Understanding DID Auth

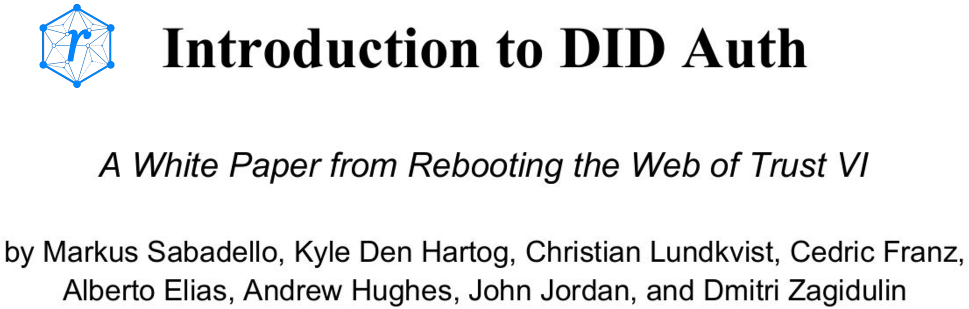
Markus Sabadello
Danube Tech, DIF, Sovrin,
W3C CCG, W3C VCWG, OASIS XDI TC

https://danubetech.com/

W3C Workshop on Strong Authentication & Identification
Redmond, WA, USA – 10th December 2018

DID Auth

- Background
  - Decentralized Identifiers (DIDs)
  - Decentralized Public Key Infrastructure (DPKI)
- DID Auth
  - Prove that the DID subject controls its DID
  - A concept, with different architectures and implementations



Introduction to DID Auth

A White Paper from Rebooting the Web of Trust VI

by Markus Sabadello, Kyle Den Hartog, Christian Lundkvist, Cedric Franz,
Alberto Elias, Andrew Hughes, John Jordan, and Dmitri Zagidulin

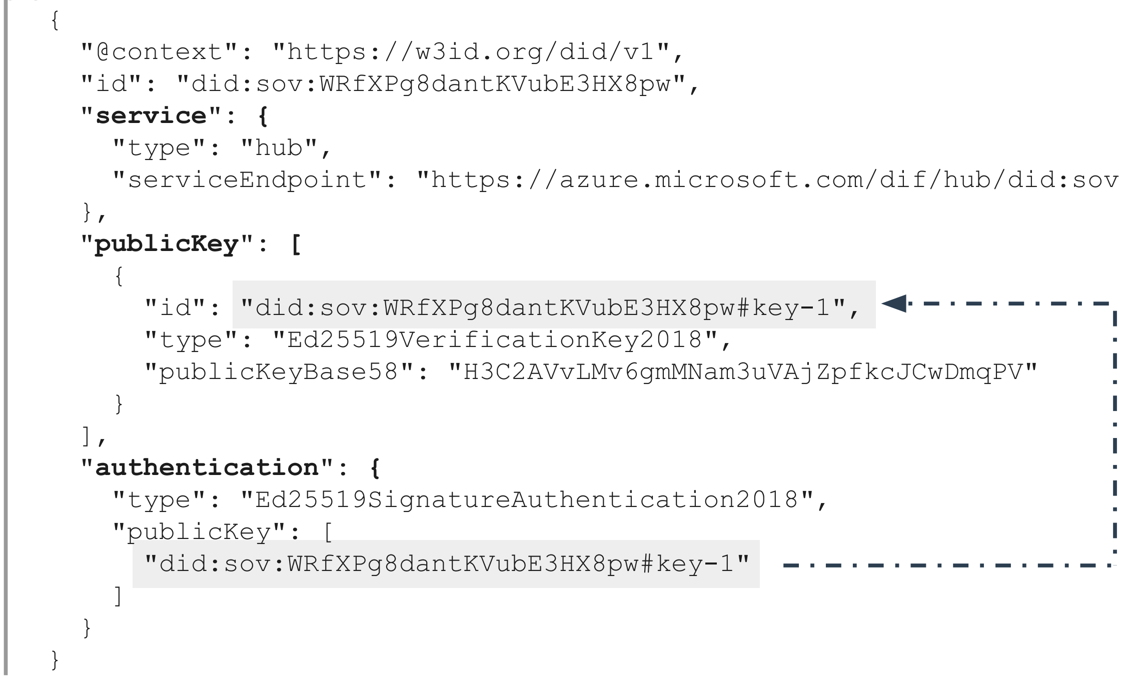
DID Document

- DID Document tells us how control of the DID can be proven
- DID Document contains service endpoints, public keys, authentication methods

```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:sov:WRfXPg8dantKVubE3HX8pw",
  "service": {
    "type": "hub",
    "serviceEndpoint": "https://azure.microsoft.com/dif/hub/did:sov:WRfXPg8dantKVubE3H"
  },
  "publicKey": [
    {
      "id": "did:sov:WRfXPg8dantKVubE3HX8pw#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAjZpfkcJCwDmqPV"
    }
  ],
  "authentication": {
    "type": "Ed25519SignatureAuthentication2018",
    "publicKey": [
      "did:sov:WRfXPg8dantKVubE3HX8pw#key-1"
    ]
  }
}
```

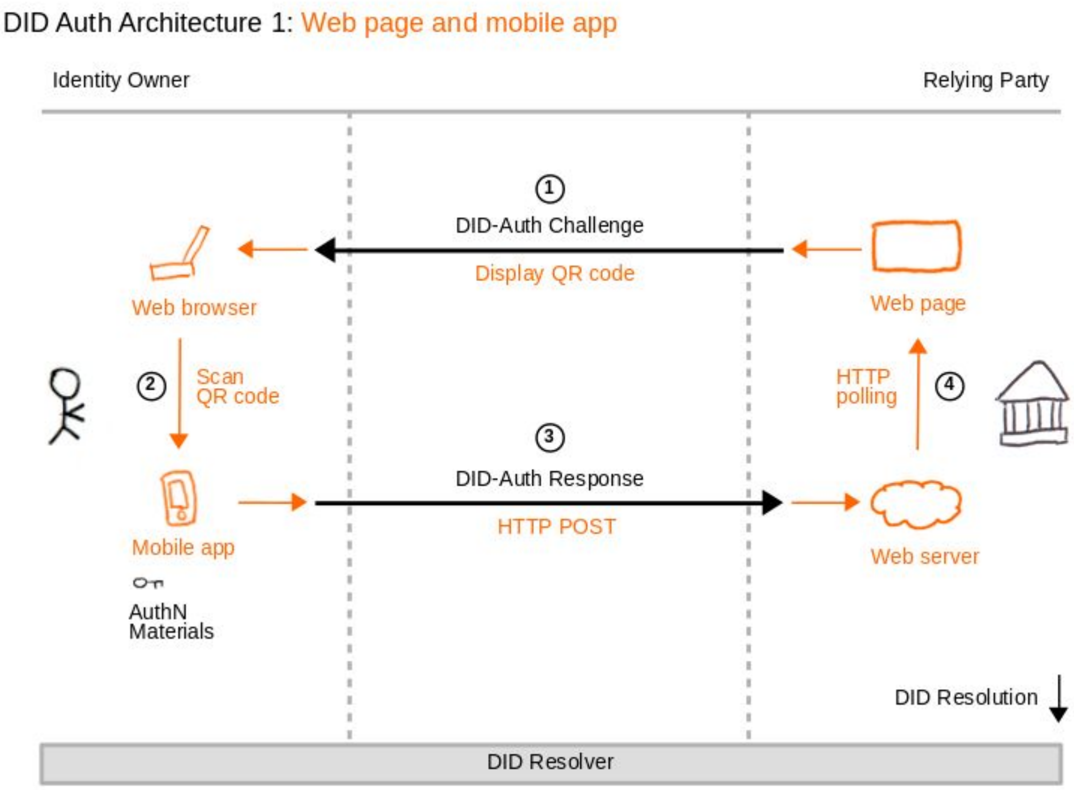
DID Document

- DID Document tells us how control of the DID can be proven
- DID Document contains service endpoints, public keys, authentication methods



```
{
  "@context": "https://w3id.org/did/v1",
  "id": "did:sov:WRfXPg8dantKVubE3HX8pw",
  "service": {
    "type": "hub",
    "serviceEndpoint": "https://azure.microsoft.com/dif/hub/did:sov:WRfXPg8dantKVubE3H"
  },
  "publicKey": [
    {
      "id": "did:sov:WRfXPg8dantKVubE3HX8pw#key-1",
      "type": "Ed25519VerificationKey2018",
      "publicKeyBase58": "H3C2AVvLMv6gmMNam3uVAjZpfkcJCwDmqPV"
    }
  ],
  "authentication": {
    "type": "Ed25519SignatureAuthentication2018",
    "publicKey": [
      "did:sov:WRfXPg8dantKVubE3HX8pw#key-1"
    ]
  }
}
```

DANUBE TECH GMBH



DID Auth Example Architecture

Challenges, Responses, Transports

- Challenge-response flow to prove that the DID subject controls its DID.

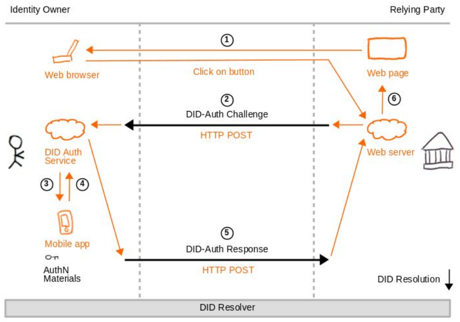
- **Challenge:**
  - Identity owner's DID may or not be known.
  - May or may not contain proof of control of a DID of the relying party.

- **Response:**
  - Linked to a challenge (e.g. using a nonce).
  - Contains proof of control of a DID of the identity owner.

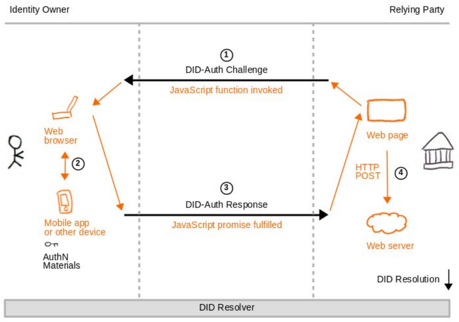
- **Transports:** HTTP POST, QR code, Mobile deep link, JavaScript browser API, Bluetooth, NFC, etc.
- Transports may require additional information such as endpoint URIs that may be included in the challenge, or discoverable from a DID.

DID Auth Example Architectures

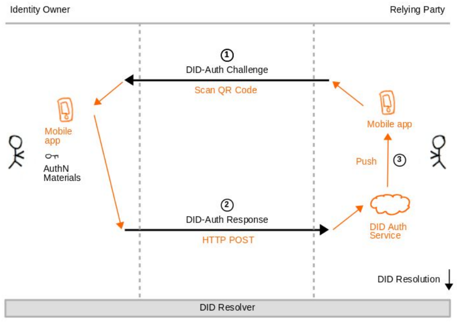
DID Auth Architecture 4: Web page and DID Auth service (2)



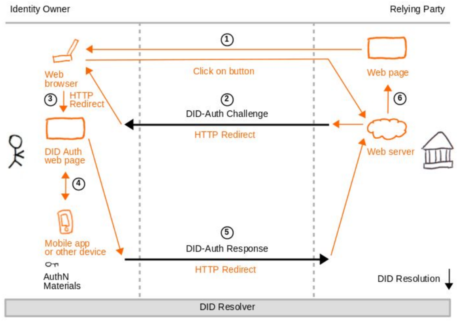
DID Auth Architecture 6: Web page and web browser



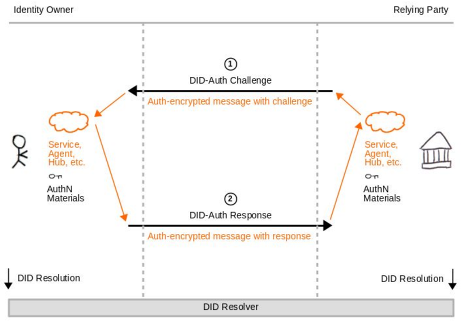
DID Auth Architecture 7: Mobile apps and DID Auth service



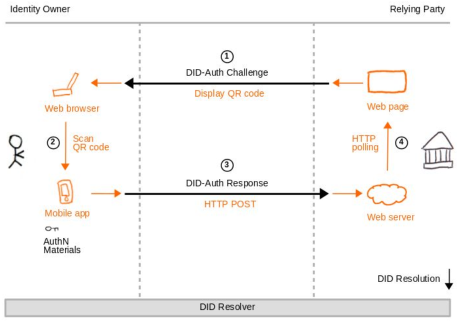
DID Auth Architecture 5: Web page and DID Auth web page



DID Auth Architecture 10: Authenticated Encryption



DID Auth Architecture 1: Web page and mobile app



DID Auth Data Formats

- Example JWT:

```
{
        "header": {
                "typ": "JWT",
                "alg": "ES256"
        },
        "payload": {
                "iss": "did:example:123456789abcdefg",
                "sub": "did:example:123456789abcdefg",
                "iat": 1479850830,
                "exp": 1511305200,
        },
        "signature": "..."
}
```

- Example JSON-LD VC:

```
{
        "type": ["Credential"],
        "issuer": "did:example:123456789abcdefg",
        "issued": "2018-03-07",
        "credentialSubject": {
                "id": "did:example:123456789abcdefg",
                "publicKey": "did:example:123456789abcdefg#keys-2"
        },
        "proof": {
                "type": "Ed25519Signature2018",
                "created": "2018-01-01T21:19:10Z",
                "creator": "did:example:123456789abcdefg#keys-2",
                "nonce": "c0ae1c8e-c7e7-469f-b252-86e6a0e7387e",
                "signatureValue": "..."
        }
}
```

DANUBE
TECH GMBH

DID Auth Data Formats

- ## Example JWT:

```
{
        "header": {
                "typ": "JWT",
                "alg": "ES256"
        },
        "payload": {
                "iss": "did:example:123456789abcdefg",
                "sub": "did:example:123456789abcdefg",
                "iat": 1479850830,
                "exp": 1511305200,
        },
        "signature": "..."
}
```

- ## Example JSON-LD VC:

```
{
        "type": ["Credential"],
        "issuer": "did:example:123456789abcdefg",
        "issued": "2018-03-07",
        "credentialSubject": {
                "id": "did:example:123456789abcdefg",
                "publicKey": "did:example:123456789abcdefg#keys-2"
        },
        "proof": {
                "type": "Ed25519Signature2018",
                "created": "2018-01-01T21:19:10Z",
                "creator": "did:example:123456789abcdefg#keys-2",
                "nonce": "c0ae1c8e-c7e7-469f-b252-86e6a0e7387e",
                "signatureValue": "..."
        }
}
```

Relation to OIDC, WebAuthn

OIDC + DID

- Self-Issued OpenID Provider
- Discover OIDC endpoint from DID

```
{
   "@context": "https://w3id.org/did/v1",
   "id": "did:example:123456789abcdefg",
   "service": [{
      "id": "did:example:123456789abcdefg;openid",
      "type": "OpenIdConnectVersion1.0Service",
      "serviceEndpoint": "https://openid.example.com/"
   }]
}
```

WebAuthn + DID

- Registration

  `Register(Account, Origin)`

- Registration Response (without DID)

  `RegisterResponse(`**`PublicKeyCredential`**`, Attestation, Origin)`

- Registration Response (with DID)

  `RegisterResponse(`**`DIDCredential`**`, Attestation, Origin)`

And more! DID-TLS, DID-HTTP-Signatures, DID-PGP, DID-SSH

For the Workshop

- Come up with a list of core DID Auth principles
  1) The identifier that is being authenticated is a DID.
  2) All elements of the DID Document can change, the DID stays the same.
  3) DID Resolution is performed to discover how to authenticate the DID.
  4) ... more?

- **Workshop Question #1:** Relation to OIDC, FIDO, WebAuthn?
- **Workshop Question #2:** Relation to VC exchange protocols?

Community Resources

- **W3C Credentials Community Group**
  https://www.w3.org/community/credentials/
- **Decentralized Identity Foundation**
  http://identity.foundation/
- **Rebooting-the-Web-of-Trust**
  http://www.weboftrust.info/
- **Internet Identity Workshop**
  http://internetidentityworkshop.com/

Thank You

- **Markus Sabadello**
- Danube Tech
  https://danubetech.com/
- markus@danubetech.com

The Internet Identity Workshop

W3C Credentials Community Group

Rebooting the Web of Trust

DANUBE
TECH GMBH

Backup Slides

Verifiable Credentials

DKMS, DID Auth

Hubs, Agents, XDI

Yadis, XRI, XRD, XRDS,
JRD, Webfinger

W3C Web Payments CG

OASIS XDI TC

DIDs: W3C Credentials CG
v0.11 Draft Community Report

DIDs: W3C DID WG
Charter now being written

Rebooting-the-Web-of-Trust
Internet Identity Workshop

DID registered
prov. URI scheme

DID method specs

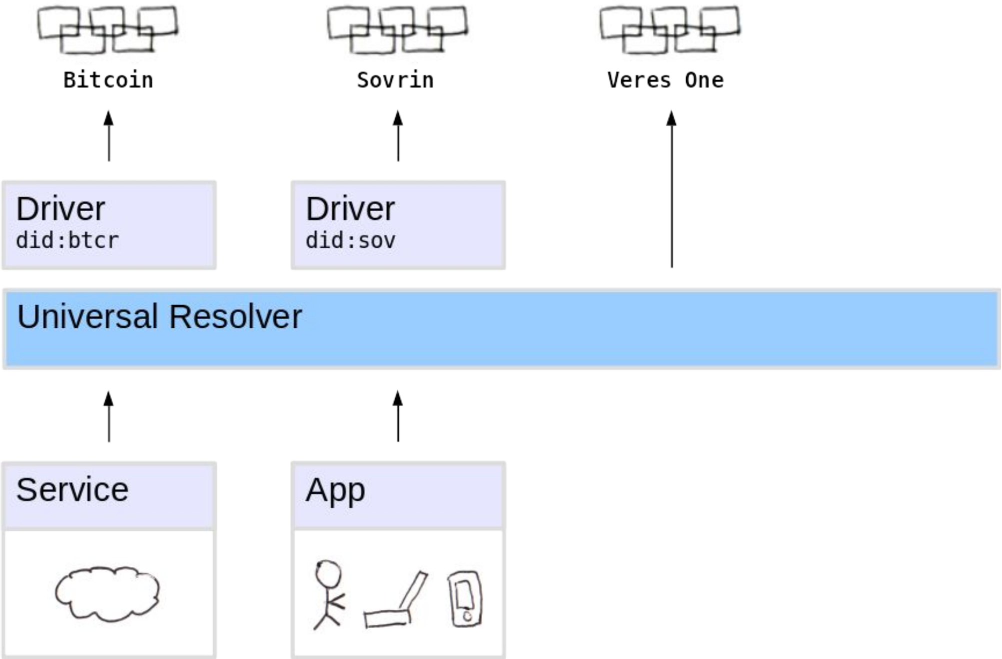
W3C JSON-LD 1.1

W3C Cryptographic Suites

RFC 7517: JWK

DID Universal Resolver

- Looks up ("resolves") DID to its DID Document.
- Provides a universal API that works with all DID methods.
- Uses a set of configurable "drivers" that know how to connect to the target system.
- **https://uniresolver.io/**



Bitcoin
Sovrin
Veres One
Driver
did:btcr
Driver
did:sov
Universal Resolver
Service
App