Digital Identity

Credits Carlo Mazzocca - UNISA
cmazzocca@unisa.it

Nicolò Romandini

Post-doc @DISI

Learning Goals

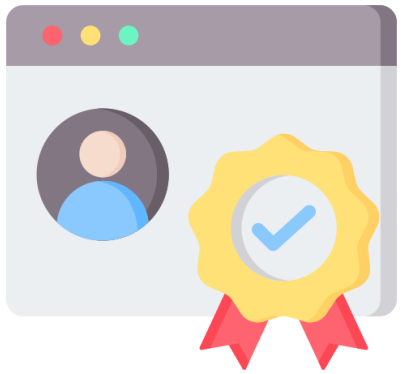
- Evolution of digital identity models
- Self-Sovereign Identity
- Implementing the Self-Sovereign Identity paradigm

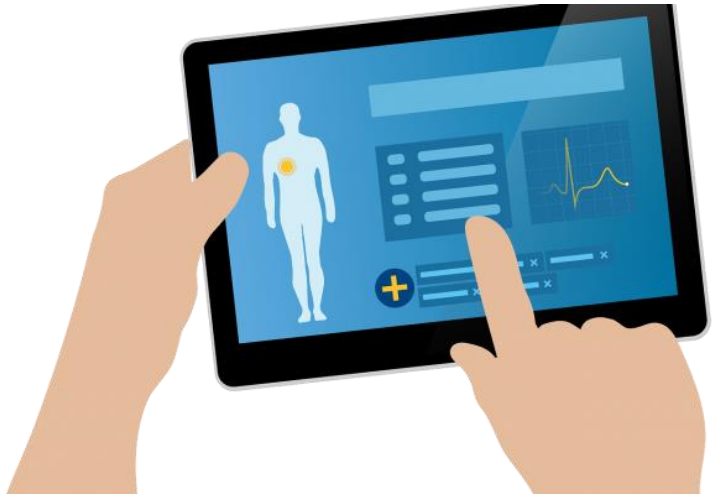
Digital Identity Evolution

Digital Identity

The recent proliferation of digital services has resulted in an **unprecedented increase** in the number of digital identities

Digital identity refers to the collection of information, attributes, and credentials that uniquely represent an individual, organization, or entity online

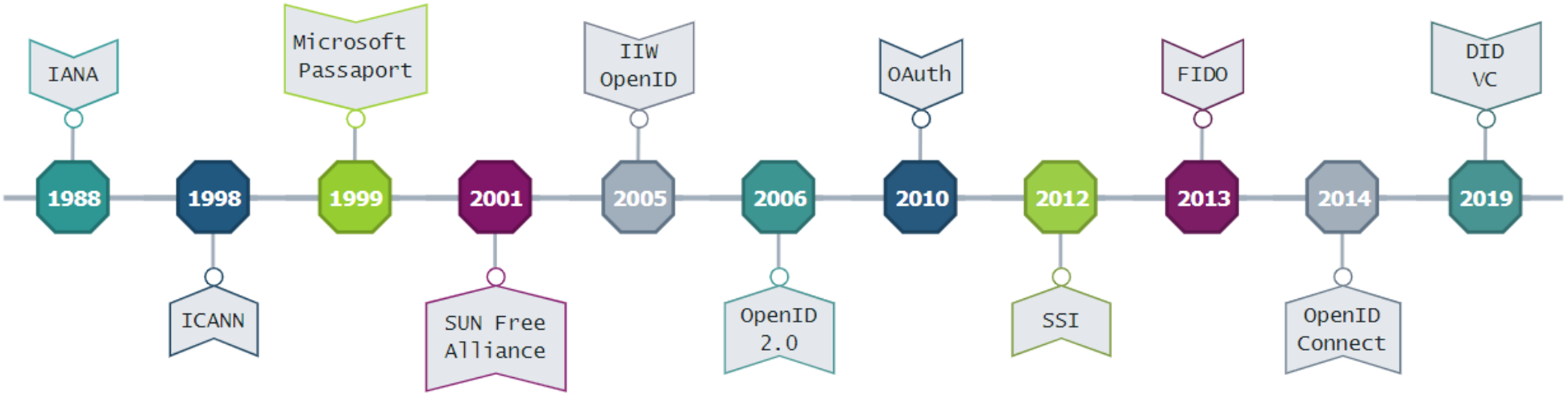




ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Many Eras

Digital identification has gradually shifted from centralized to **decentralized models**



Centralized Identity

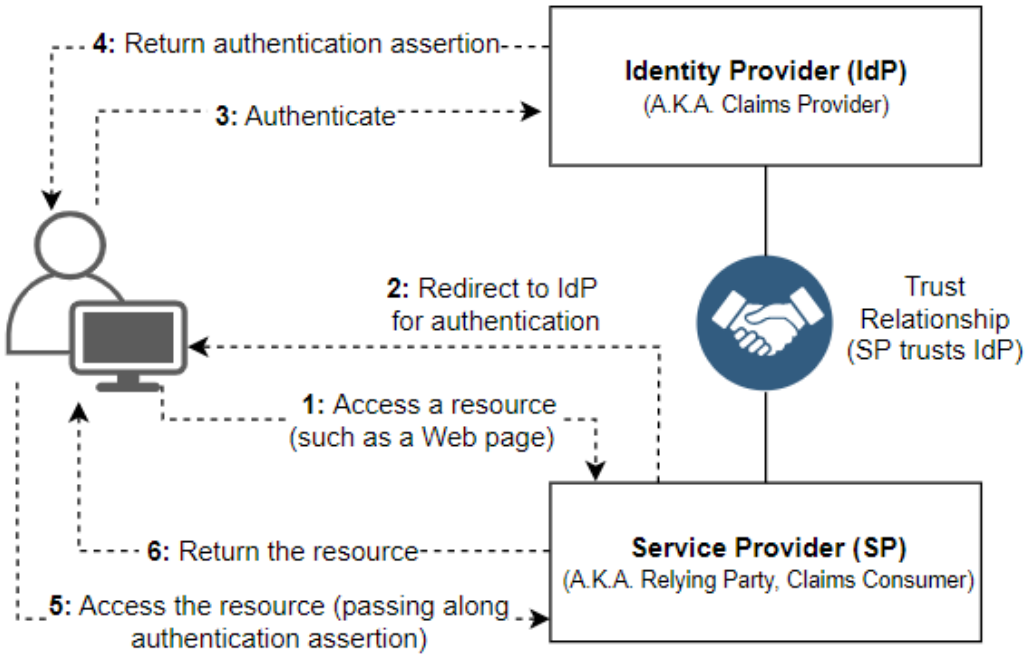
- Central authority → **data breaches** and **single point of failure**
- Combination of **username and password** → many vulnerabilities, **user prefer convenience over security**
- As many identities as the number of services → **poor usability**
- Service providers must **securely store**, **preserve**, and **safeguard user data** → **companies can be charged**


Mark Zuckerberg
16 hrs · Menlo Park, CA, United States ·
I want to share an update on the Cambridge Analytica situation -- including the steps we've already taken and our next steps to address this important issue.
We have a responsibility to protect your data, and if we can't then we don't deserve to serve you. I've been working to understand exactly what happened and how to make sure this doesn't happen again. The good news is that the most important

Federated Identity

A group of trusted entities share authentication responsibilities, allowing **users to access multiple services with a single identity**

- Individuals can use the same identity across multiple services
- Users are not required to create different accounts
- Reduces the number of centralized entities involved



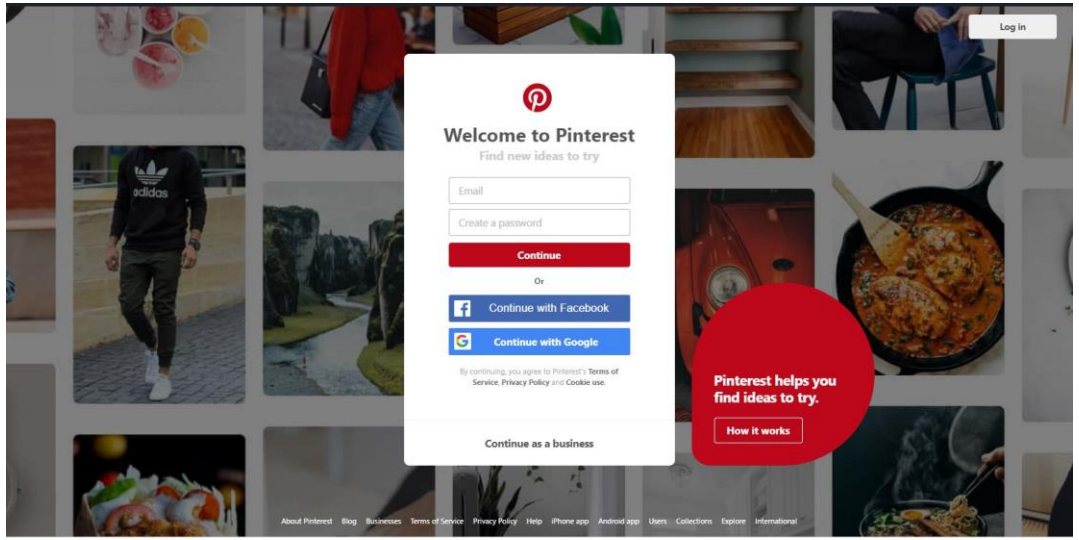
Drawbacks?

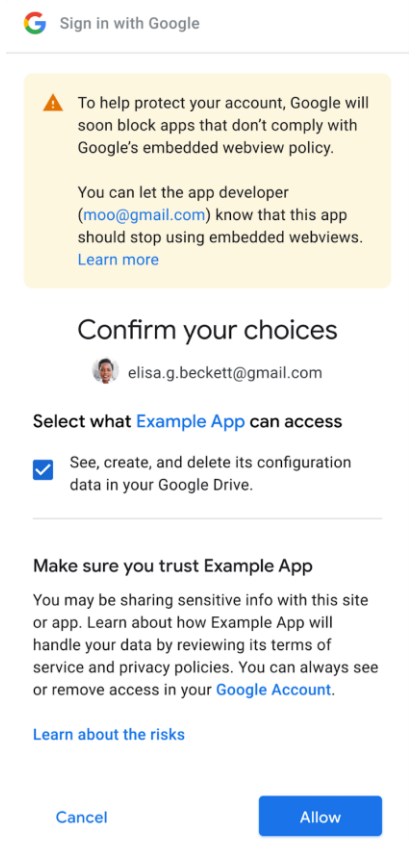
Federated Identity Limitations

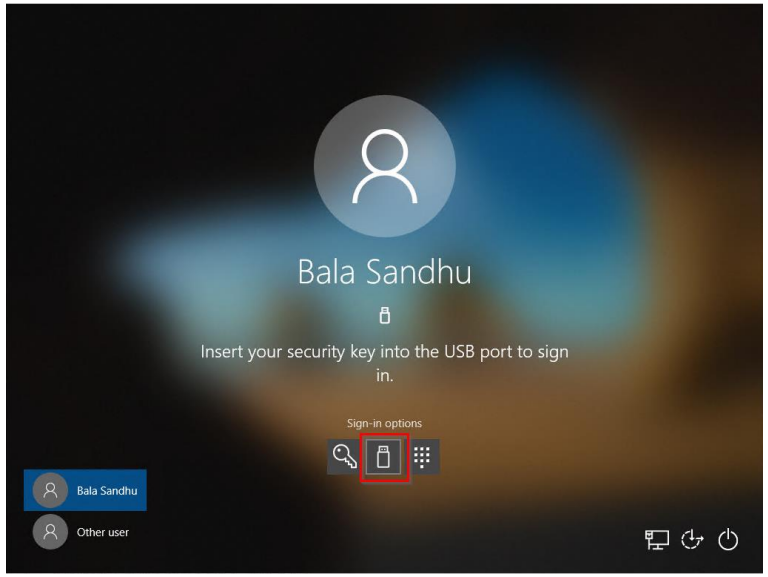
- Relies on **trusted relationships between different parties** to authenticate and authorize users
- The Identity Provider (IDP) knows when and where a user logs in, leading to privacy concerns
- Users and organizations may become dependent on a single IdP (e.g., Google, Microsoft) without an easy way to migrate

Enabling Technologies

Oauth, OpenID Connect, and FIDO 2 are key protocols to implement the federated identity model

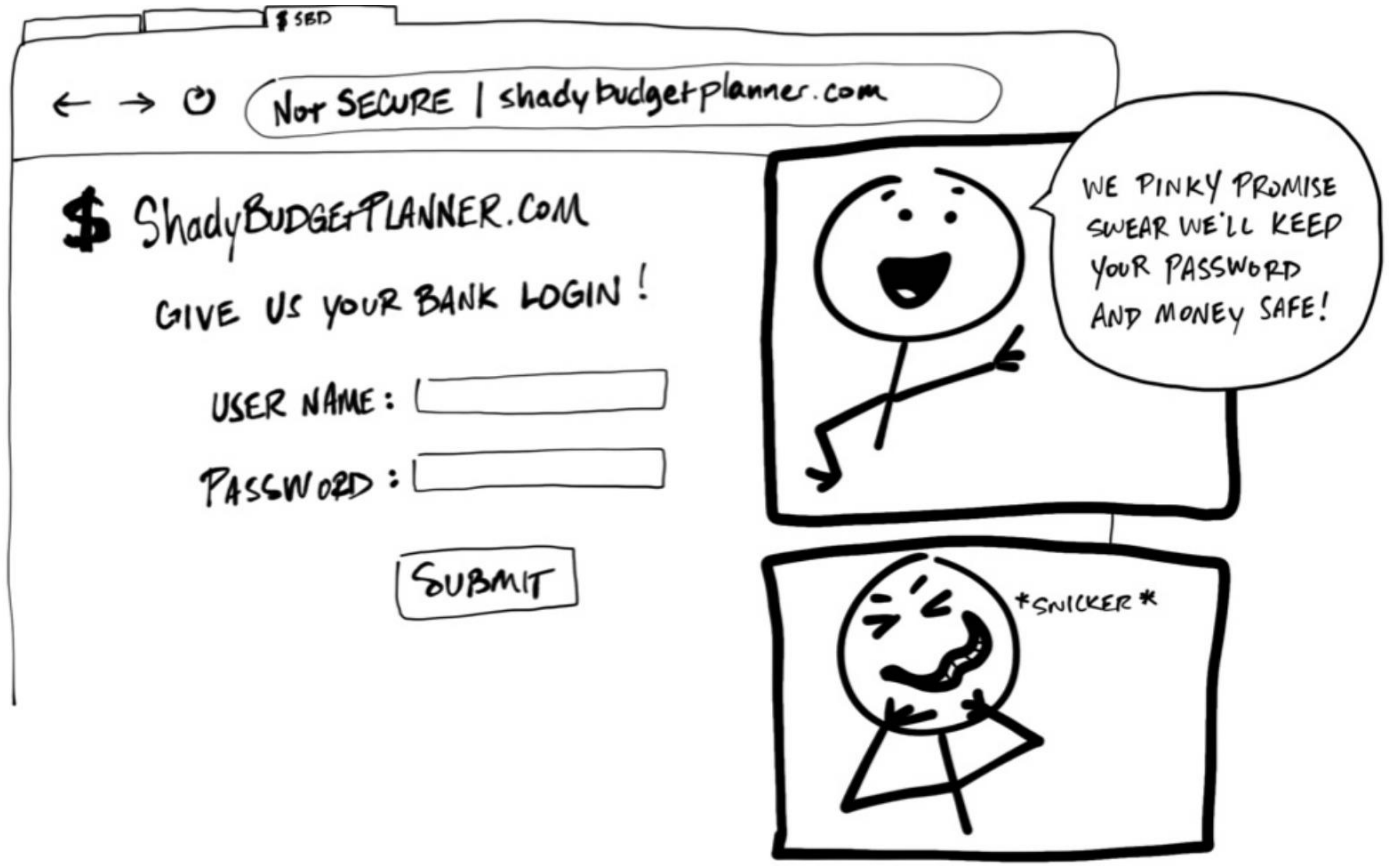






ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

“Stone Age” Data Sharing



Would you trust the service provider?

Many Challenges

- There's **no guarantee** that the organization will safely keep your credentials
- More information than necessary
- Need for **granular** access control

OAuth

- Allows one service to **access data managed by other applications**
- Applications should be **only provided with the information needed** to accomplish their task
- Decouples authentication from authorization

Let’s Make It Simple!

1. App requests authorization from a user
2. User authorizes App and delivers a proof
3. App presents a proof of authorization to server to get a token
4. Token is limited to only access what the user authorized

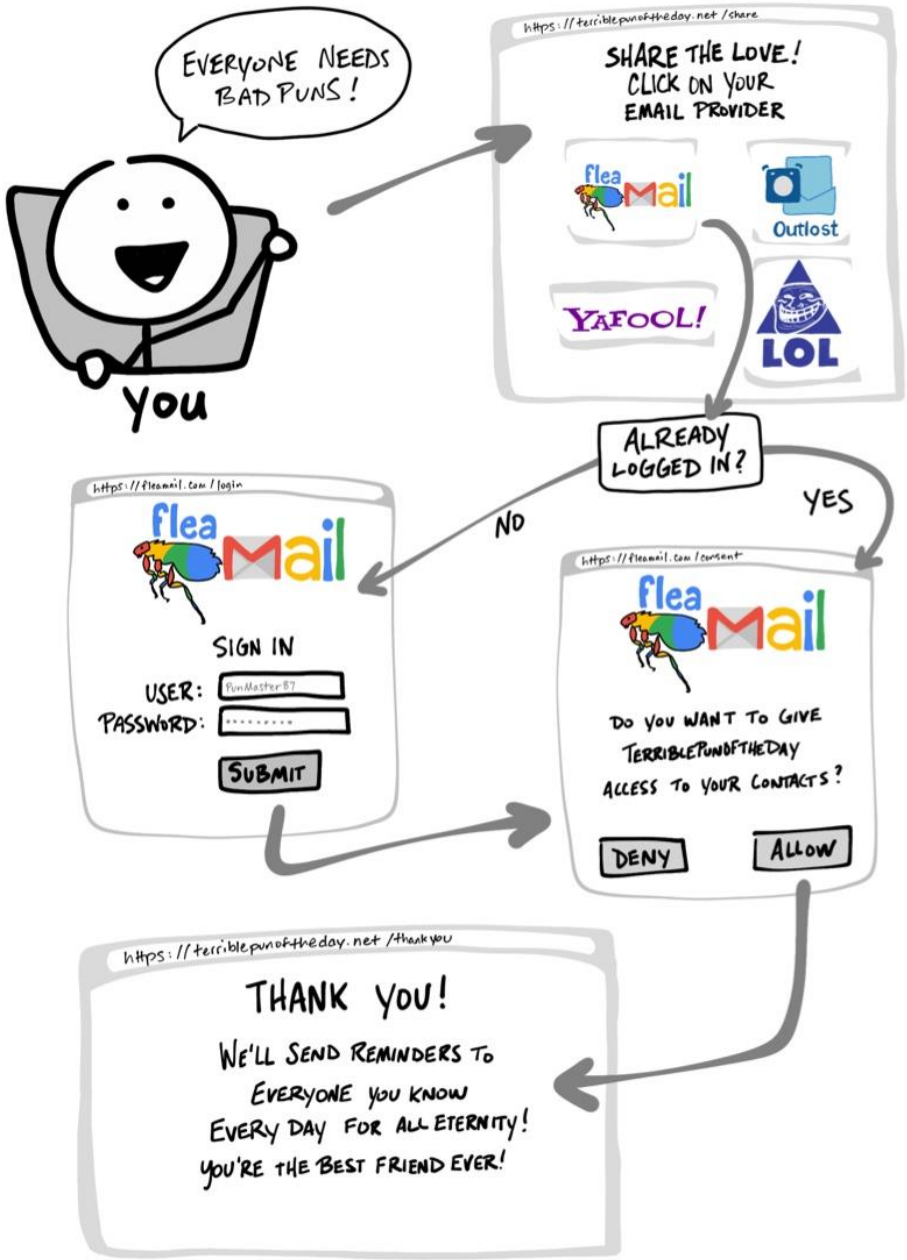
Example

Suppose you've discovered a website named "Terrible Pun of the Day" and create an account to have it send awful pun jokes as text messages every day

If you want to share this site with all your friends, instead of writing an email to every person in your contacts lists, you can use the feature provided by the application that invite your friends

You'll have to grant "Terrible Pun of the Day" access to your email contacts and send out emails for you!

OAuth Flow



Terminology

Resource Owner: the user who owns resources (name, surname email, etc.)

Client: the application that wants to access data or perform action on behalf of the Resource Owner

Authorization Server: the application where the Resource Owner has already an account

Resource Server: the Application Programming Interface (API) or service the Client wants to use

Terminology

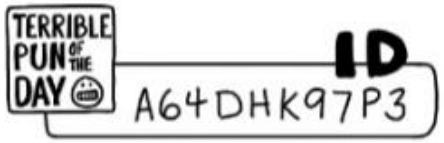
Redirect URI: the URL the Authorization Server will redirect the Resource Owner back to after granting permission to the Client

Response Type: the type of information the Client expects to receive. The most common is *code*, where an Authorization Code is returned

Scope: the granular permissions required by the Client

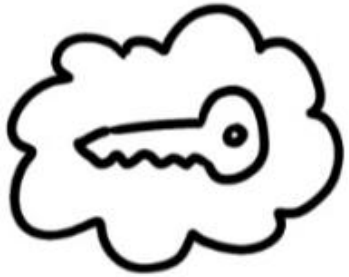
Consent: the Authorization Server takes the Scopes the Client is requesting and asks the Resource Owner to grant or deny the request

Terminology



Client ID: identify the Client with the Authorization Server

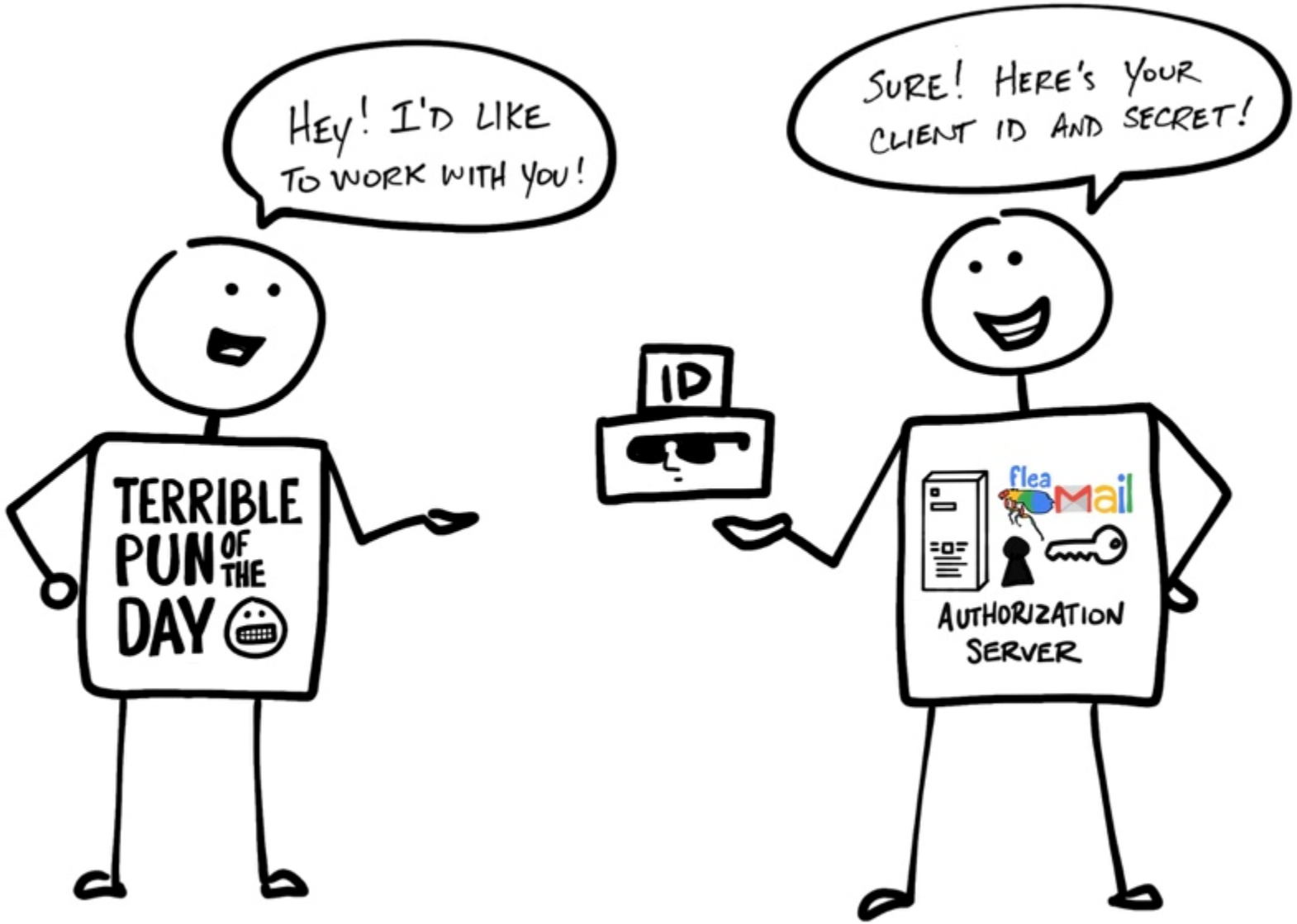
Client Secret: a secret password that only the Client and Authorization Server know



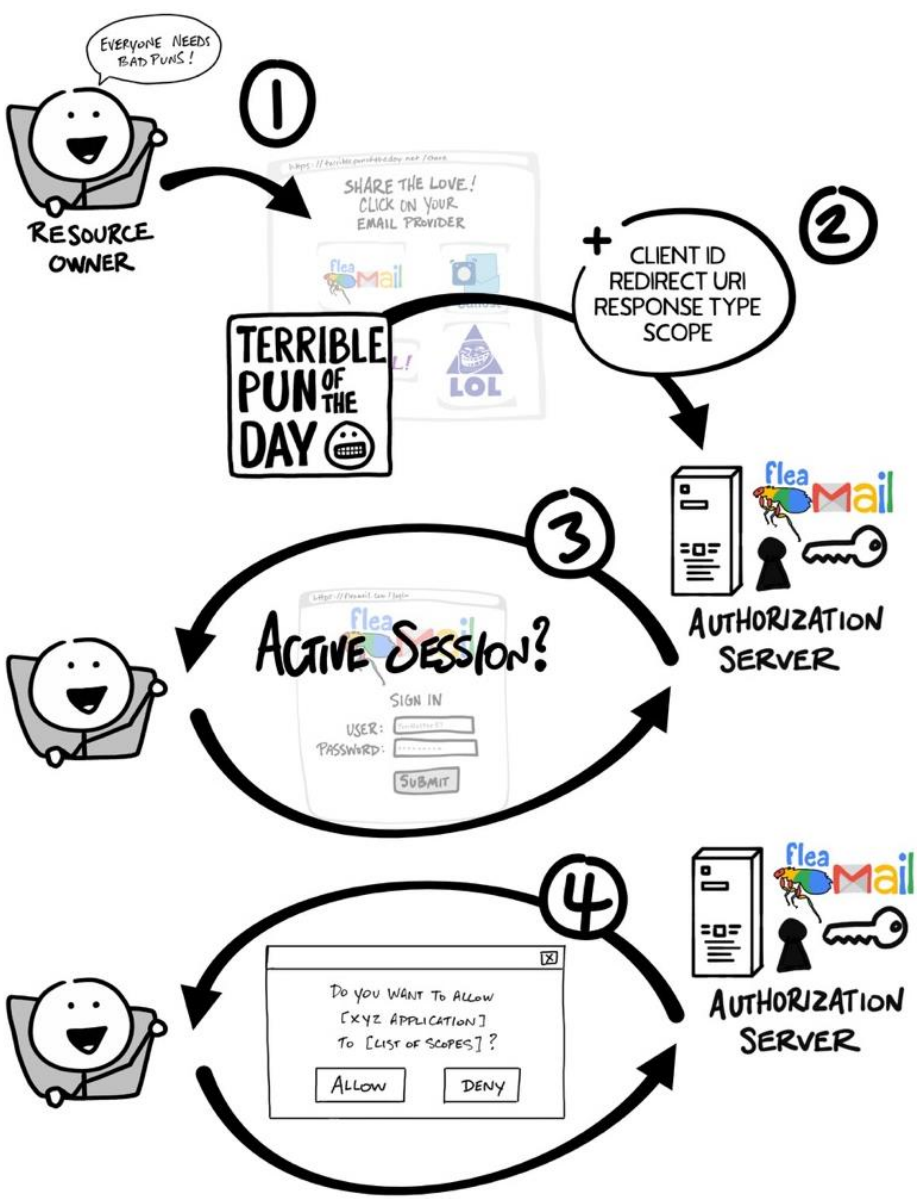
Authorization Code: a short-lived temporary code the Client gives the Authorization Server in turn of an Access Token

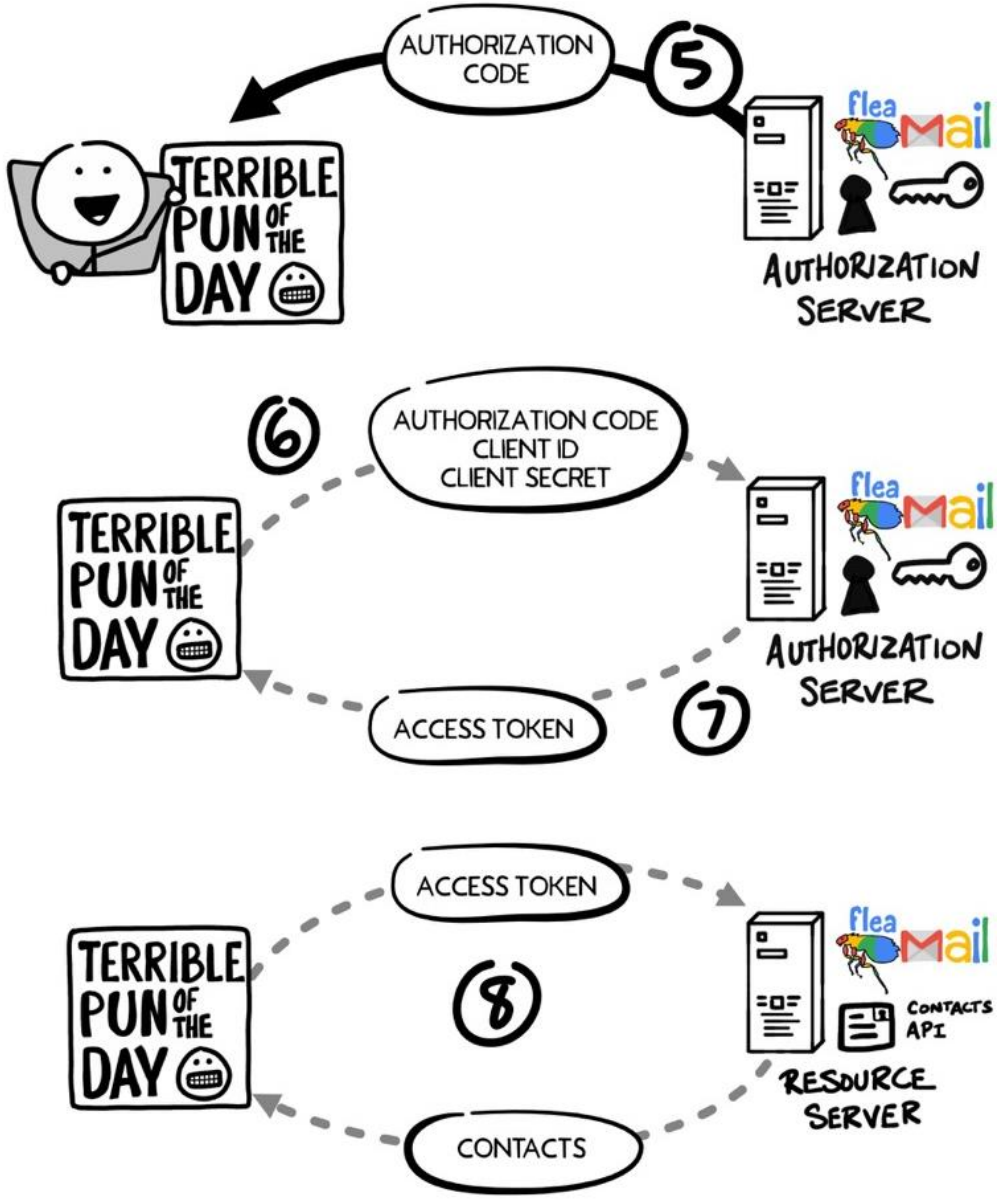
Access Token: the key used by the Client to communicate with the Resource Server

ClientID and Secret



OAuth Flow





Federated Identity

Federated identity refers to linking and using an electronic identity that the user has across several systems

An application **does not necessarily need to obtain in-store credentials** to authenticate users

It can use an identity management system that is already storing a user's electronic identity to authenticate the user as long as the application trusts that identity management system

OpenID Connect

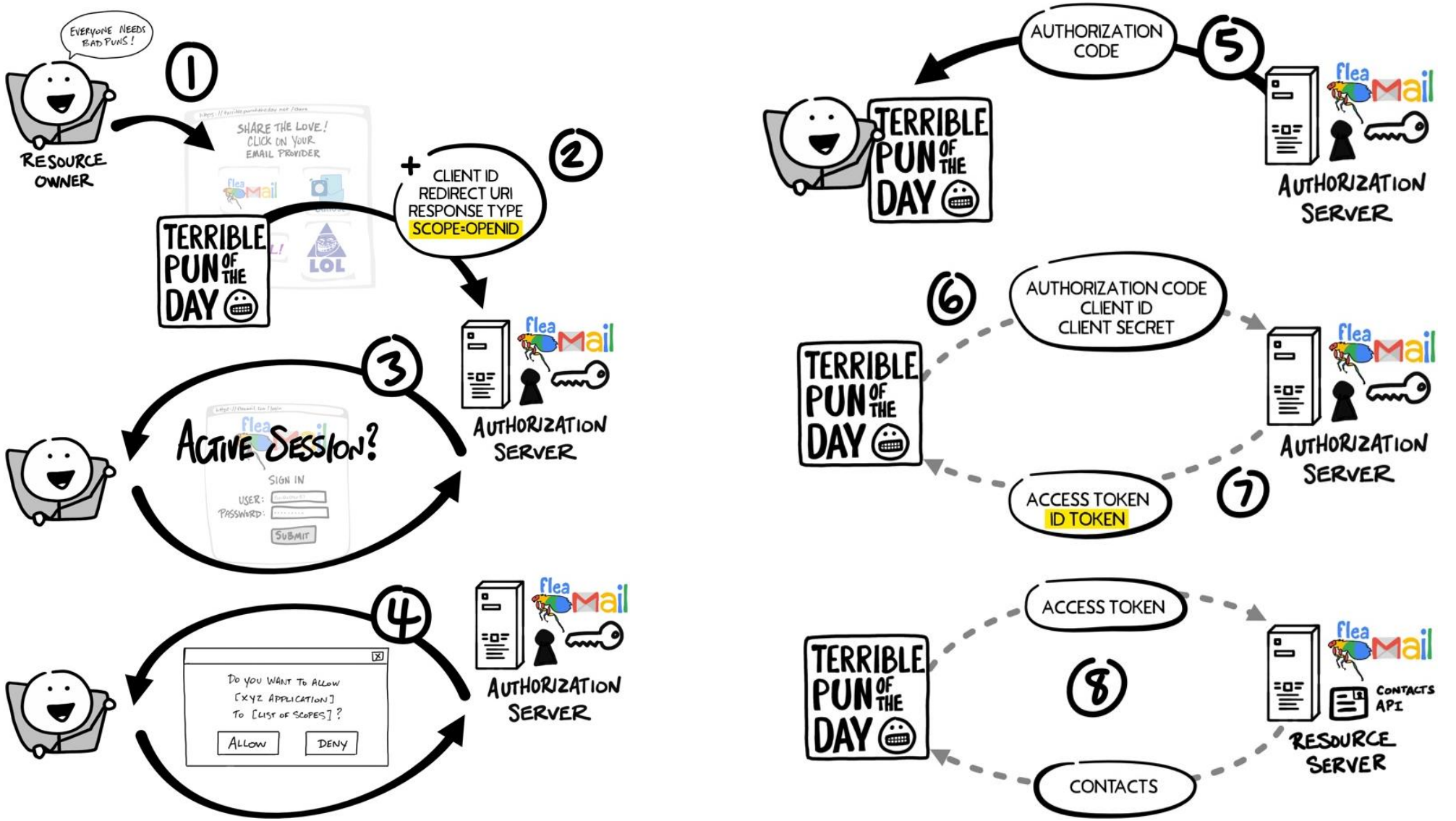
- OpenID Connect (OIDC) is a thin layer that sits on top of OAuth
- Adds login and profile information about the person who is logged in
- An Authorization Server that supports OIDC is often called Identity Provider
- Enables scenarios where one login can be used across multiple applications such as Single Sign On (SSO)

OIDC vs OAuth

1. OIDC uses `openid` as scope

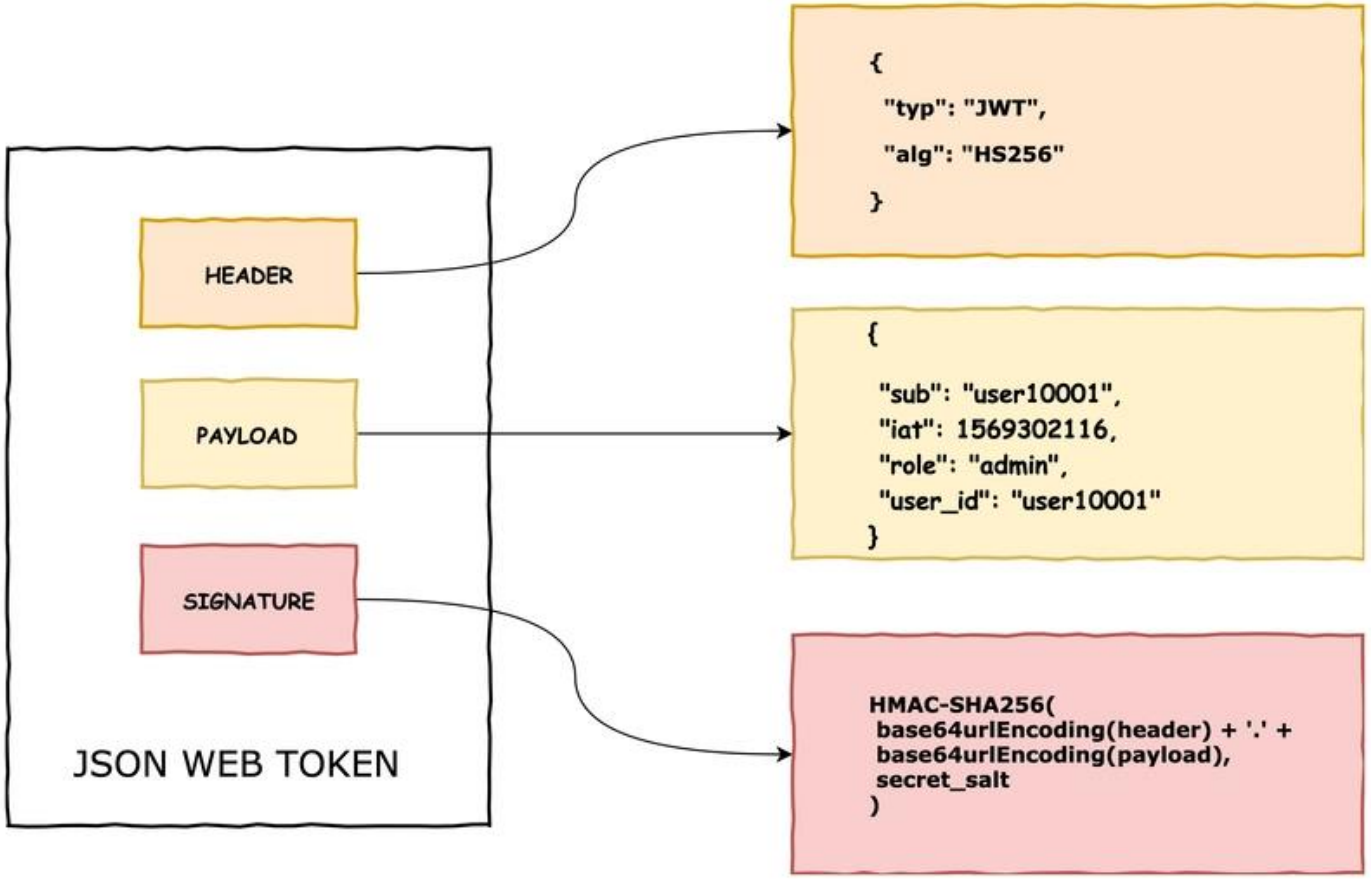
2. In the final exchange, the Client receives both an Access Token and an IDToken

OIDC Flow



ID Token

- The ID Token is a JSON Web Token (JWT)



Self-Sovereign Identity

Electronic Identification, Authentication, and Trust Services

In 2014, Europe introduce the Electronic Identification, Authentication, and Trust Services

- **Cross-border recognition** of national electronic IDs (eIDs) within EU member states
- Establishes a legal framework for **electronic signatures, seals, timestamps, and certificates**
- Facilitates secure online transactions and document authentication
- Primarily focuses on **government-to-citizen (G2C)** and **business-to-government (B2G)** interactions
- **Centralized and Federated Model**: eIDAS relies on member states issuing eIDs, meaning users do not fully control their digital identity

eIDAS 2.0

eIDAS 2.0 is an update to the original regulation, introduced in 2022, aiming to expand digital identity usage, enhance privacy, and introduce SSI elements

- European Digital Identity Wallet (EUDIW) **directly managed by users**
- While in eIDAS participation was optional, now every EU country **must provide** citizens access to the EUDIW
- More use cases and interoperability
- **Stronger privacy guarantees** → higher compliance with General Data Protection Regulation (GDPR)

European Digital Identity Framework

In May 2024, the EU Regulation 2024/1183 established a European Digital Identity Frame-work, aiming to enhance secure and user-controlled digital identity

A core component of this framework is the European Digital Identity Wallet (EUDIW), projected to drive a significant shift toward digital identity, with 500 million smartphone users expected to regularly use the EUDIW by 2026

Identità e portafoglio digitale: entrano in vigore le novità Ue. Cosa cambia per cittadini e aziende

di Alessandro Longo

Gartner Predicts At Least 500 Million Smartphone Users Will Be Using a Digital Identity Wallet by 2026

LONDON, U.K., September 24, 2024

The Shift Toward Portable Digital Identity Is Underway

Gartner, Inc. predicts that by 2026, at least 500 million smartphone users will be regularly making verifiable claims using a digital identity wallet (DIW).

Il 20 maggio scatta Eidas 2.0, la revisione del regolamento su "identificazione elettronica e servizi fiduciari per le transazioni elettroniche nel mercato interno" europeo. Una delle principali novità è il portafoglio digitale europeo. L'Italia ne prepara il lancio a gennaio, ma siamo al terzo rinvio

Self-Sovereign Identity

"Self-Sovereign Identity is the next step beyond user-centric identity and that means it begins at the same place: the user must be central to the administration of identity." – Christopher Allen

Individuals are granted **full control over their information**, having the power to decide when, if, and how they wish to disclose or modify their data

Mazzocca, C., Acar, A., Uluagac, S., Montanari, R., Bellavista, P., & Conti, M. (2025). A survey on decentralized identifiers and verifiable credentials. IEEE Communications Surveys & Tutorials.

SSI Features

- User control and ownership
- Decentralization
- Portability and interoperability
- Selective disclosure and minimal disclosure

Decentralized Identifier

DID is a digital identifier recently formalized and standardized by the World Wide Web Consortium (W3C)

A DID uniquely identifies a DID Subject (not only individuals) and comprises:

- Uniform Resource Identifier
- Specific identifier for a DID method
- Method-specific identifier for the DID

Key Components

Namespace: The namespace part of the DID (like ethr, sov, btcr, etc.) identifies the method

Method Specification: This is a document that outlines the rules and operations for DIDs that use this method. It specifies how to:

- Create a new DID
- Resolve a DID to its associated DID Document (which contains public keys and other information)
- Update a DID Document
- Deactivate a DID

Real Example

Let’s break down a concrete example using the ethr (Ethereum) method:

- Scheme: did
- Method: ethr
- Identifier: 0x123456789abcdef0abcd123456789abcdef0abcd

did:ethr:0x123456789abcdef0abcd123456789abcdef0abcd

DID Document

Each DID resolves to a DID Document, a machine-readable JSON-LD that contains various information about the DID Subject

A DID Document comprises **public keys**, service endpoints, authentication parameters, timestamps, and additional metadata

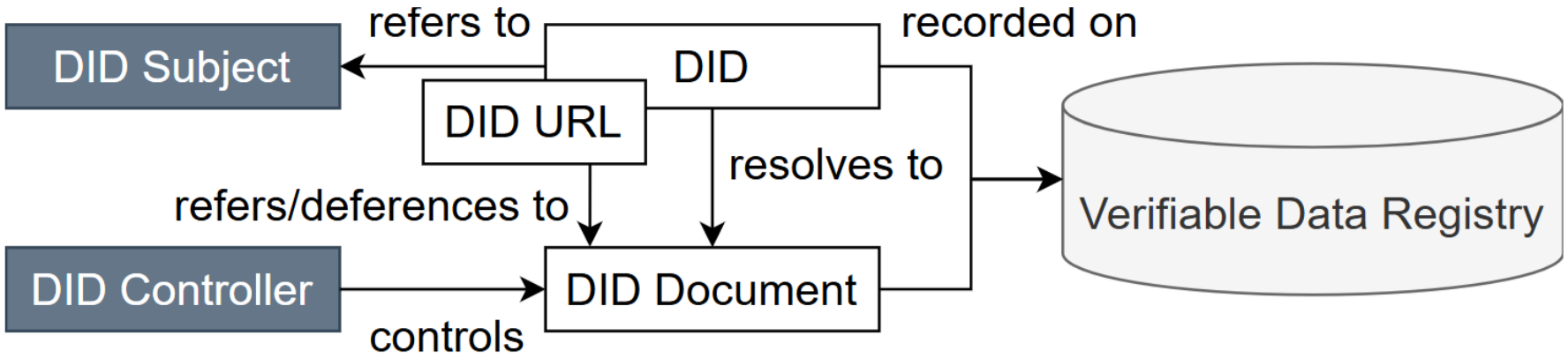
did:example:123456789abcdefghi

```
{
  "@context": [
   "https://www.w3.org/ns/did/v1",
   "https://w3id.org/security/suites/ed25519-2020/v1"
]
"id": "did:example:123456789abcdefghi",
"authentication": [{
  "id": "did:example:123456789abcdefghi#keys-1",
  "type": "Ed25519VerificationKey2020",
  "controller": "did:example:123456789abcdefghi",
  "publicKeyMultibase": "zH3C2AVvLMv6gmMNam3uVAjZp..."
  }]
}
```

DID-based Architecture

An entity can **prove the ownership** of a DID by leveraging the private key corresponding to public key included in the DID Document

To **verify the proof**, the verifier will access the DID Document shared through a Verifiable Data Registry (VDR)



Different Types

The DID specification introduces three types:

- **Anywise DIDs** can be utilized with an unspecified number of parties, typically strangers. They allow broad usage without limiting the number of relationships that can be established

- **Pairwise DIDs** are only known by their subject and one other party, such as a service provider. Pairwise DIDs address privacy concerns by ensuring that each relationship has a unique DID, minimizing the risk of correlation between different interactions

- **N-wise DIDs** are designed to be known by strictly N parties, including the subject. They encompass pairwise DIDs as a special case when N equals 2

Verifiable Credential

A Verifiable Credential (VC) is an interoperable data structure capable of representing claims and properties of an identity owner (DID Subject) in a **cryptographically verifiable** and temper-proof manner

Credentials are stored in a personal wallet, typically on the identity owner's device


```
{
  "@context": [
    "https://www.w3.org/2018/credentials/v1",
    "https://www.w3.org/2018/credentials/examples/v1"
  ],
  "id": "http://example.edu/credentials/1872",
  "type": ["VerifiableCredential", "AlumniCredential"],
  "issuer": "https://example.edu/issuers/565049",
  "issuanceDate": "2010-01-01T19:23:24Z",
    "credentialSubject": {
    "id": "did:example:ebfeb1f712ebc6f1c276e12ec21",
    "alumniOf": {
      "id": "did:example:c276e12ec21ebfeb1f712ebc6f1",
      "name": [{
        "value": "Example University",
        "lang": "en"
      }, {
        "value": "Esempio di Università",
        "lang": "it"
      }]
    }
  },
  "proof": {
    "type": "RsaSignature2018",
    "created": "2017-06-18T21:19:10Z",
    "proofPurpose": "assertionMethod",
    "verificationMethod":
"https://example.edu/issuers/565049#key-1",
    "jws":
"eyJhbGciOiJSUzI1NiIsImI2NCI6ZmFsc2UsImNyaXQiOlsiYjY0..."
  }
}
```

VC Structure

A VC also consist of several elements:

- URI of the Subject
- URI of the Issuer responsible for the claims
- URI that uniquely identifies the credential
- Cryptographic signatures
- Expiration conditions

The W3C has also defined the concept of Verifiable Presentation (VP), which specifies the methods for signing and presenting VCs by the holder

Comparison with X.509 Certificates

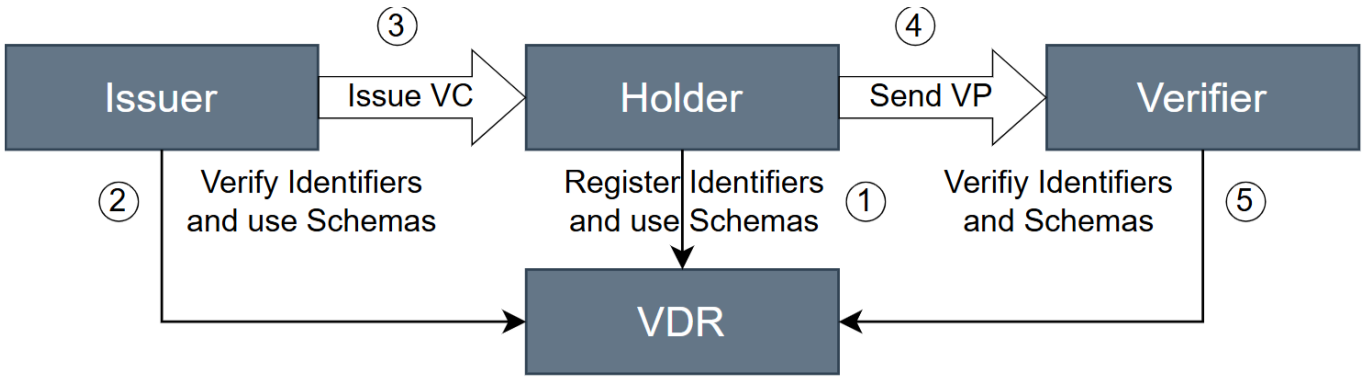
X.509 certificates have a **very specific format**, and the content has been agreed upon over time. It ties data to a public key

VC can contain anything from a simple statement to a complex structure, as long as this statement conforms to a certain formats (JSON-LD or JSON-JWT)

The only thing that remains is the condition that a verifier must be able to interpret the content of a statement

Main Actors

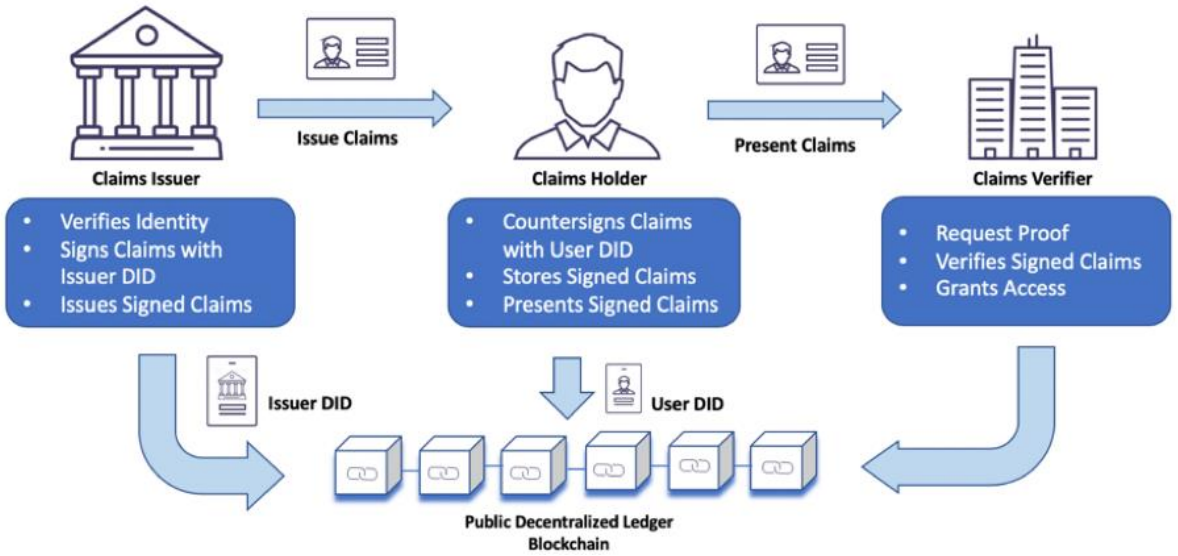
- The **holder** refers to the identity owner that controls one or more VC
- The **issuer** is a trusted organization (e.g., governments or municipalities) responsible for creating and issuing VCs
- The **verifier** receives one or more VCs, verify them, and grant access to services accordingly



Verifiable Data Registry

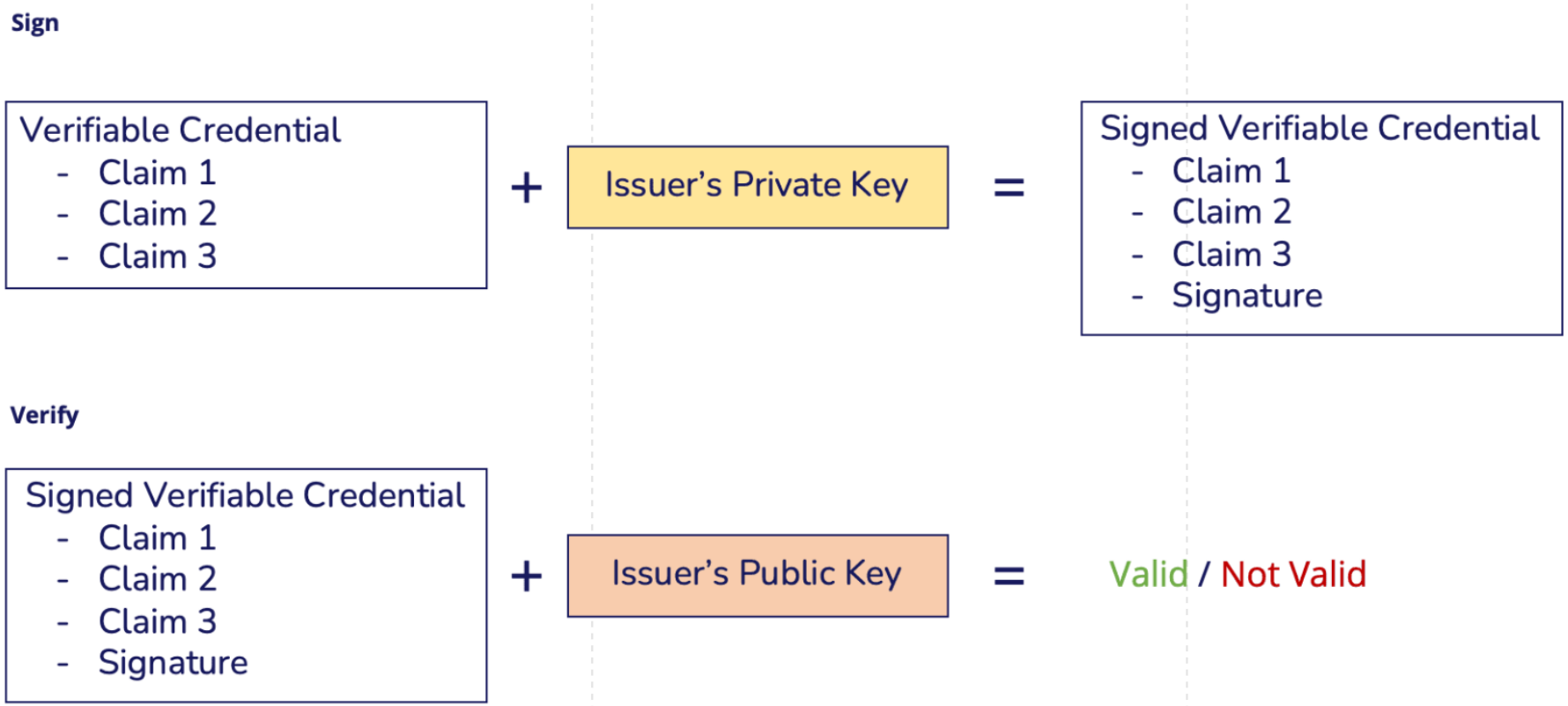
Verifiable Data Registry (VDR) facilitates the creation, management, and verification of identifiers, keys, credentials, schemas

The W3C's standards lack explicit specifications about implementing VDR, most of the proposed approaches are based on DLTS such as blockchain



VC Signature Scheme

VC can be automatically verified by entities that wish to confirm that you are whom you claim to be online



Selective Disclosure

Why should a verifier access to all the claims in a VC if it only requires one?

No information should be disclosed beyond what is necessary to execute a transaction

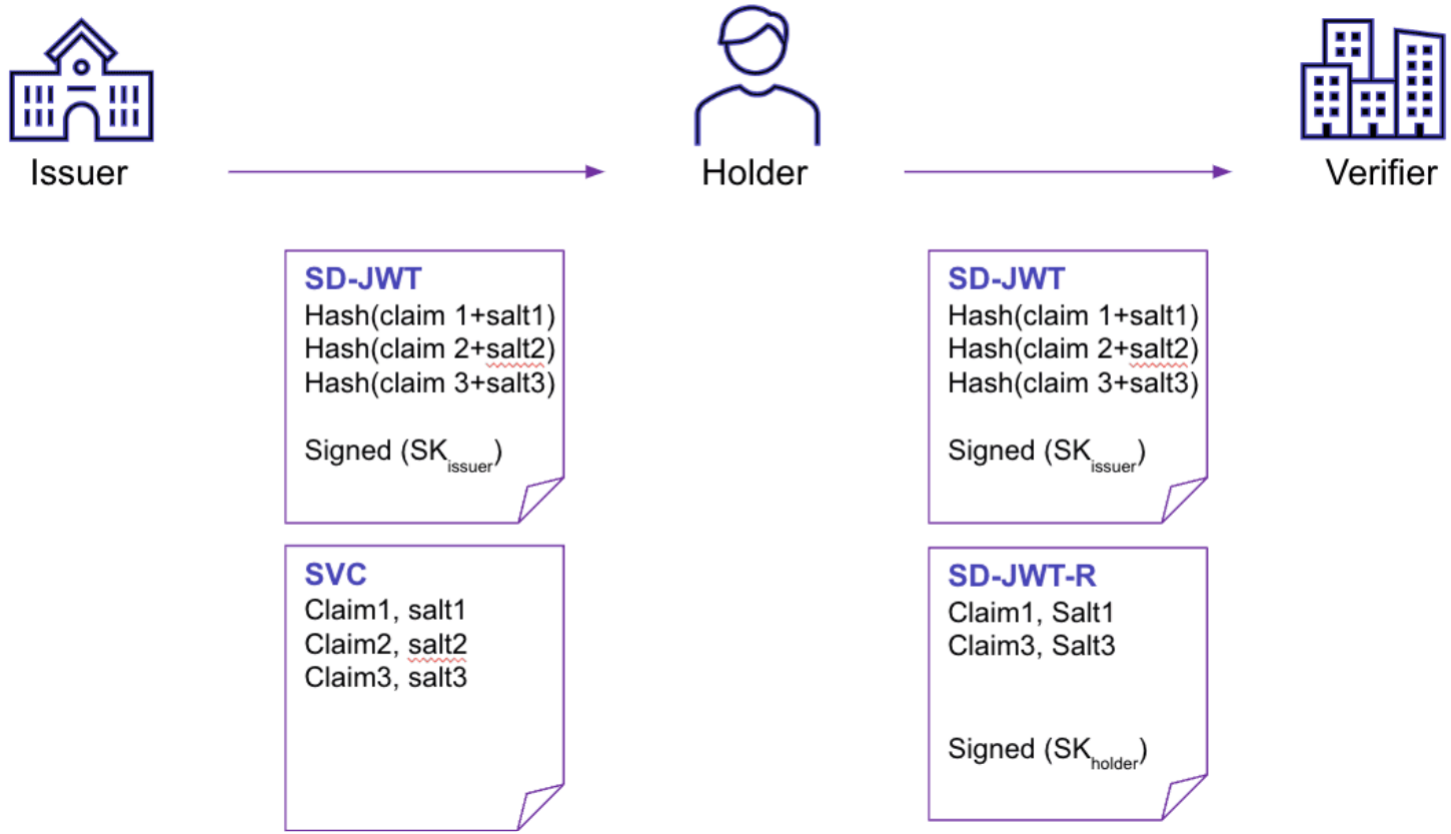
Do I need to disclose my address to buy alcoholic?

Security and Privacy Properties of Selective Disclosure

- **Minimization**: revealing only the minimum amount of information, including metadata
- **Predicate disclosure**: prove that a claim satisfies a specific condition without revealing the actual value
- **Unlinkability**: given two presentation from the same users, the verifier cannot determine whether they are originating from the same credential
- **Unobservability**: the issuer cannot determine when, where, and to which a credential is presented
- **Untraceability**: colluding issuers and verifiers cannot trace a user's credential across multiple interactions
- **Non-transferability**: credentials cannot be reused or delegated to another party
- **Unforgeability**: a holder cannot forge a fraudulent credential that have not been legitimately issued

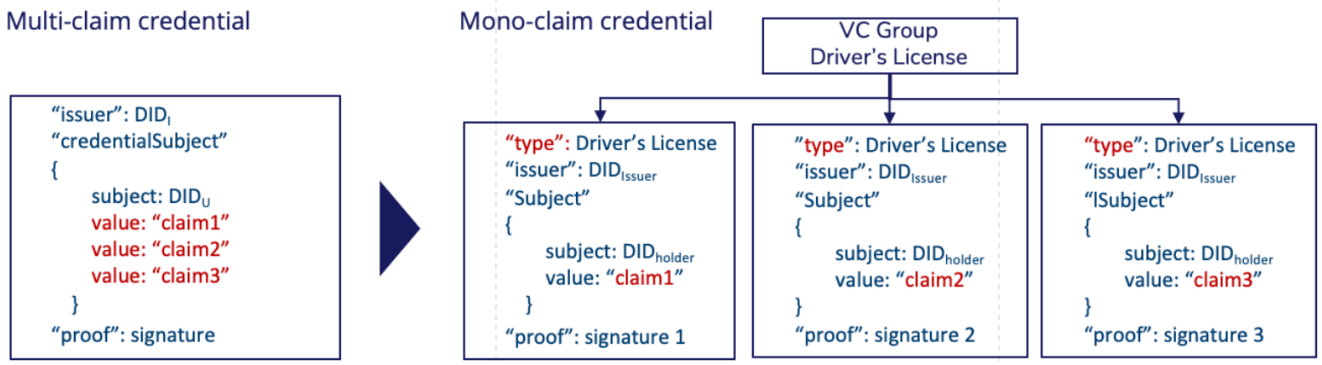
Selective Disclosure for JWT

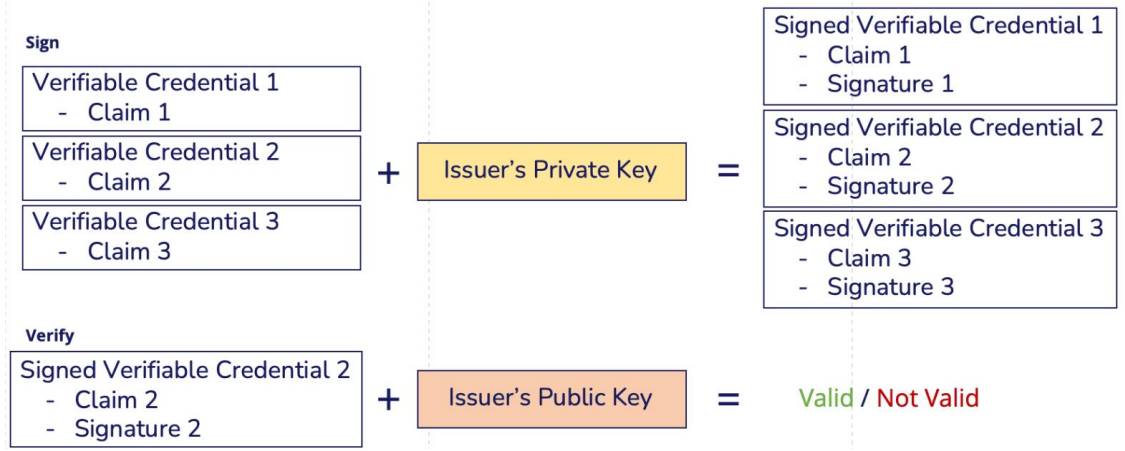
Selective Disclosure for JWT (SD-JWT) is a mechanism that allows a user to selectively disclose the contents of a JWT to a service provider (it is a building block in EUDI Wallet)



Selective Disclosure via Mono-claim Credentials

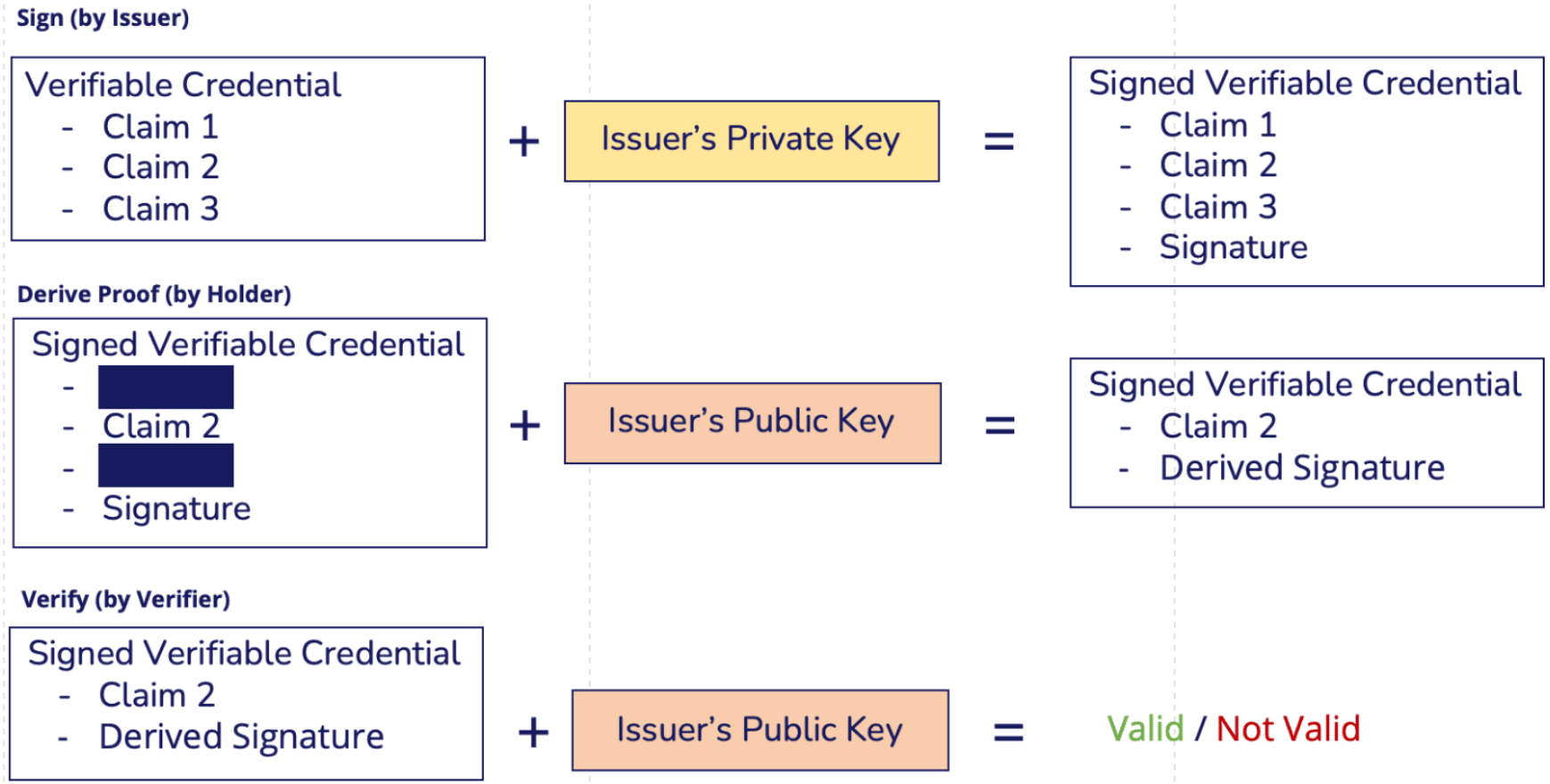
We can divide a multi-claim credential into a subset of mono-claim credentials





Selective Disclosure via BBS+

BBS+ signatures allow a VC holder to derive proofs from the original signature



Threat Model

1. An adversary may perform various attacks, such as phishing, malware, or direct key theft, to compromise the private key associated with a DID or gain unauthorized access to credentials, impersonating the legitimate identity owner

2. A malicious user may collude with others or steal a valid credential from a legitimate user, presenting it to a verifier and acting on behalf of the holder

3. An attacker may attempt to gain access to a service by forging a legitimate VC or presenting a credential issued by an entity that lacks the authority to issue VCs

4. Although always verifiable, the validity of VCs can change over time due to loss of privileges or expiration. Consequently, a verifier must verify the validity of a credential before accepting it

Threat Model

5. A VC may contain more claims than necessary for accessing a specific service. Consequently, a service provider might acquire more information than required. Malicious service providers could exploit personal data for financial gain through individual profiling, while honest service providers might inadvertently elevate privacy loss risks

6. Selective disclosure techniques could still allow service providers to link claims to the same individual across subsequent presentations or collude with other providers

7. An adversary may intercept a VP and reuse it to a different verifier, impersonating the legitimate holder and gaining unauthorized access

Available Implementations

There are some available implementation some are more general, while others are tailored for specific domains

Library	DIDKit	IOTA Identity Framework	Hyperledger Aries	Microsoft Entra Wallet	Veramo
Main Target Platform	Multi-platforms	Multi-platforms/IoT Devices	Multi-platforms	Mobile Applications	Multi-platforms
Programming Language	Rust, C, Java, Android, Python, JavaScript	Rust, Node.js	Python, JavaScript, Go, .NET	Android, iOS	JavaScript
W3C Compliance	✓	✓	✓	✓	✓
Credential Format	JSON-LD/JWT	JWT	JSON-LD	JWT	JSON-LD/JWT
Key & Wallet Management	-	Stronghold	-	Azure Key Vault	-
Verifiable Proof Types	RSA/EdDSA/ECDSA/ EIP712/JWS2020	EdDSA/ECDSA	BBS+/EdDSA	EdDSA/ES256K/ECDSA P-256	EdDSA, ECDH, ECDSA
Selective Disclosure	SD-JWT	SD-JWT/ZKSD	SD-JWT	SD-JWT	SD-JWT Through Plugin Interface
Verifiable Data Registry	-	Tangle	Indy Ledger	-	-
Learning Curve	↓	≈	↑	≈	↓
Open Source	✓	✓	✓	✓	✓

GDPR Compliance

- **Right to Be Informed**: Data sharing is fully under the control of the user, who decides when and with whom to share their credentials

- **Right to Rectification**: Identity owner can request the issuance of the credential with the updated information

- **Right to Be Forgotten**: User can revoke a VC rendering it unusable. Moreover, the use of selective disclosure minimize the amount of information shared

European Blockchain Services Infrastructure

The European blockchain services infrastructure (EBSI) consists of a peer-to-peer network of interconnected nodes running a blockchain-based services infrastructure

Each member of the European Blockchain Partnership (EBP) – the 27 EU countries, Norway, Liechtenstein and the European Commission – will run at least one node. The infrastructure is made up of different layers including:

- A base layer containing the basic infrastructure, connectivity, the blockchain and necessary storage
- A core services layer that will enable all EBSI-based use cases and applications
- Additional layers dedicated to use cases and specific applications

European Blockchain Services Infrastructure

We cannot deploy custom smart contracts as it is a government-backed, permissioned network

It supports predefined smart contracts for specific use cases

- VC issuance and verification
- Diploma certification
- Cross-border business registration

Use Cases of EUDIW

ACCESSING GOVERNMENT SERVICES

Save time accessing digital public services

Access digital public services (nationally and across borders) by using your wallet to securely identify and authenticate yourself.

PILOTED AND TESTED BY

Potential
For European Digital Identity

Visit Potential ›

MOBILE DRIVING LICENSE

Request a digital version of your driving licence

Download, store, and share your mobile driving licence; eliminating the need for a physical copy of it.

PILOTED AND TESTED BY

Potential
For European Digital Identity

Visit Potential ›

PAYMENTS

Make your online transactions easier

Your wallet offers a single way to identify yourself to all your bank accounts. Easily authorise payments through your wallet.

PILOTED AND TESTED BY

NOBID CONSORTIUM

Visit Nobid ›

EWC

Visit EWC ›

EDUCATION CERTIFICATION

Never lose the diploma you worked so hard for again

Easily store and share all your education credentials when applying to a new job or university.

PILOTED AND TESTED BY

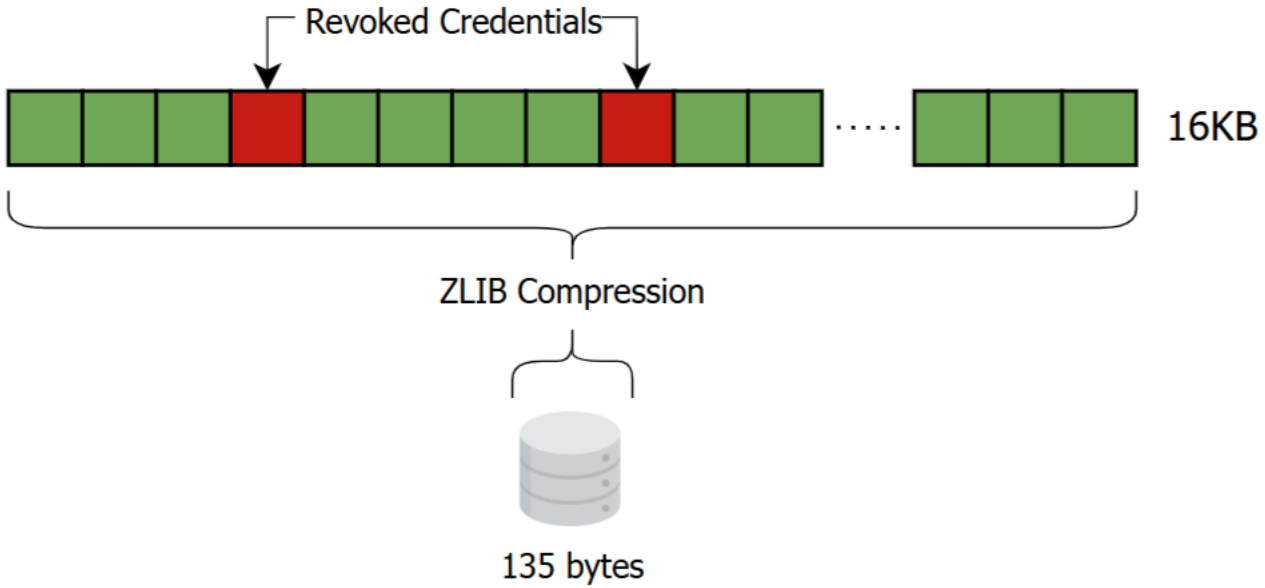
DC4EU

Visit DC4EU ›

Credential Revocation

We may need to revoke credential over time. There is only one W3C specification *Revocation List 2020*

Each credential includes an index, which refers to a position within the bitstring. We can use it to assess credential validity



References

- Mazzocca, C., Acar, A., Uluagac, S., Montanari, R., Bellavista, P., & Conti, M. (2025). A survey on decentralized identifiers and verifiable credentials. IEEE Communications Surveys & Tutorials. https://ieeexplore.ieee.org/abstract/document/10891701

- David Neal. “An Illustrated Guide to OAuth and OpenID Connect”. https://developer.okta.com/blog/2019/10/21/illustrated-guide-to-oauth-and-oidc

- Gataca, “SSI Essentials: Zero Knowledge Proof (ZKP) and Selective Disclosure, till death do us part?”. https://gataca.io/blog/ssi-essentials-which-selective-disclosure-protocol-will-succeed/

- Mazzocca, C., Acar, A., Uluagac, S., & Montanari, R. (2024). {EVOKE}: Efficient Revocation of Verifiable Credentials in {IoT} Networks. In 33rd USENIX Security Symposium (USENIX Security 24) (pp. 1279-1295). https://www.usenix.org/conference/usenixsecurity24/presentation/mazzocca