Andreas Put and Bart De Decker

imec-DistriNet, KU Leuven

andreas.put@kuleuven.be

bart.dedecker@kuleuven.be

IoTSEAR: a System for Enforcing Access control Rules with the IoT

KU LEUVEN DistriNet

IARIA

About

Andreas Put is a postdoctoral researcher in the imec-DistriNet research group at KU Leuven. During his PhD, his research focused on privacy-enhancing technologies, anonymous authentication, and e-Voting. However, his research in recent years centers around enhancing security and privacy specifically in IoT environments.

Outline

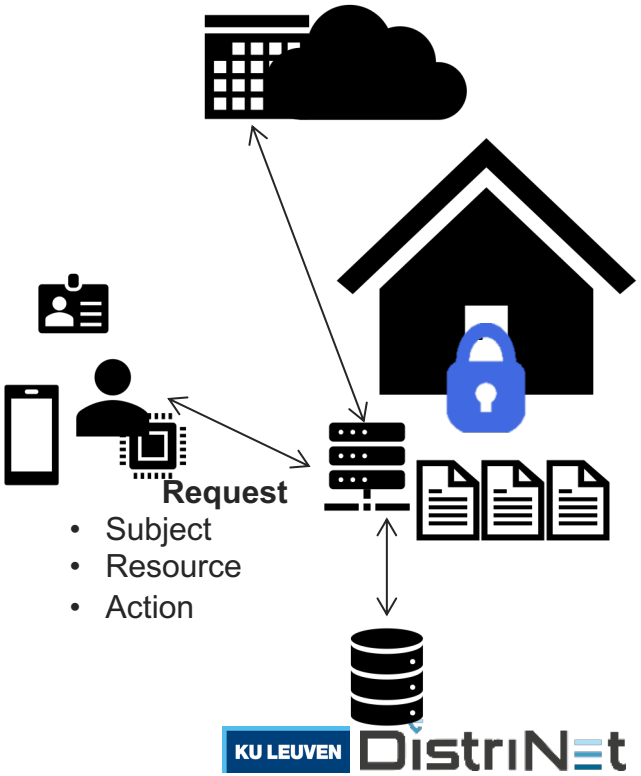
- Introduction
- Context Model
- IoTSEAR
- Conclusion

Introduction

Introduction

Context-aware access control

- Context types
  - System state
  - IoT Device context
  - 3rd party/Cloud service
    - Incl. federated identity management
- Context security requirements!
  - Integrity, authenticity, ownership



Introduction

IoTSEAR scope

- How to specify context security requirements?

  → **Generic model for context**

- How to specify context aware access permissions?

  → policy language [1]

- How to enforce access permissions & security requirements?

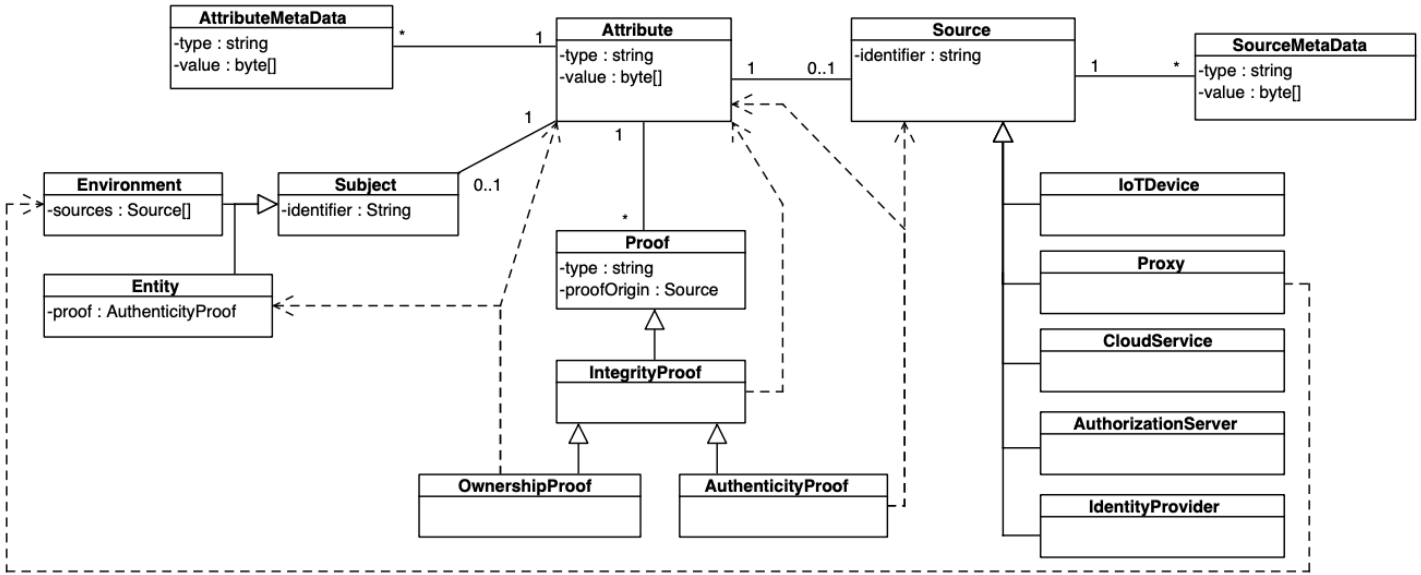
  → **IoTSEAR middleware**

[1] A. Put and B. De Decker, “Attribute-based privacy-friendly access control with context,” in International Conference on E-Business and Telecommunications. Springer, 2016, pp. 291–315

Context Model

Context Model

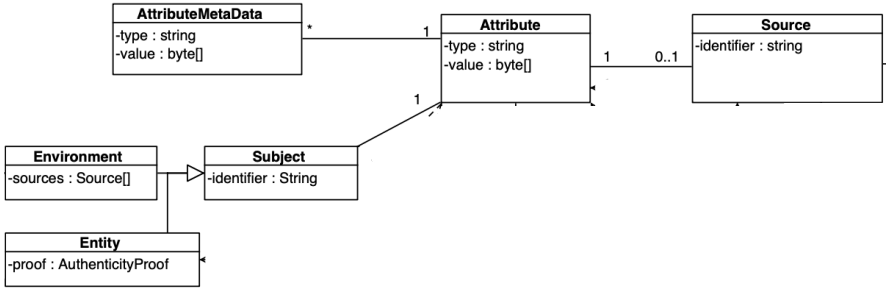
Overview



Context Model

Attribute & Proof

- Attribute:
  - Raw data
    sensor output, identity/authorization token, …
  - Metadata: timestamp, encoding, …
  - Source
  - Subject
- Proof:
  Universally verifiable object
  - IntegrityProof & AuthenticityProof
    Verify attribute Integrity & source authenticity
  - OwnershipProof
    Verify link between Subject & Attribute



Instances of the context model

Applied to a sensor reading

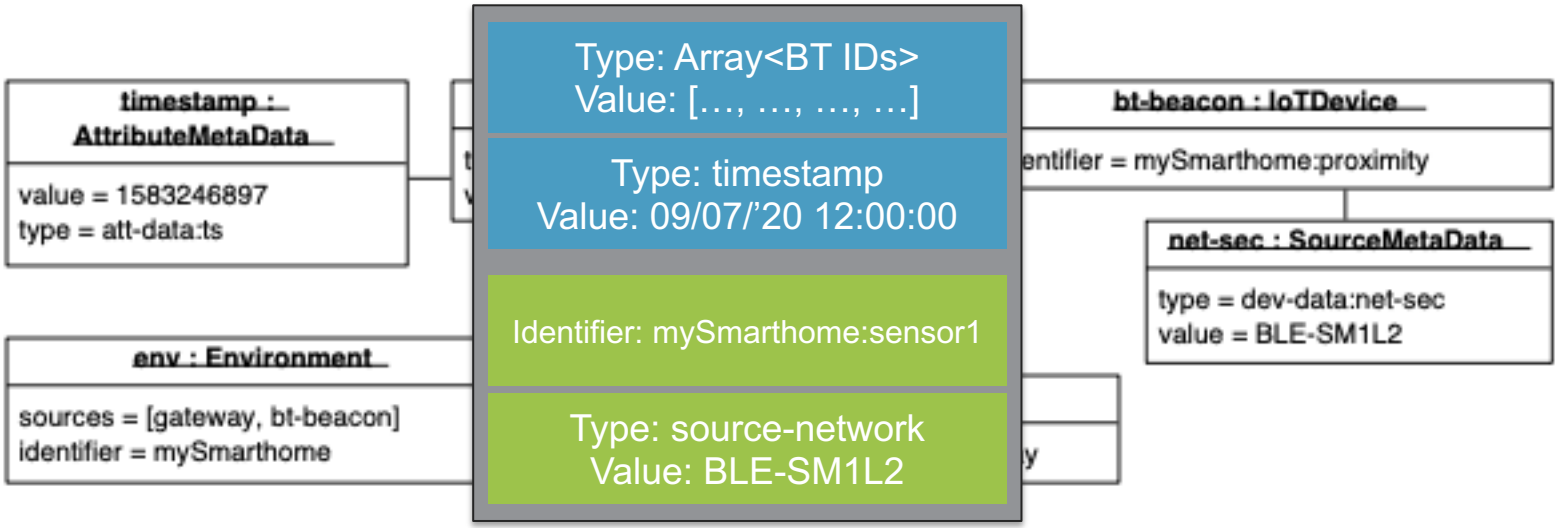
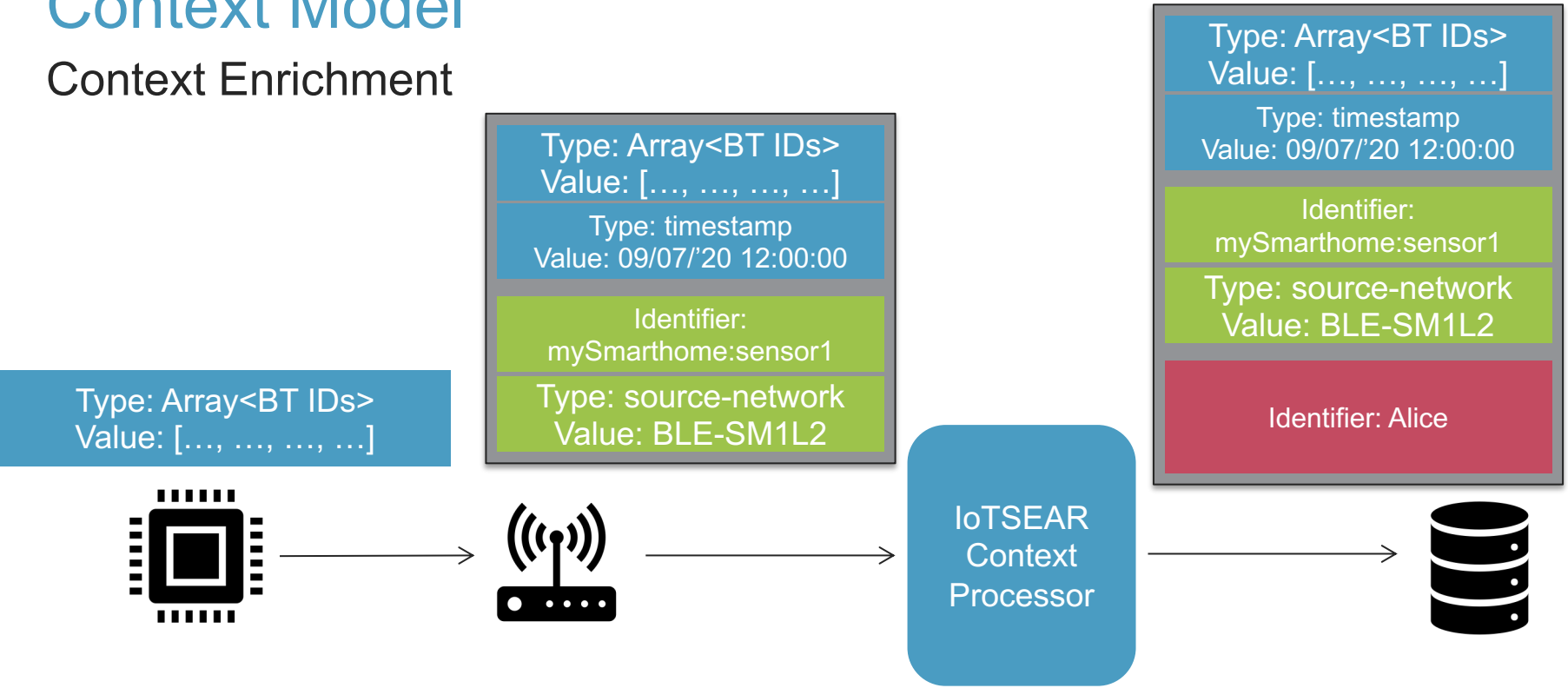


Figure 2. The context model applied to a proximity sensor reading

IoTSEAR

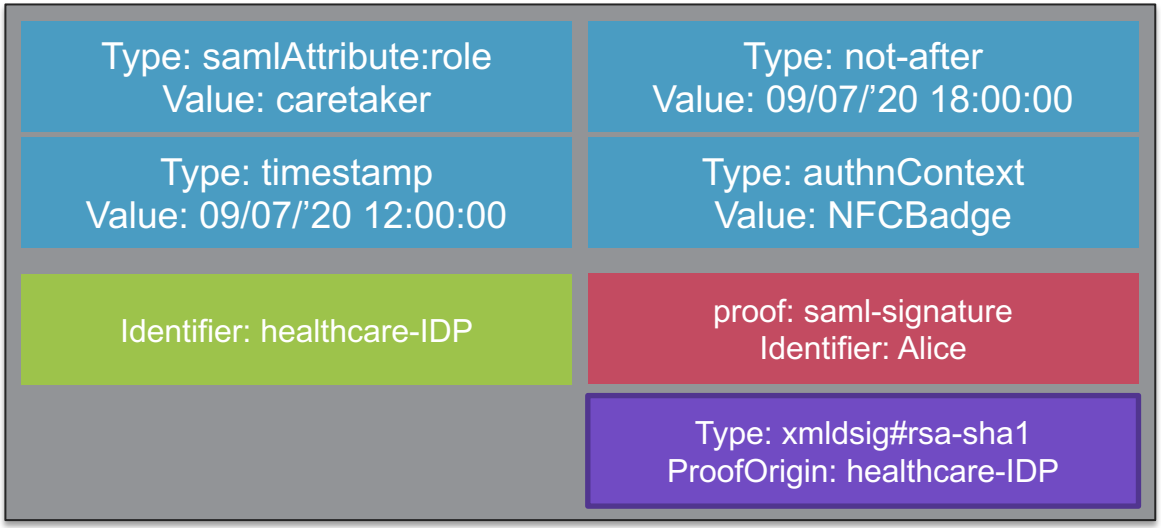
Context Model

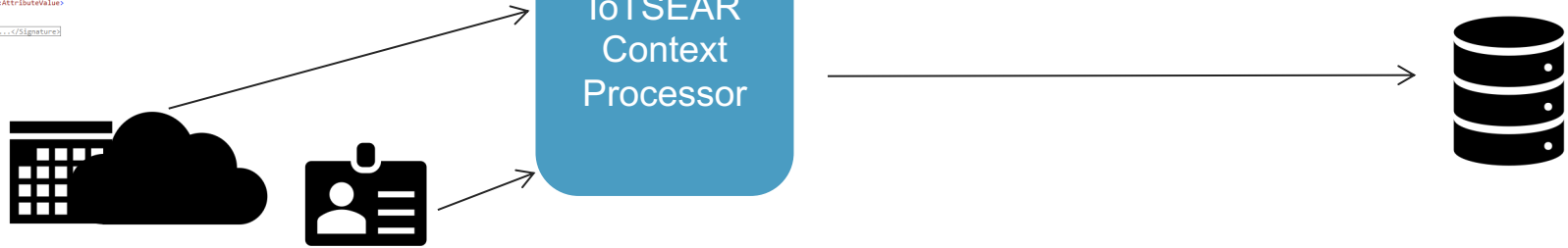
Context Enrichment



Context Model

Context Enrichment

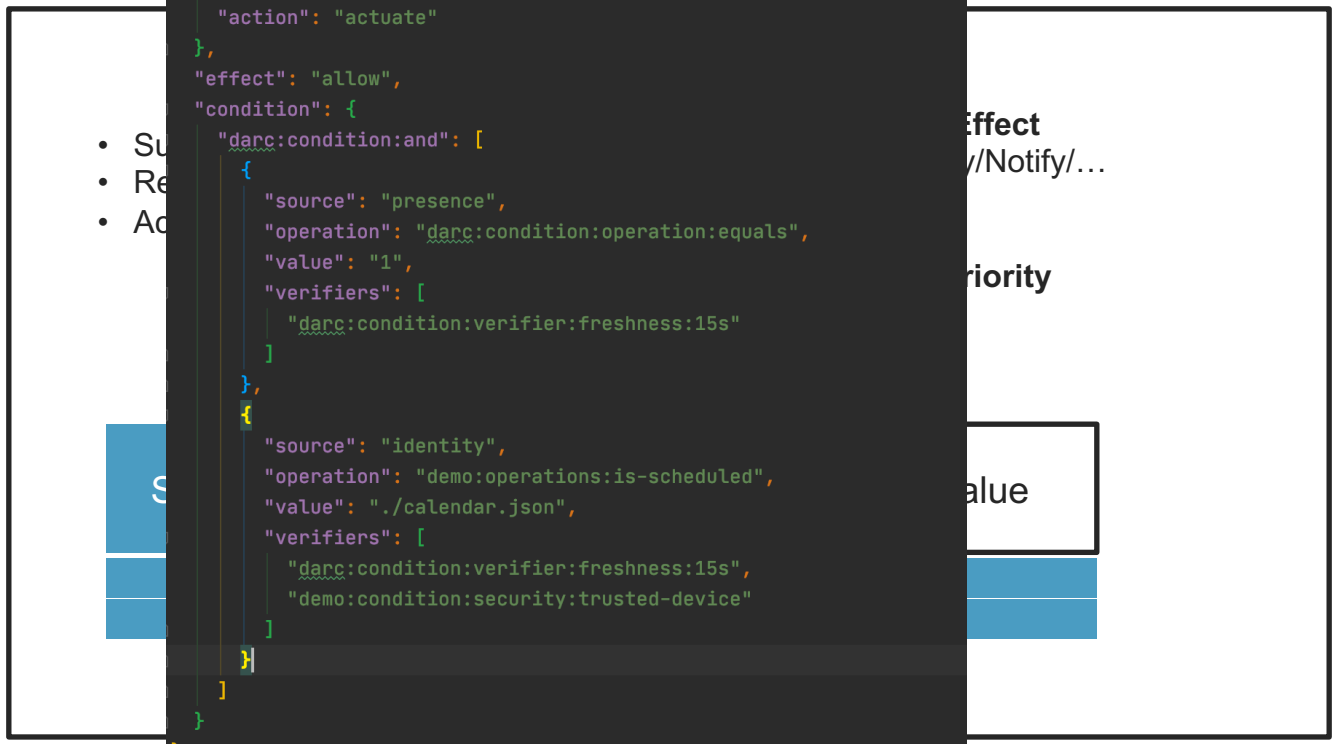




KU LEUVEN DistriNet

IoTSEAR

Policy representation



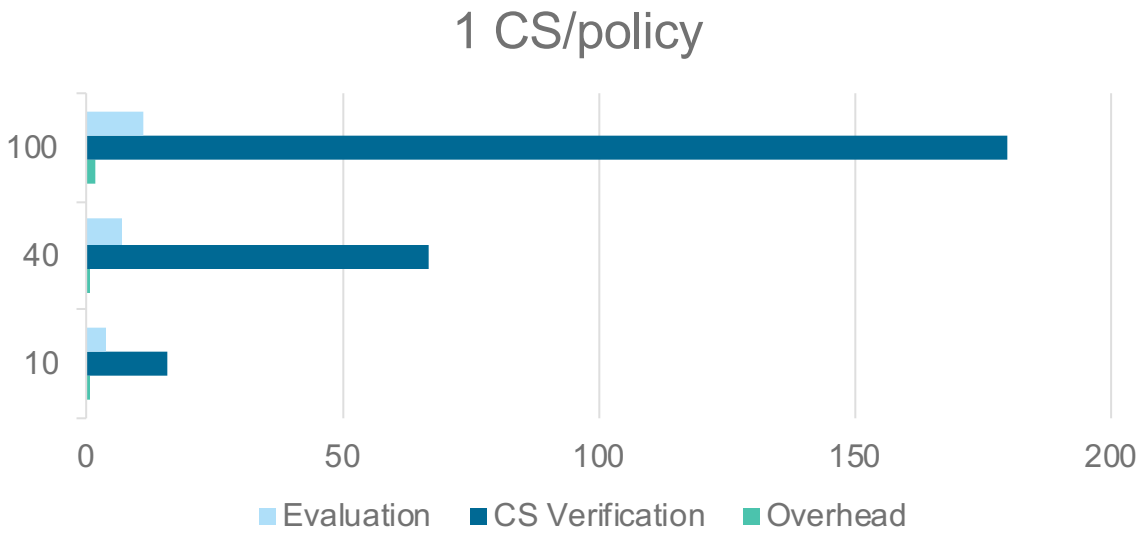

```
{
  "identifier": "smartlock-actuate-normal",
  "priority": 1,
  "target": {
    "subject": "AnySubject",
    "resource": "smartlock-1",
    "action": "actuate"
  },
  "effect": "allow",
  "condition": {
    "darc:condition:and": [
      {
        "source": "presence",
        "operation": "darc:condition:operation:equals",
        "value": "1",
        "verifiers": [
          "darc:condition:verifier:freshness:15s"
        ]
      },
      {
        "source": "identity",
        "operation": "demo:operations:is-scheduled",
        "value": "./calendar.json",
        "verifiers": [
          "darc:condition:verifier:freshness:15s",
          "demo:condition:security:trusted-device"
        ]
      }
    ]
  }
}
```

IoTSEAR

Context verifiers

- Middleware component
  - Selected through identifiers in policy
  - Used to filter useable context objects
    - Input: Context object
    - Output: Boolean
- E.g. Freshness, known-devices, basic/substantial/high, …
  - Often application dependent

IoTSEAR performance



Conclusion

Conclusion

- Generic model for context
  - Allows (third parties) to verify custom security requirements
- IoTSEAR middleware
  - Policy enforcement & context management
  - Application specific security requirements
  - Acceptable performance overhead

Thank you!

andreas.put@kuleuven.be

bart.dedecker@kuleuven.be