UiO : Department of Informatics
University of Oslo

Semantic - Attribute Based Access Control

Uğur Bayram
University of Oslo

Master of Science - Short Thesis - 30 ECTS

Supervisors:

Christian Johansen, Hamed Arshad

UiO : Department of Informatics
University of Oslo

Semantic - Attribute Based Access Control
Master Thesis

Uğur Bayram
University of Oslo

Master of Science (MSc) - **30 ECTS**

Supervisors:

Christian Johansen, Hamed Arshad

Thesis Presentation Outline

Introduction

01

Subject:
Semantic - Attribute Based Access Control

02

Why did I choose

03

SCOTT

SCOTT
Secure Connected Trustable Things

04

Goals & Objectives

Motivation

Access Control

- Access control in general
- Access control in IoT

ABAC

- Fine-grained:
  - Flexibility,
  - Dynamicity,
  - Granularity

Limitations

- Centralized (?)
- System complexity
  - Storage
  - Processor
  - Network

[1]https://www.sciencedirect.com/science/article/pii/S1877050918305064?via%3Dihub

[2]https://ieeexplore.ieee.org/document/8632578

Research Objectives

Objective #1 : ABAC on fog-area

External access control systems authorizing distributed devices over network is not ideal.

Objective #2 : Semantic-ABAC

Plain ABAC has limitations on context extraction. Semantic reasoner helps to solve this.

Objective #3 : Decentralized Access Control

Changing the trend centralized approach to decentralized and running improved S-ABAC on resource-constrained IoT gateway.

Research Questions

01

Question#1

What is the importance of the fine-grained access control for IoT settings?

02

Question#2

Can ABAC work for simple IoT devices on the fog-area?

03

Question#3

What is the ideal gateway device for this need?

04

Question#4

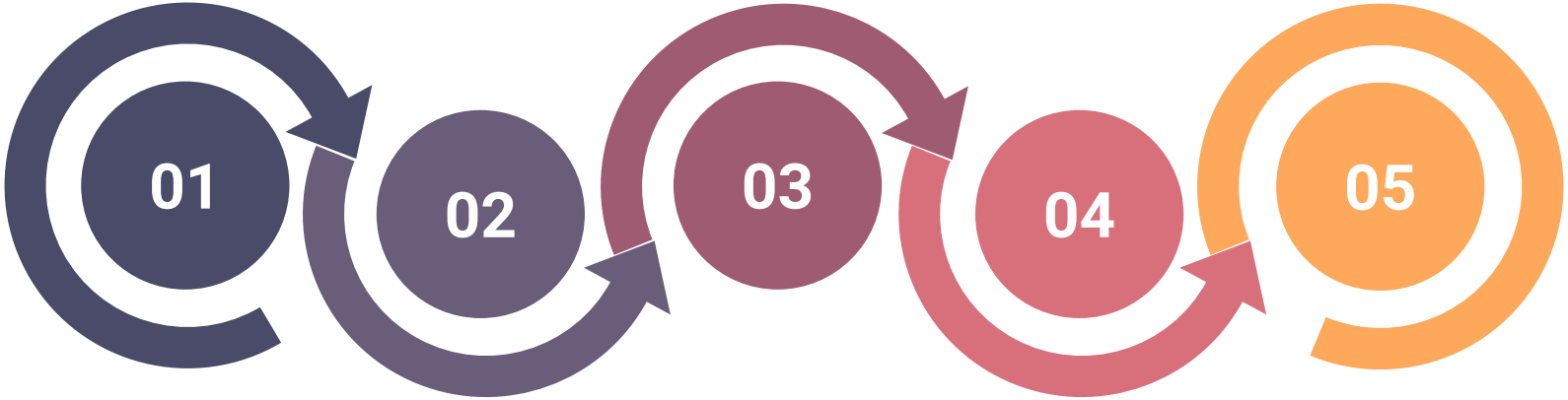
How can we link an ABAC engine to a Semantic Reasoner having a simple ontology?

Methodology

Step 01
Identify the areas: IoT Security, Access Control Models, ABAC, Semantic Technology, Ontology

Step 03
Define high-level system architecture, review IoT Gateways and select the ideal one.

Step 05
Prototype assessment, performance tests for ABAC & S-ABAC on cloud & fog area,



Step 02
Literature review on journals, articles, previous thesis on similar fields, external blogs and internet resources.

Step 04
Implement Semantic Reasoner and integrate with ABAC engine

Background - Access Control Systems

Discretionary Access Control (DAC)

- Resource owner
- Access control list
- Popular in OS resource and directory control
- Weakest access control system
- Ideal for small teams, companies

Mandatory Access Control (MAC)

- Controlling well-characterized information flow: hierarchical
- User cannot change rules
- Suitable for high level confidentiality required organizations: military

Role Based Access Control (RBAC)

- Subject role defines permission
- Can be customized per-application
- Easy to define roles and permissions
- User-Role-Permission combinations may go out of control for large organizations

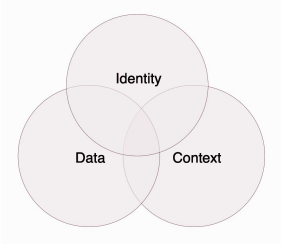
Attribute Based Access Control (ABAC)

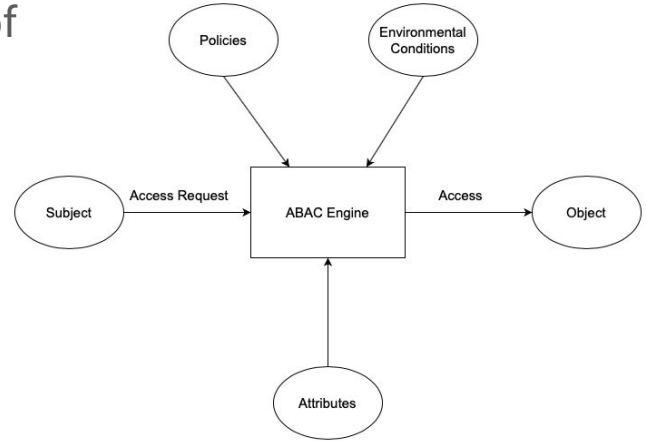
- As per NIST, evolved from IBAC, RBAC
- Subject attributes, object attributes, environmental conditions define the policy
- Complex infrastructure
- Extensive flexibility, granularity

https://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.SP.800-162.pdf

ABAC - I

- Multidimensional characteristic
- ABAC entities: Subject, object, operation, environmental conditions, policy
- ABAC makes decision by combining all entities
- Higher number of discrete inputs results larger set of possible combinations; makes ABAC more flexible
- ABAC Result: Access, Deny, Not Applicable, Indeterminate





https://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.SP.800-162.pdf

ABAC - II

PROS

- **Fine-Grained, Attribute-Based:** Attributes are building blocks
- **Externalized Authorization:** Decoupled business logic
- **Centralized Authorization:** Easy deployment, easy maintenance, reduced cost, large hardware capacity (on cloud)
- **Policy-Driven Authorization:** Configurable policy definitions

CONS

- **Doesn't fit to IoT:** Distributed environments with several different domains, collaborating with each other
- **Context awareness:** Surrounded with set of logical functions

Extending ABAC Capabilities

Semantic Tech & Ontology

Semantic Technology:

- Machine readable to machine understandable
- IoT example: smart sensor sending information to any platform
- RDF vs XML

Ontology:

- Providing a shared understanding of common domains
- Hospital domain: departments, clinics, staff
  - Staff: Technical, administrative, operational
- OWL

S-ABAC

S-ABAC = ABAC + Semantic Technologies

- An ontology management system
  - provides the extended user and resource attributes
- An access control system
  - uses the extended attributes for access evaluation[1]

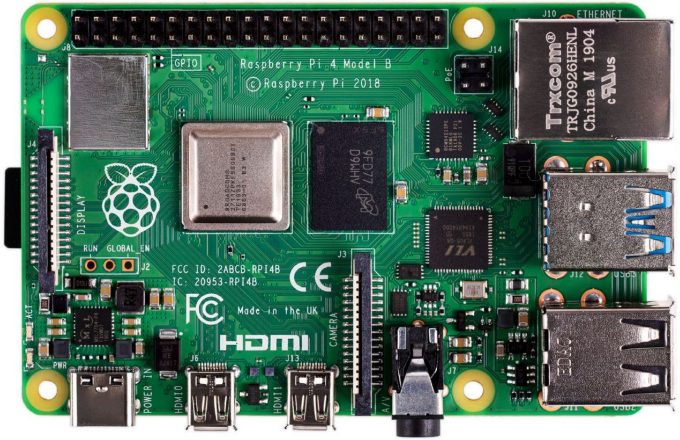
[1] https://its-wiki.no/images/2/28/SABAC-presentation.pdf

Access Control Comparison

	DAC	MAC	RBAC	ABAC	S-ABAC
Simplicity	Simple	Simple-Medium	Medium	Complex	Complex
Flexibility	Least	Less	More	Most	Most
Strength	Weakest	Strong	Strong	Strong	Strong
Granularity	No	No	No	Yes	Yes
Context-awareness	No	No	No	No-Yes	Yes

Popular IoT Boards

Raspberry Pi	**Microcomputer** CPU: Quad-core 1.5GHz RAM: 2GB, 4GB or 8GB Dimensions: 88mm × 58mm × 19.5mm WiFi, Bluetooth Unix-Debian, community support, convenient development environment



Arduino	**Microcontroller** CPU: Single-core 16MHz Flash Memory: 32 KB Dimensions: 68.6 x 53.4 mm WiFi No Operating system Arduino programming language

Prototype Implementation - Overview

ABAC

FIWARE Authzforce ABAC project

- Java based, open-source project
- OASIS XACML compliant
- Offers RESTful authorization server

Minimum Hardware Requirements

- CPU frequency: 2.6 GHz
- RAM: 4GB min
- Disk space: 10 GB min

Software Requirements:

- JRE 8 and above
- Tomcat 8.x

RPi

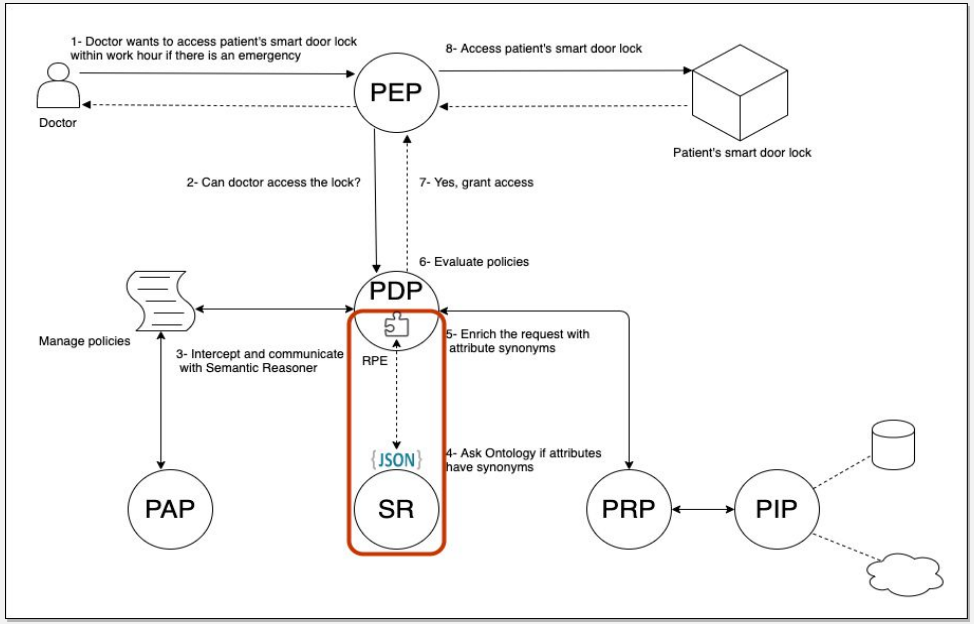
Raspberry Pi Model 4B

- CPU: Quad-core 1.5GHz
- RAM: 4GB
- Memory: 16GB Micro SD Card
- Display: Mini-HDMI Port
- USB Ports: Allowing to use keyboard and mouse

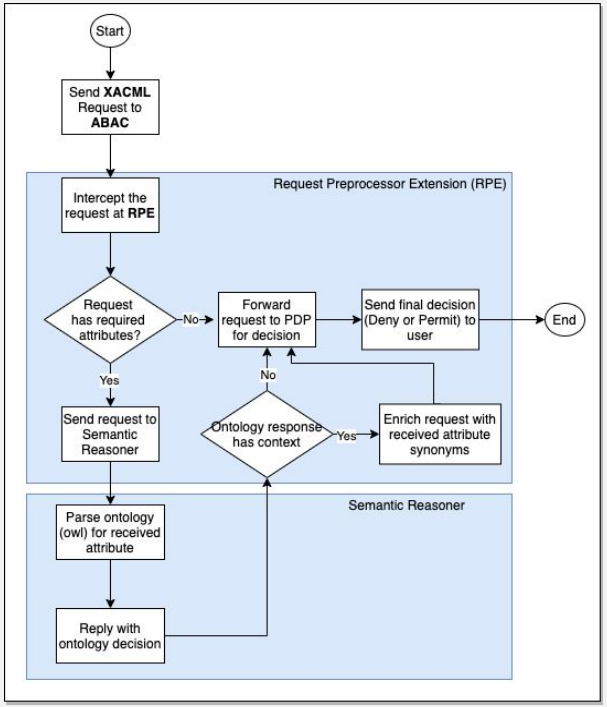
Turns to be an ideal development platform

Prototype Implementation - I

High Level Architecture



Flow Diagram



Prototype Implementation - II

ABAC Engine Setup on Raspberry Pi 4

Install Tomcat and Java

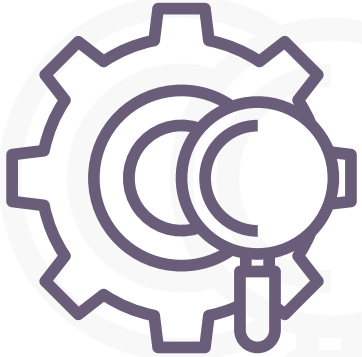
```
> $ sudo apt install openjdk-8-jre
> $ sudo apt install tomcat8
```

Download and Instal ABAC

```
> $ wget
https://repo1.maven.org/maven2/org/ow2/authzforce/authzforce-ce-server-dist/8.1.0/authzforce-ce-server-dist-8.1.0.deb
> $ sudo aptitude install gdebi curl
> $ sudo gdebi authzforce-ce-server-dist-M.m.P.deb
```

Test ABAC and Deploy Policy

```
> $ curl http://localhost:8080/authzforce-ce/?_wadl
> curl -X POST \
http://localhost:8080/authzforce-ce/domains/{domain-id}/pap/policies \ -H 'Content-Type: application/xml' \ -d
'<PolicySet>...etc</PolicySet>'
```



Policy

$\underbrace{\textbf{Doctor}}_{\text{subject}}$ can $\underbrace{\textbf{access}}_{\text{action}}$ $\underbrace{\textbf{patients' smart door lock}}_{\text{object}}$ in $\underbrace{\textbf{emergency state}}_{\text{environmental condition}}$ during $\underbrace{\textbf{working hours}}_{\text{environmental condition}}$

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns5:PolicySet xmlns="http://authzforce.github.io/core/xmlns/pdp/6.0" xmlns:ns2="http://www.w3.org/2005/Atom" xmlns:ns3="http://authzforce.github.io/rest-api-model/xmlns/authz/5"
xmlns:ns4="http://authzforce.github.io/pap-dao-flat-file/xmlns/properties/3.6" xmlns:ns5="urn:oasis:names:tc:xacml:3.0:core:schema:wd-17" PolicySetId="root" Version="1.0"
PolicyCombiningAlgId="urn:oasis:names:tc:xacml:3.0:policy-combining-algorithm:permit-overrides">
    <ns5:Target/>
    <ns5:Policy PolicyId="urn:oasis:names:tc:xacml:3.0:example:MyPolicyNew" Version="1.0" RuleCombiningAlgId="urn:oasis:names:tc:xacml:3.0:rule-combining-algorithm:permit-overrides">
        <ns5:Target/>
        <ns5:Rule RuleId="urn:oasis:names:tc:xacml:3.0:example:MyRule" Effect="Permit">
            <ns5:Target>
                <ns5:AnyOf>
                    <ns5:AllOf>
                        <ns5:Match MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
                            <ns5:AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">doorLock</ns5:AttributeValue>
                            <ns5:AttributeDesignator Category="urn:oasis:names:tc:xacml:3.0:attribute-category:resource" AttributeId="urn:oasis:names:tc:xacml:1.0:resource:resource-id"
                            DataType="http://www.w3.org/2001/XMLSchema#string" MustBePresent="false"/>
                        </ns5:Match>
                    </ns5:AllOf>
                </ns5:AnyOf>
            </ns5:Target>
            <ns5:Condition>
                <ns5:Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:and">
                    <ns5:Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-at-least-one-member-of">
                        <ns5:AttributeDesignator Category="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject" AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
                        DataType="http://www.w3.org/2001/XMLSchema#string" MustBePresent="false"/>
                        <ns5:Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
                            <ns5:AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">doctor</ns5:AttributeValue>
                        </ns5:Apply>
                    </ns5:Apply>
                    <ns5:Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-at-least-one-member-of">
                        <ns5:AttributeDesignator Category="urn:oasis:names:tc:xacml:3.0:attribute-category:action" AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
                        DataType="http://www.w3.org/2001/XMLSchema#string" MustBePresent="false"/>
                        <ns5:Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-bag">
                            <ns5:AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">access</ns5:AttributeValue>
                        </ns5:Apply>
                    </ns5:Apply>
                </ns5:Apply>
            </ns5:Condition>
        </ns5:Rule>
        <ns5:Rule RuleId="Deny-Rule" Effect="Deny"/>
    </ns5:Policy>
</ns5:PolicySet>
```

Prototype Implementation - V

PDP Request Preprocessor Extension

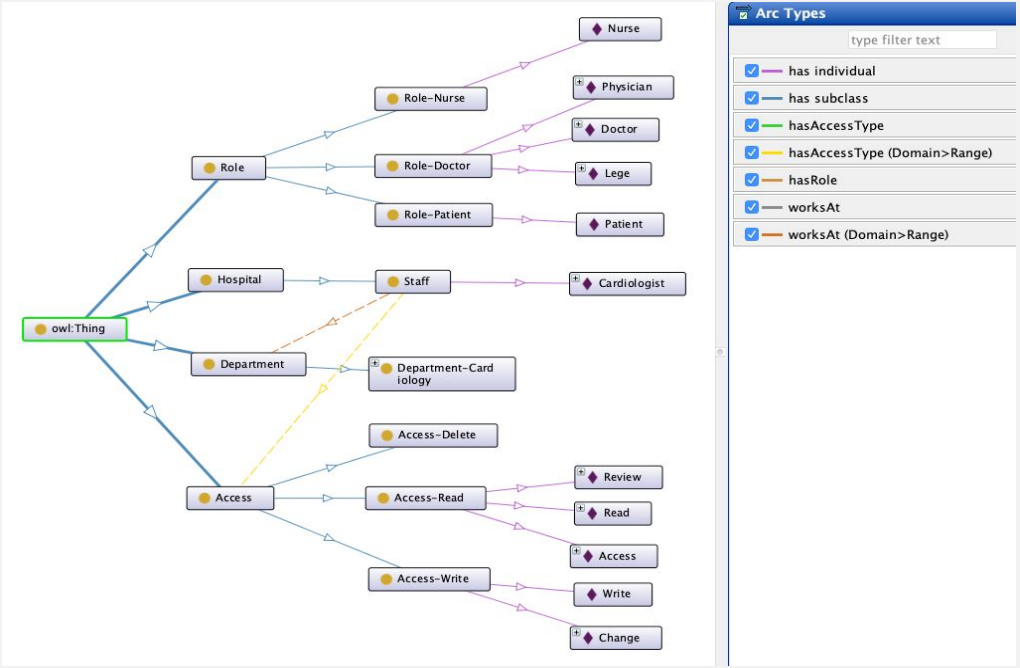
- PDP Extension Plugin to implement non-core PDP behaviour
- Java based plugin, deployed on ABAC engine
- Intercept incoming request before evaluation on PDP
- Modify if necessary and resume the flow
- REST Client to communicate with Semantic Reasoner

Goals:

- Manage integration bw ABAC-Semantic Reasoner
- Manage request modification

Prototype Implementation - III

Hospital Staff Ontology



Prototype Implementation - IV

Semantic Reasoner Tool

- A tool to parse relationships in ontology and extract synonyms
- Java, Spring boot, OWLApi, Hermit Reasoner, deployed on Tomcat web-server
- Tool offers endpoints:
  - /query-by-role/{roleName}
  - /query-by-access-type/{accessTypeName}
  - /query-by-access-type/{accessTypeName}/for/{roleName}
- Subject synonyms: Doctor, Lege, Physician
- Access type synonyms: Access, Read, Review

Goals:

- Integrate with ABAC and offer API
- Parse the ontology and extract knowledge
- Forward back the knowledge

```xml
<Request xmlns="urn:oasis:names:tc:xacml:3.0:core:schema:wd-17"
xmlns:xacml="urn:oasis:names:tc:xacml:3.0:core:schema:wd-17"  xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance"  xmlns:xf="http://www.w3.org/2005/xpath-functions"
xmlns:md="http:www.med.custom.com/schemas/record.xsd" CombinedDecision="false"
ReturnPolicyIdList="false">
   <Attributes Category="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject">
     <Attribute AttributeId="urn:oasis:names:tc:xacml:3.0:custom:attribute:role" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">doctor</AttributeValue>
     </Attribute>
   </Attributes>
   <Attributes Category="urn:oasis:names:tc:xacml:3.0:attribute-category:resource">
     <Attribute AttributeId="urn:oasis:names:tc:xacml:3.0:content-selector" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">doorLock</AttributeValue>
     </Attribute>
   </Attributes>
   <Attributes Category="urn:oasis:names:tc:xacml:3.0:attribute-category:environment" >
     <Attribute AttributeId="urn:oasis:names:tc:xacml:3.0:custom:attribute:status"
     IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">emergency</AttributeValue>
     </Attribute>
   </Attributes>
   <Attributes Category="urn:oasis:names:tc:xacml:3.0:attribute-category:action">
     <Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">read</AttributeValue>
     </Attribute>
   </Attributes>
</Request>
```

```xml
<Request xmlns="urn:oasis:names:tc:xacml:3.0:core:schema:wd-17"
xmlns:xacml="urn:oasis:names:tc:xacml:3.0:core:schema:wd-17"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"  xmlns:xf="http://www.w3.org/2005/xpath-functions"  xmlns:md="http:www.med.custom.com/schemas/
record.xsd" CombinedDecision="false" ReturnPolicyIdList="false">
   <Attributes Category="urn:oasis:names:tc:xacml:1.0:subject-category:access-subject">
     <Attribute AttributeId="urn:oasis:names:tc:xacml:3.0:custom:attribute:role" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">doctor</AttributeValue>
     </Attribute>
     <Attribute AttributeId="urn:oasis:names:tc:xacml:3.0:custom:attribute:role" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">lege</AttributeValue>
     </Attribute>
     <Attribute AttributeId="urn:oasis:names:tc:xacml:3.0:custom:attribute:role" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">physician</AttributeValue>
     </Attribute>
   </Attributes>
   <Attributes Category="urn:oasis:names:tc:xacml:3.0:attribute-category:resource">
     <Attribute AttributeId="urn:oasis:names:tc:xacml:3.0:content-selector" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">doorLock</AttributeValue>
     </Attribute>
   </Attributes>
   <Attributes Category="urn:oasis:names:tc:xacml:3.0:attribute-category:environment" >
     <Attribute AttributeId="urn:oasis:names:tc:xacml:3.0:custom:attribute:status" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">emergency</AttributeValue>
     </Attribute>
   </Attributes>
   <Attributes Category="urn:oasis:names:tc:xacml:3.0:attribute-category:action">
     <Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">read</AttributeValue>
     </Attribute>
     <Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">review</AttributeValue>
     </Attribute>
     <Attribute AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id" IncludeInResult="false">
        <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">access</AttributeValue>
     </Attribute>
   </Attributes>
</Request>
```

Evaluation

1. **Outcome**

   Decentralized S-ABAC with ontology on fog node

1. **Performance**

Load Tests (ms)			
Cloud		RPi-4B	
ABAC	S-ABAC	ABAC	S-ABAC
52.36	1537.68	69.05	1713.14

1. **Cost**

   - RPi Model 4B = 729kr
   - Mini-HDMI Adapter = 95kr
   - **Total = 824kr**

2. **Limitation**

   - Mock requests for object (smart door lock)
   - Basic ontology works perfect on RPi 4B
   - Further tests can be apply on more inclusive ontology

Discussion

Discussion #1

Function Extension vs Request Preprocessor Extension

Discussion #2

Cloud ABAC is the trend due to complexity

Discussion #3

Centralized approach advantages and disadvantages

Discussion #4

Cloud (S-)ABAC vs GDPR

Discussion #5

Decentralized S-ABAC

Future Work

Suggestion #1

Synonym repository or dataset is the key for not having very long policy requests

Suggestion #2

PDP Response Postprocessor Extension for future analytics policy responses.

Suggestion #3

Turn the setup into microservices by using docker container for each component.

Back to Research Questions & Objectives

Objective #1 : ABAC on fog-area

Objective #2 : Semantic-ABAC

Objective #3 : Decentralized Access Control

RQ #1: What is the importance of the fine-grained access control for IoT settings?

RQ #2: Can ABAC work for simple IoT devices on the fog-area?

RQ #3: What is the ideal gateway device for this need?

RQ #4: How can we link an ABAC engine to a Semantic Reasoner having a simple ontology?

Conclusion

S-ABAC is a successor of other access control systems, including ABAC

S-ABAC on fog-area **IS POSSIBLE** without losing performance

Thank you for your attention.

QUESTIONS

THANK YOU !

Semantic - Attribute Based Access Control
Master Thesis

Uğur Bayram - ugurb@ifi.uio.no

University of Oslo

Master of Science (MSc) - 30 ECTS

Supervisors:

Christian Johansen, Hamed Arshad