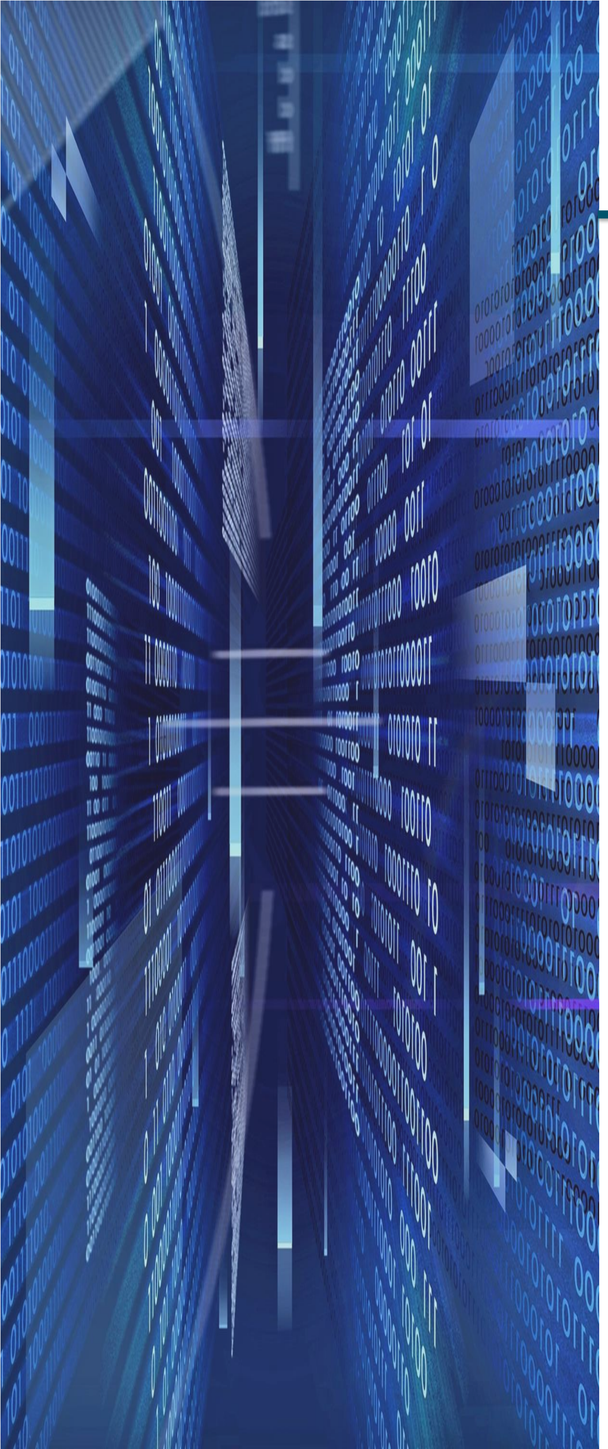
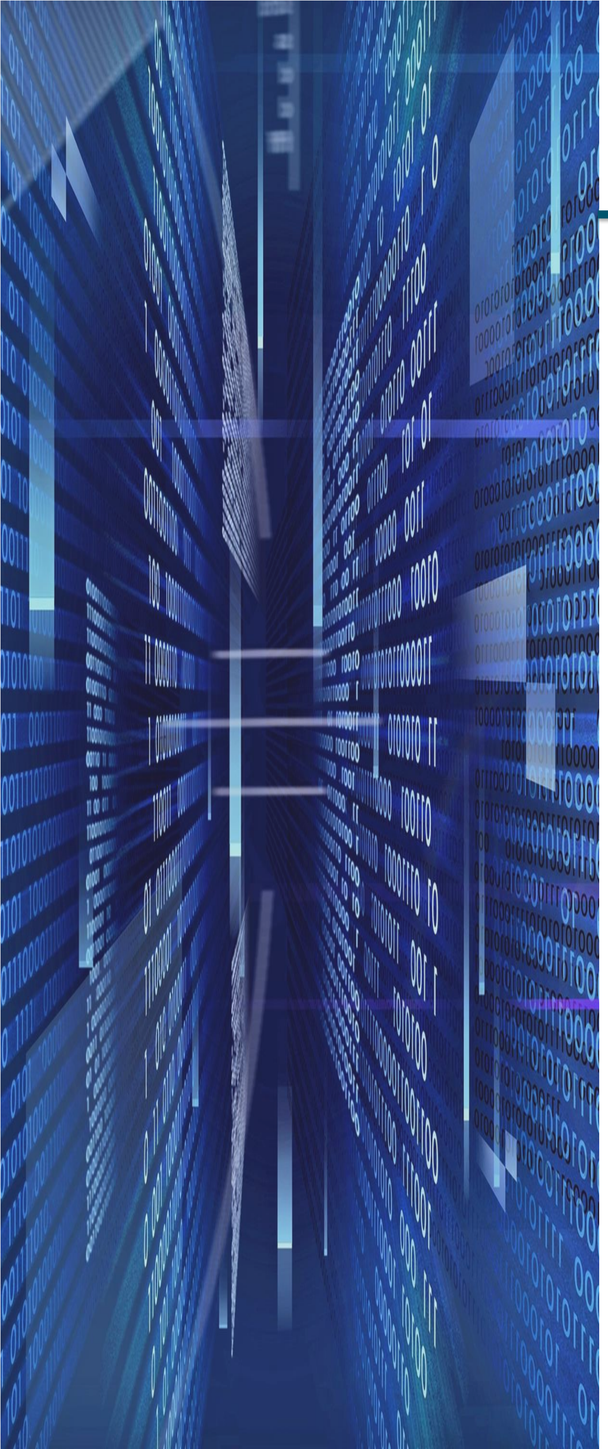


Digital Identity Management

Prof Federica Paci

Lecture Outline

- Passwordless authentication
  - FIDO2 Alliance Protocols
- Digital identity management
  - Digital identity
  - Single sign on
  - Federated Identities
  - Standards and Protocols
    - SAML
    - Shibboleth
    - OpenID Connect



FIDO2 Protocols

FIDO2 Protocols

- Standards based on **public-key cryptography** to easily authenticate users to online services in mobile and desktop environments
- Thet support passwordless, second-factor and multi-factor authenticationwith **embedded (or bound) authenticators** (such as biometrics or PINs) or external (or roaming) **authenticators** (such as FIDO Security Keys, mobile devices, wearables, etc.).
- The protocols guarantee:
  - Resistance against phishing and replay attacks
  - User privacy

• Obbiettivo è ridurre la dipendenza dalle password tradizionali e migliorare la sicurezza delle autenticazioni online

Main Actors

- The **user** who has to authenticate to the online service
- The **relying party** is the organization responsible for registering and authenticating the user
- The **client platform** includes the client and the client device
- The **authenticator** provides key management and cryptographic signatures
  - It is assigned by the maker a unique identifier called AAGUID (Authenticator Attestation Global Unique Identifier)
  - It is also assigned a pair of private and public keys that can be used for attestation

Main specifications

- WebAuth
  - A standard web API that is being built into browsers and platforms to enable support for FIDO Authentication.

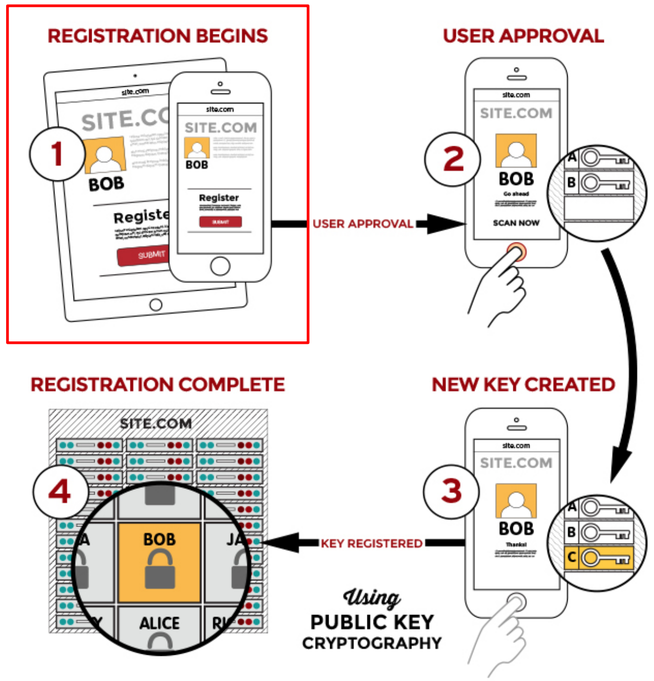
    ↳ Specifica che fa parte di FIDO2. Consente agli utenti di autenticarsi su siti web senza la necessità di password. Invece di utilizzare password, WebAuthn utilizza chiavi crittografiche asimmetriche. Quando un utente si autentica su un sito web compatibile con WebAuthn, il browser genera una coppia di chiavi pubbliche e private. La chiave pubblica viene memorizzata sul server, mentre la chiave privata rimane sul dispositivo dell'utente
- CTAP2
  - A protocol to interact with external authenticators (FIDO Security Keys, mobile devices) for authentication on FIDO2-enabled browsers and operating systems over USB, NFC, or BLE for a passwordless, second-factor or multi-factor authentication experience

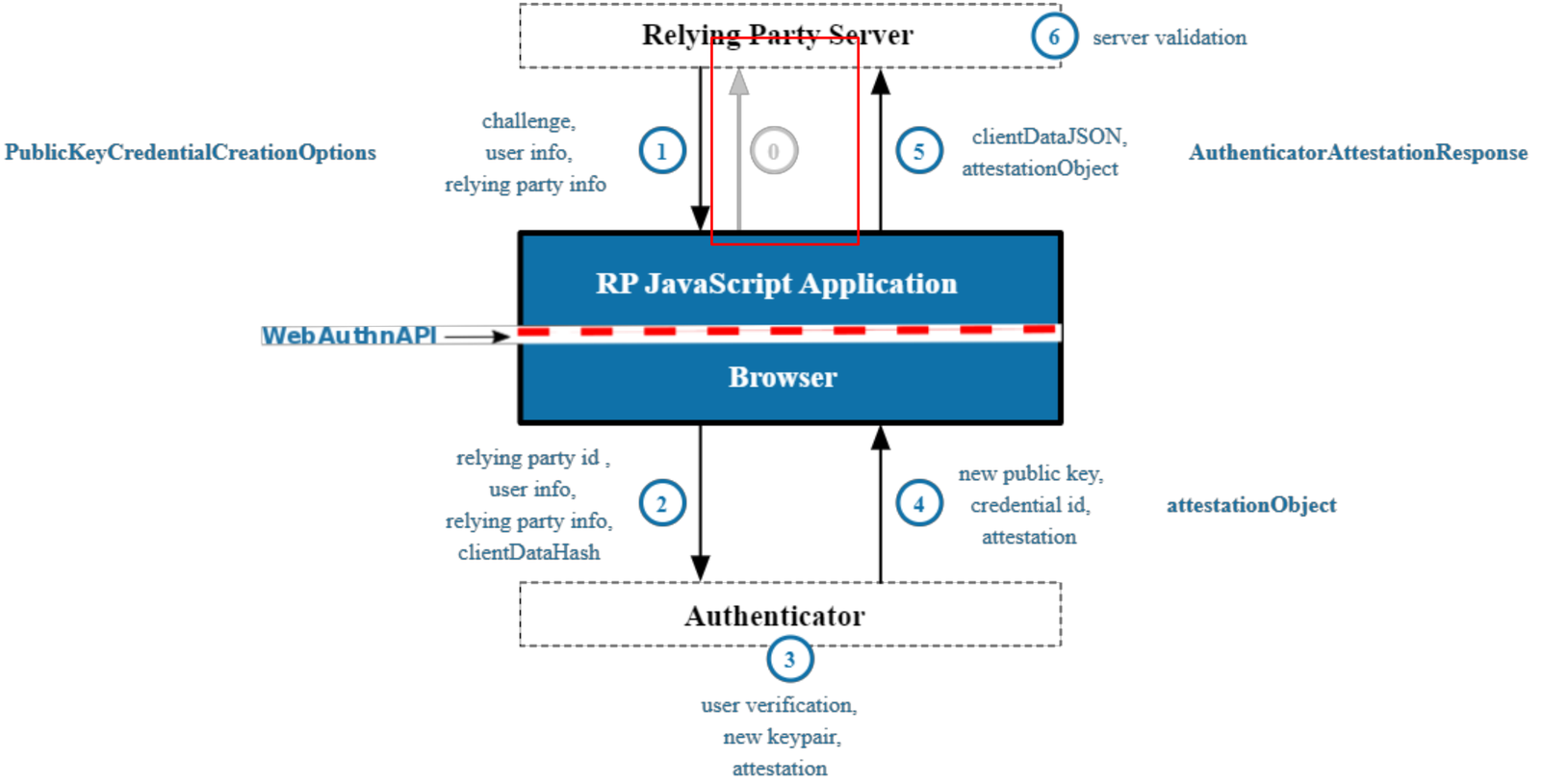
    ↳ CTAP è un protocollo che consente la comunicazione tra il client (solitamente un browser) e l'autenticatore (ad es una chiavetta di sicurezza). Questo protocollo facilita il processo di autenticazione forte e sicura.

Registration ceremony

In un sistema di autenticazione basato su password un server presenterà un modulo a un utente richiedendo utente e password

In WebAuthn un server deve fornire dati che collegano un utente a una credenziale (una coppia di chiavi pubblica e privata) I dati includono identificatori per utente e organizzazione





PublicKeyCredentialCreationOptions Object

```
const publicKeyCredentialCreationOptions = {
    challenge: Uint8Array.from(
        randomStringFromServer, c => c.charCodeAt(0)),
    rp: {
        name: "Duo Security",
        id: "duosecurity.com",
    },
    user: {
        id: Uint8Array.from(
            "UZSL85T9AFC", c => c.charCodeAt(0)),
        name: "lee@webauthn.guide",
        displayName: "Lee",
    },
    pubKeyCredParams: [{alg: -7, type: "public-key"}],
    authenticatorSelection: {
        authenticatorAttachment: "cross-platform",
    },
    timeout: 60000,
    attestation: "direct"
};
```

buffer di byte crittograficamente casuali generato sul server e serve a prevenire attacchi di replay

sta per relying party, può essere considerato come la descrizione dell'organizzazione responsabile della registrazione e dell'autenticazione dell'utente. L'ID deve essere un sottoinsieme del dominio attualmente nel browser

è un'informazione sull'utente attualmente in fase di registrazione L'authenticator utilizza l'ID per associare una credenziale all'utente. Si suggerisce di non utilizzare informazioni personali identificative come ID, poiché potrebbero essere memorizzate in un autenticatore

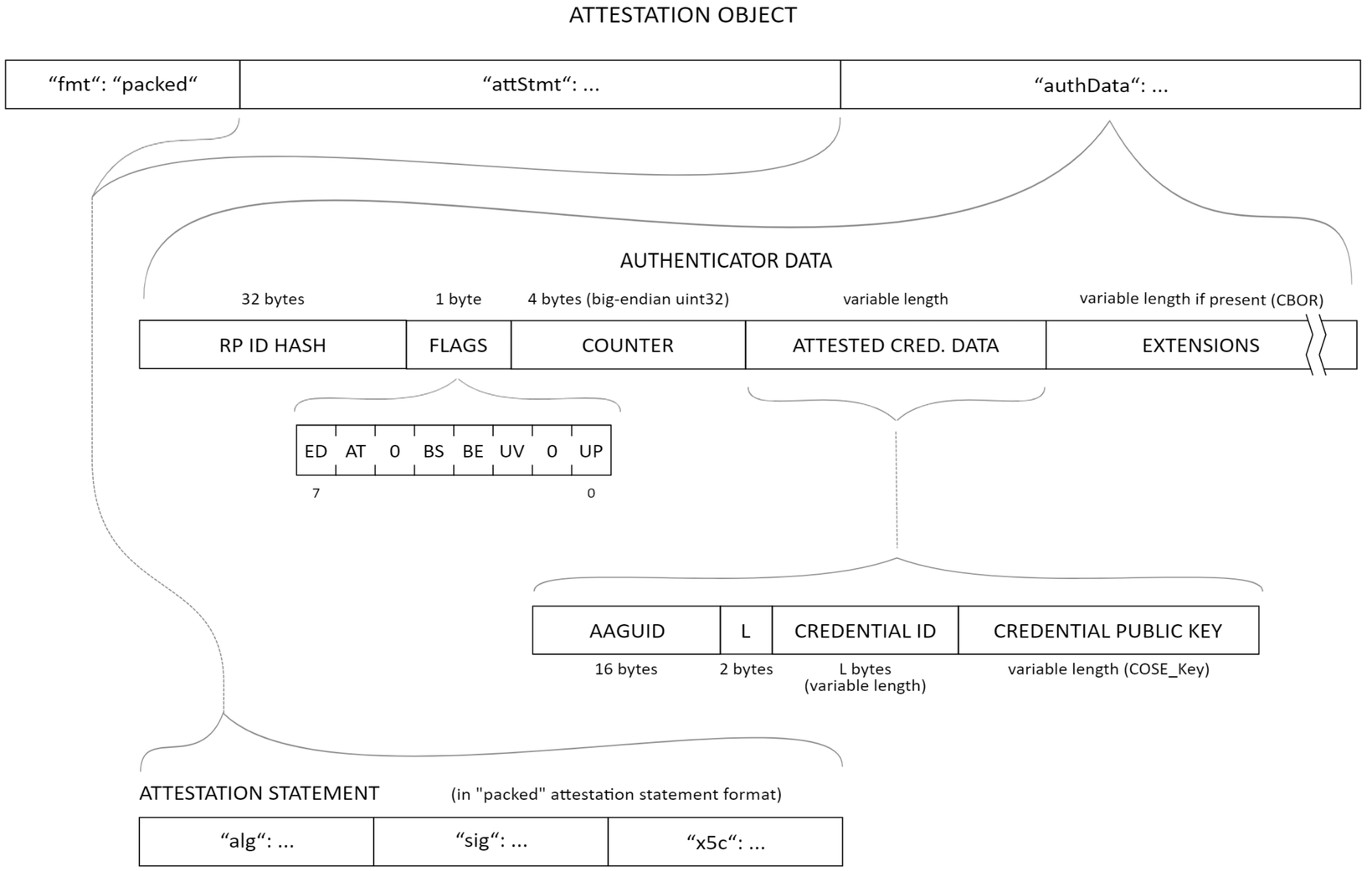
Questo è un array di oggetti che descrivono quali tipi di chiavi pubbliche, pubbliche sono accettabili per un server. L'alg è un numero che indica tale informazione

tempo in millisecondi che l'utente ha per rispondere a una richiesta di registrazione prima che venga restituito un errore

Questo oggetto opzionale aiuta le relying parties a fare ulteriori restrizioni sul tipo di autenticatori consentiti per la registrazione

I dati di attestazione restituiti dall'authenticator contengono informazioni che potrebbero essere utilizzate per tracciare gli utenti. Questa opzione consente ai server di indicare quanto sia importante per questo evento di registrazione tali dati di attestazione.
None indica che al server non interessa l'attestazione
Indirect significa che il server accetterà dati di attestazione anonimizzati
Direct significa che il server desidera ricevere i dati di attestazione direttamente dall'authenticator

Attestation Object

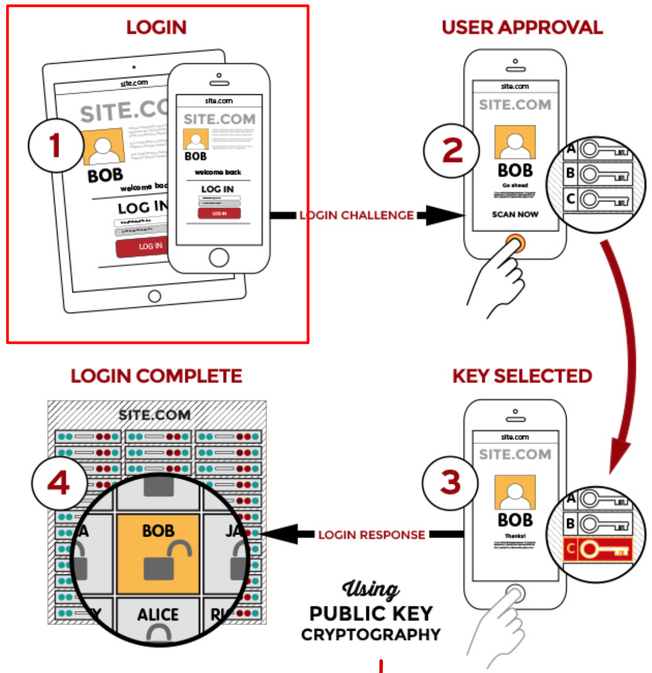


ClientDataJSON

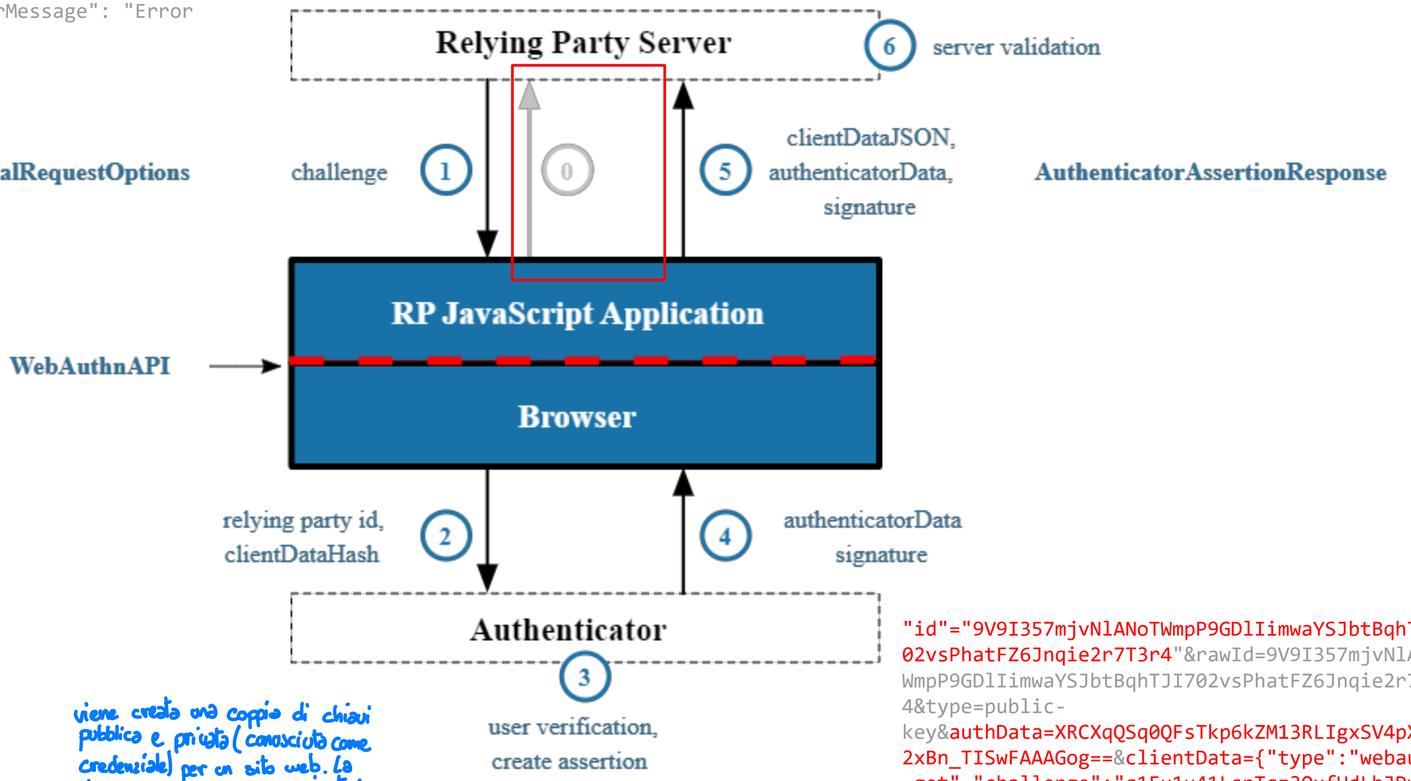
```
{
    challenge: "p5aV2uHXr0AOqUk7HQitvi-Ny1....",
    origin: "https://webauthn.guide",
    type: "webauthn.create"
}
```

The Authentication ceremony


```
{"challenge": "q1Fu1u41LspTqz3QxfUdLbJPydbcuclI",
"timeout": 60000, "rpId": "www.pastery.net",
"allowCredentials": [], "userVerification":
"required", "extensions": {"txAuthSimple":
"FIDO", "txAuthGenericArg": {"contentType":
"text/plain", "content": "RklETw=="}, "uvi":
true}, "status": "ok", "errorMessage": "Error
while logging in."}
```

↳ Questa API consente ai server di autenticare gli utenti utilizzando la crittografia a chiave pubblica invece di una password

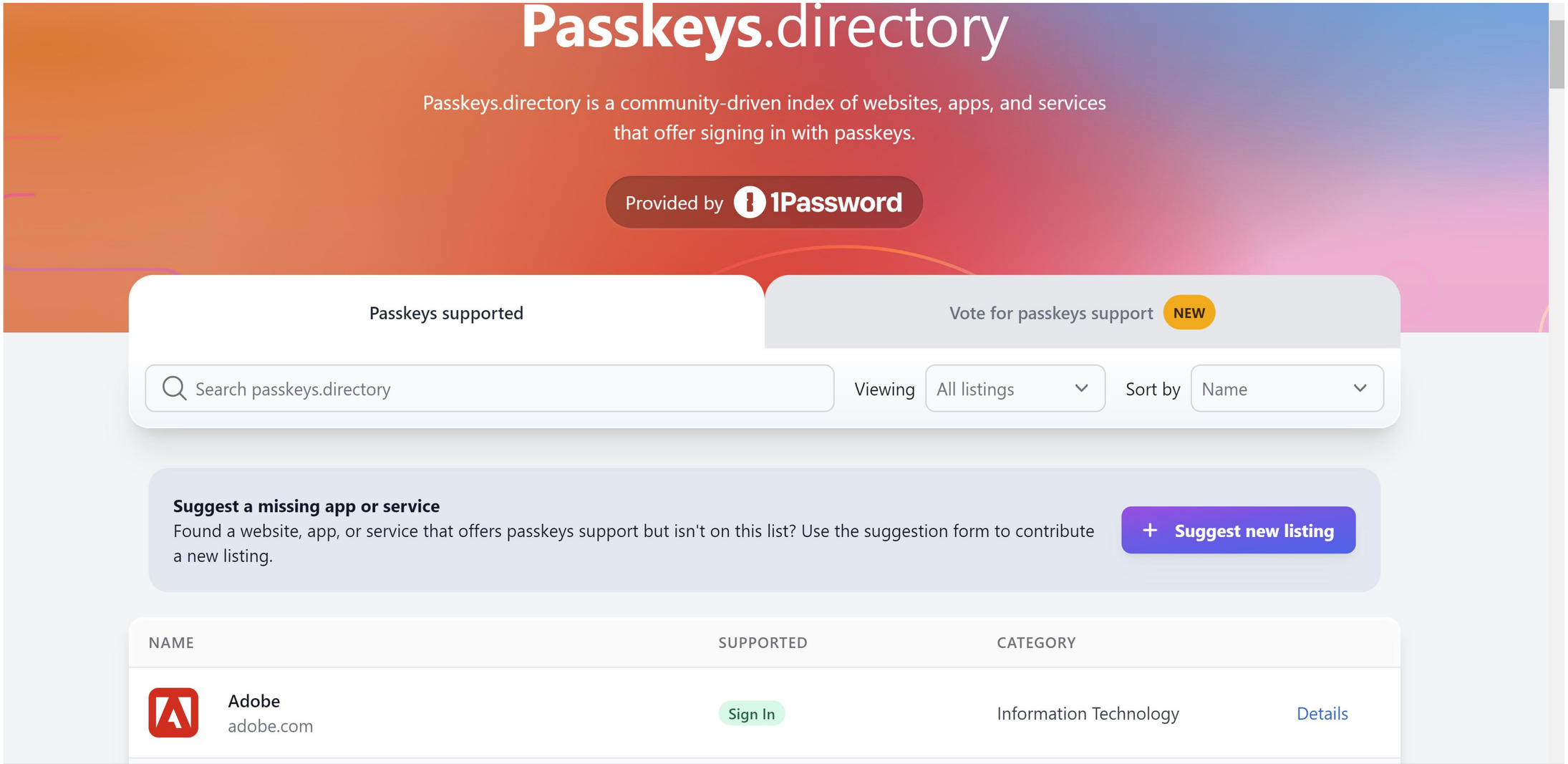


viene creata una coppia di chiavi pubblica e privata (conosciuta come credenziale) per un sito web. La chiave privata viene memorizzata in modo sicuro sul dispositivo dell'utente, una chiave pubblica e un ID di credenziale generato casualmente vengono inviati al server per l'archiviazione. Il server può usare quella chiave pubblica per dimostrare l'identità dell'utente

La chiave pubblica non è segreta perché è inutile senza quella privata
Il fatto che il server non riceva nessun segreto ha implicazioni significative per la sicurezza, i server non sono più attraenti per gli hacker, poiché le chiavi pubbliche non sono utili per loro


```
"id"="9V9I357mjvNlANoTWmpP9GDlIimwaYSJbtBqhTJI7
02vsPhatFZ6Jnqie2r7T3r4"&rawId=9V9I357mjvNlANoT
WmpP9GDlIimwaYSJbtBqhTJI702vsPhatFZ6Jnqie2r7T3r
4&type=public-
key&authData=XRCXqQSq0QFsTkp6kZM13RLIgxSV4pXxBe
2xBn_TISwFAAAGog==&clientData={"type":"webauthn
.get","challenge":"q1Fu1u41LspTqz3QxfUdLbJPydbc
uclI","origin":"https://www.pastery.net","cross
Origin":false}=&signature=3045022100d9847028155
9c385cbe56526caaa900c75db268c9dfd77ab05c73cfb47
56cb530220131191228f0a273d58a80c8de8359c44d20f4
248ba2f99e3468c8fec46e646d4&assertionClientExte
nsions={}
```

Supported implementations

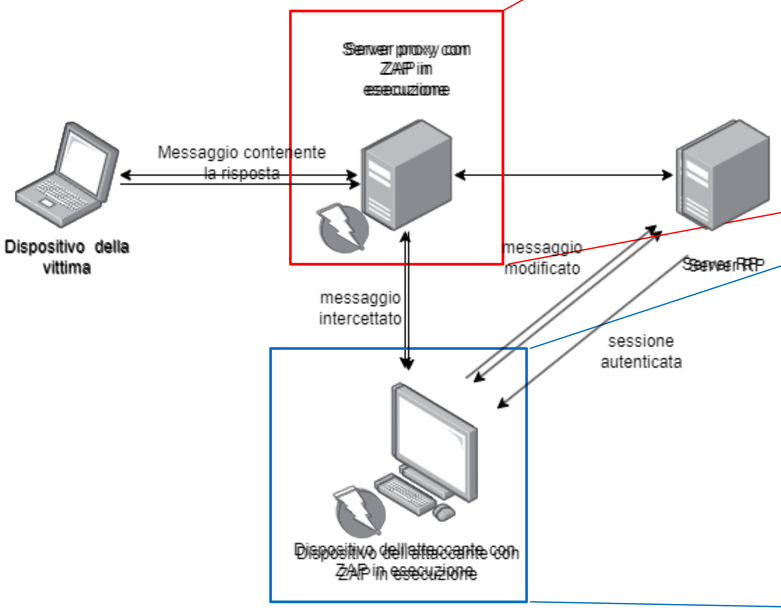


Is passwordless and two factor authentication based on WebAuthn API secure?

↳ USB dispositivi USB o sensori di impronte digitali per verificare l'identità dell'utente

Methodology

"id"="9V9I357mjvNIANoTWmpP9GDIlimwaYSJbtBqhTJI702vsPhatFZ6Jnqie2r7T3r4"&rawId=9V9I357mjvNIANoTWmpP9GDIlimwaYSJbtBqhTJI702vsPhatFZ6Jnqie2r7T3r4&type=public-key&authData=XRCXqQSq0QFsTkp6kZM13RLIgxSV4pXxBe2xBn_TISwFAAAGog==&clientData={"type":"webauthn.get","challenge":"q1Fu1u41LspTqz3QxfUdLbJPydbcucII","origin":"https://www.pastery.net","crossOrigin":false}&signature=3045022100d98470281559c385cbe56526caaa900c75db268c9dfd77ab05c73cfb4756cb530220131191228f0a273d58a80c8de8359c44d20f4248ba2f99e3468c8fec46e646d4&assertionClientExtensions={}



"id"="9V9I357mjvNIANoTWmpP9GDIlimwaYSJbtBqhTJI702vsPhatFZ6Jnqie2r7T3r4"&rawId=9V9I357mjvNIANoTWmpP9GDIlimwaYSJbtBqhTJI702vsPhatFZ6Jnqie2r7T3r4&type=public-key&authData=XRCXqQSq0QFsTkp6kZM13RLIgxSV4pXxBe2xBn_TISwFAAAGog==&clientData={"type":"webauthn.get","challenge":"q1Fu1u41LspTqz3QxfUdLbJPydbcucII","origin":"https://www.pastery.net","crossOrigin":false}&signature=3045022100d98470281559c385cbe56526caaa900c75db268c9dfd77ab05c73cfb4756cb530220131191228f0a273d58a80c8de8359c44d20f4248ba2f99e3468c8fec46e646d4&assertionClientExtensions={}

Websites Analyzed

Passwordless Authentication

- Adobe
- Authgear Demo
- Binance
- Bridgecrest
- Corbado
- Dinero
- Docusign
- FormX.ai
- GitHub
- Haeppie
- Hanckock
- Hanko
- Mangadex
- Microsoft
- NVIDIA
- Passkeys.guru
- Pastery.net
- Porkbun
- Sinology
- Zoho

Two Factor Authentication

- Docusign
- GitHub
- Mangadex
- INDIVIA
- Porkbun
- Yahoo|
- Zoho

Identified vulnerabilities

V1 - No TLS binding

V2 - No invalid challenge

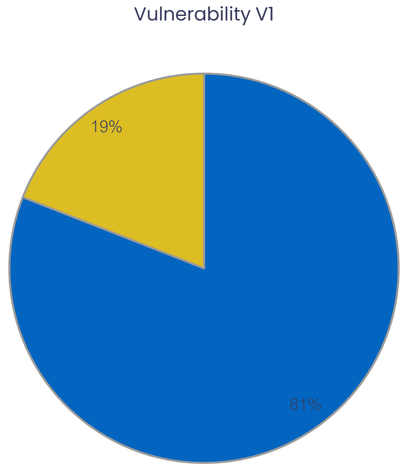
V3 - No verification that the owner of the credentials and of the account are the same entity

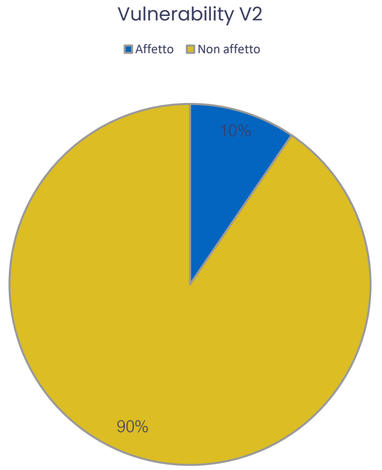
V4 - No verification that the owner of the credentials and of the account are the same entity 2FA

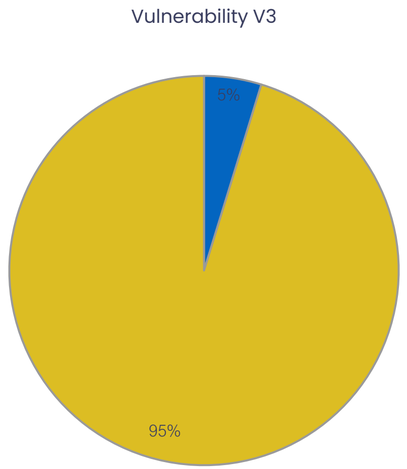
Identified vulnerabilities

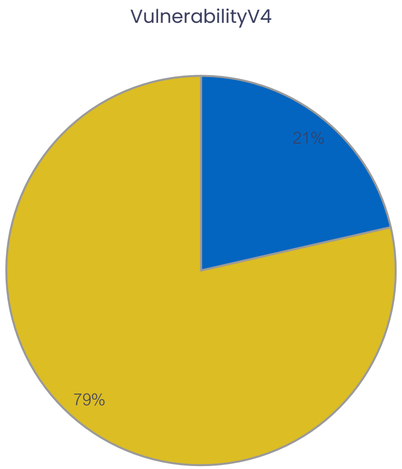
Passwordless Authentication	Vulnerabilities		
	V1	*V2*	*V3*
Adobe	✓	✗	✗
Authgear	✓	✗	✗
Binance	✓	✗	✓
Bridgecrest	✓	✗	✗
Corbado	✓	✗	✗
Dinero	✓	✗	✗
Docusign	✓	✗	✗
FormX	✓	✗	✗
GitHub	✓	✗	✗
Haeppie	✓	✗	✗
Hanckock	✗	✗	✗
Hanko	✓	✗	✗
Mangadex	✓	✗	✗
Microsoft	✗	✗	✗
NVIDIA	✗	✗	✗
Passkeys.guru	✓	✗	✗
Pastery.net	✓	✗	✗
Porkbun	✓	✓	✗
Sinology	✗	✗	✗
Zoho	✓	✓	✗

Two Factor Authentication	Vulnerabilities			
	V1	*V2*	*V3*	*V4*
Docusign	✓	✗	✗	✓
GitHub	✓	✗	✗	✓
Mangadex	✓	✗	✗	✓
NVIDIA	✗	✗	✗	✗
Porkbun	✓	✓	✗	✓
Yahoo	✓	✗	✗	✓
Zoho	✗	✗	✗	✗

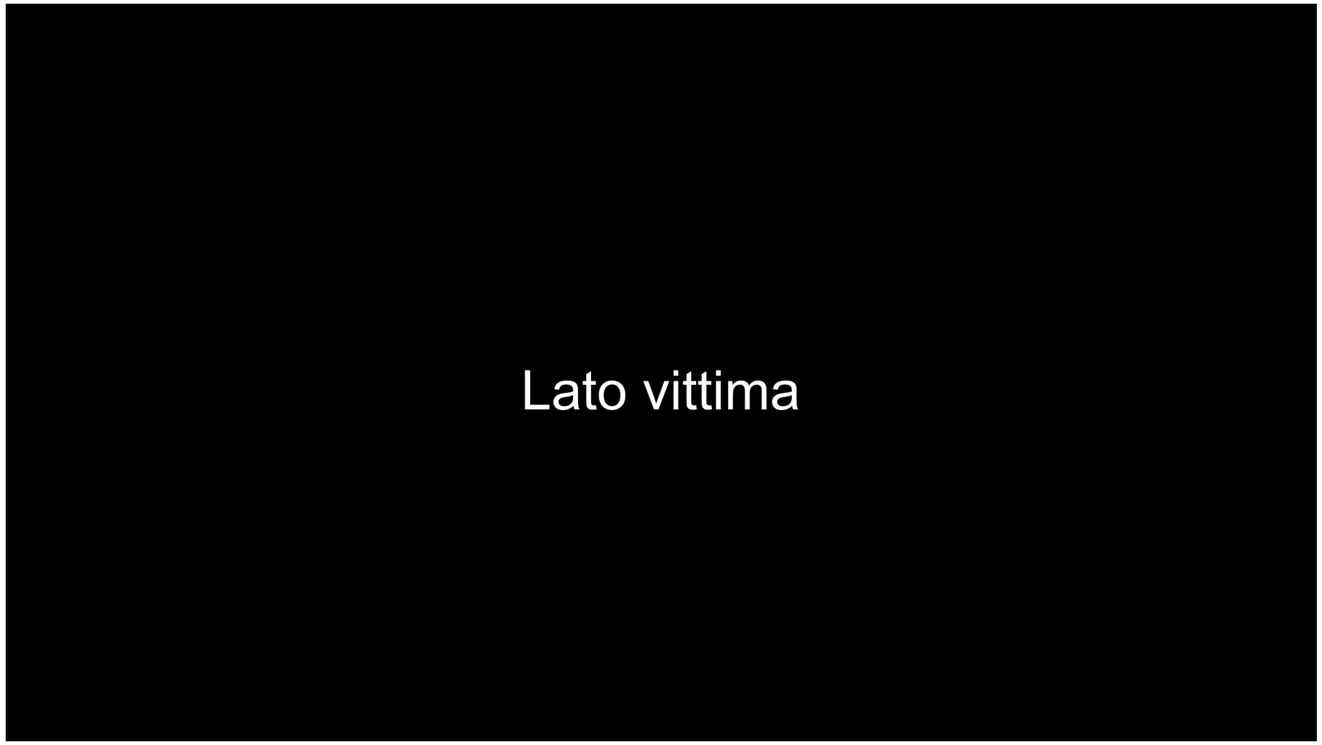


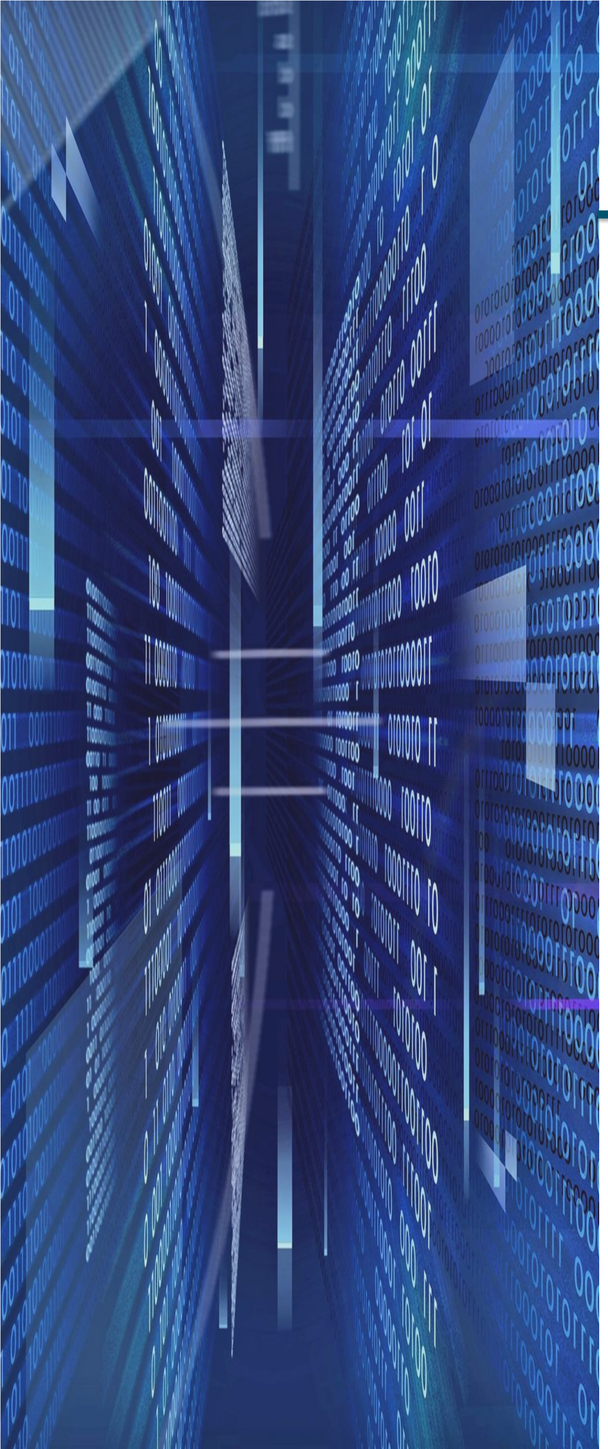






Exploitation of vulnerability V1

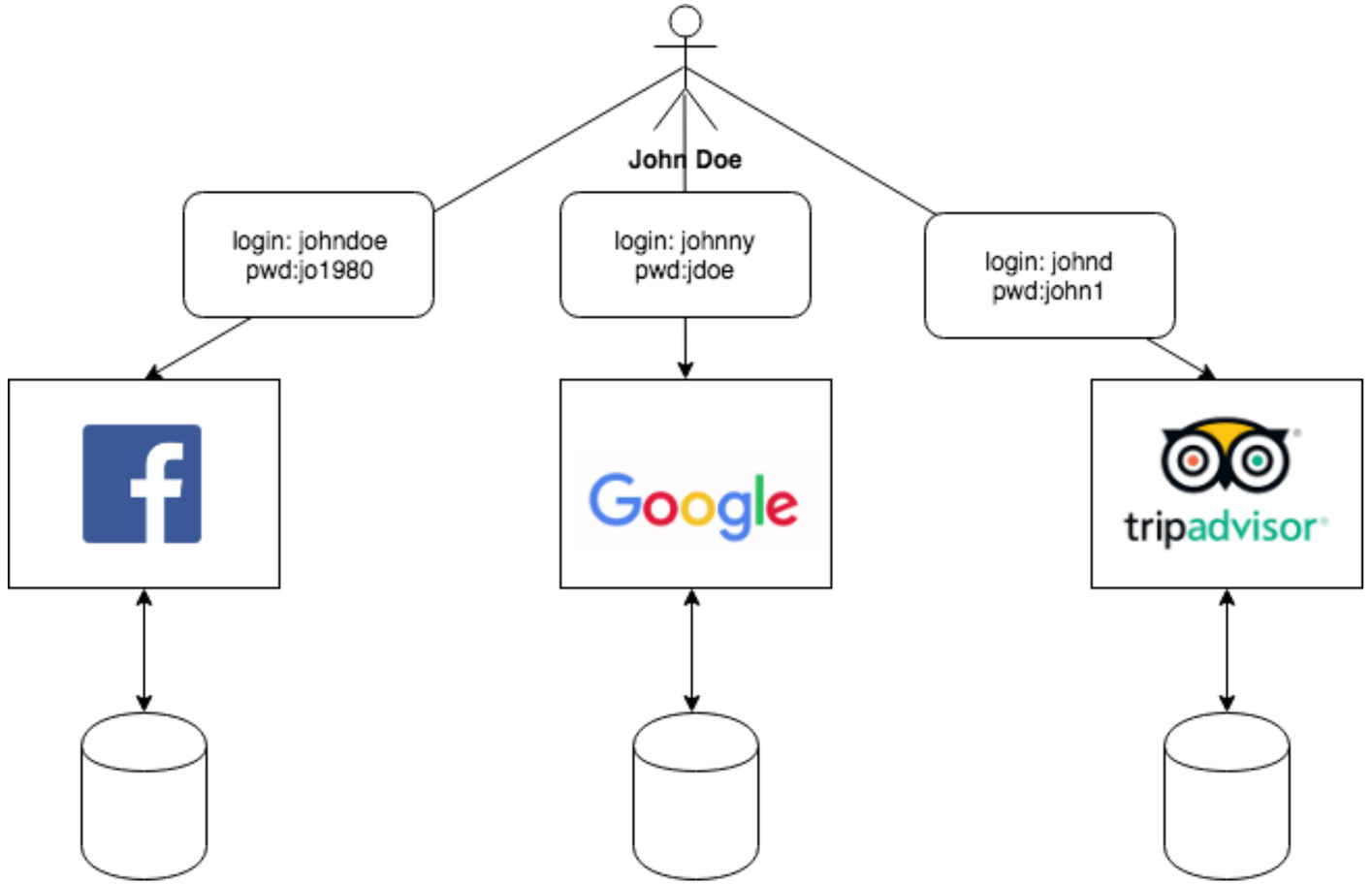




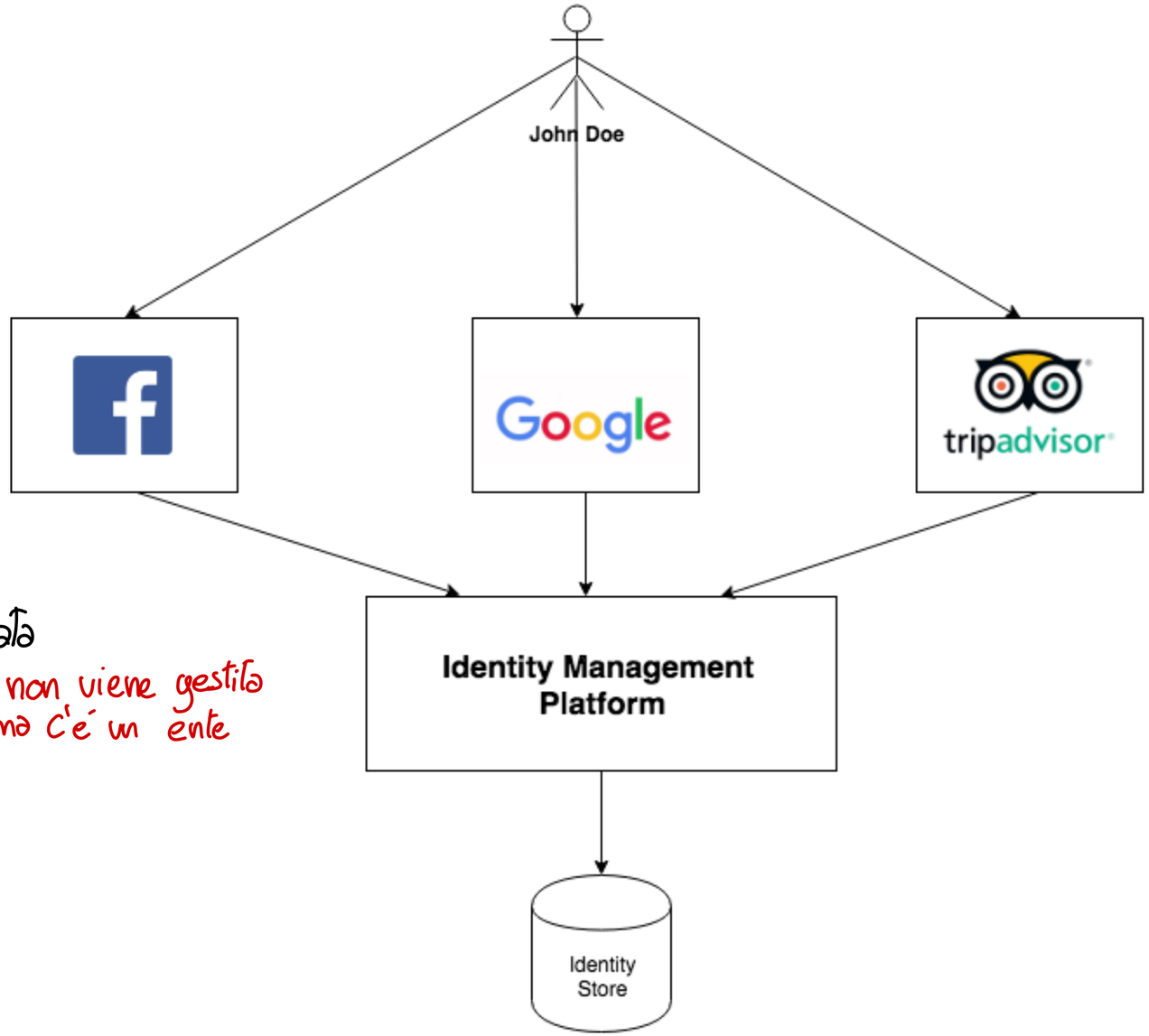
Single Sign On Protocols

Traditional authentication systems

Ogni sito provede al proprio sistema di autenticazione



The solution



Soluzione centralizzata
l'autenticazione non viene gestita dai singoli siti ma c'é un ente preposto

Digital Identity Management

- A **digital identity** is the digital representation of the information know about a specific individual
  - Name and last name
  - National insurance number
  - Home address
  - Job title
  - User id and password
- A digital identity management system provides a centralized solution that manages users' digital identities and user access to resources/services.
  - Maintain identity of the user and associates attributes to this identity
  - Verify identity of the user based on his/her identity attributes

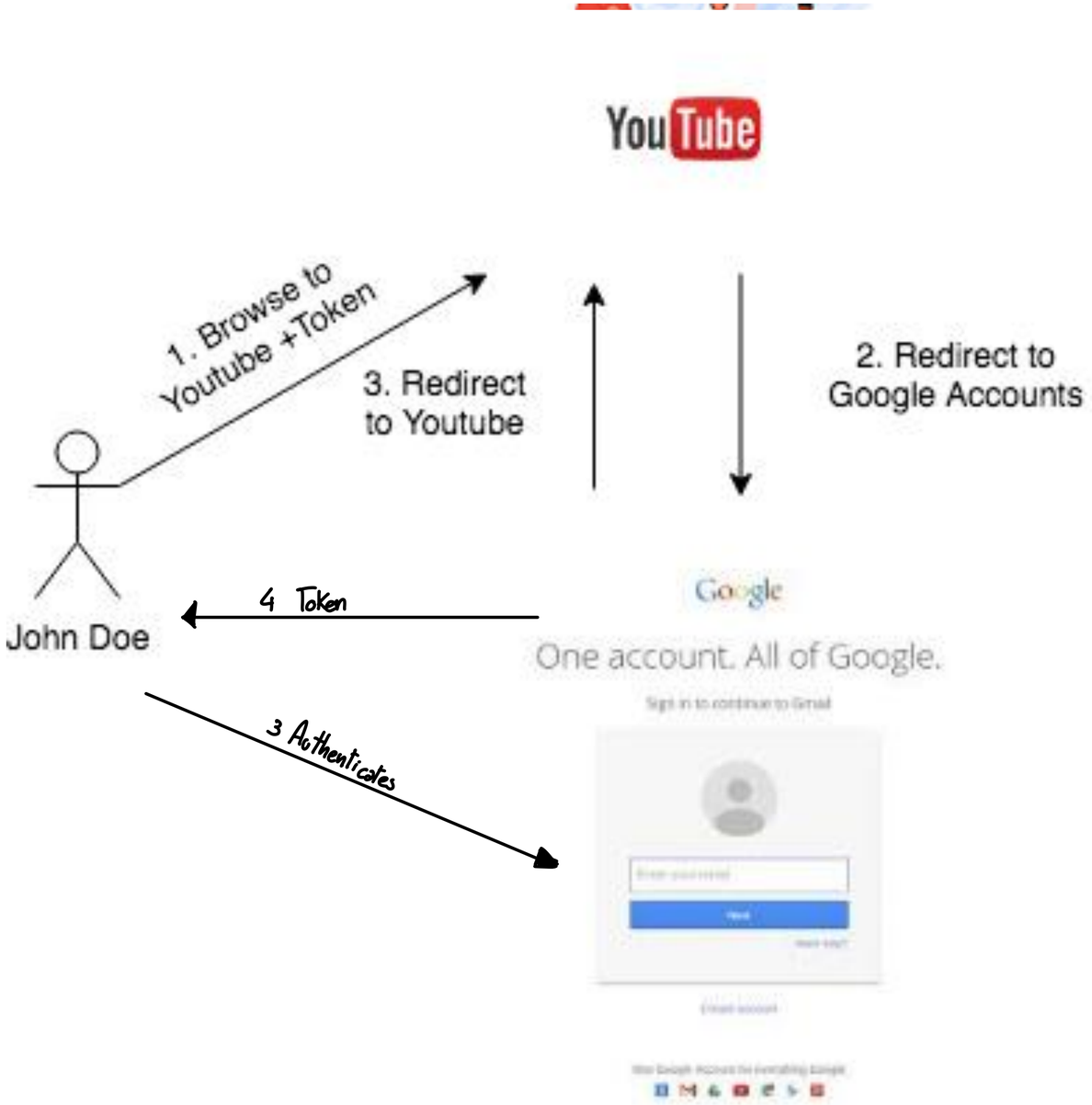
Main Players

- Subject or User
  - System entity about which something can be asserted
- Asserting Party or Identity Provider
  - System entity that creates assertions about a subject
- Relying Party or Service Provider
  - System entity that consumes assertions about a subject

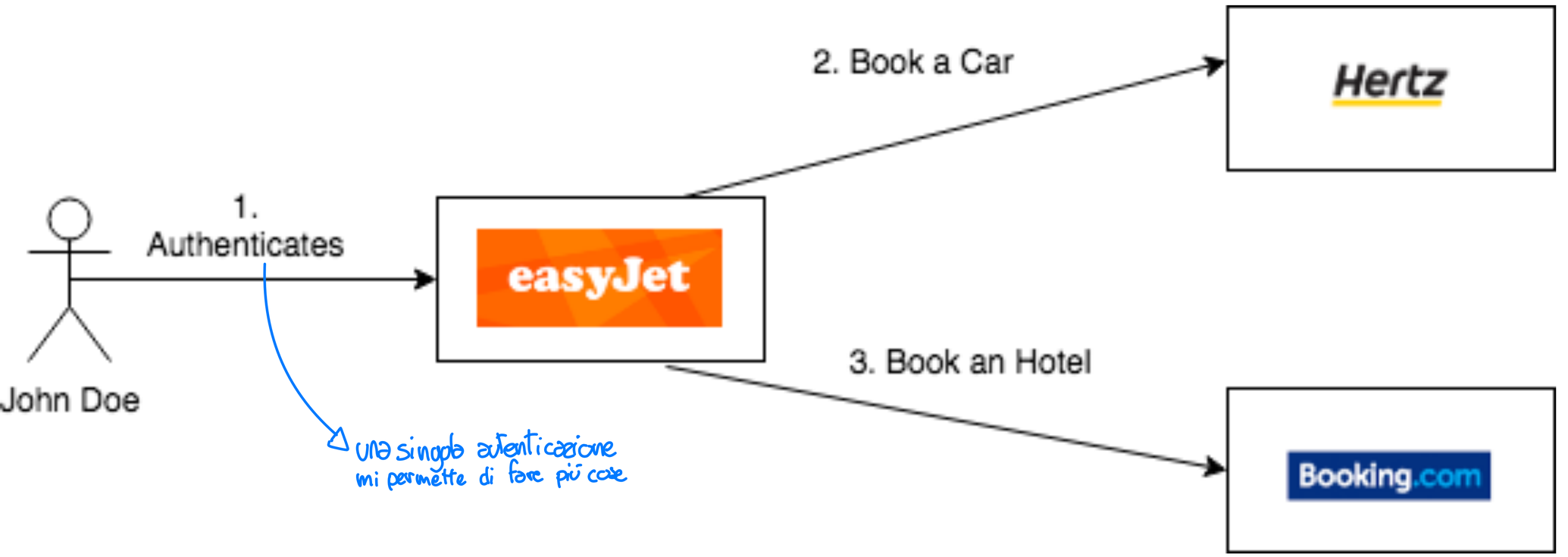
What is Single-Sign On

- User authenticates once and then access all the resources the user is authorized to use
- Authentication to the individual resources is handled by the Identity Provider in a manner that is transparent to the user
- Identity Provider maintains the identity information of the user
- When the user has to authenticate again for a resource, the Identity Provider does the job for the user

An Example of SSO



An Example of Cross Domain SSO



Federated Identity

- Organizations reach an agreement and establish a common, shared identifier to refer to a subject
- Facilitate sharing on identity information across different organisations
- Facilitates SSO
  - An end-user that "logs into" any member of the federation has effectively logged into all of the members
- Reduces costs of maintaining and managing identities
  - No need independently and maintain collect identity information

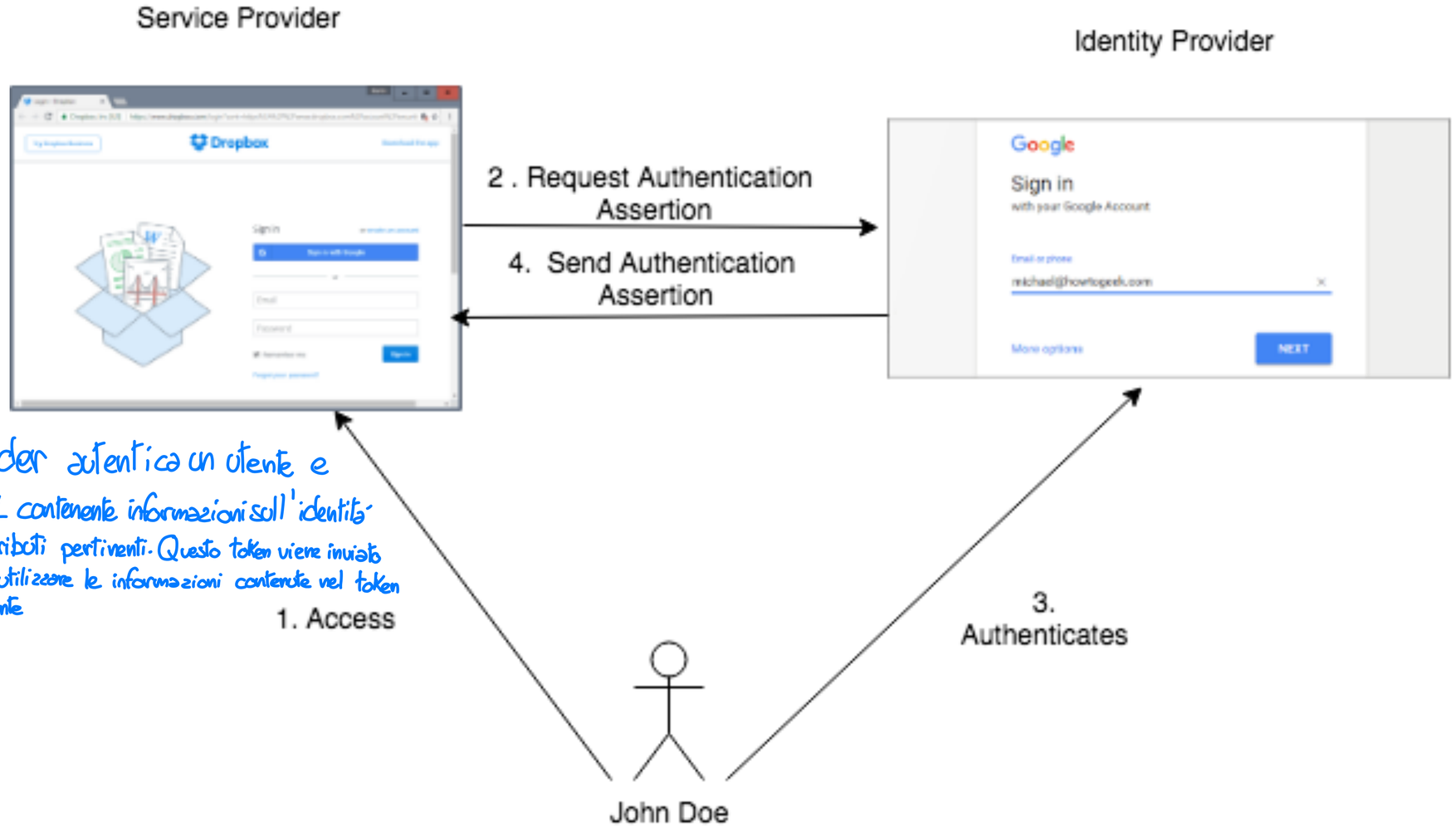
SAML

- SAML enables SSO and Identity Federation by providing a standard representation for attribute assertions and authentication assertion
- SAML Asserting Party/Identity Provider verifies identity of a user and issues an authentication assertion
- The user can present to a Service Provider the authentication assertion without authenticating again

Il SAML é uno standard XML utilizzato per lo scambio di informazioni di autenticazione e autorizzazione tra parti. In particolare tra servizi di identità e servizi di fornitura servizi. Progettato per facilitare autenticazione e autorizzazione basate su token in ambienti distribuiti.

SAML Use Cases – SP Initiated SSO

ampiamente usato in sistemi SSO in scenari aziendali



Un identity Provider autentica un utente e genera un token SAML contenente informazioni sull'identità dell'utente, i ruoli e altri attributi pertinenti. Questo token viene inviato al service provider che può utilizzare le informazioni contenute nel token per concedere l'accesso all'utente

SAML Assertions

- Declaration of facts about a subject that an asserting party claims to be true
- Three types of assertions
  - Authentication statements
    - describe the means used to authenticate a subject
  - Attribute statements
    - list attributes that a subject has
  - Authorization statements
    - Define subject's permissions

SAML Authentication Request <AuthnRequest>

- The following attributes and elements have to be present:
  - **ID** Newly generated number for identification
  - **Version**, SAML current version which is 2.0
  - **IssueInstant** the time when the request was issued UTC format
  - **AssertionConsumerServiceURL** The SAML URL interface of the service provider, where the Identity provider sends the authentication token
  - **<Subject>** to specify the user to be authenticated
  - **<Issuer>** to indicate the unique identifier of the Service Provider who generated the request
  - **<NameIDPolicy>** it indicates the level of security required for the user authentication

SAML <Response>

- It must contain the following attributes and elements
  - **ID** unique identifier of the response
  - **Version**, SAML version
  - **IssueInstant** the time the response has be issued in UTC format;
  - **InResponseTo**, the ID of the authentication request;
  - **Destination**, the URI of the Service provider
  - **<Status>** indicates whether the authentication was successfull or not
  - **<Issuer>** the identifier of the Identity Provider
  - **<Assertion>** contains information about the authentication method and eventually identity attributes of the user
  - **<Signature>** contains the digital signature of the Identity Provider

SAML Authentication Assertion

- It has to contain contain the following attributes and elements:
  - **ID** unique identifier
  - **Version,** SAML version
  - **IssueInstant** issuance time in UTC format
  - **<Subject>** indicates the user to be authenticated
  - **<Issuer>** indicates the Identity Provider
  - **<Conditions>** specifies the time interval when the assertion is valid
  - **<AudienceRestriction>** specifies the Service Provider that is intended to consume the assertion
  - l'elemento **<AuthStatement>** specifies the authentication context
  - **<AttributeStatement>** lists the identity attributes certified by the Identity Provider
  - **<Signature>** contains the signature of the Identity Provider

SPID


INPS
L'Istituto
Dati, ricerche e bilanci
Avvisi, bandi e fatturazione
INPS Comunica
Prestazioni e servizi
Amministrazione trasparente
Assistenza
Contatti
Dichiarazioni di accessibilità
Indietro
Vai a MyINPS
Cerca
/ Prestazioni e Servizi / Autenticazione
Autenticazione
PIN
SPID
CIE
CNS
SPID è il sistema di accesso che consente di utilizzare, con un'identità digitale unica, i servizi online della Pubblica Amministrazione e dei privati accreditati. Se sei già in possesso di un'identità digitale, accedi con le credenziali del tuo gestore. Se non hai ancora un'identità digitale, richiedila ad uno dei gestori.
Maggiori informazioni su SPID
Entra con SPID
spid
AgID Agenzia per l'Italia Digitale

Main Players

- **Agenzia per l'Italia Digitale (AgID):** the entity that monitors and authorizes entities to issuing SPID
- **Identity Provider:** private and public entity certified by AgID which has to verify the identity of user and issue SPID
- **Service Provider**: private or public entity which offers online services
- **Attribute Provider:** entity which issues to users the qualifying attributes
- **User:** owner of SPID who uses it to access online services

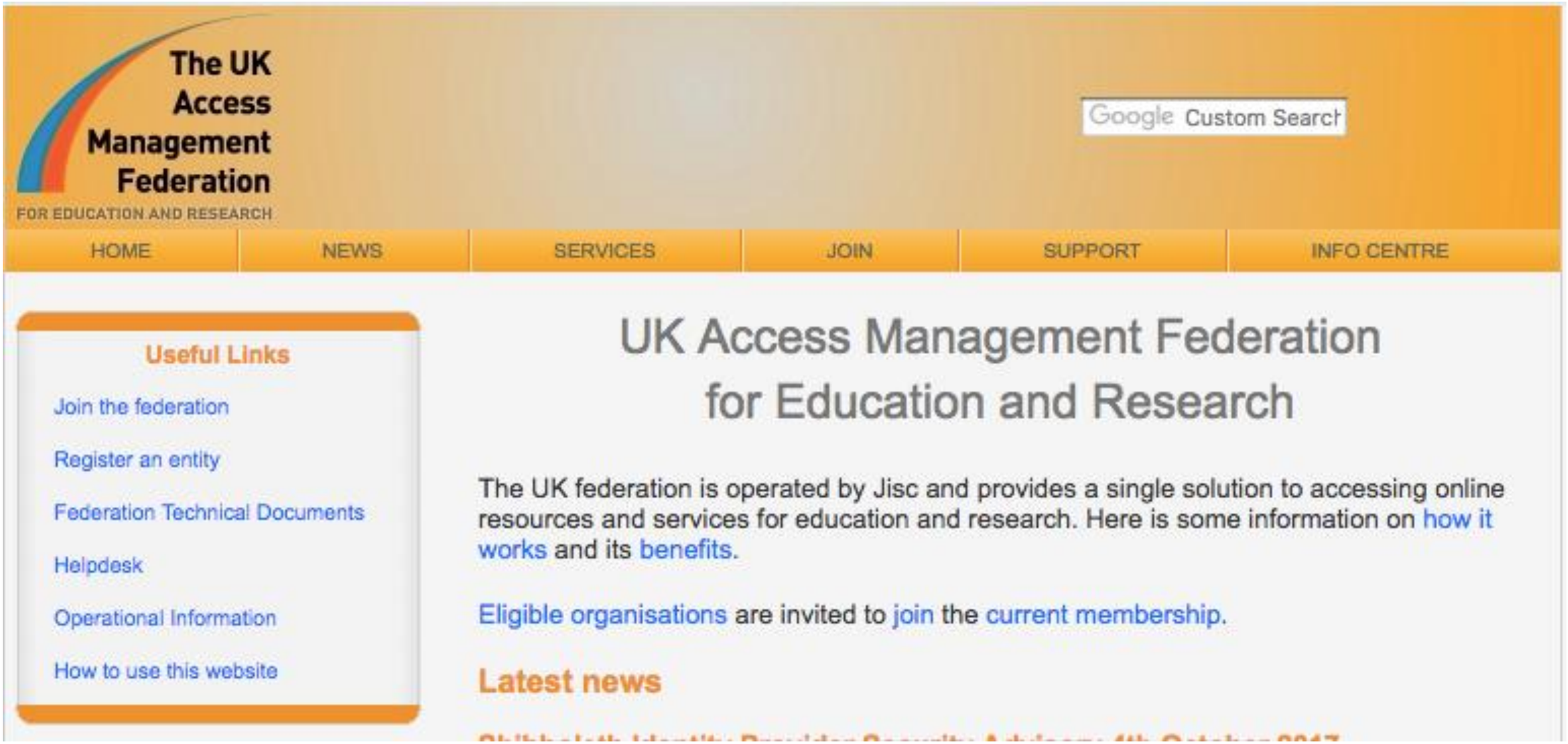
SPID levels of secure authentication

- **level 1**, allows access with username and password;
- **level 2**, allows access with SPID level 1 credentials and the generation of a temporary OPT access code (one time password) or the use of an APP accessible from smartphone or tablet;
- **level 3**, allows access with SPID credentials and the use of additional security solutions and any physical devices (e.g. smart card) that are granted by the identity provider

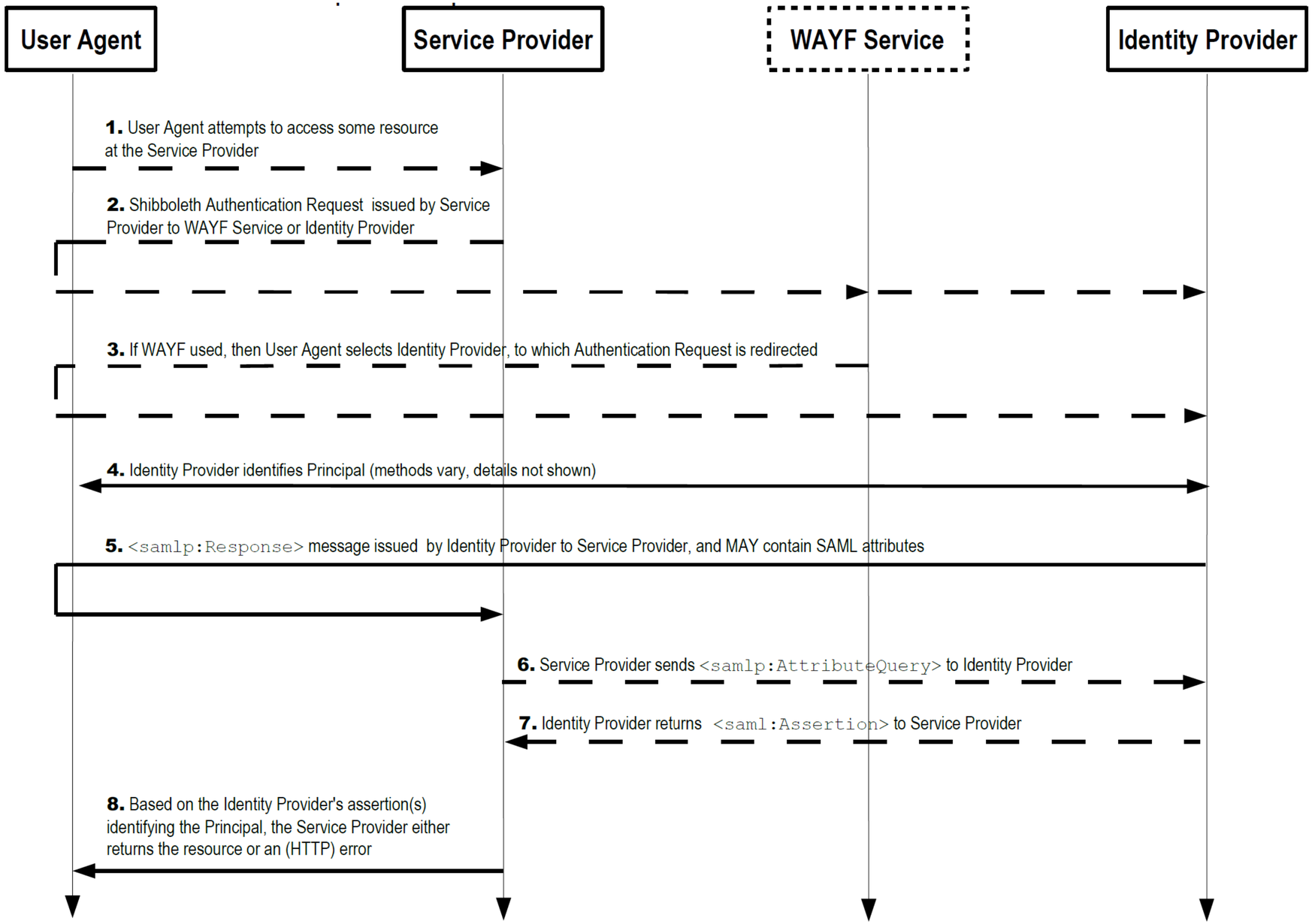
Shibboleth

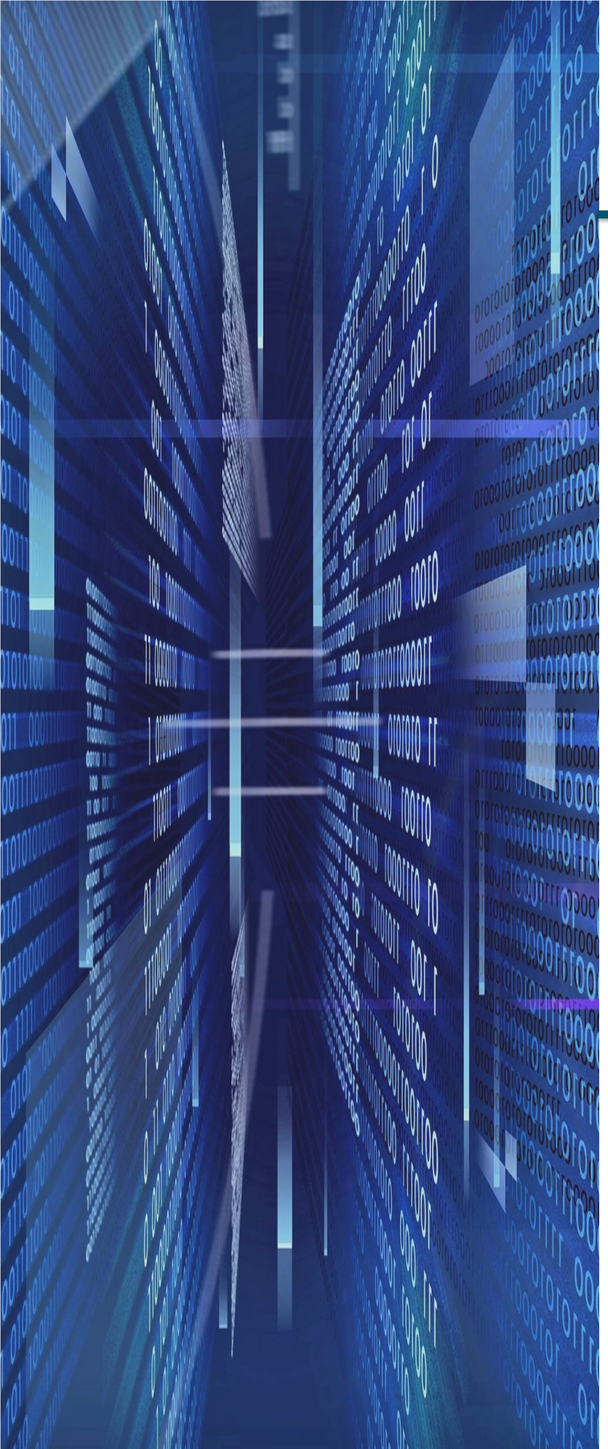
- Shibboleth is an Internet2 consortium project that enables universities to share resources and research across institutional boundaries.
- Shibboleth enable students, faculty, and staff to access resources at partner institutions without needing to create separate, local IDs and passwords for each one

An example of federation



Shibboleth authentication flow





OpenID Connect

OpenID Connect

- OpenID Connect (OIDC) is an open authentication protocol that profiles and extends OAuth 2.0 to add an identity layer
- OIDC allows clients to confirm an end user's identity using authentication by an authorization server
- Implementing OIDC on top of OAuth 2.0 creates a single framework that promises to secure APIs, mobile native applications, and browser applications in a single, cohesive architecture

progettato per aggiungere un layer di identità standardizzato e sicuro alle applicazioni web
offre un modo per autenticare gli utenti e ottenere informazioni sull'identità dell'utente in modo sicuro.

A simple scenario

Resource Owner

Client Application

Authorization Server/
Resource Server

Protected Resources

What is OAuth?

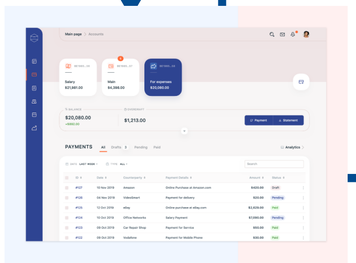
- It is a standard **authorization protocol** that allows a third-party application to access protected resource hosted on a HTTP server
- It requests an **access token** from the Authorization Server
- Then the third-party application uses the access token to request access to the protected resources

Actors

- **Resource Owner:** entity capable of granting access to a protected resource
- **Resource Server**: the server that stores that resource owner resources
- **Authorization Server:** the server issuing access token to the client after authenticating the resource owner and obtaining its authorization
- **Client:** a third-party application that requests access to protected resources on-behalf of resource owner and with its approval

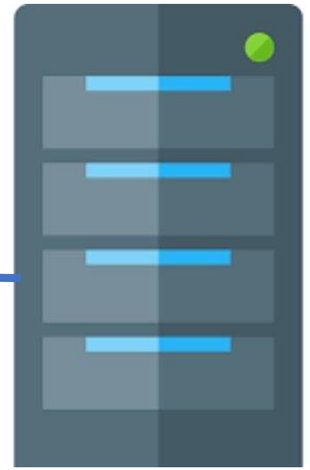
OAuth -The Authorization Code Flow

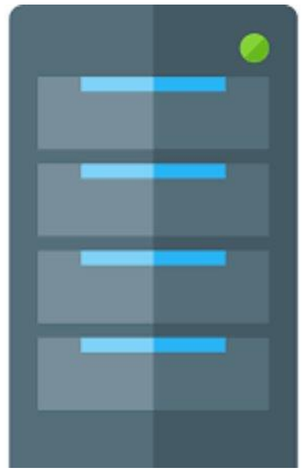
Alice – Resource Owner



My Bank App – Client Application

Credit Union Bank - Authorization Server





Credit Union Bank - Resource Server

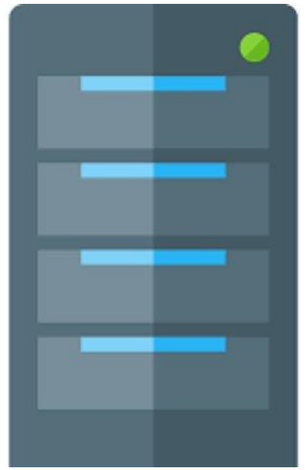
https://example.com/oauth2/register?
application_name=Mybank&
redirect_uri=https://example.com/callback&
grant_type=code

client_id & client_secret

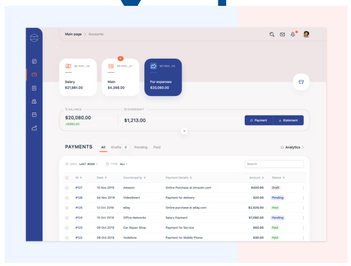
OAuth - The Authorization Code Flow

Alice – Resource Owner

Credit Union Bank - Authorization Server

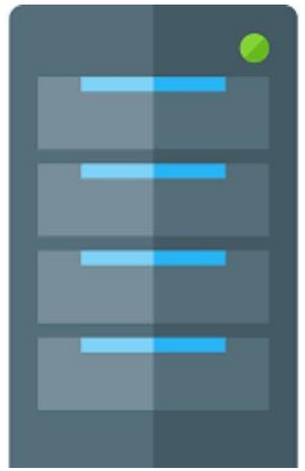


Alice Authenticates



My Bank App – Client Application

```
https://example.com/oauth2/authorize?
response_type=code&
client_id=<client id>&
redirect_uri=https://example.com/callback&
scope=bankaccount&
state=123xyz
```

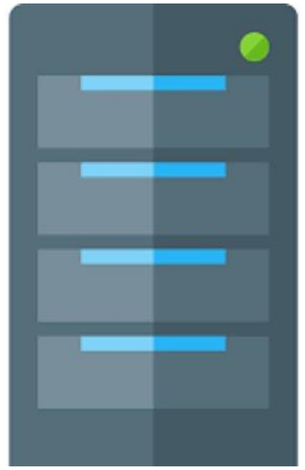


Credit Union Bank - Resource Server

OAuth -The Authorization Code Flow

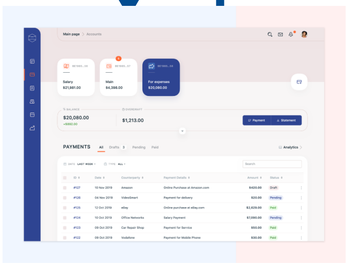
Alice – Resource Owner

Credit Union Bank - Authorization Server

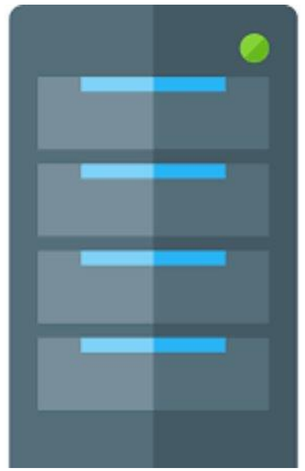


Authorization Code

https://example.com/oauth2/authorize?
code=8042d4cad58c66d0a5edfad89521192&
state=123xyz



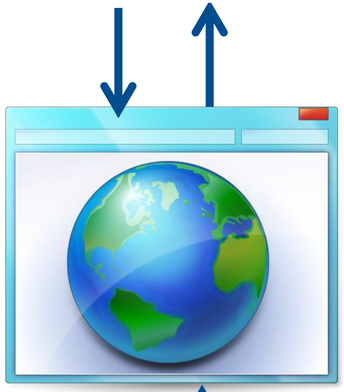
My Bank App – Client Application

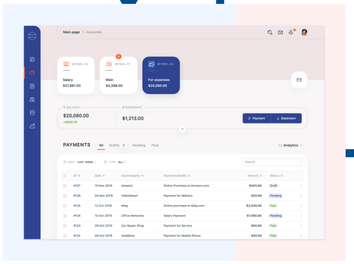


Credit Union Bank - Resource Server

OAuth - The Authorization Code Flow

Alice – Resource Owner



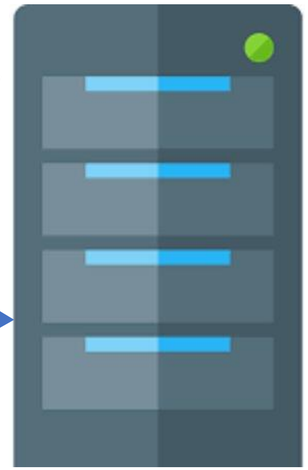


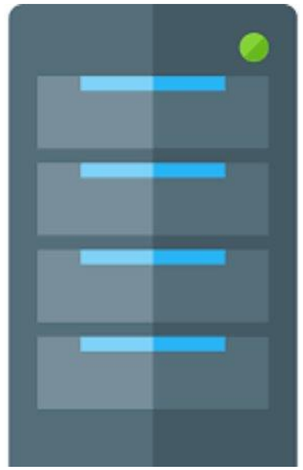
My Bank App – Client Application

POST https://example.com/oauth2/token
grant_type=authorization_code&
code=8042d4cad58c66d0a5edfad89521192&
redirect_uri=https://example.com/callback&
client_id=<client id>&
client_secret=<client secret>

Authorization Code & Redirection URI

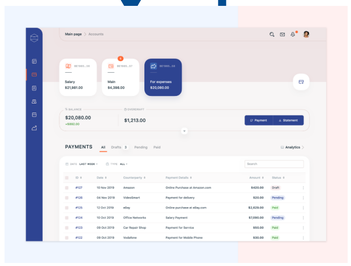
Credit Union Bank - Authorization Server





Credit Union Bank - Resource Server

OAuth -The Authorization Code Grant Flow

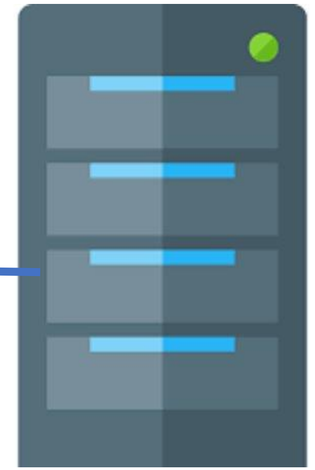


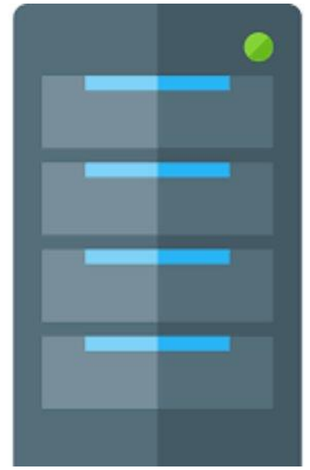
My Bank App –
Client Application

```
HTTP/1.1 200 OK
 Content-Type: application/json;charset=UTF-8
     {
       "access_token":"2YotnFZFEjr1zCsicMWpAA",
       "token_type":”bearer",
       "expires_in":3600,
       "refresh_token":"tGzv3JOkF0XG5Qx2TlKWIA”
      }
```

Access token & Refresh token

Credit Union Bank -
Authorization Server



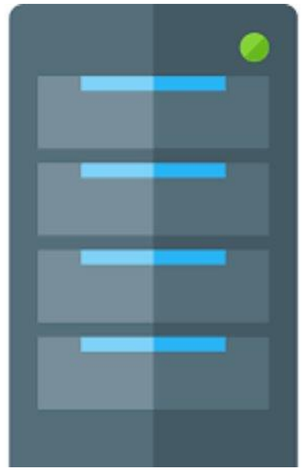


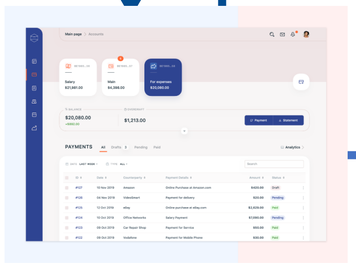
Credit Union Bank -
Rsource Server

OAuth - The Authorization Code Flow

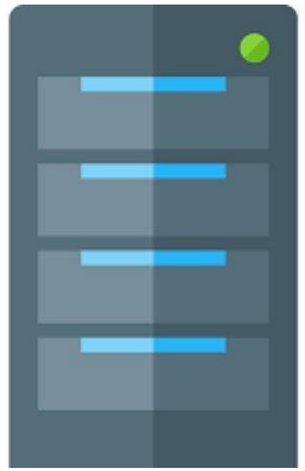
Alice – Resource Owner

Credit Union Bank - Authorization Server





Access token



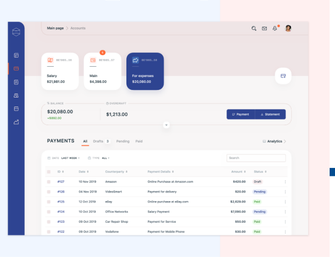
Credit Union Bank - Resource Server

OpenID Connect-The Authorization Code Flow

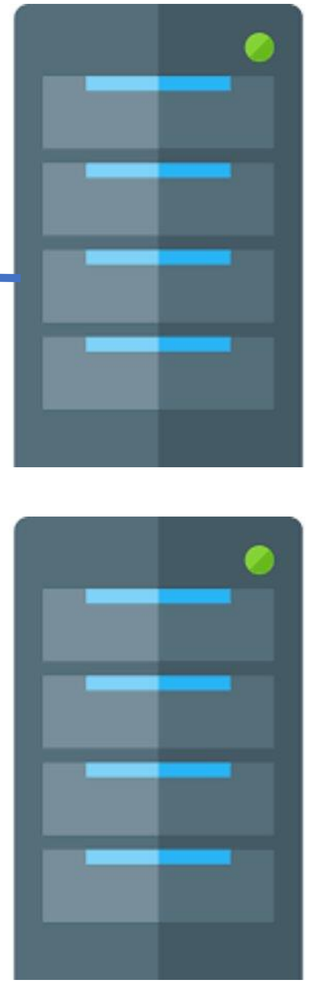
Alice – Resource Owner

Credit Union Bank - Authorization Server

https://example.com/oauth2/register?
application_name=Mybank&
redirect_uri=https://example.com/callback&
grant_type=code



client_id & client_secret



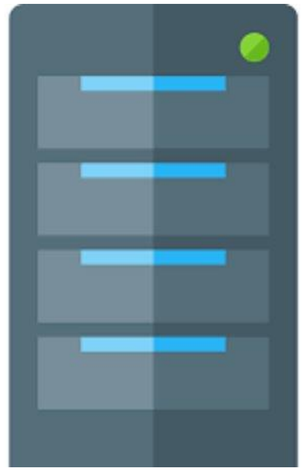
My Bank App – Client Application

Credit Union Bank - Resource Server

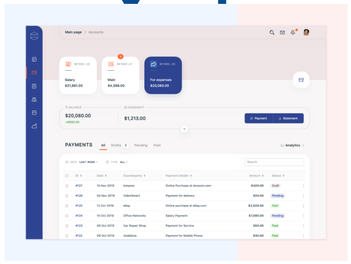
OpenID Connect - The Authorization Code Flow

Alice – Resource Owner

Credit Union Bank - Authorization Server

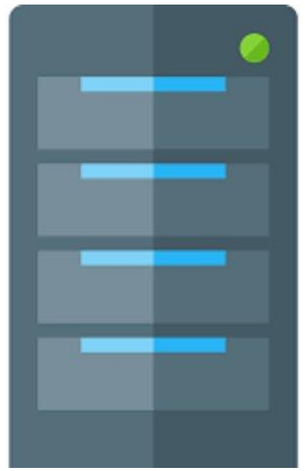


Alice Authenticates



My Bank App – Client Application

https://example.com/oauth2/authorize?
response_type=code&
client_id=<client id>&
redirect_uri=https://example.com/callback&
scope=open-id&profile&email
state=123xyz

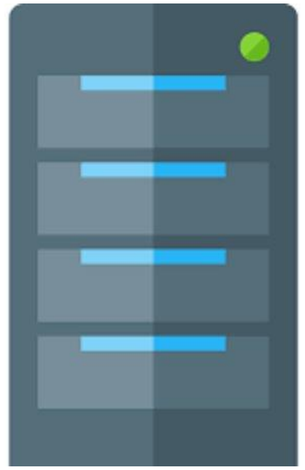


Credit Union Bank - Resource Server

OpenID Connect -The Authorization Code Flow

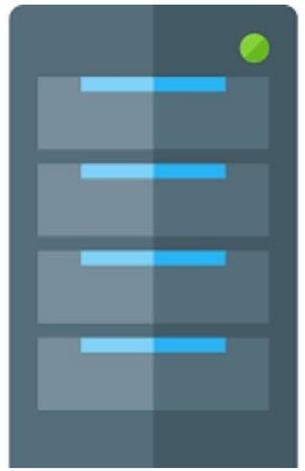
Alice – Resource Owner

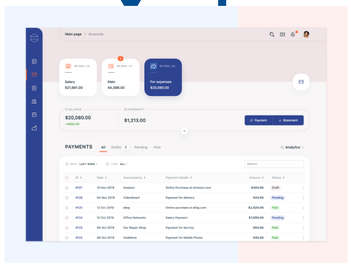
Credit Union Bank - Authorization Server



Authorization Code

https://example.com/oauth2/authorize?
code=8042d4cad58c66d0a5edfad89521192&
state=123xyz



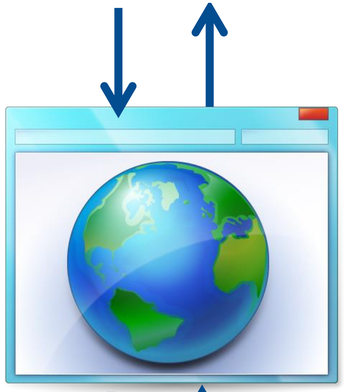


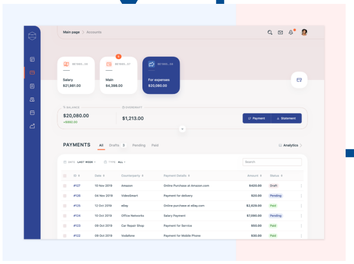
My Bank App – Client Application

Credit Union Bank - Resource Server

OpenID Connect - The Authorization Code Flow

Alice – Resource Owner



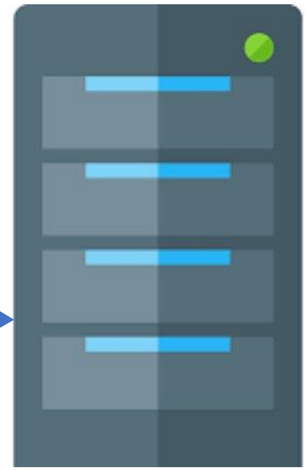


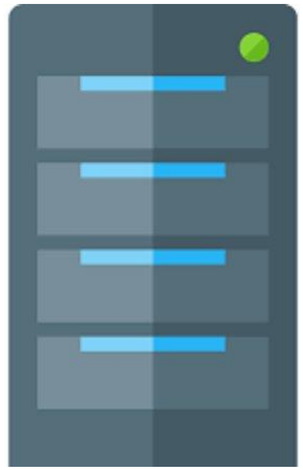
My Bank App – Client Application

POST https://example.com/oauth2/token
grant_type=authorization_code&
code=8042d4cad58c66d0a5edfad89521192&
redirect_uri=https://example.com/callback&
client_id=<client id>&
client_secret=<client secret>

Authorization Code & Redirection URI

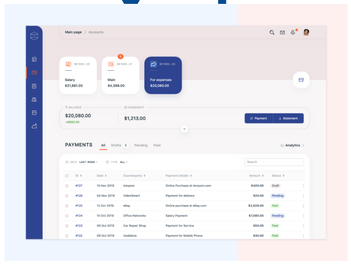
Credit Union Bank - Authorization Server





Credit Union Bank - Resource Server

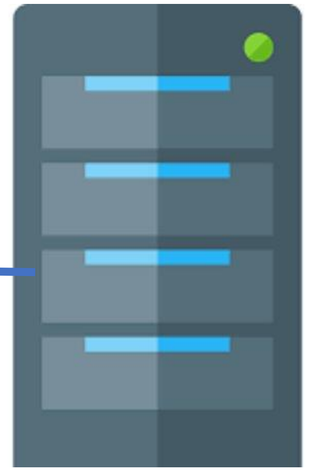
OpenID Connect -The Authorization Code Grant Flow

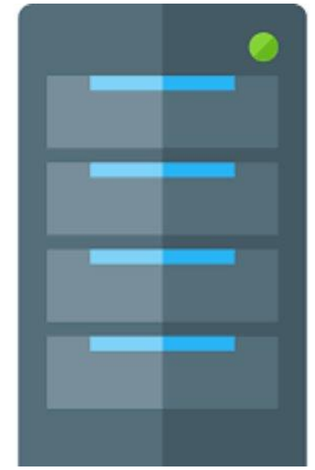


```
HTTP/1.1 200 OK
 Content-Type: application/json;charset=UTF-8
     {
       "access_token":"2YotnFZFEjr1zCsicMWpAA",
        id_token: ":"KLotnFZFEjr1zCsicMWpAA"
       “token_type":”bearer",
       "expires_in":3600,
            }
```

Access token & IDToken

Credit Union Bank - Authorization Server





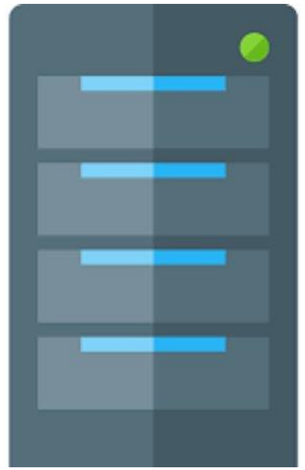
Credit Union Bank - Rsource Server

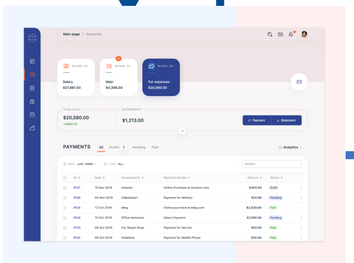
My Bank App – Client Application

OpenID Connect - The Authorization Code Flow

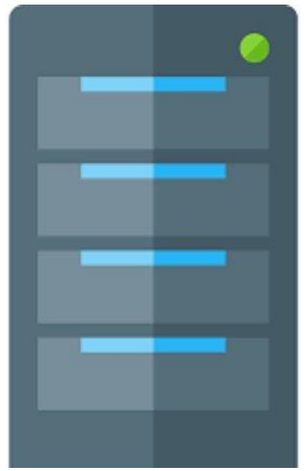
Alice – Resource Owner

Credit Union Bank - Authorization Server





Access token



My Bank App – Client Application

Credit Union Bank - Resource Server

Id Token

```
{
  "sub"       : "alice",
  "iss"       : "https://openid.c2id.com",
  "aud"       : "client-id",
  "nonce"     : "n-0S6_WzA2Mj",
  "auth_time" : 1311280969,
  "acr"       : "c2id.loa.hisec",
  "iat"       : 1311280970,
  "exp"       : 1311281970
}
```

chi ha verificato la validità

inizio

fine validità

Summary

- SSO simplifies user authentication across different web sites
  - User login once at a web site and then can access resources to other web sites
- Federated SSO allows users to access websites from other service providers
- Federated SSO is implemented through standardization efforts
  - SAML: XML-based framework
  - Shibboleth: SAML-based protocol for educational institutions
  - OpenId Connect: single sign on protocol for web, mobile and cloud apps

Resources

- WebAuth guide
  - https://webauthn.guide/
- SAML
  - https://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.html
- OAuth Web site
  - https://oauth.net/2/