

Usage Control with The Eclipse Dataspace Connector (EDC)

Overview and Results

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

Theo Dimitrakos

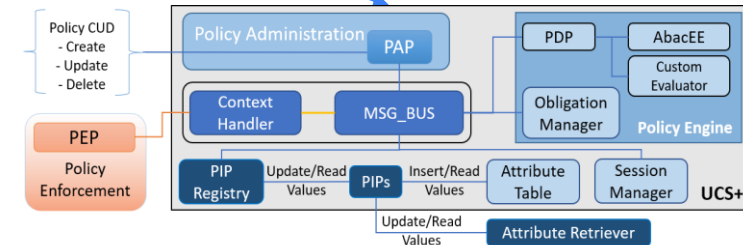
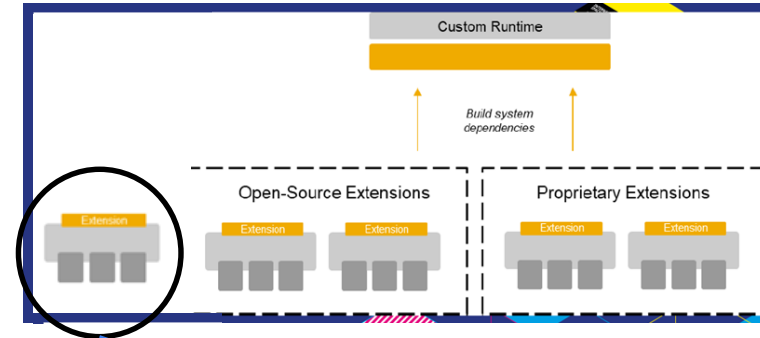
German Research Center, Huawei Technologies, Munich

** 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

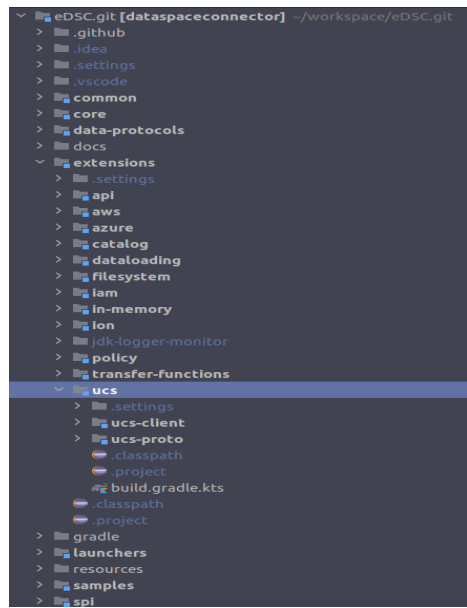
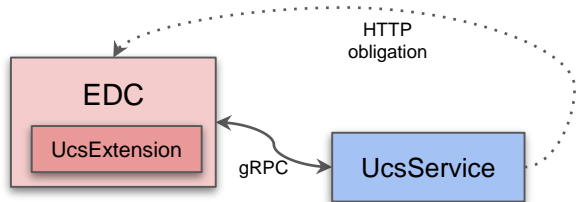
- ✓ Follows the EDC style to extend the default policy behavior with EDC

✓ 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

The screenshot displays a project management interface with two main columns: 'Open' and 'Closed'. The 'Open' column contains six tasks, with the third task, 'Support for revoke', highlighted by a red rectangle. The 'Closed' column contains seven tasks. Each task includes a title, a unique identifier, and one or more colored tags representing categories or priorities.

| Task Title | ID | Tags | Status |
|---|-----|--|--------|
| Split ucsextenstion in anotner codebase | #15 | LowPriority, Refactoring, UCS | Open |
| Rename ucs-client project to something like ucs-core | #14 | MediumPriority, Refactoring, UCS | Open |
| Draw reference architecture | #5 | Documentation | Open |
| Support for revoke | #3 | PEP, UCS | Open |
| Obligation for data anonymisation is fully integrated in Streaming sample | #19 | LowPriority, PEP, StreamingController, UCS | Open |
| Move ucs extension and example to another repo that uses this eDSC mirror as submodule/dependency | #24 | | Open |
| execution | #23 | Demo | Closed |
| EPIC: Obligation for data anonymisation is in place | #18 | HighPriority, PEP, UCS | Closed |
| Policy with anonymisation is present | #22 | HighPriority | Closed |
| ObligationAnonym. creates an anonymised file on provider premises | #21 | HighPriority, PEP | Closed |
| Obligation Controller is present on provider side | #20 | HighPriority, PEP | Closed |
| Streaming compiles | #16 | MediumPriority, StreamingController | Closed |
| UCS code is compliant with checkstyle | | | Closed |

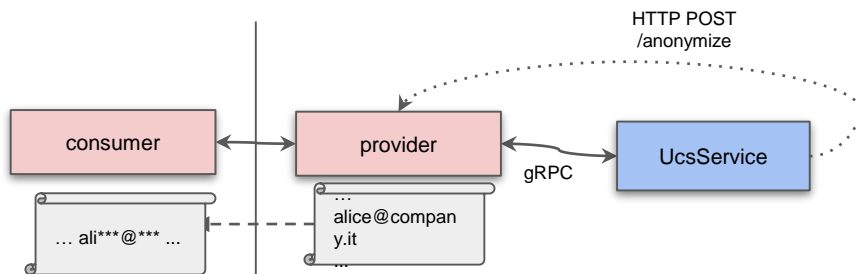
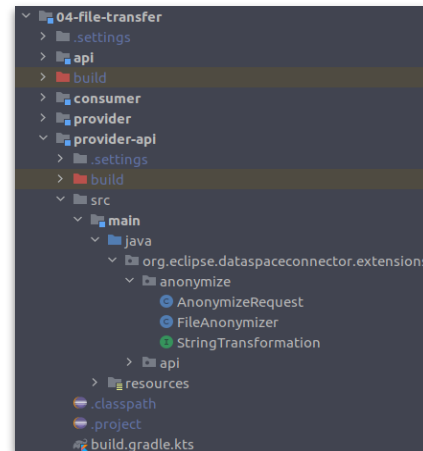
Custom PEP and Obligation for File Transfer sample

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

```
roarc at static in ~/workspace
./run-file-transfer-curl.sh
```

```
INFO 2021-11-25T15:37:29.682149179 edc.cd43f0e4-7482-4661-9cbd-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 ri {
    permit
    on permit {
      obligation http {
        url = 'http://localhost:8181/api/anonymize'
        method = 'POST'
        body = '{"type": "email-anonymize"}'
        headers = 'Content-Type:application/json'
      }
    }
  }
}
```



```
1 test123 ale***@gmail.com, .... qwerty
2 ant***@security-forge.com
3 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

