Usage Control with The Eclipse Dataspace Connector (EDC)

Overview and Results

Amjad Ibrahim, Antonio La Marra ⋆*, Alessandro Rosetti* ⋆*,*

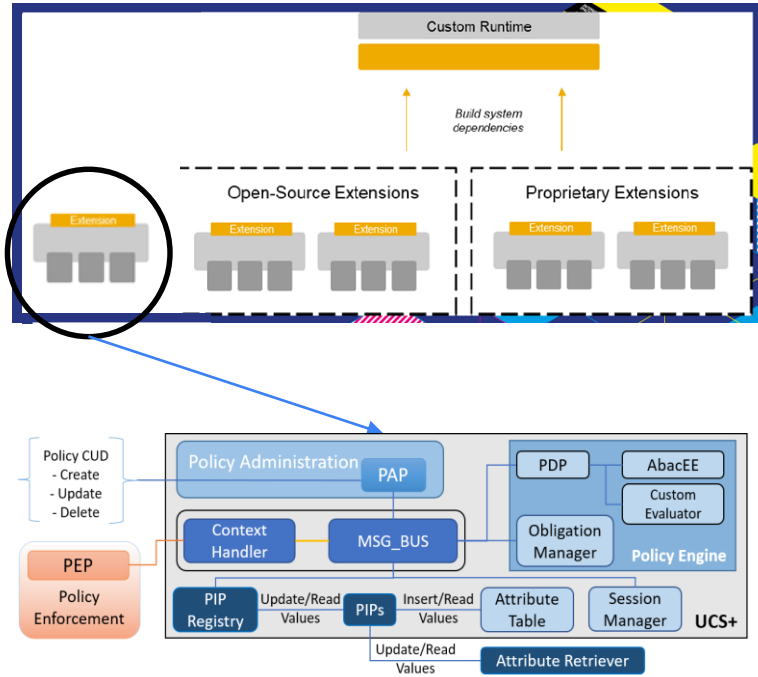
Theo Dimitrakos

German Research Center, *Huawei Technologies, Munich*

⋆ *Security Forge*, Pisa

Overview: Bringing Usage Control (UCON) into EDC

- UCON extends the ABAC with addition of
  - > **continuous** monitoring of **mutable** attributes
  - > authorization session management
  - > a change-driven re-evaluation of authorizations
- “specified usage restrictions and obligations are realized even after access to data has been granted”
  - > As a **hospital patient**, I want my medical data to be anonymized before being shared with **local officials**



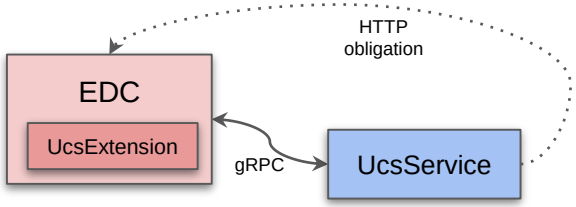
- EDC has an access control mechanism that directly evaluates ODRL policies. We plan to extend it with a modern usage control technology UCON

  ✓ first phase is to get to an ABAC-like initial support into EDC

  ✓ a light flavor of UCON with continuity, obligations, and notifications (no revocation)

  > continuous enforcement with full UCON capabilities.

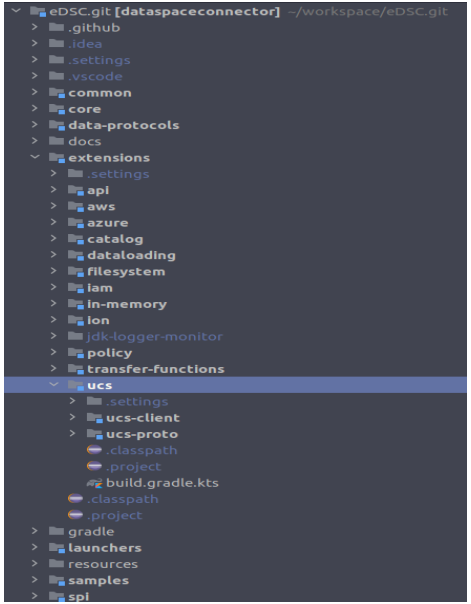
Hackathon Progress



✓EDC-UCS extension

- ✓ Follows the EDC style to extend the default policy behavior with EDC

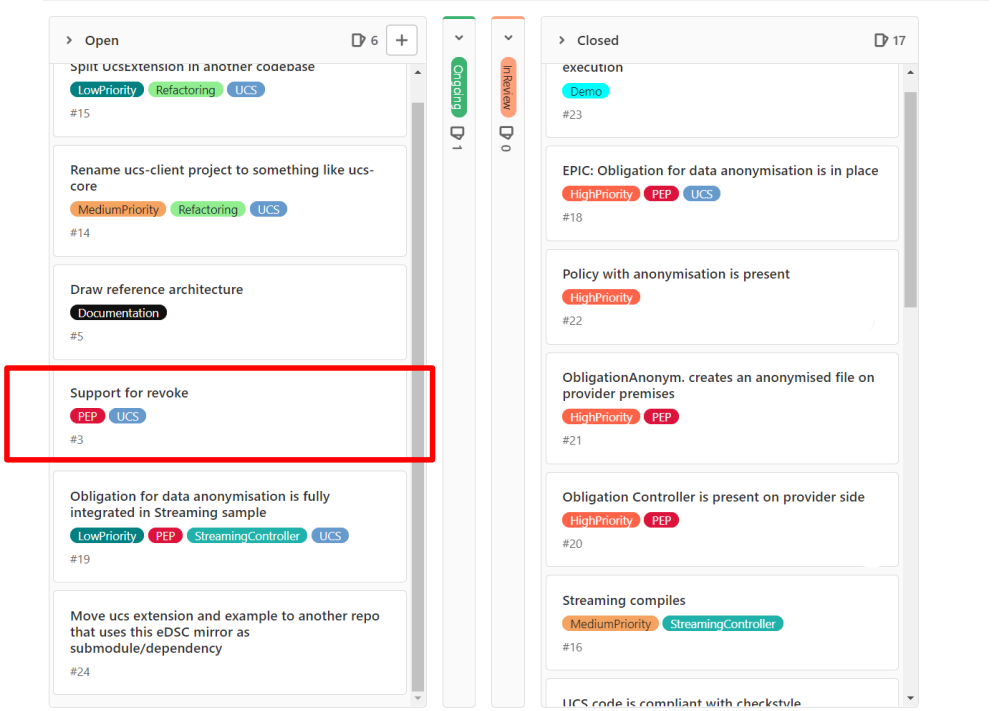
- ✓ Technical:
  - ✓ UcsExtension: registers UcsIdsPolicyService that replaces the default IdsPolicyService.
  - ✓ UcsClient: connects to the UcsService through gRPC.
  - ✓ UCS module contains the protobuf files.
  - ✓ **UCS** evaluates an **ALFA** policy containing obligations that are executed in the UCS itself.



✓Custom Policy Enforcement Points (PEP) and privacy-preserving obligation for 2 EDC samples

- ✓ File transfer
- ✓ Streaming

Progress and Further Steps



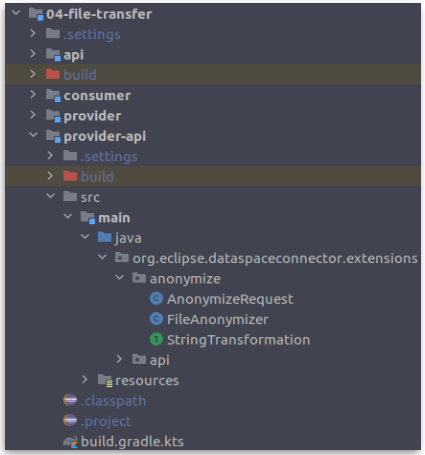
Custom PEP and Obligation for File Transfer sample

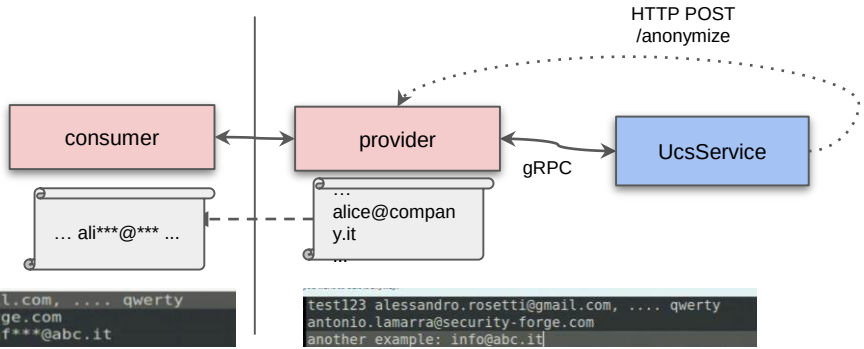
- Simple Policy
- The file contents will be anonymized by anonymizing any email text with a regex **alice@company.it** -> ali***@***


```
INFO 2021-11-25T15:37:29.002149179 edc-cd43fbe4-7482-4061-9cbd-80f1870ec9e6 ready
INFO 2021-11-25T15:37:39.565588944 Received request for file test-document against provider http://localhost:8181/
DEBUG 2021-11-25T15:37:44.889291481 Request approved and acknowledged for process: 85d3e96c-ee6a-46b1-905b-cc98c2000d42
DEBUG 2021-11-25T15:37:49.210584958 Process 85d3e96c-ee6a-46b1-905b-cc98c2000d42 is now IN PROGRESS
DEBUG 2021-11-25T15:37:49.211243651 Process 85d3e96c-ee6a-46b1-905b-cc98c2000d42 is now COMPLETED
```

```
policy anonymize {
    apply firstApplicable
    rule r1 {
        permit
        on permit {
            obligation http {
                url = 'http://localhost:8181/api/anonymize'
                method = 'POST'
                body = '{"type" : "email-anonymize"}'
                headers = 'Content-Type:application/json'

            }
        }
    }
}
```






```
test123 ale***@gmail.com, .... qwerty
ant***@security-forge.com
another example: inf***@abc.it
```

```
test123 alessandro.rosetti@gmail.com, .... qwerty
antonio.lamarra@security-forge.com
another example: info@abc.it
```

Custom PEP and Obligation for Streaming sample

Added a simple UI that leverages the existing WebSocket infrastructure

Two users subscribed to the same topic “hello”

The usage policy **permits** them to exchange information around it but any shared personal information (e.g., emails) must be **masked** for others