Open APIs for Open Minds

FIWARE Data Spaces

A Practical Introduction Into Roles and Components

Stefan Wiedemann - Technical Lead & Architect - FIWARE Foundation e.V.

FIWARE

Agenda

1. What is a Data Space?
2. The Data Space and its participants
    a. The Trust Anchor
    b. The Data Provider
    c. The Data Consumer
    d. The Marketplace
3. Live Demo

What is a Data Space?

- a “decentralized infrastructure for trustworthy data sharing and exchange in data ecosystems based on commonly agreed principles” - OPEN DEI initiative

- “A federated, open infrastructure for sovereign data sharing based on common policies, rules, and standards.” - Gaia-X

- “A data space is a secure and standardized digital infrastructure that enables trusted data exchange and data-based services among various stakeholders.” - IDSA

- “A data space can be defined as a data ecosystem built around commonly agreed building blocks enabling an effective and trusted sharing of data among participants for the creation of value.” - DSBA

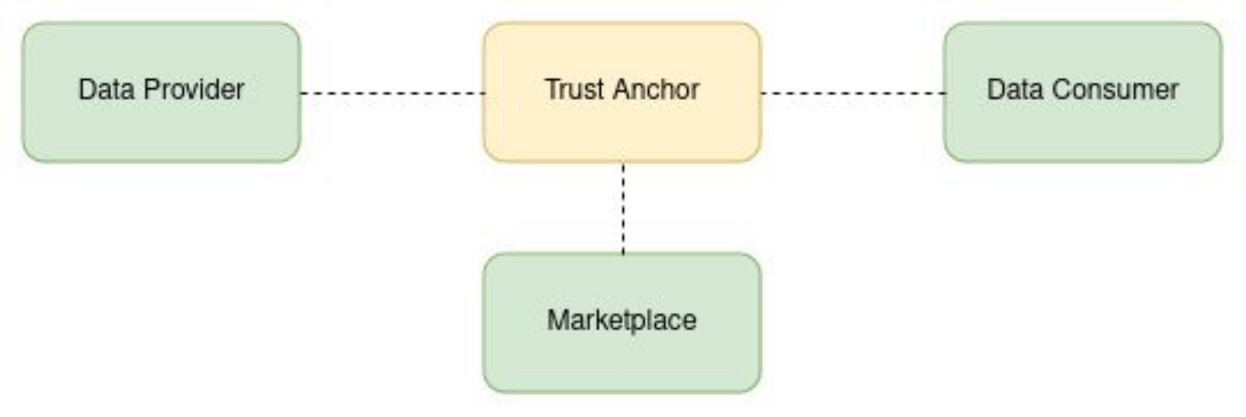
What is a Data Space?

- no single definition, but a common view
- different participants(organizations) form a Data Space together
- agree on common rules, standards and policies
- technical solution to share data and services
- technical solution is accompanied by legal solutions

The Data Space

- Participants will fulfill various roles
  - Data Provider
  - Data Consumer
  - Marketplace

- Trust Anchor(s) to ensure trust between the participants



The Trust Anchor

- Responsibilities:
  - provides the capabilities to identify participants
  - ensures independence from single participants

- often seen as part of the “Data Space Operator”
  - Data Space should define criteria for participation
  - might provide capabilities for on-boarding

- various options:
  - single Trust Anchor
    - centralized
    - decentralized
  - multiple Anchors for different interactions

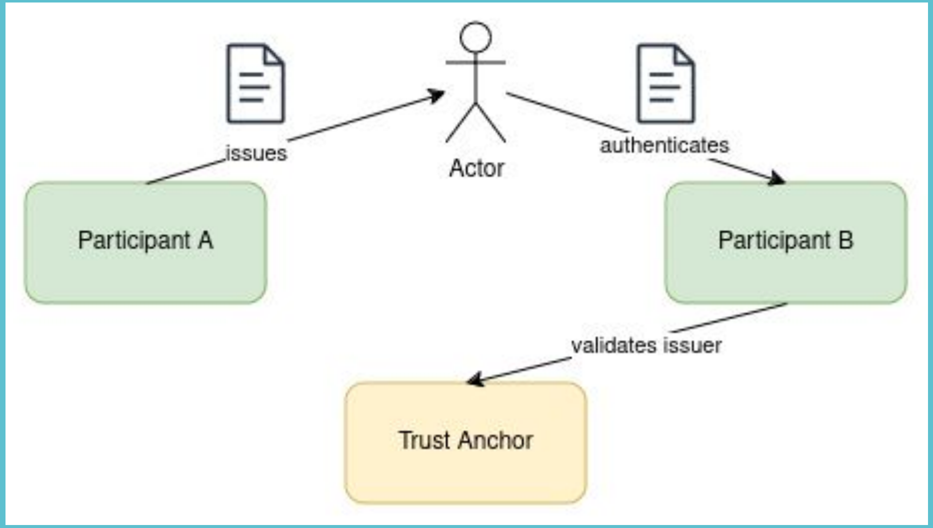
- has to participate in the authentication between participants

Authentication in a FIWARE Data Space

- Based on Decentralized Identifiers and Verifiable Credentials
- Decentralized Identifiers:
  - new type of identifier, that enables a verifiable, decentralized identity
  - the identifier can be resolved to a did-document that can express:
    - verification methods
    - cryptographic material
    - services to prove control of the did
  - examples:
    - did:key:zDnaeep661sHagxq47tMuJndWmmVngxEeaFmwD6uZzuoNDwSB - public key contained as part of the identifier
    - did:web:animalgoods.dsba.fiware.dev:did - did-document can be resolved through a well-known endpoint
- Verifiable Credentials:
  - signed json-documents
  - can contain any information, trust depends on the issuer
- combination of both allows decentralized, secure authentication in the Data Space

Authentication in a FIWARE Data Space

- participants issues Verifiable Credentials to actors
  - VC contains claims about the actor, the participant or other informations
  - signed with the Decentralized Identity of the participant
- Actor authenticates with the VC at another participant(using OID4VP)
- receiving participant verifies the credential signature and checks the issuer at the trust anchor
- API required from the Trust Anchor: EBSI Trusted Issuers Registry



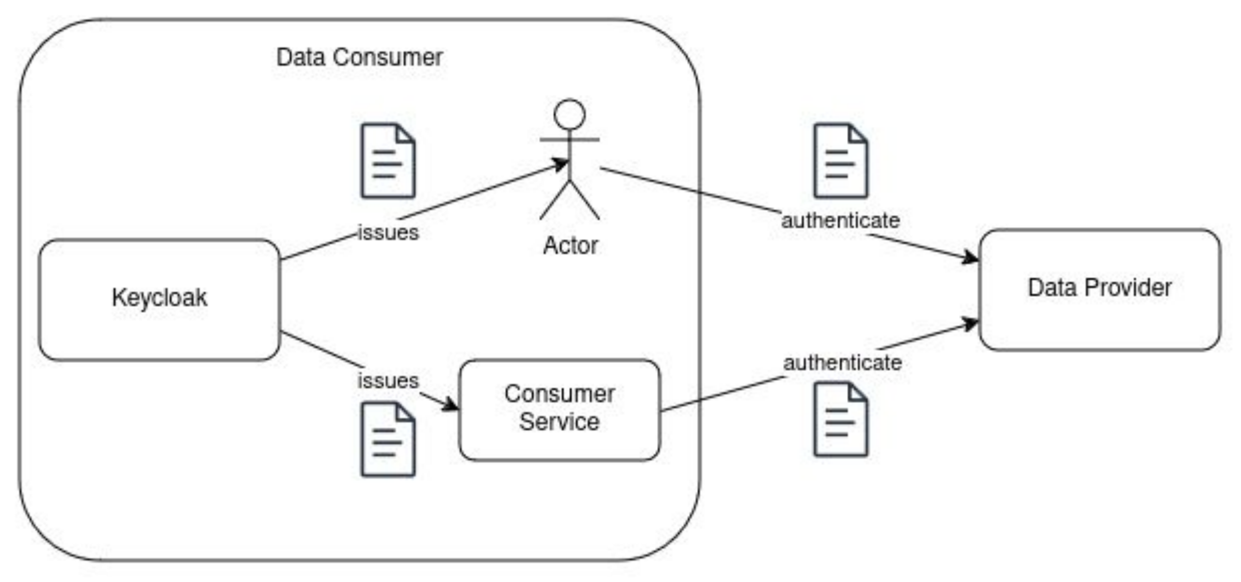
FIWARE

The Data Consumer

- wants to consume data from other participants or use their data services
- has to be registered at the Trust Anchor
- needs to properly authenticate
  - issue Verifiable Credentials to its actors
  - provide the right claims for user, service and purpose

The Data Consumer

- Keycloak as issuer of Verifiable Credentials - OID4VCI
- credentials can be issued to users or services to act on behalf of the organization
- claims per actor - allow fine grained definition of permissions for the actors
- currently no additional requirements for a pure consumer



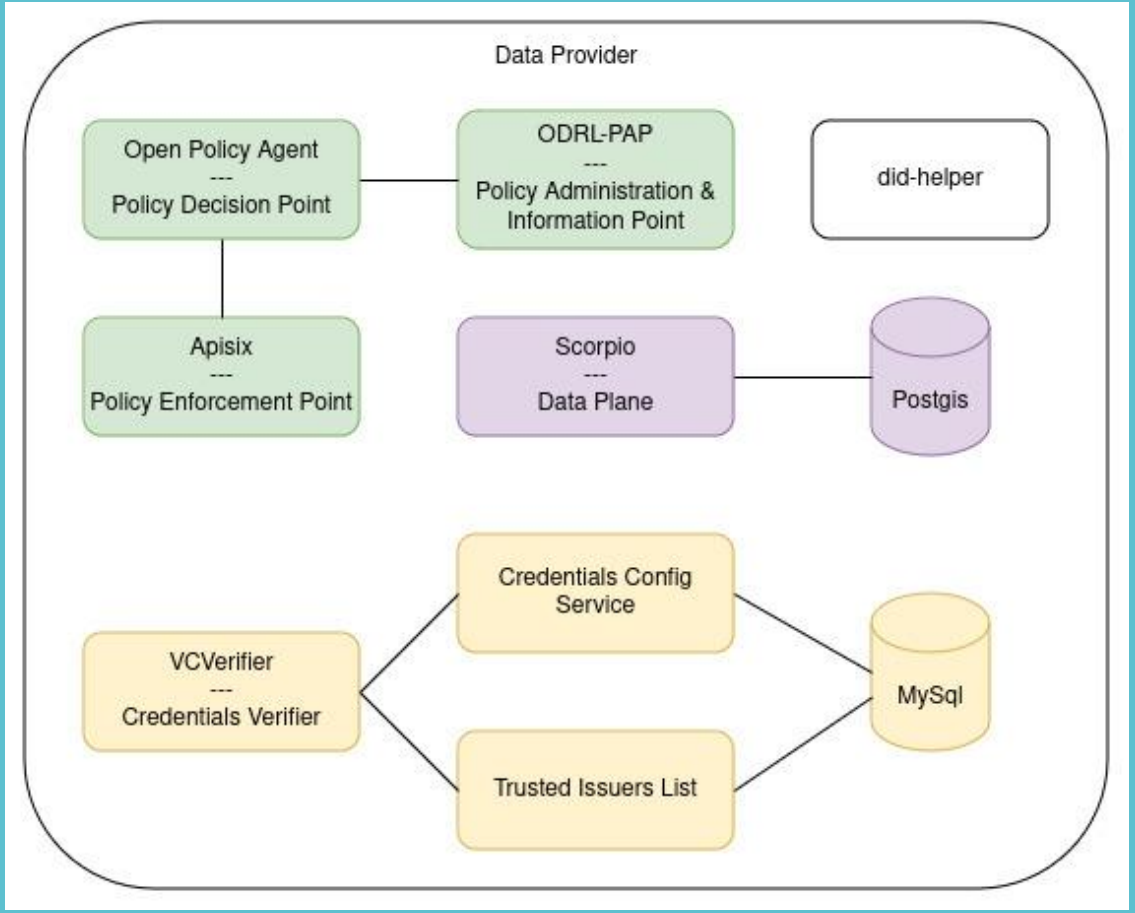
The Data Provider

- offers Data or Services
- wants to restrict access to participants of the Data Space
- defines access policies and levels and enforces them
- wants fine grained control about the access

The Data Provider

Components for Authentication

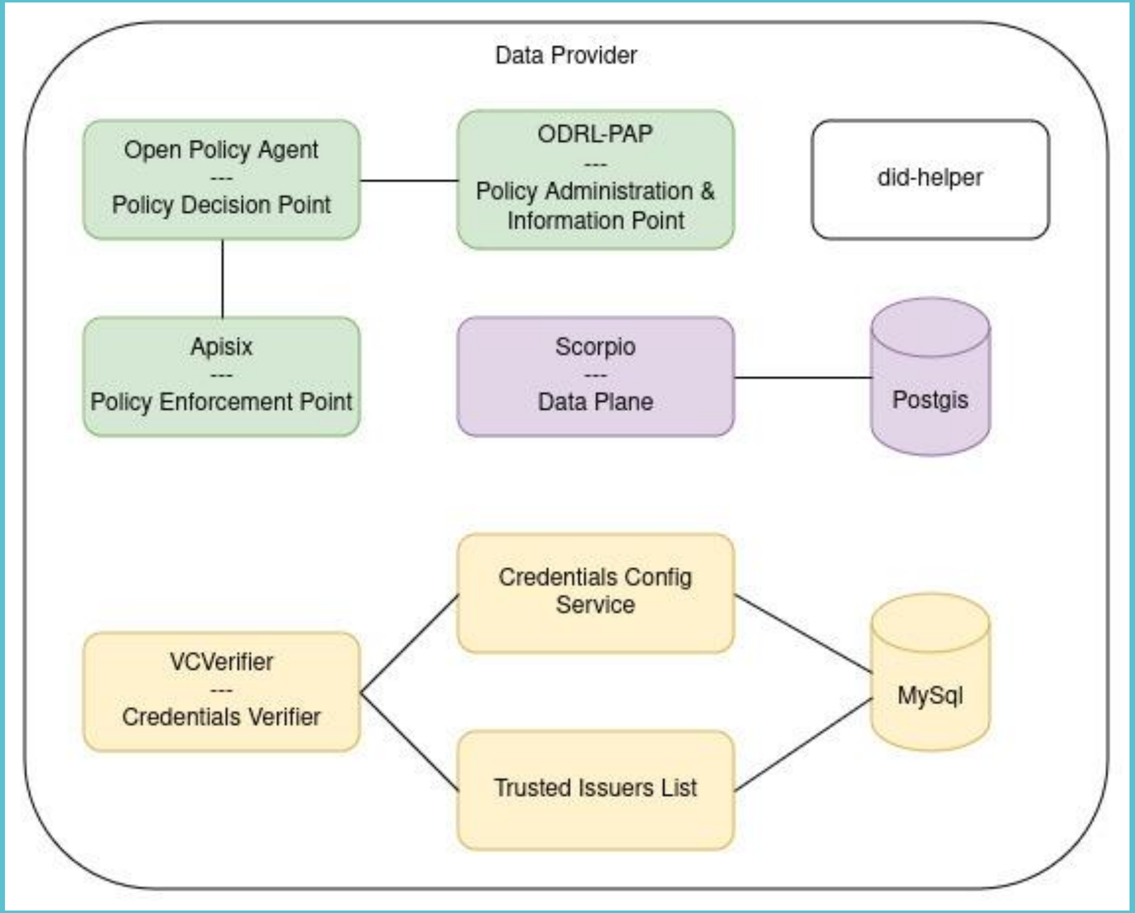
- Verifier
  - □ offers OID4VP compatible endpoints
  - □ verifies the signature of the credential
  - □ verifies that the issuer is a participant
  - □ verifies that only credential types and claims included that are allowed for the issuer
  - □ returns a JWT containing the claims
- Credentials Config Service
  - □ allows to configure the responsible Trusted Issuers Registry and List
- Trusted Issuers List
  - □ provides the allowed types and claims for the individual participant


Data Provider
Open Policy Agent --- Policy Decision Point
ODRL-PAP --- Policy Administration & Information Point
did-helper
Apisix --- Policy Enforcement Point
Scorpio --- Data Plane
Postgis
Credentials Config Service
VCVerifier --- Credentials Verifier
MySql
Trusted Issuers List

The Data Provider

Components for Authorization

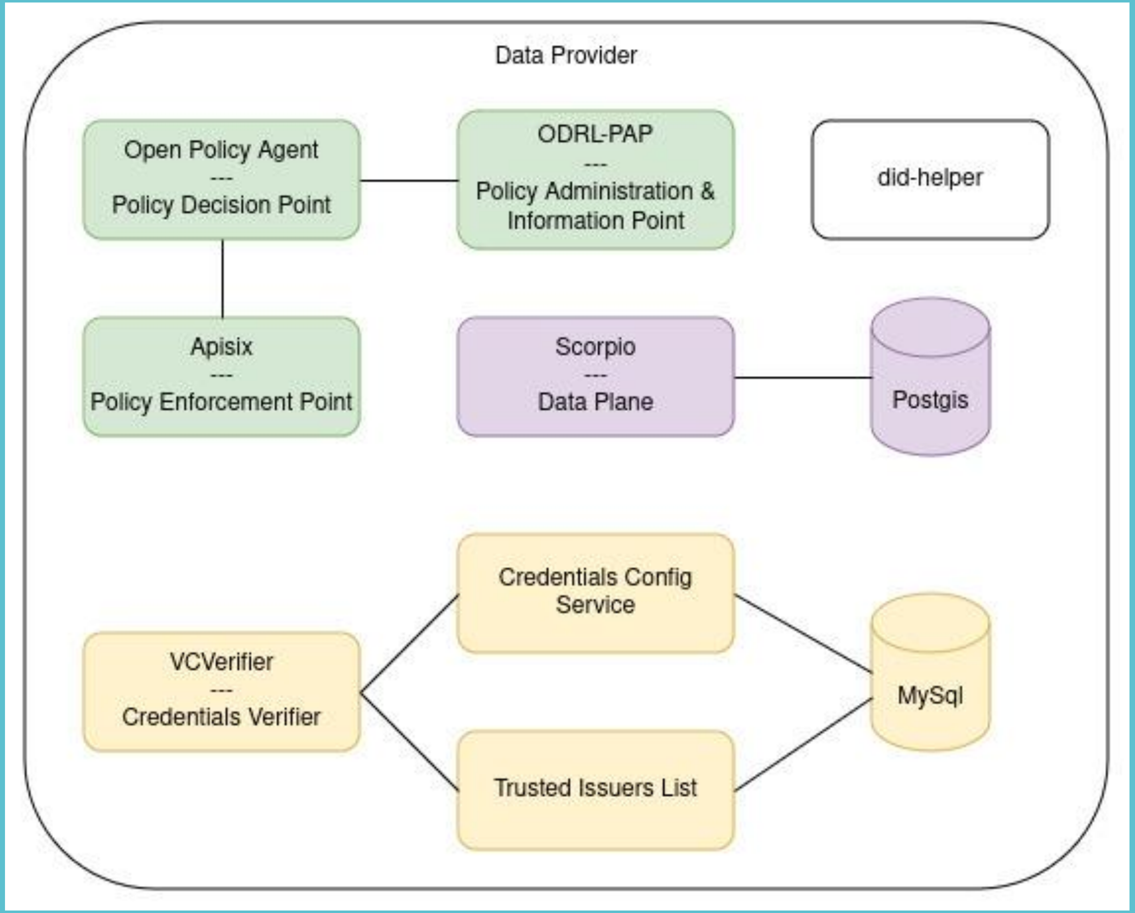
- Policy Enforcement Point
  - □ entry point for all requests to data or services
  - □ requests decisions from the Policy Decision Point
  - □ Apisix used as PEP, can provide additional functionality like access logging, rate limiting
- Policy Decision Point
  - □ decides about the request by evaluating the
  - □ policies and additional information
  - □ retrieves policies from the PAP
  - □ Open Policy Agent is used as PEP
- ODRL-PAP
  - □ Policy Management based on ODRL
  - □ translates policies into rego and offers them to the PDP
  - □ allows extension with individual profiles



The Data Provider

Data Plane

- typically an NGSI-LD compatible ContextBroker
- can essentially be any service
- currently only support for Rest-based communication, but other protocols are possible
- not restricted to a single Data Plane

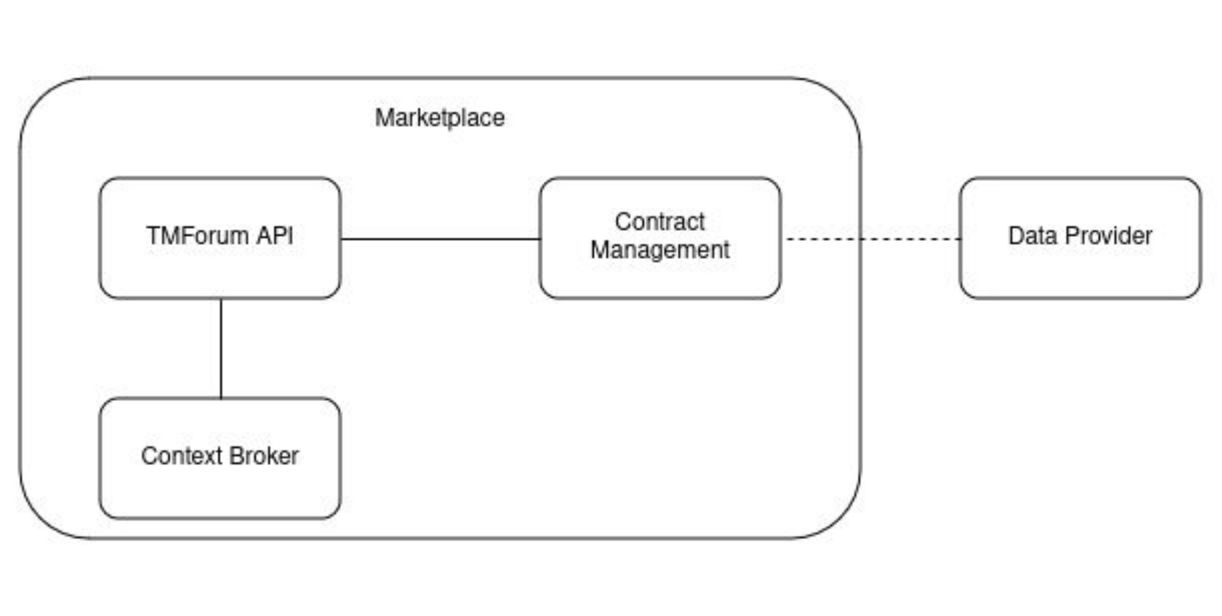

Data Provider
Open Policy Agent --- Policy Decision Point
ODRL-PAP --- Policy Administration & Information Point
did-helper
Apisix --- Policy Enforcement Point
Scorpio --- Data Plane
Postgis
Credentials Config Service
VCVerifier --- Credentials Verifier
MySql
Trusted Issuers List

The Marketplace

- provides a catalogue of available assets and services
- supports contract negotiations
- functionality to offer, sell and buy access to data and services

The Marketplace

- TMForum-API implementations to provide a standardized API
- Product, Offering etc. data stored inside the Context Broker
- Contract Management listens for product orders and enables access at the Data Provider
- Integrates with FIWARE Components like the BAE Marketplace



Demo

Local Minimal Viable Dataspace

Slides

https://github.com/wistefan/presentations

Thank you!

http://fiware.org
Follow @FIWARE on Twitter

FIWARE