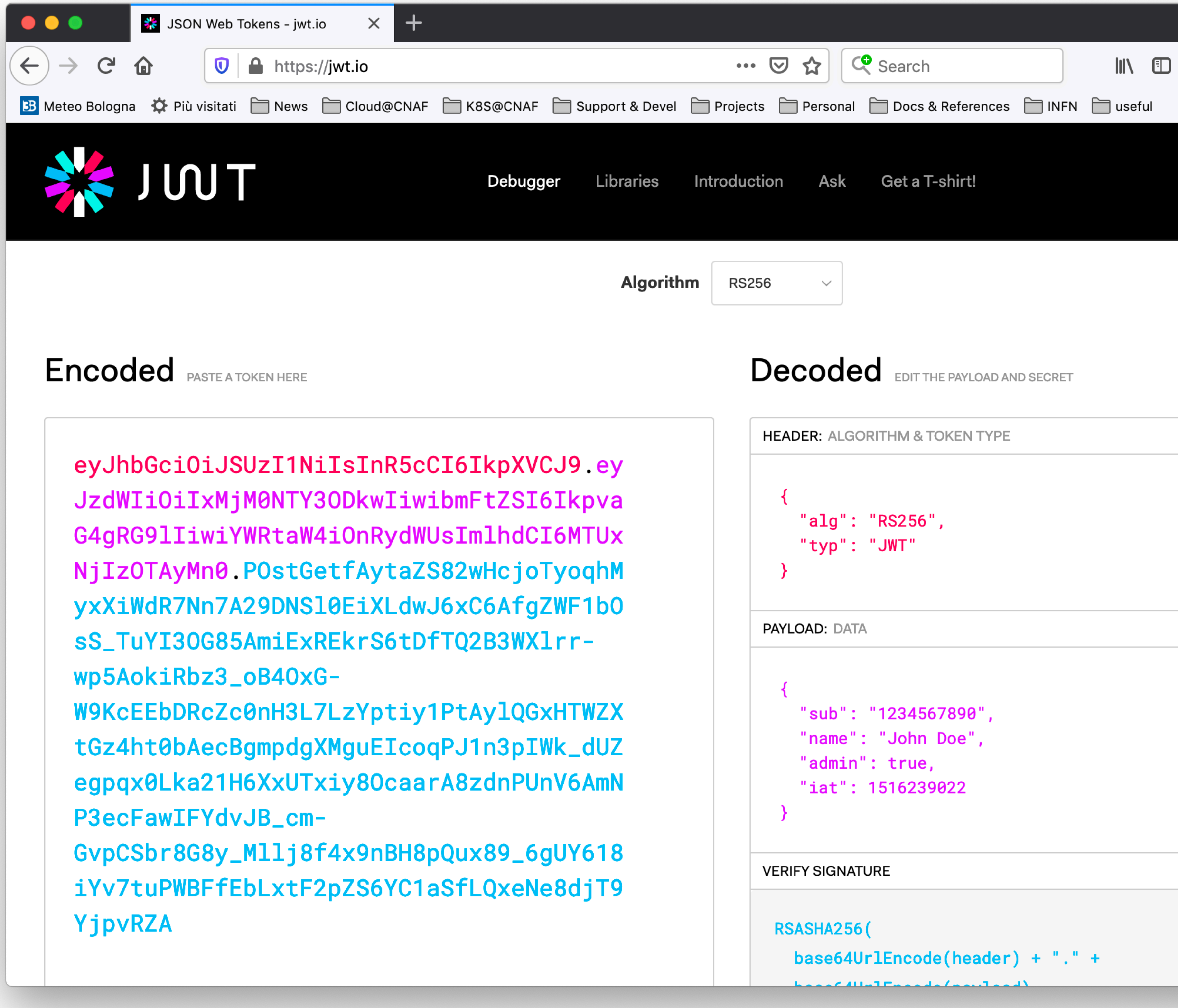
JWT claim names

- Registered claim names (i.e. a set of basic claims defined by the JWT standard
  - “iss” (Issuer): the principal that issued the JWT (e.g., IAM ESCAPE)
  - “sub” (Subject): the principal that is the subject of the JWT (e.g., a unique id linked to an IAM account)
  - “aud” (Audience): identifies the recipients that the JWT is intended for (e.g., RUCIO)
  - “exp” (Expiration time): identifies the expiration time on or after which the JWT MUST NOT be accepted for processing
  - “nbf” (Not before): identifies the time before which the JWT MUST NOT be accepted for processing
  - “iat” (Issued at): identifies the time at which the JWT was issued
  - “jti” (JWT ID): provides a unique identifier for the JWT
- Public claim names
  - Either a registered public claim name or one that has a collision-resistant name
- Private claim names
  - Claim names that are not registered or public (i.e. are not collision-resistant)

Funded by the European Union’s
Horizon 2020 - Grant N° 824064

Exercise: JWT Signature verification

- Get a token from the IAM ESCAPE instance
- Go to jwt.io, paste the token in token field
- Go to https://iam-escape.cloud.cnaf.infn.it/jwt, grab the key and paste it in the verify signature box



Algorithm RS256

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.ey
JzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6Ikpva
G4gRG9lIiwiYWRtaW4iOnRydWUsImlhdCI6MTUx
NjIzOTAyMn0.POstGetfAytaZS82wHcjoTyoqhM
yxXiWdR7Nn7A29DNSl0EiXLdwJ6xC6AfgZWF1bO
sS_TuYI3OG85AmiExREkrS6tDfTQ2B3WXlrr-
wp5AokiRbz3_oB4OxG-
W9KcEEbDRcZc0nH3L7LzYptiy1PtAylQGxHTWZX
tGz4ht0bAecBgmpdgXMguEIcoqPJ1n3pIWk_dUZ
egpqx0Lka21H6XxUTxiy8OcaarA8zdnPUnV6AmN
P3ecFawIFYdvJB_cm-
GvpCSbr8G8y_Mllj8f4x9nBH8pQux89_6gUY618
iYv7tuPWBFfEbLxtF2pZS6YC1aSfLQxeNe8djT9
YjpvRZA
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{
  "alg": "RS256",
  "typ": "JWT"
}
```

PAYLOAD: DATA

```
{
  "sub": "1234567890",
  "name": "John Doe",
  "admin": true,
  "iat": 1516239022
}
```

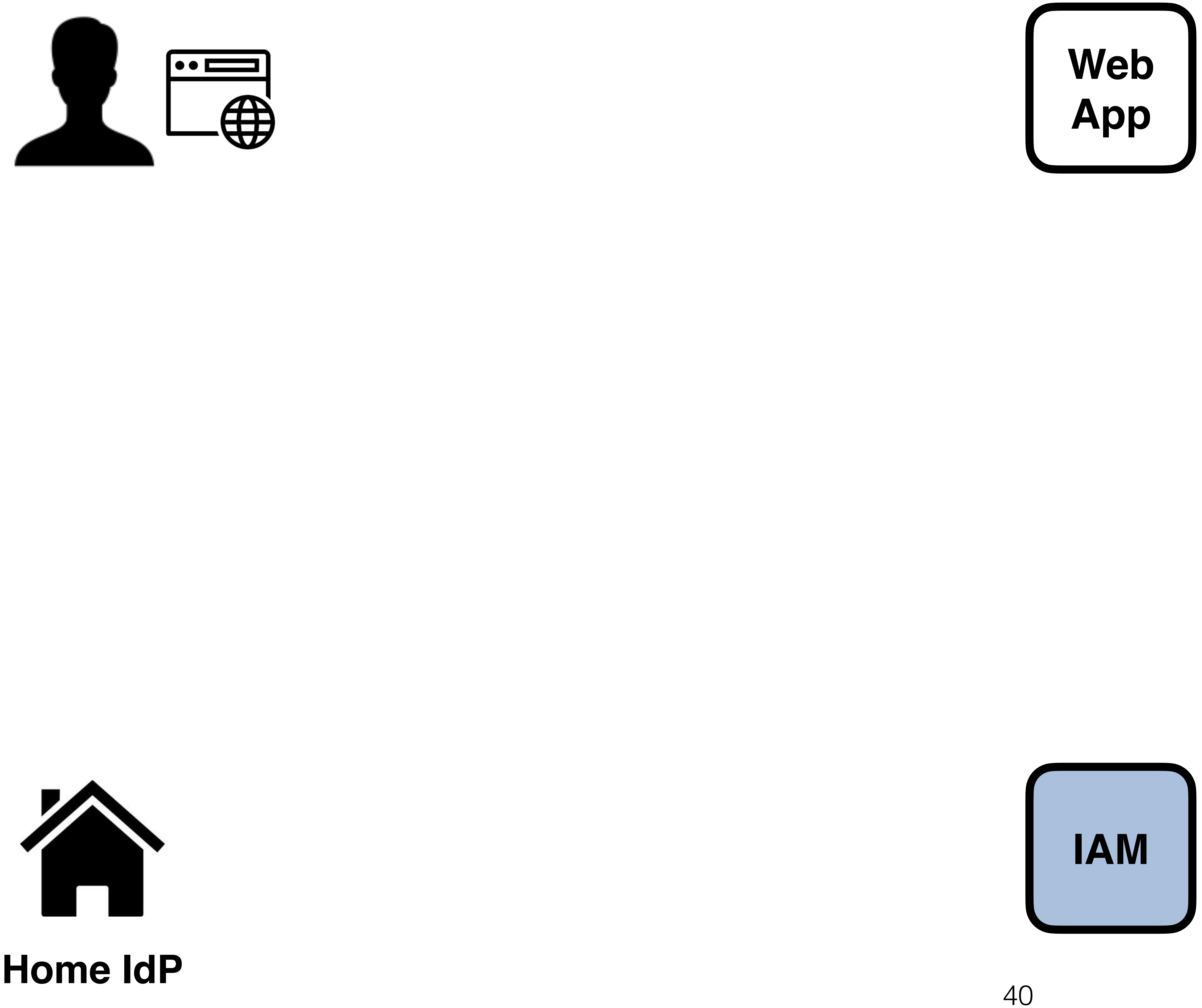
VERIFY SIGNATURE

```
RSASHA256(
  base64UrlEncode(header) + "." +
```

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Web application integration scenario

Web application: authorization code flow



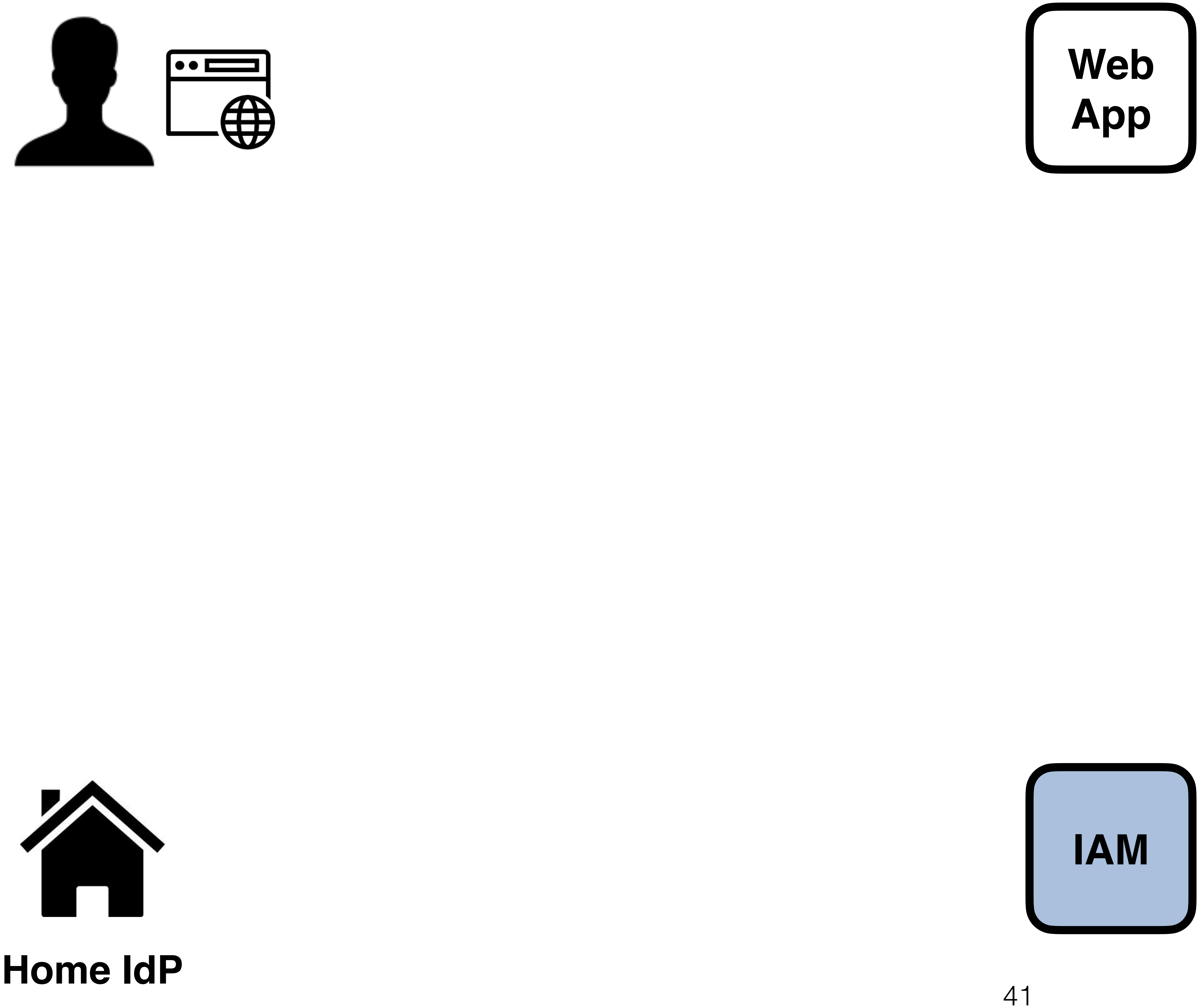
Web App

IAM

Home IdP

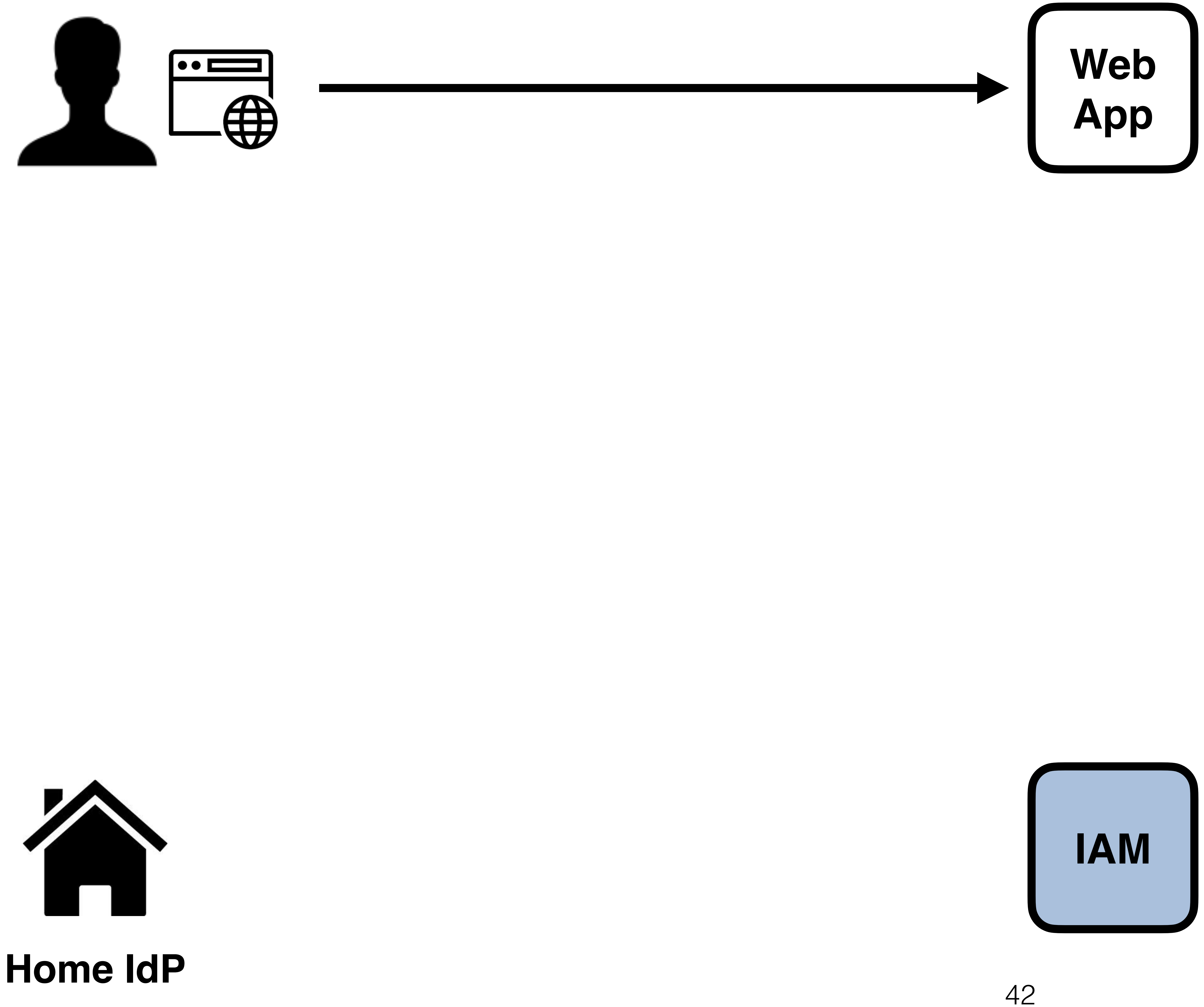
A Web App integrates with IAM to **delegate user authentication management** and **obtain authorization** information

Web application: authorization code flow



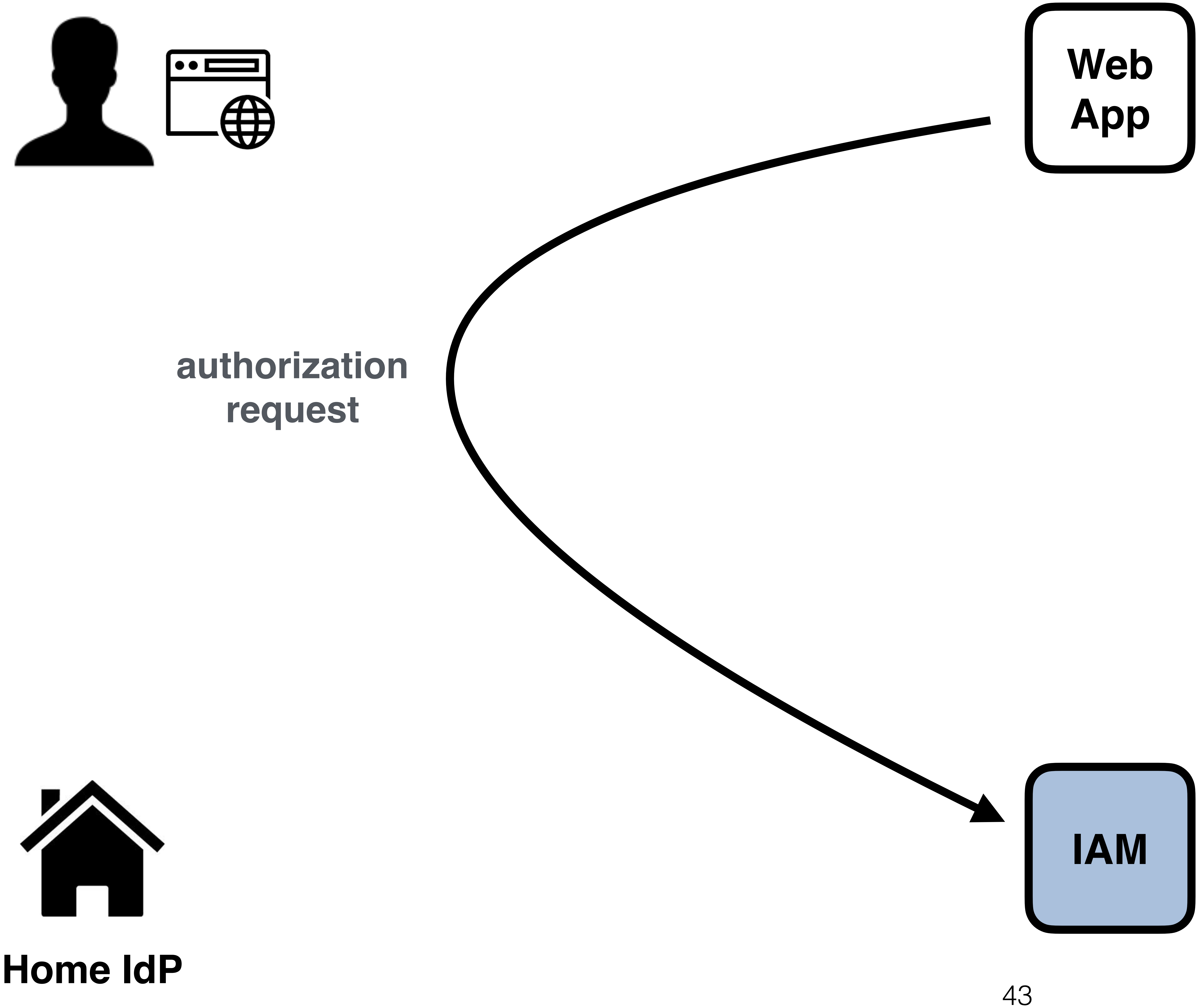
OAuth and OpenID connect provide the **authorization code flow** in support of this integration use case

Web application: authorization code flow



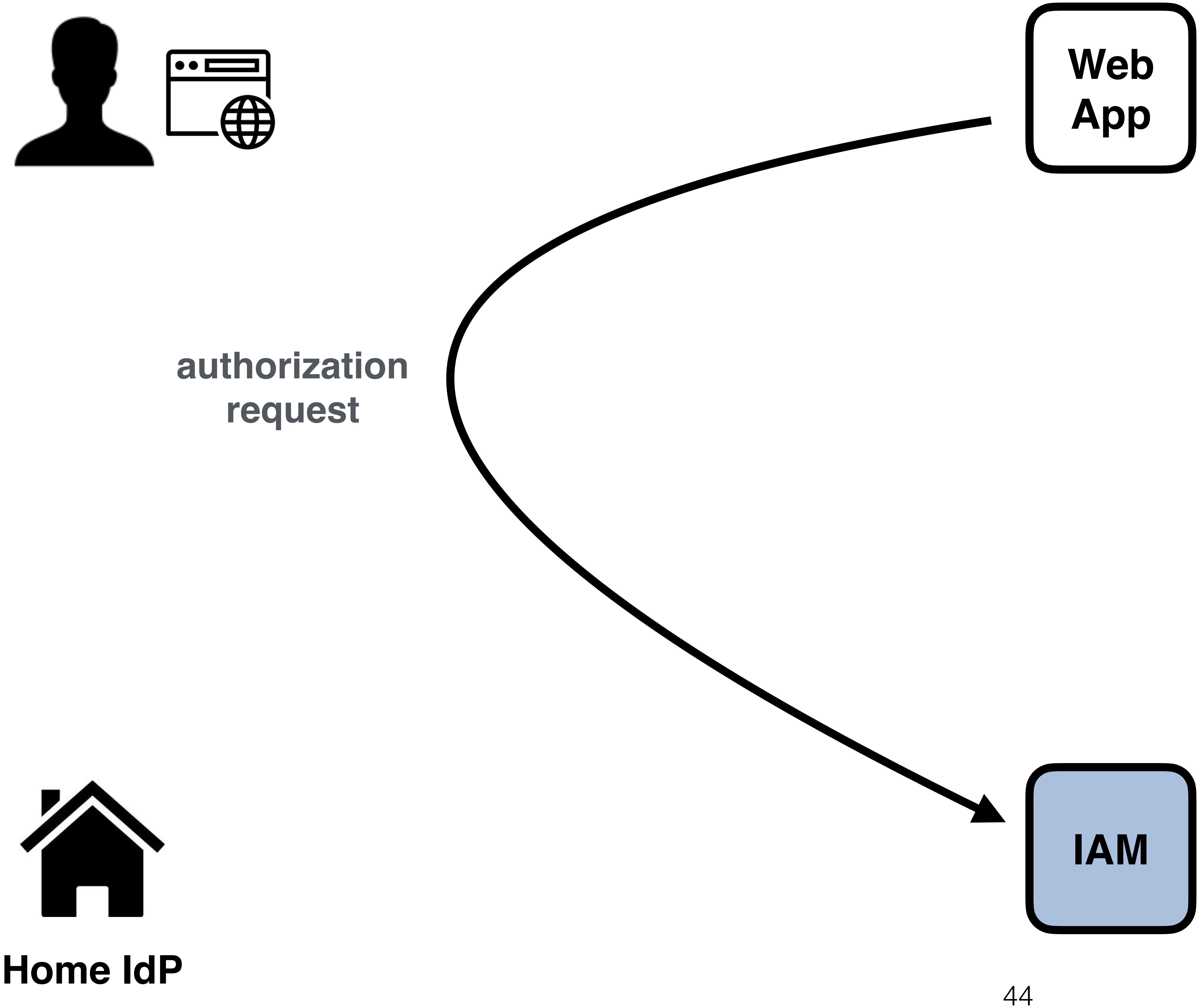
User points its browser to web app, which redirects back to IAM for authentication

Web application: authorization code flow



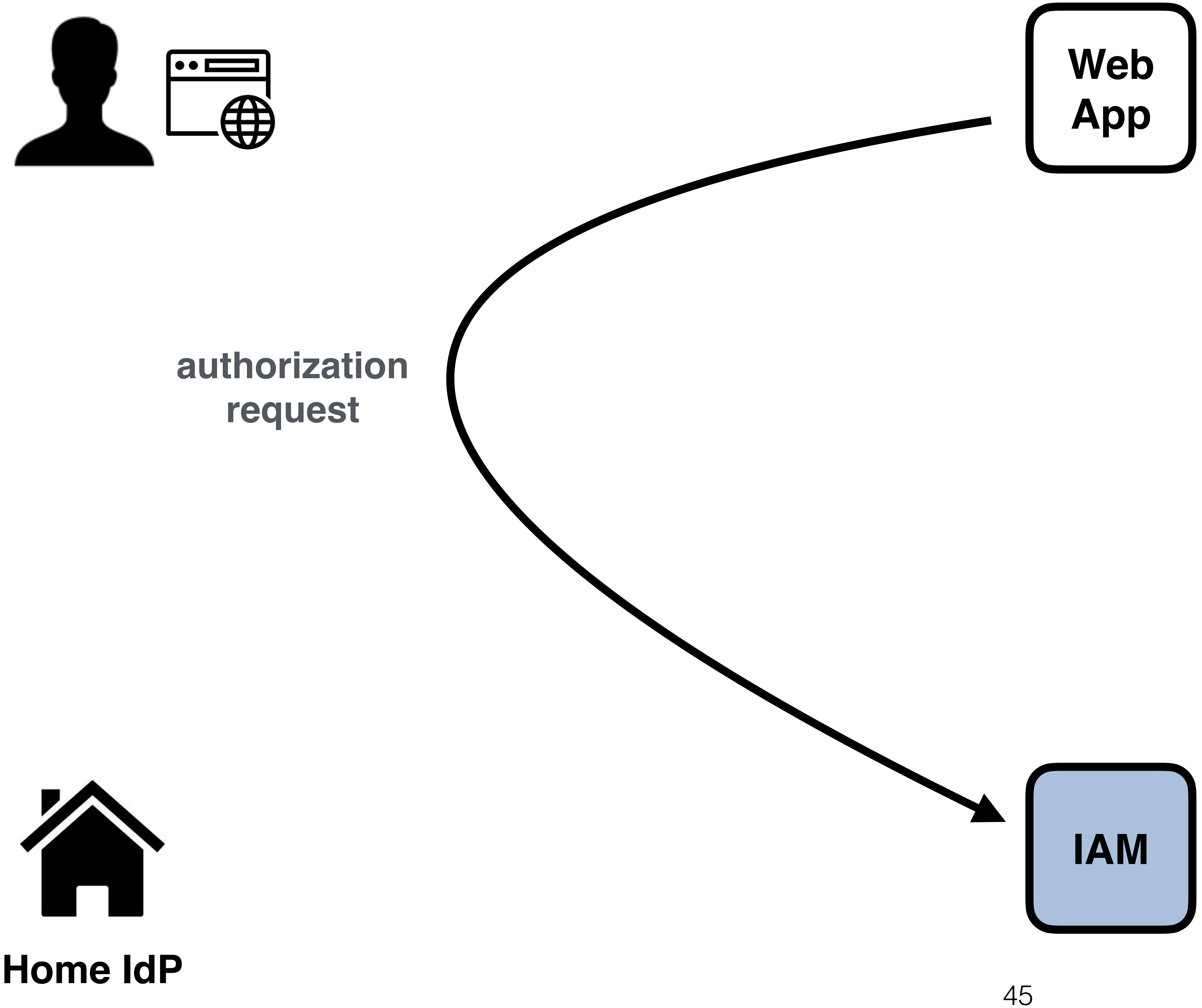
User points its browser to web app, which redirects back to IAM for authentication

Web application: authorization code flow



This authorization request starts the authorization flow, and includes parameters (e.g., OAuth scopes) that will influence which information is returned by IAM

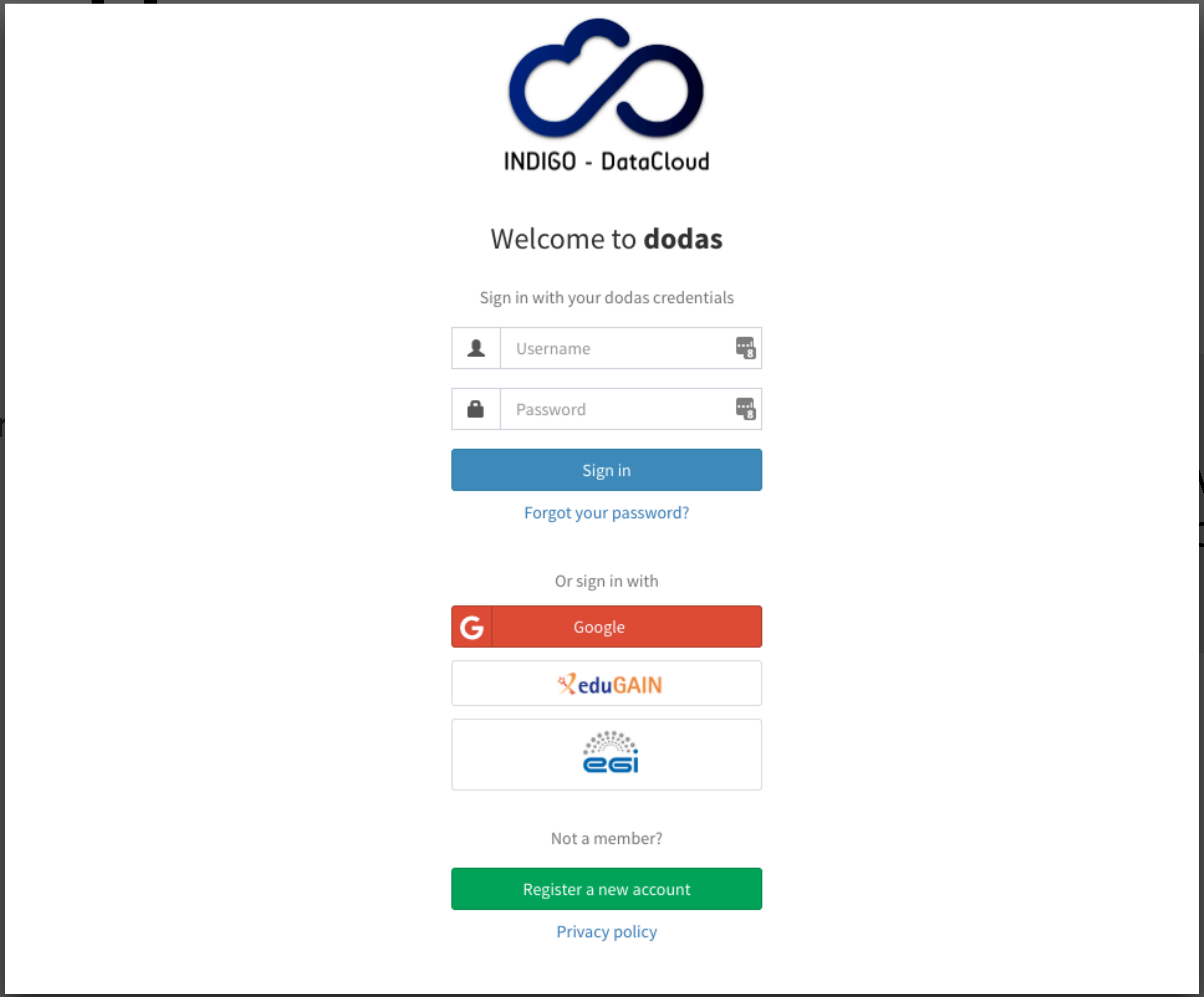
Web application: authorization code flow



User does not have a valid session at IAM, so IAM shows the login page

Web application: authorization code flow

authorizatior
request



INDIGO - DataCloud

Welcome to **dodas**

Sign in with your dodas credentials

Username

Password

Sign in

Forgot your password?

Or sign in with

Google

eduGAIN

egi

Not a member?

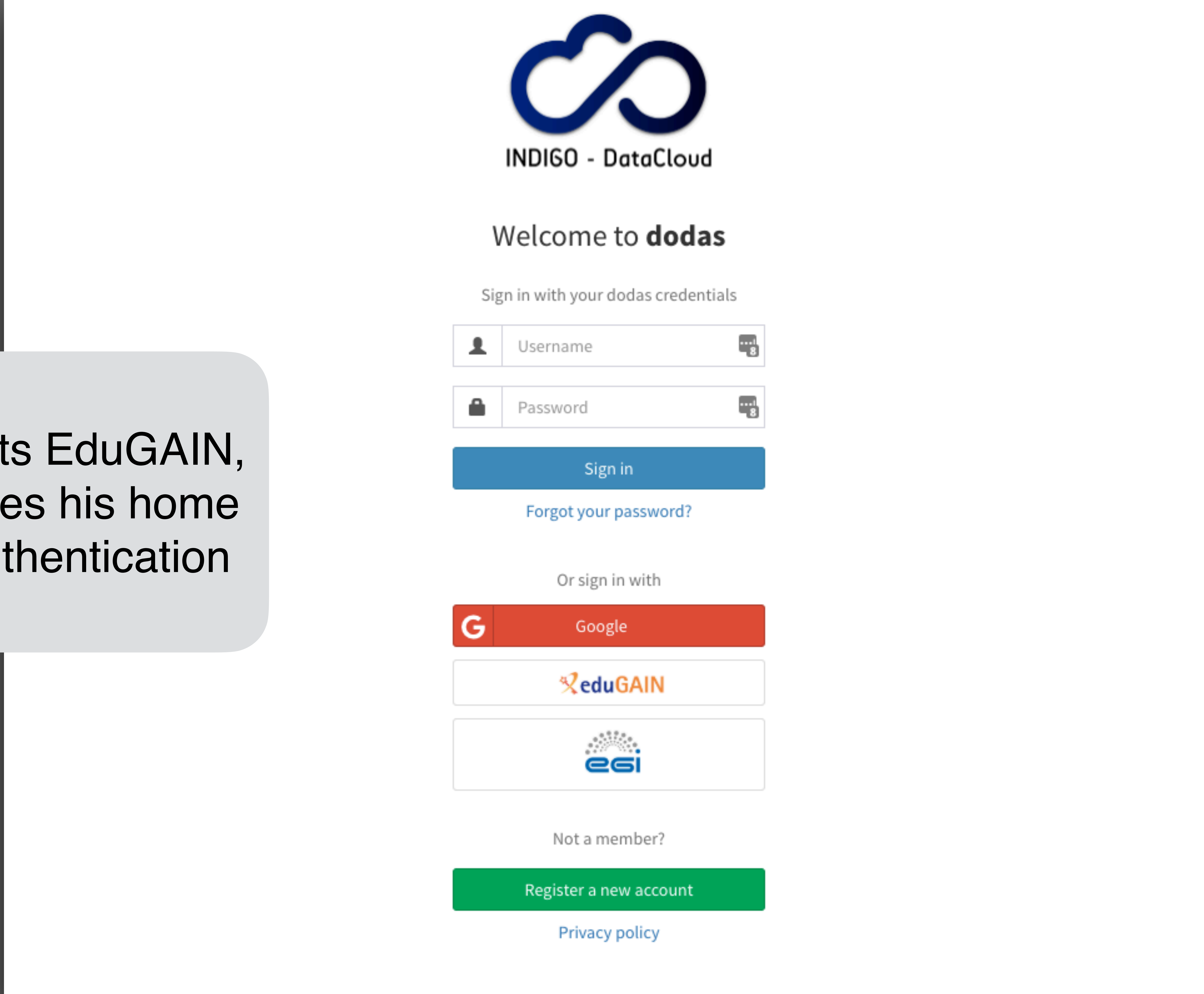
Register a new account

Privacy policy

ve a valid session at
ows the login page

Home IdP

Web application: authorization code flow



INDIGO - DataCloud

Welcome to **dodas**

Sign in with your dodas credentials

Username

Password

Sign in

Forgot your password?

Or sign in with

Google

eduGAIN

egi

Not a member?

Register a new account

Privacy policy

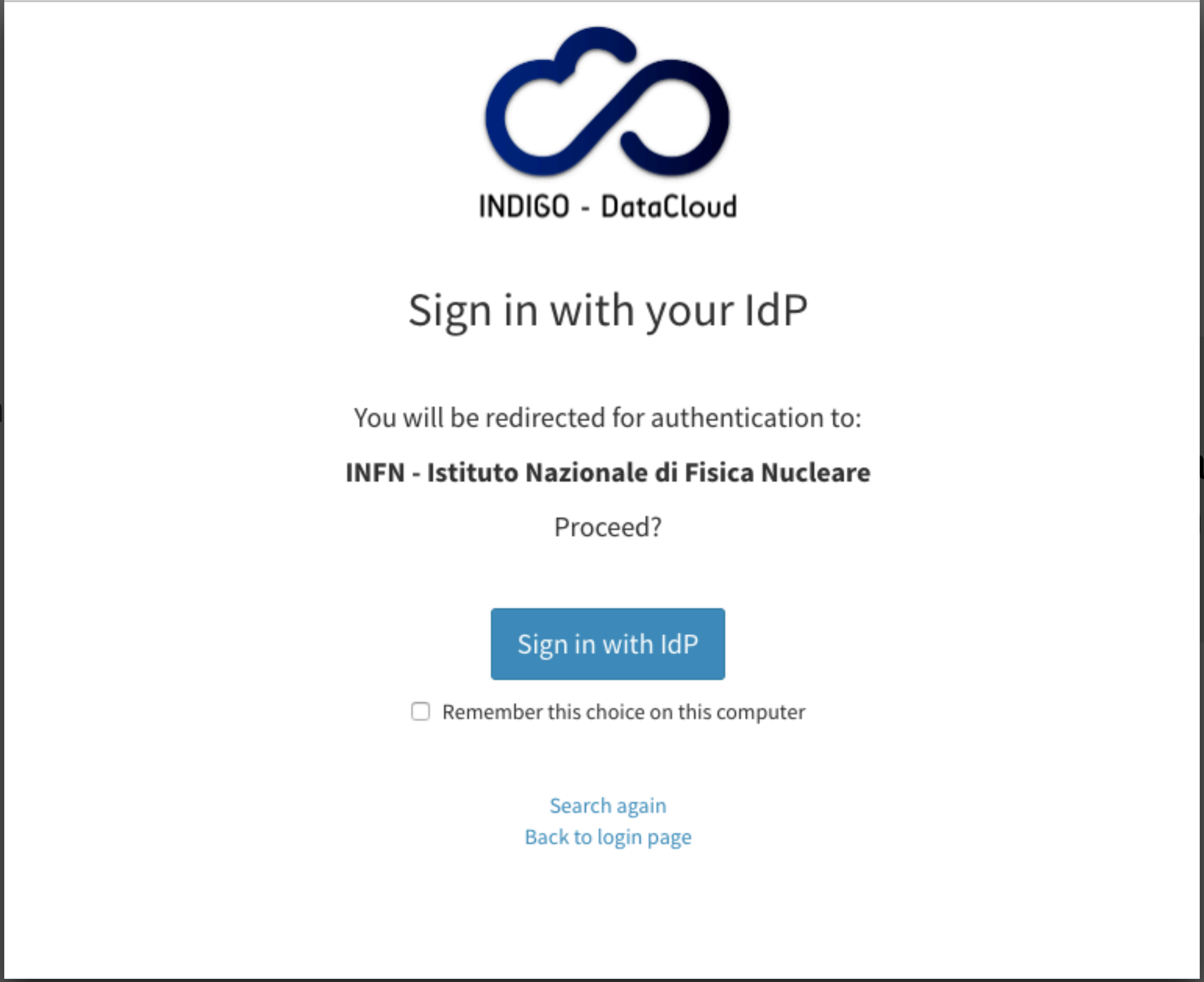
User selects EduGAIN, and chooses his home IDP for authentication

ve a valid session at
ows the login page

Home IdP

Web application: authorization code flow

authorization request



INDIGO - DataCloud

Sign in with your IdP

You will be redirected for authentication to:

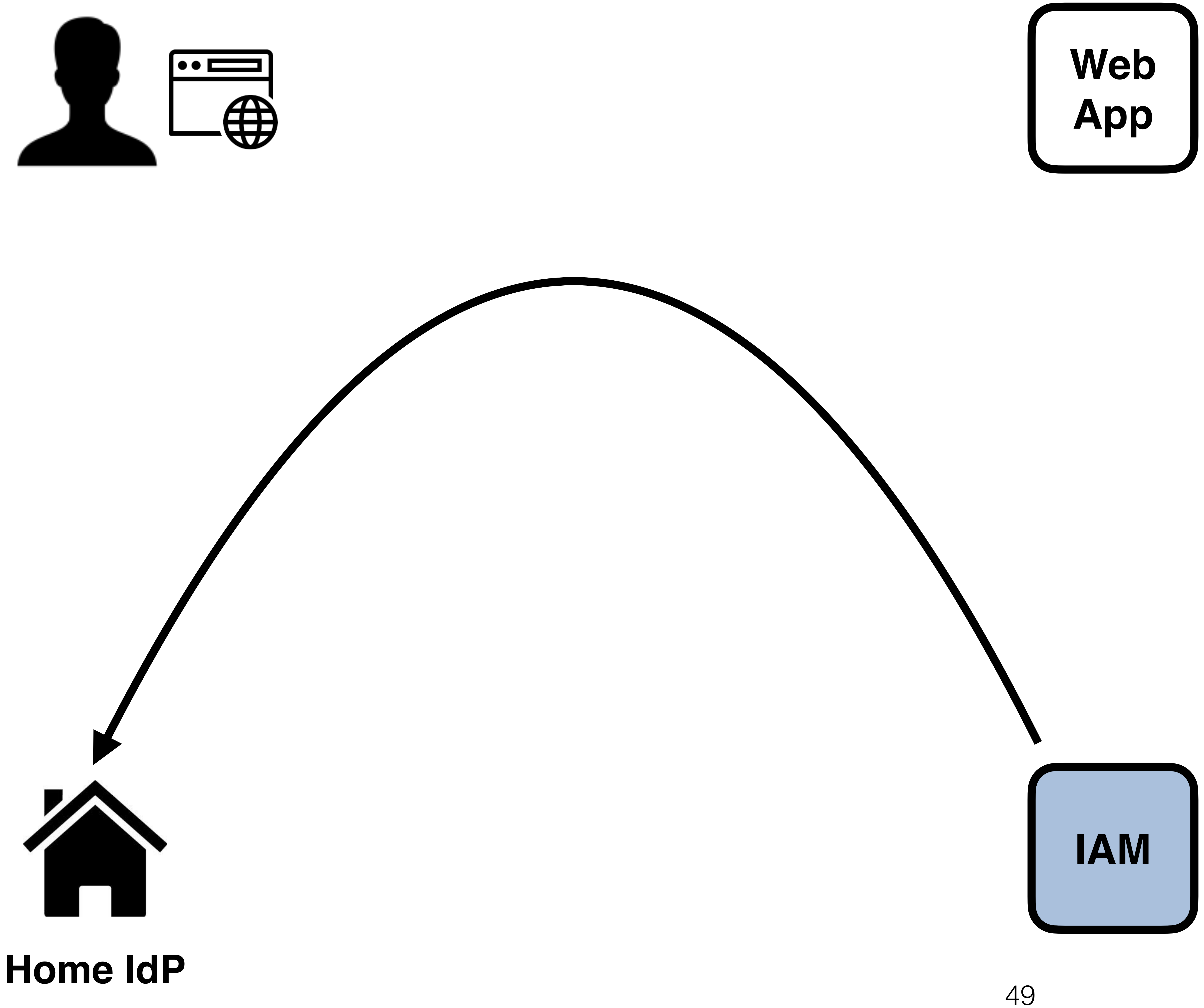
INFN - Istituto Nazionale di Fisica Nucleare

Proceed?

ave a valid session at
ows the login page

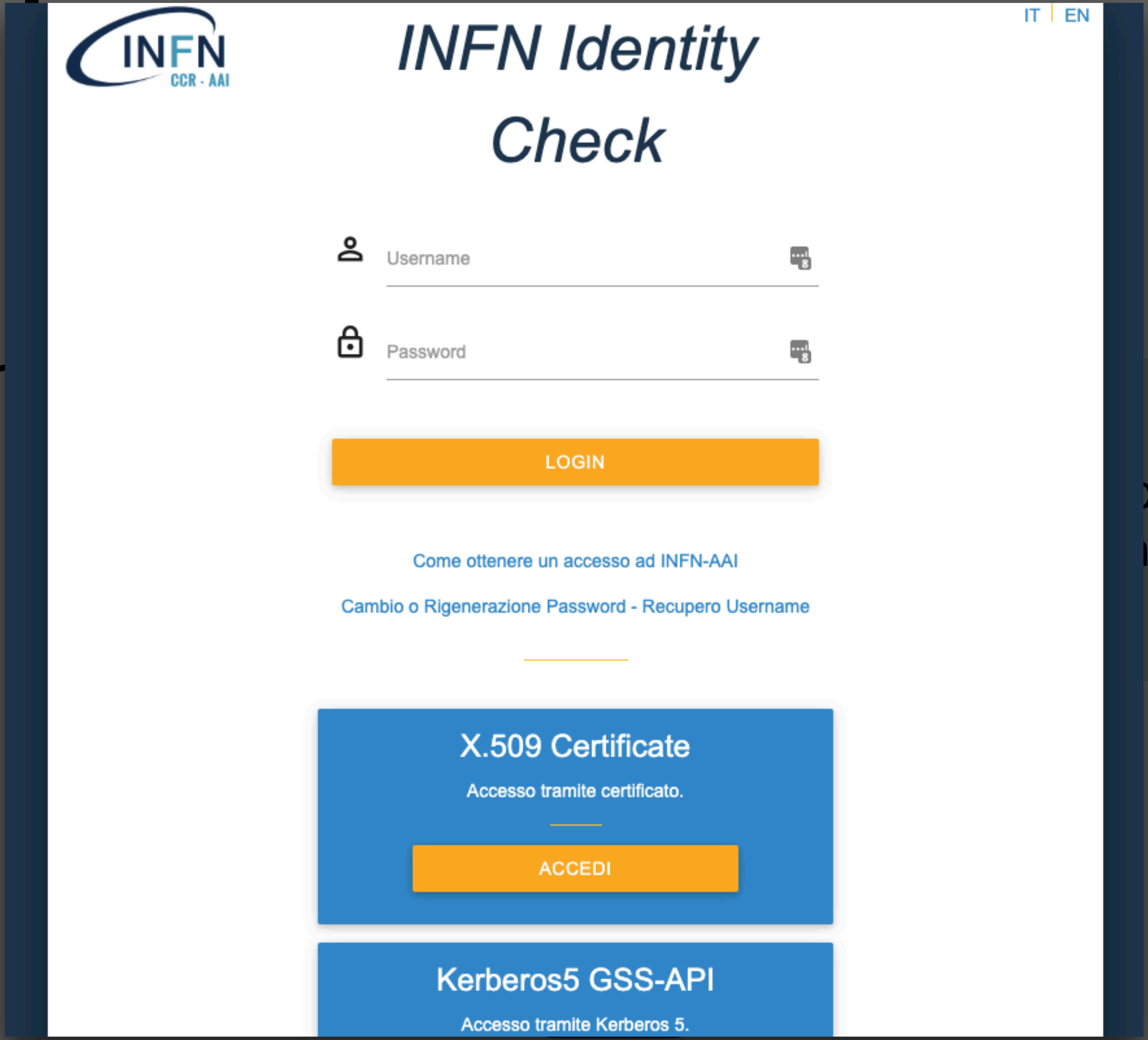
Home IdP

Web application: authorization code flow



User is redirected to home IDP for authentication

Web application: authorization code flow



IT | EN

INFN Identity Check

Username

Password

LOGIN

Come ottenere un accesso ad INFN-AAI

Cambio o Rigenerazione Password - Recupero Username

X.509 Certificate

Accesso tramite certificato.

ACCEDI

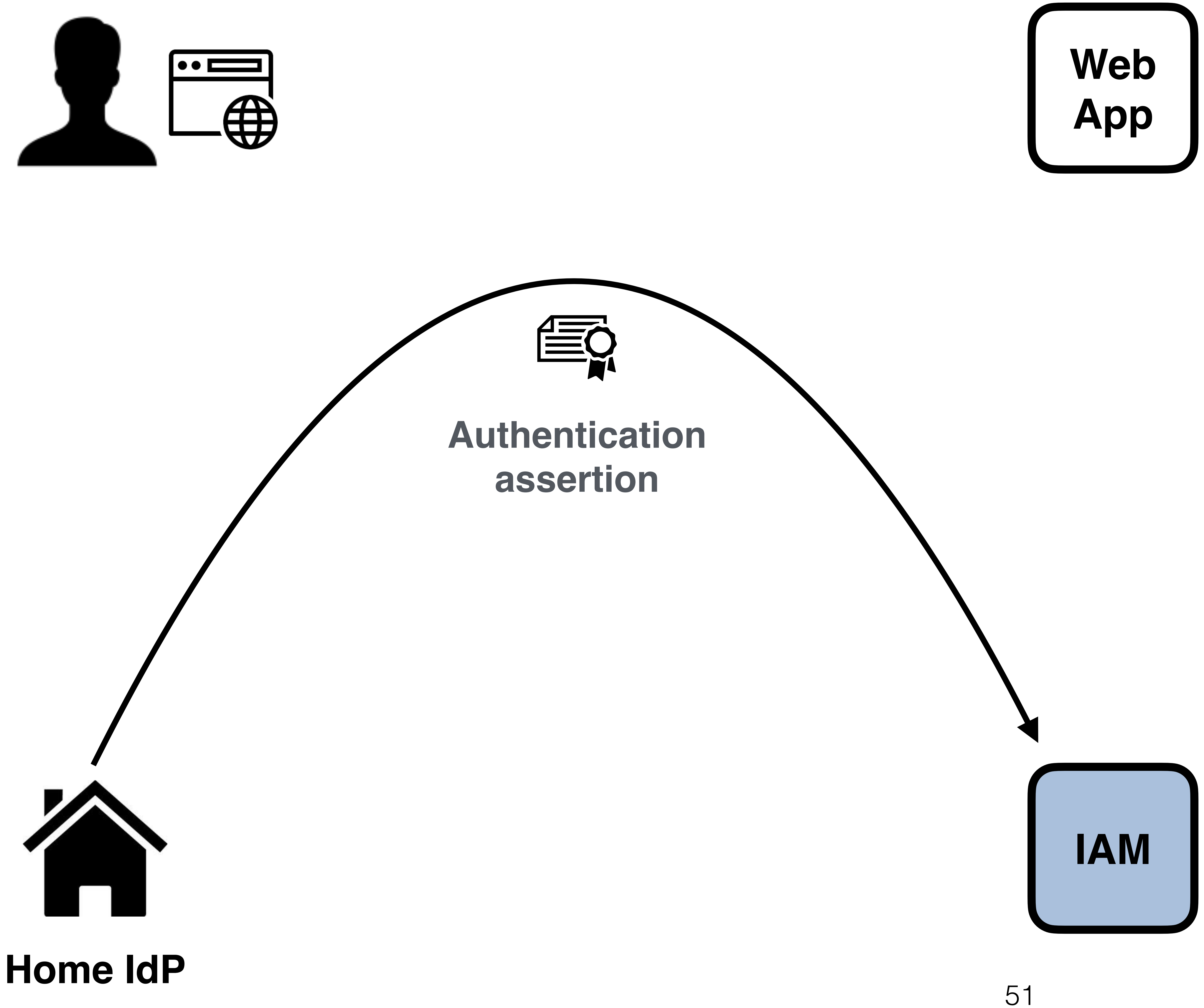
Kerberos5 GSS-API

Accesso tramite Kerberos 5.

cted to home IDP
entication

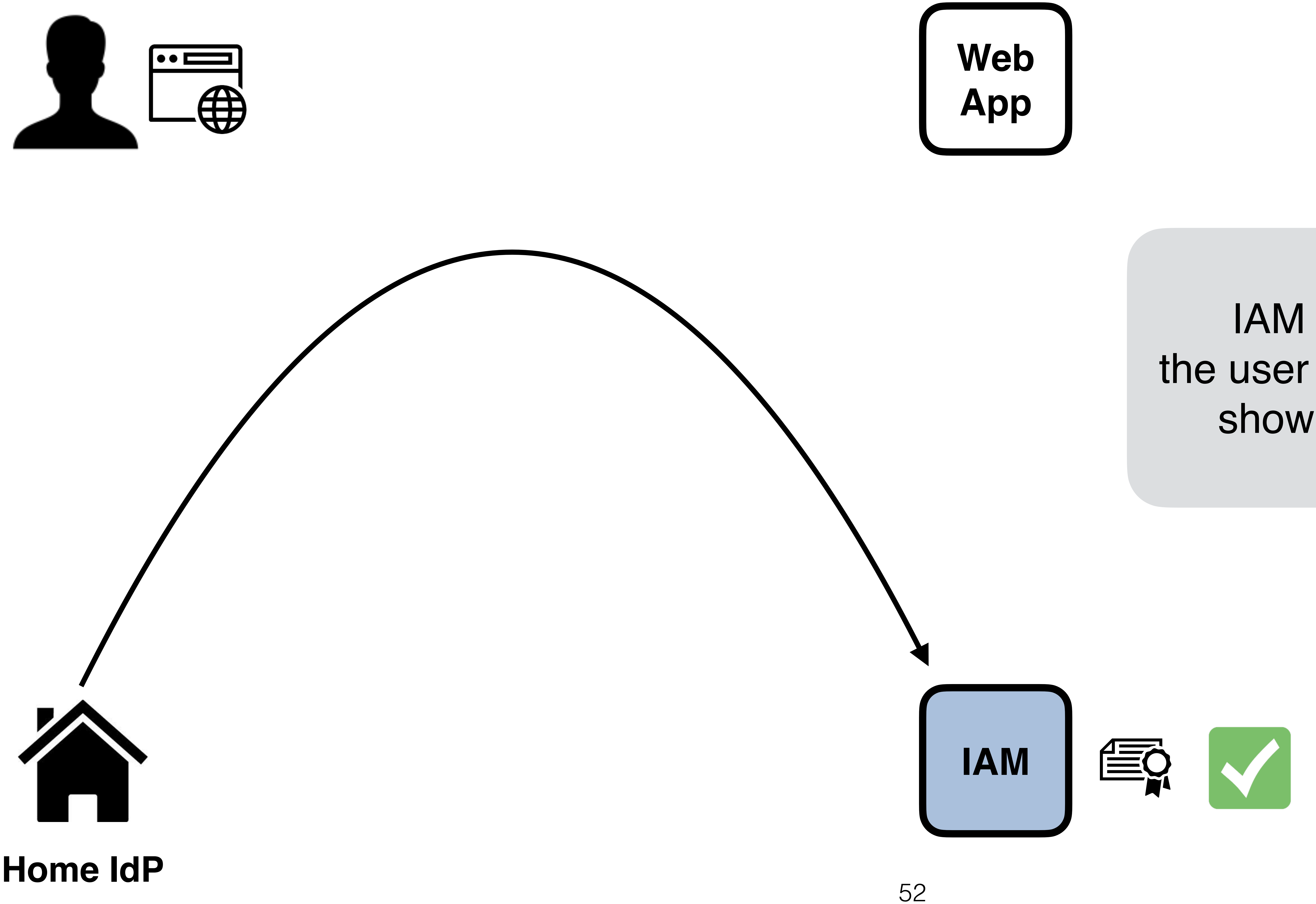
Home IdP

Web application: authorization code flow



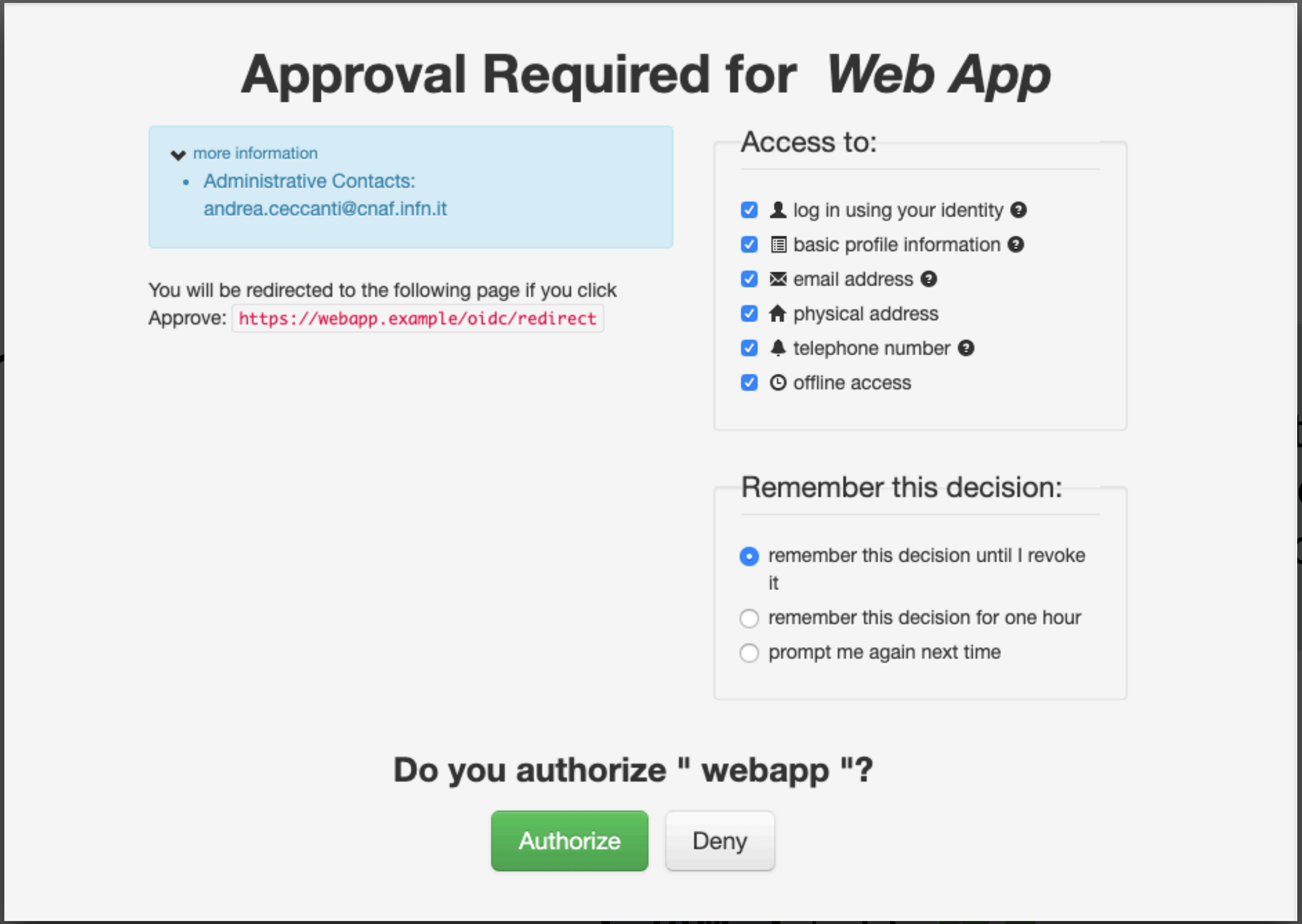
Home IDP authenticates user and sends back an authentication assertion, via redirection and possibly other interactions between IAM and the IDP

Web application: authorization code flow



IAM validates the assertion, the user is a registered one, so IAM shows a “Give consent” page

Web application: authorization code flow



Approval Required for *Web App*

more information

- Administrative Contacts: andrea.ceccanti@cnaf.infn.it

You will be redirected to the following page if you click Approve: `https://webapp.example/oidc/redirect`

Access to:

- ☑ log in using your identity
- ☑ basic profile information
- ☑ email address
- ☑ physical address
- ☑ telephone number
- ☑ offline access

Remember this decision:

- ● remember this decision until I revoke it
- ○ remember this decision for one hour
- ○ prompt me again next time

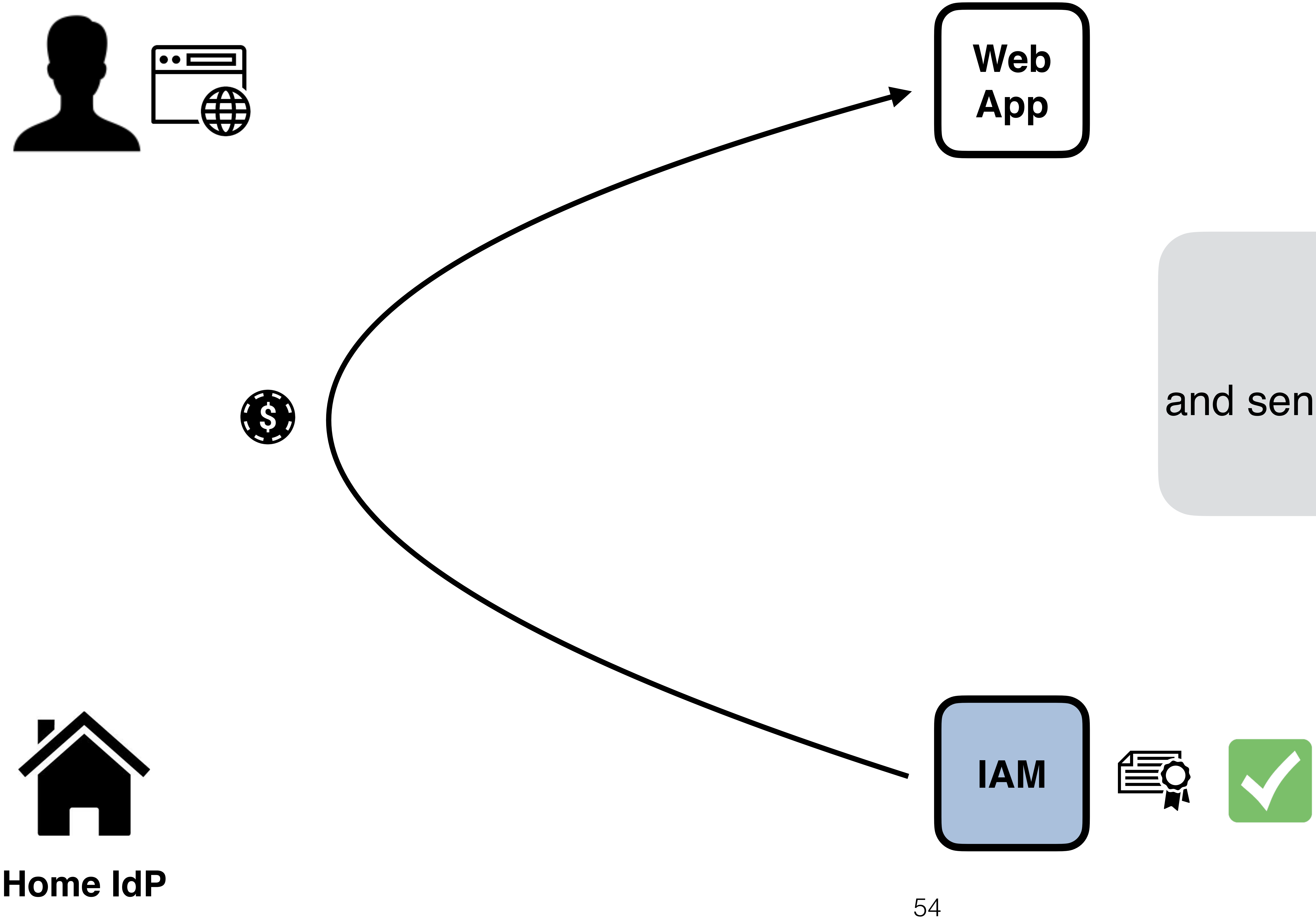
Do you authorize " webapp "?

the assertion,
ered one, so IAM
consent" page

IAM

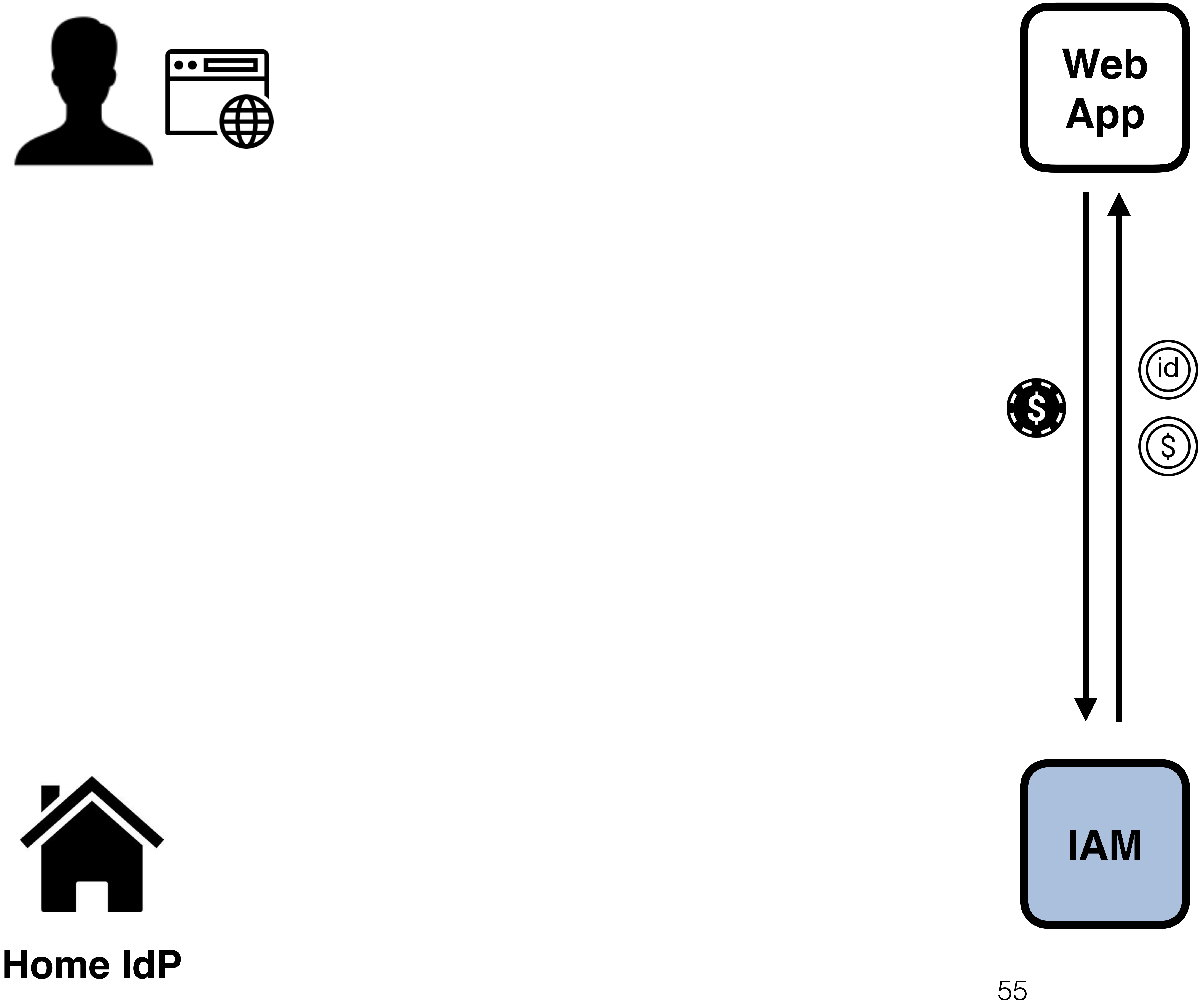
Home IdP

Web application: authorization code flow



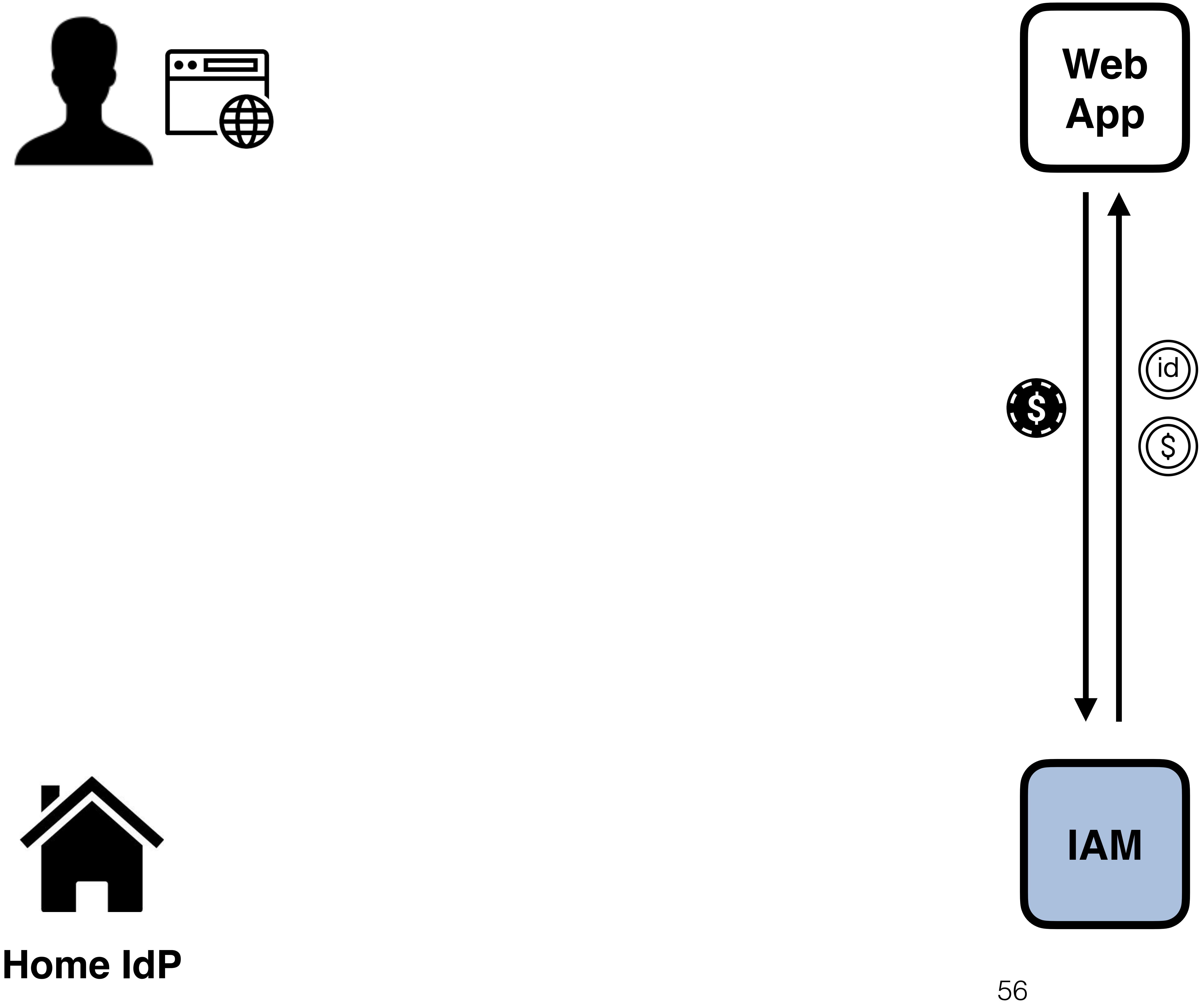
IAM generates an **authorization code** and sends it back to web app using an HTTP redirect

Web application: authorization code flow



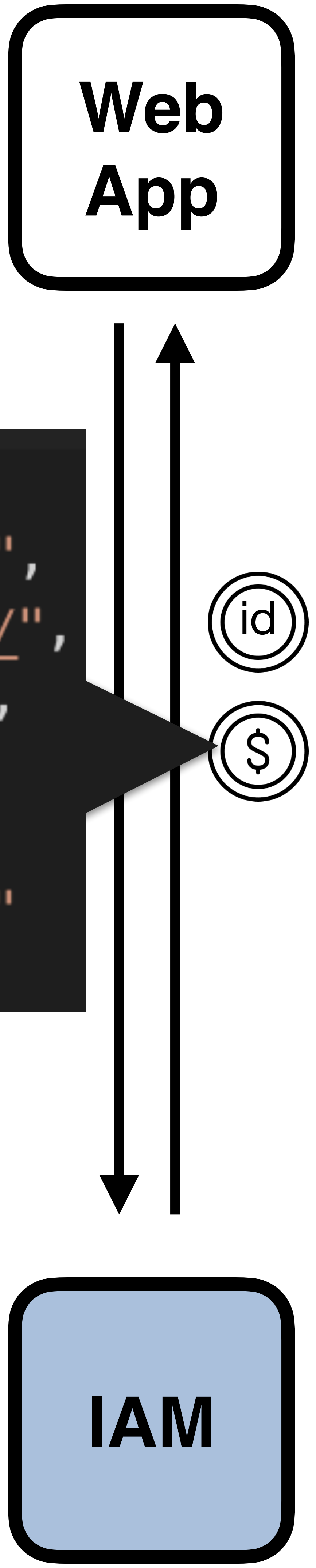
The Web App exchanges the **authorization code** with a couple of tokens: an **access token** and an **id token**

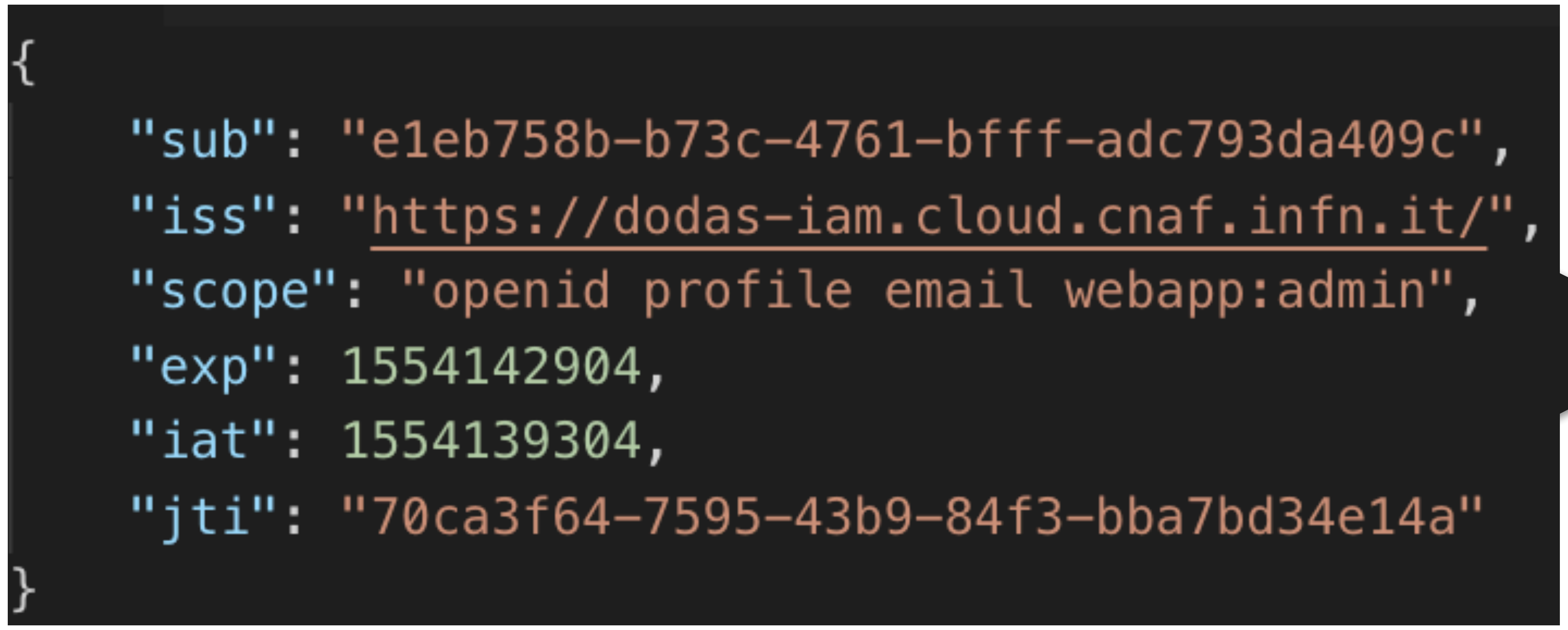
Web application: authorization code flow



In IAM,
both tokens are
JWT tokens.

Web application: authorization code flow



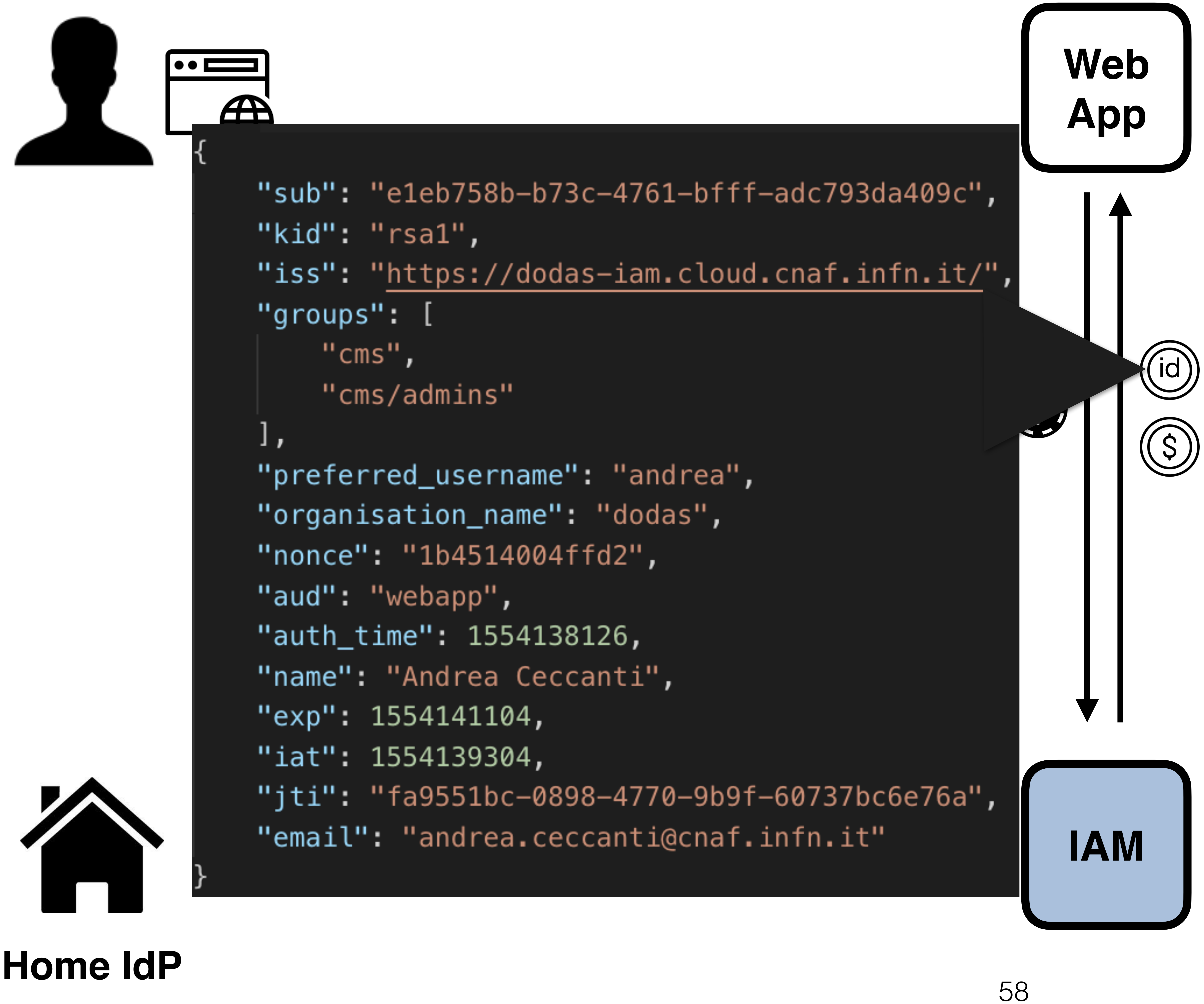


```
{
    "sub": "e1eb758b-b73c-4761-bfff-adc793da409c",
    "iss": "https://dodas-iam.cloud.cnaf.infn.it/",
    "scope": "openid profile email webapp:admin",
    "exp": 1554142904,
    "iat": 1554139304,
    "jti": "70ca3f64-7595-43b9-84f3-bba7bd34e14a"
}
```

The **access token** provides (mainly) authorization information

Home IdP

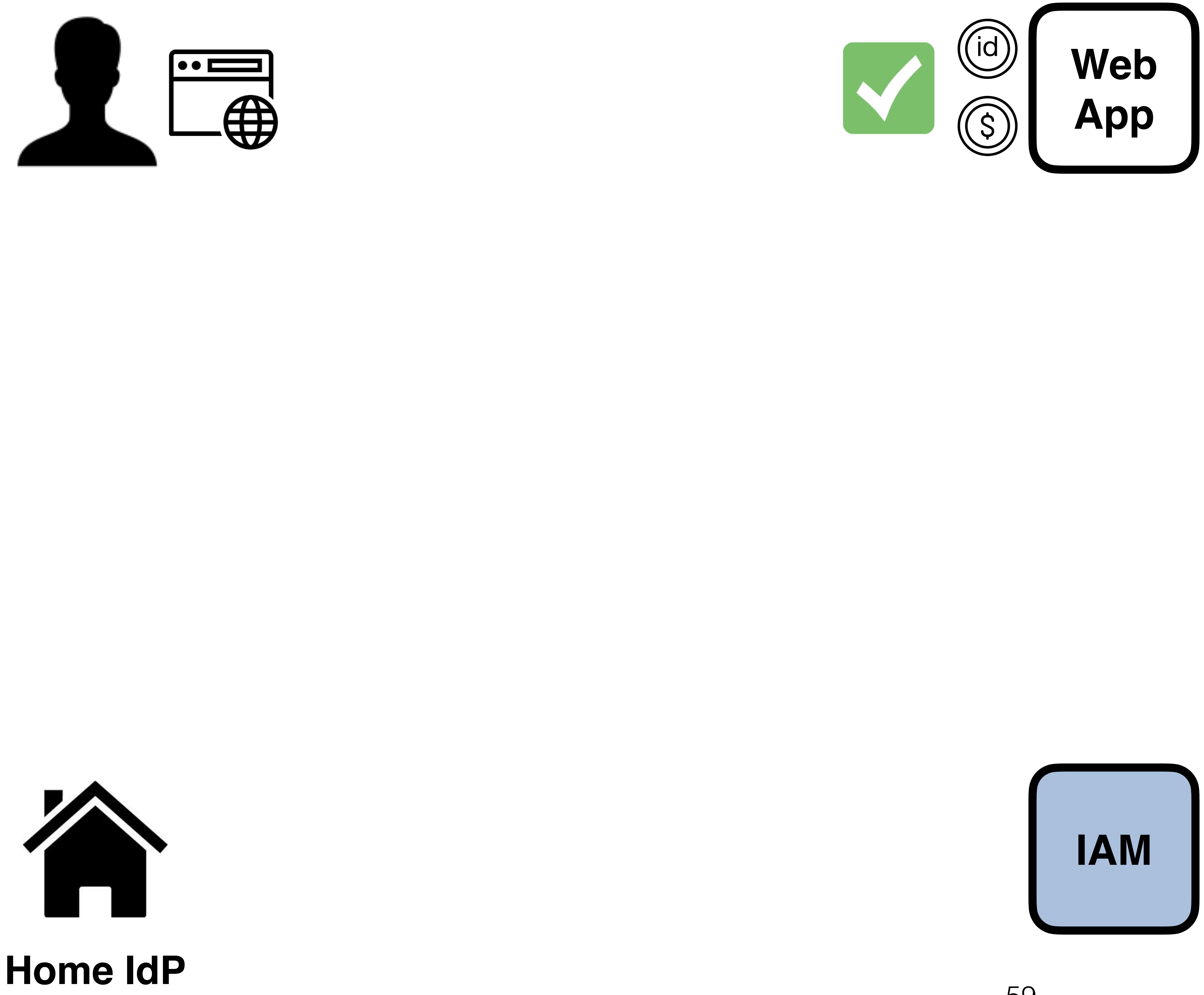
Web application: authorization code flow



```
{
    "sub": "e1eb758b-b73c-4761-bfff-adc793da409c",
    "kid": "rsa1",
    "iss": "https://dodas-iam.cloud.cnaf.infn.it/",
    "groups": [
        "cms",
        "cms/admins"
    ],
    "preferred_username": "andrea",
    "organisation_name": "dodas",
    "nonce": "1b4514004ffd2",
    "aud": "webapp",
    "auth_time": 1554138126,
    "name": "Andrea Ceccanti",
    "exp": 1554141104,
    "iat": 1554139304,
    "jti": "fa9551bc-0898-4770-9b9f-60737bc6e76a",
    "email": "andrea.ceccanti@cnaf.infn.it"
}
```

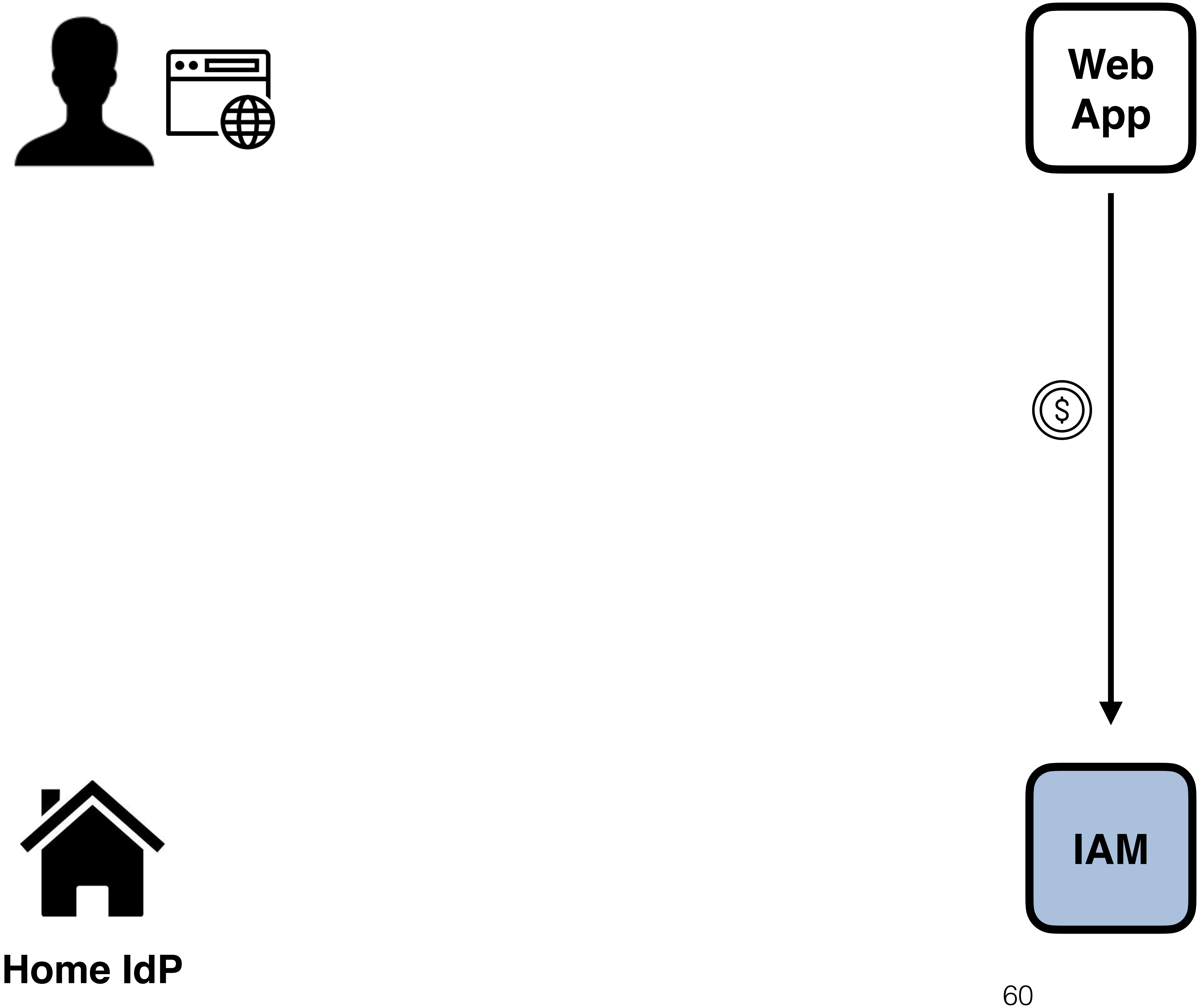
The **id token** provides (mainly) authentication information

Web application: authorization code flow



Both tokens are **validated** following to the JWT and OpenID Connect guidelines, checking **temporal validity**, **token signature**, **audience**, etc…

Web application: authorization code flow



Additional information about the user can be requested by querying the **/userinfo** endpoint and providing the just obtained **access token** for authentication/authorization purposes

Authorization code flow in practice

- In practice, decent OAuth/OpenID Connect client libraries implement all the above **behind the scenes.**

- As an example, Apache mod_auth_openidc requires the following information to enable a working OpenID Connect integration
  - The OpenID Connect provider discovery/metadata URL
  - Client credentials

- The library then takes care of exchanging messages with the OpenID provider, implementing verification checks, and provides the obtained authentication/authorization information to the protected web application
  - typically via env variables or HTTP headers

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Refresh token flow

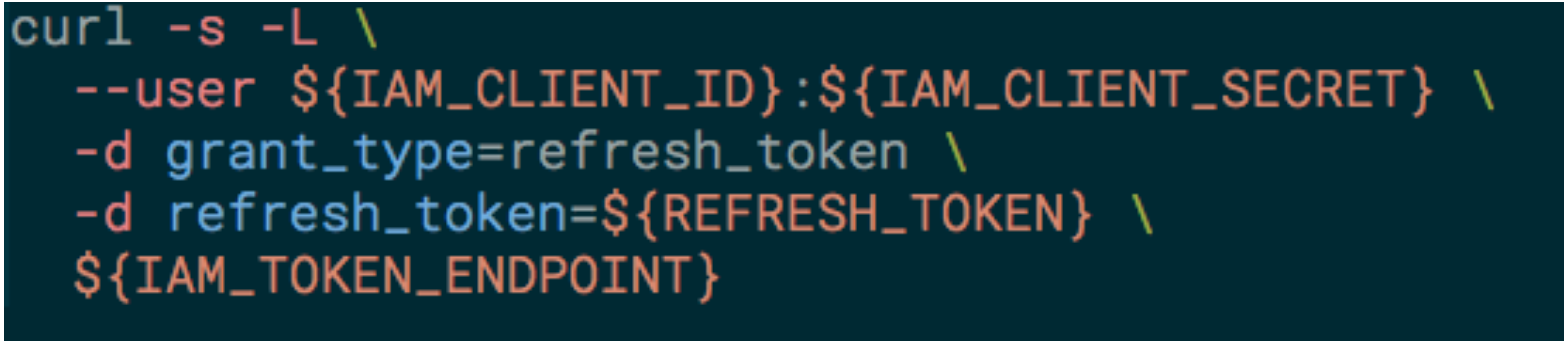
- Used by a client to refresh an access token that is about to expire using a refresh token obtained in a former authorization flow

- Authenticated call to the IAM/AS token endpoint
  - Produces a new access token and possibly an updated refresh toke

- The scope request parameter can be used to attenuate the token privileges, by requesting a subset of the scopes linked to user authorization grant

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Refresh token flow request: example



```
curl -s -L \
  --user ${IAM_CLIENT_ID}:${IAM_CLIENT_SECRET} \
  -d grant_type=refresh_token \
  -d refresh_token=${REFRESH_TOKEN} \
  ${IAM_TOKEN_ENDPOINT}
```

Client credentials flow

- Used when client applications need an access token to act on behalf of themselves

- Authenticated call to the IAM/AS token endpoint
    - authentication is by client credentials

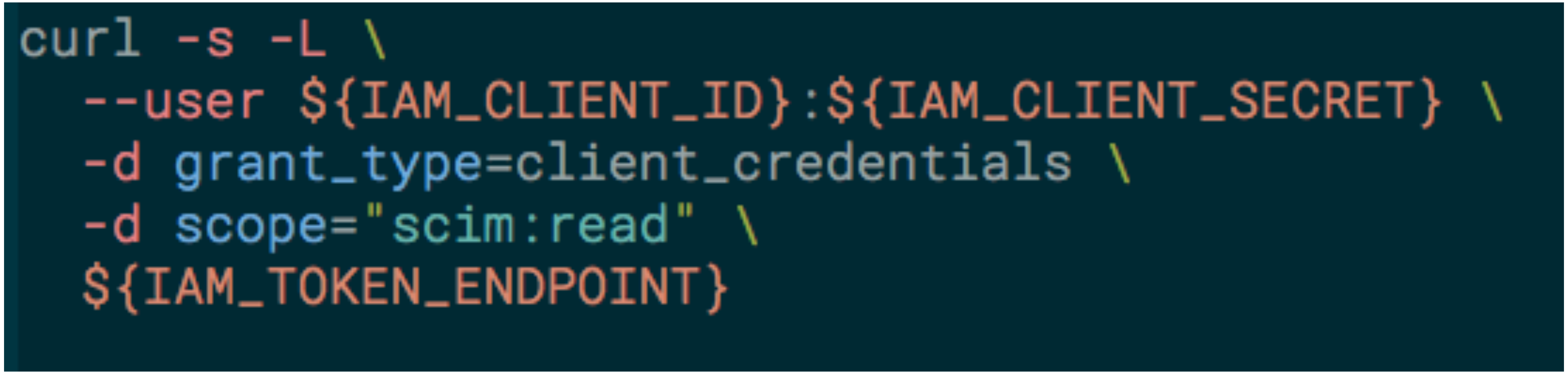
- Produces an access token

- No user identity linked to this OAuth flow, the client is acting on behalf of itself
    - aka as Service accounts

- Refresh tokens also do not apply: the client can get a new token anytime

Funded by the European Union's
Horizon 2020 - Grant N° 824064

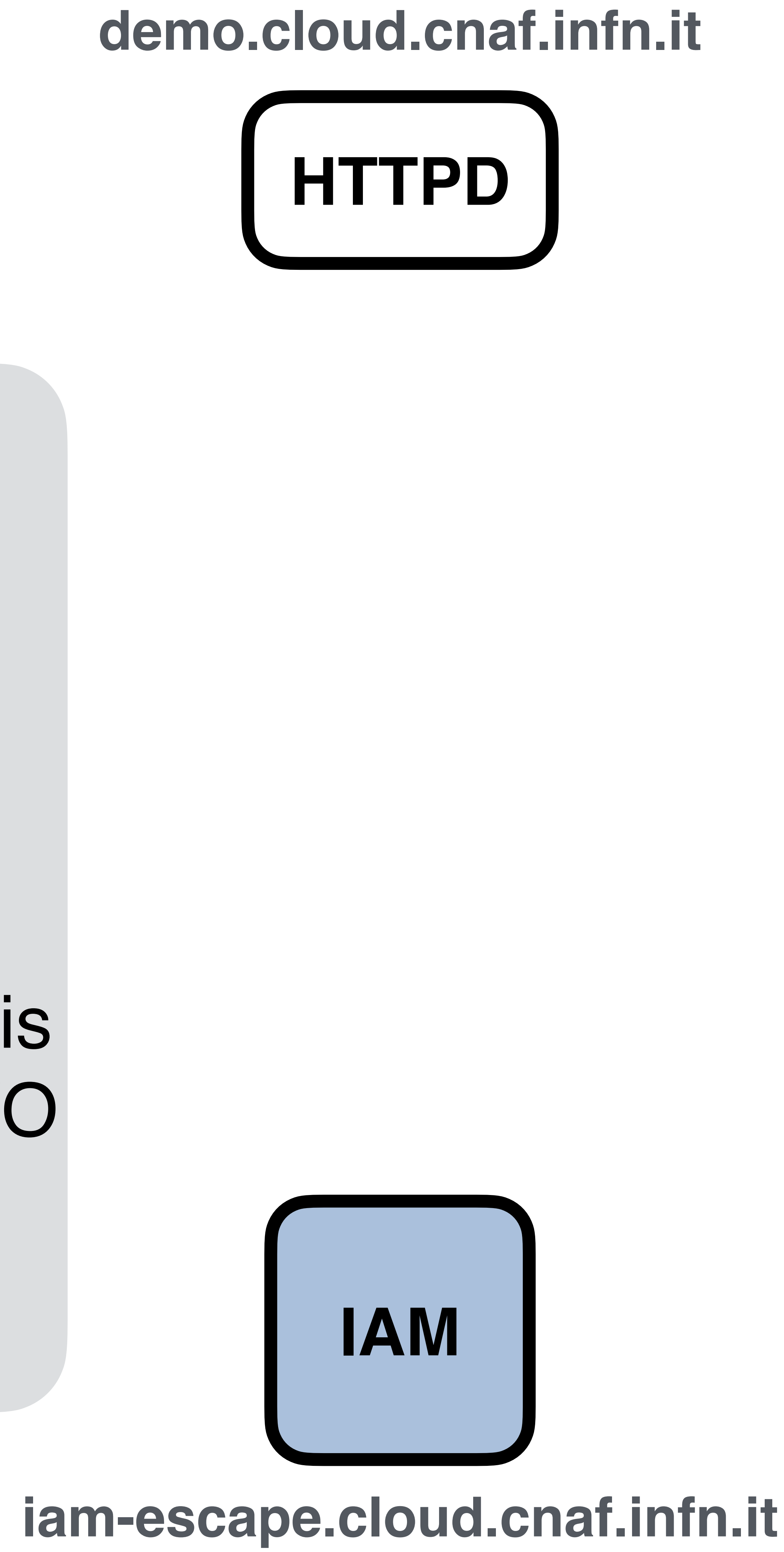
Client credentials flow: request



```
curl -s -L \
  --user ${IAM_CLIENT_ID}:${IAM_CLIENT_SECRET} \
  -d grant_type=client_credentials \
  -d scope="scim:read" \
  ${IAM_TOKEN_ENDPOINT}
```

Integration demo

Integration Demo setup



HTTPD
is an Apache server configured with
mod_auth_openidc

We want to showcase group-based authorization, so that access to resources is authorized taking into account ESCAPE VO membership

Integration Demo setup

demo.cloud.cnaf.infn.it

HTTPD

HTTPD
is an Apache server configured with
mod_auth_openidc

We want to showcase group-based authorization, so that access to resources is authorized taking into account ESCAPE VO membership

Access policies

/escape is accessible from all members of the **ESCAPE** organization

/lofar is accessible from members of the **/escape/lofar** group in the ESCAPE organization

IAM

iam-escape.cloud.cnaf.infn.it

Apache mod_auth_openidc configuration

```
ServerName demo.cloud.cnaf.infn.it

<VirtualHost _default_:80>

  OIDCProviderMetadataURL https://iam-escape.cloud.cnaf.infn.it/.well-known/openid-configuration
  OIDCClientID demo_client
  OIDCClientSecret *****
  OIDCScope "openid email profile"
  OIDCRedirectURI https://demo.cloud.cnaf.infn.it/oidc/redirect_uri
  OIDCCryptoPassphrase *****

  <Location /escape>
    …
    AuthType openid-connect
    Require valid-user
    LogLevel debug
  </Location>

…
```

Apache mod_auth_openidc configuration

```
   <Location /lofar>
    …
    AuthType openid-connect
    Require claim groups:escape/lofar
  </Location>

</VirtualHost>
```

Thanks for your attention!

Questions?

Useful references

- IAM ESCAPE docs: https://indigo-iam.github.io/escape-docs
- IAM on GitHub: https://github.com/indigo-iam/iam
- IAM documentation: https://indigo-iam.github.io/docs
- IAM in action video: https://www.youtube.com/watch?v=1rZlvJADOnY
- Apache integration demo: https://github.com/andreaceccanti/iam-tutorial/tree/master/apache-integration-demo
- Contacts:
  - andrea.ceccanti@cnaf.infn.it
  - enrico.vianello@cnaf.infn.it
  - indigo-aai.slack.com

Funded by the European Union's
Horizon 2020 - Grant N° 824064