# Writing Open Source SunXACML Access Control in Electronic Health Record with Acceptable Performances

Snezana Sucurovic[1] and Dejan Simic[2]

[1] Institute Mihajlo Pupin, Volgina 15, 11 000 Belgrade, Serbia

[2] Faculty of Organisational Sciences, Computer Science Department
Jove Ilica 154, 11 000 Belgrade, Serbia

**Abstract.** OASIS is a non–for-profit consortium that drives the development convergence and adoption of open standards for the global information society. It involves more than 600 organizations and individuals as well as IT leaders Sun, Microsoft, IBM and Oracle. One of it's standard is XACML which appears a few years ago and now there are about 150 000 hits on Google. XACML (eXtensible Access Control Markup Language) is not technology related. Sun published in 2004 open source Sun XACML which is in compliance with XACML 1.0. specification and now worked to be in compliance with XACML 2.0. The heart of XACML are attributes values of defined type and name that is to be attached to a subject, a resource, an action and an environment in which subject request action on resource. On that way XACML is to replace Role Based Access Control which dominated for years. The paper examines performances in CEN 13 606 and ISO 22 600 based healthcare system which use XACML for access control.

## 1 Related Work on using XACML

At the RSA 2008 Conference, members of the OASIS open standards consortium, in cooperation with the Health Information Technologies Standards Panel (HITSP), demonstrated interoperability of the Extensible Access Control Markup Language (XACML) version 2.0. Simulating a real world scenario provided by the U.S. Department of Veterans Affairs, the demo showed how XACML ensures successful authorization decision requests and the exchange of authorization policies. The XACML Interop at the RSA 2008 conference utilizes requirements from Health Level Seven (HL7), ASTM International, and the American National Standards Institute (ANSI). The demo features role-based access control (RBAC), privacy protections, structured and functional roles, consent codes, emergency overrides and filtering of sensitive data. Vendors show how XACML obligations can provide capabilities in the policy decision making process. The use of XACML obligations and identity providers using the Security Assertion Markup Language (SAML) are also highlighted. According to the ANSI/HITSP announcement, the multi-vendor demonstrations "highlight the use of OASIS standards in HITSP-approved guidelines, known as

Sucurovic S. and Simic D. (2009).
Writing Open Source SunXACML Access Control in Electronic Health Record with Acceptable Performances .
In *Proceedings of the 1st International Workshop on Open Source in European Health Care: The Time is Ripe*, pages 59-68
DOI: 10.5220/0001813400590068

'constructs,' to meet healthcare security and privacy needs. The Panel's security and privacy specifications address common data protection issues in a broad range of subject areas, including electronic delivery of lab results to a clinician, medication workflow for providers and patients, quality, and consumer empowerment. HITSP is a multi-stakeholder coordinating body designed to provide the process within which affected parties can identify, select, and harmonize standards for communicating health care information throughout the health care spectrum. As mandated by the U.S. Department of Health and Human Services (HHS), the Panel's work supports Use Cases defined by the American Heath Information Community (AHIC)[1].

## 2 Related Work on Access Control Standards

### 2.1 Commite European de Normalisation (CEN) Standard ENV 13606 [2]

In 2008 ISO has given ISO 13 606 on Electronic Healthcare Record Architecture. This standard is revised Comitte European de Normalisation standard CEN ENV 13 606. This standard is base for access control in European standards and developed systems. The main item in the CEN healthcare information system architecture (CEN, 2002) standard is an Architectural Component. The Architectural Components are organized in a hierarchical structure. Each Architectural Component has a reference to access control list for that component defined as Distribution Rules. A Distribution Rule comprises classes Who, Where, When, Why and How which define who, where, when, why and how is allowed to access the component (Table 1). To access the system a user presents his attributes that correspond to attributes of class Who, When, Where, Why and How. Classes Who, When, Where, Why and How are processed with operator AND. There can be one or more DR attached to AC and they are processed with operator OR.

### 2.2 ISO 22 600 Standard [3]

ISO/TS 22600-1:2006 is intended to support the needs of healthcare information sharing across unaffiliated providers of healthcare, healthcare organizations, health insurance companies, their patients, staff members and trading partners. It is also intended to support inquiries from both individuals and application systems. ISO/TS 22600-1:2006 supports collaboration between several authorization managers that may operate over organizational and policy borders.

**Table 1.** Distribution Rules attributes.

| Class | Attribute | Type |
|---|---|---|
| Who | Profession | String |
| | Specialisation | String |
| | Engaged in care | Boolean |
| | Healthcare agent | Class |
| When | Episode of care | String |
| | Episode reference | String |
| Where | Country | String |
| | Legal requirement | Boolean |
| Why | Healthcare process code | String |
| | Healthcare process text | String |
| | Sensitivity class | String |
| | Purpose of use | Class |
| | Healthcare party role | Class |
| How | Access method | String |
| | Consent required | Class |
| | Signed | Boolean |
| | Encrypted | Boolean |
| | Operating system security rating | String |
| | Hardware security rating | String |
| | Software security rating | String |

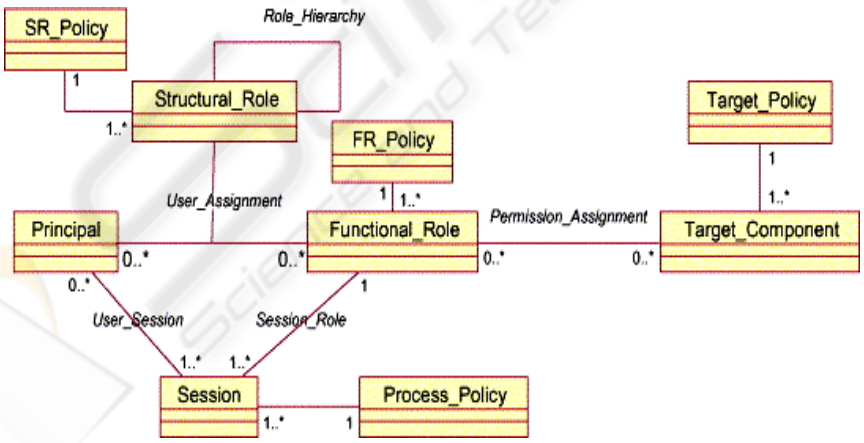


**Fig. 1.** ISO 22 600 access control policies.

# 3 eXtensible Access Control Markup Language (XACML)

XACML is an OASIS [4] standard that describes both a policy language and an access control decision request/response language (both written in XML). The typical setup is that someone wants to take some action on a resource, so the elements of request are user's atttributes, action and resource to be accessed. The request/response language lets you form a query to ask whether or not a given action should be allowed and interpret the result.

The currency that XACML deals in is attributes. Attributes are named values of known types. Specifically, attributes are characteristics of the Subject, Resource, Action or Environment in which the access request is made. A user's name, their security clearance, the file they want to access, and the time of day are all attribute values. When a request is sent to a Policy Decision Point (PDP) (which makes decision), that request is formed almost exclusively of attributes, and they will be compared to attribute values in a policy to make the access decision.

## 3.1 SunXACML

This paper presents work with Sun's version of XACML [5]. Sun's XACML has supported only XACML 1.0 and not XACML 2.0 yet. XACML 2.0 supports Security Access Control Markup Language (SAML) protocol which is enveloping protocol and provides standardized protocol for encryption and decryption of attributes and policies on the net.

Since Sun XACML is in developing phase (toward version XACML 2.0) in order to work with it, it is necessary to download form Code Version System (CVS) ( from sourceforge.net) the last version of java classes and then compile them using given build.xml file. Finally it is necessary to build jar archive using Apache's Ant tool. In that way, we can obtain the most stable version of Sun's XACML, since code for version 2.0 is added in this phase of development.

## 3.2 Writing Access Control Policies using XACML

A policy is a Policy Set which comprises one or more policies, where a policy comprises one or more rules. There are also combining algorithms for policies and rules. For example if the combining algorithm is "ordered-permit-overrides" the firstly rules that permit access have advantage. In that case the last rule is

<Rule RuleId="FinaleRule" Effect="Deny"/>

that means that if there is not previously written permit rule access is forbidden.

There is also another case when we write as the last rule

<Rule RuleId="FinaleRule" Effect="Permit"/>

Besides Rules, a Policy comprises a Target which is necessary to find correspondent policy to the given request. A target comprises <Subject> attribute, <Resource> attribute and <Action> Attributes, where subject and resource are mandatory and action can be <AnyAction/>. Subject attribute has to be of datatype RFC 822 (for

example users.example.com), and resource attribute has to be of type <anyURI>(for example http://server.example.com).

It can be several Subject and Resource attributes in Request (besides ones that are part of Target in Policy). An example of XML segment which represents Target Resource attribute has been given as follows:

```
<Resource>
  <Resource Match MatchId=anyURI-equal>
    <AttributeValue
DataType=anyURI>http://server.example.com<AttributeValue/>
    <ResourceAttributeDesignator
DataType=anyURI
AttributeID=resource_id>
  <ResourceMatch>
<Resource>
```

It means that matching evaluation operator is anyURI-equal, that attribute is of type anyURI and that its value is http://server.example.com . Also, name of the attribute is resource_id, that is how it is compared with attributes from Request.

In a Policy a Target is followed by one or several Rules. While a target is simplified condition for access decision, the heart of most Rules is a Condition which is mostly boolean or set function. If the Condition evaluates to true, then the Rule's effect (a value of Permit or Deny that is associated with successful evaluation of the Rule) is returned. A Condition can be quite complex, built from an arbitrary nesting of non-boolean functions and attributes. The following XML segment presents a complex Condition:

```
<Condition FunctionId=string-at-least-one-member-of>
  <SubjectAttributeDesignator
    DataType=string
    AttributeId=group />
  <Apply FunctionId=string-bag>
    <AttributeValue
      DataType=string>GP<AttributeValue/>
    <AttributeValue
      DataType=string>SCP<AttributeValue/>
  </Apply>
</Condition>
```

It means that this is attribute of subject with name *group* and type *string*. Set function string-at-least-one-member-of means «at least one member of set whose elements are of type string». This segment means that attribute of subject with name group has to have at least one of given values (GP, SCP) in oder to access decision evaluates to Permit.

# 4 Performance Examination

## 4.1 Related Work on Hierarchy Access Control Policies

ARTEMIS is an EU funded project that are developing a semantic web service based on P2P interoperatibility infrastructure for healthcare information systems [6]. An ARTEMIS pilot application is being clinically deployed by healthcare providers located in two European countries to demonstrate the interoperatibility of healthcare information systems across organizational and country boundaries. The pilot application includes healthcare providers South East Belfast Healthcare Trust (SEBT) in Belfast, Northern Ireland and Hacettepe Hospital in Ankara, Turkey. In ARTEMIS, healthcare providers define privacy policies that state which healthcare professionals are able to access specific medical data. The initial approach is to allow a healthcare provider to develop role ontology that defines the clinical occupations for healthcare professionals within their organization. These roles are then attached to concepts in the clinical concept ontology. As medical data is described using the clinical concepts, authorization is enforced based on the role of the healthcare professional and the clinical concept being accessed. The Clinical Concept Ontology has been derived from the Unified Medical Language System (UMLS) semantic network [7] and metathesaurus [8] to provide a rich set of terminology for describing medical data semantics. The Semantic Network consists of a set of broad subject categories, or Semantic Types, that provide a consistent categorization of all concepts represented in the UMLS Metathesaurus, and a set of useful and important relationships, or Semantic Relations, that exist between Semantic Types. The Metathesaurus is a very large, multi-purpose, and multi-lingual vocabulary database that contains information about biomedical and health related concepts, their various names, and the relationships among them. The Metathesaurus is organized by concept or meaning. In essence, it links alternative names and views of the same concept and identifies useful relationships between different concepts. There are efforts to implement mechanisms for parsing free text into UMLS concepts [12].

Event
- Activity
  - Behavior
    - Social Behavior
    - Individual Behavior
  - Daily or Recreational Activity
    - Occupational Activity
      - Healthcare Activity
        - Laboratory Procedure
        - Diagnostic Procedure
        - Therapeutic or Preventive Procedure
      - Research Activity
        - Molecular Biology Research Technique
      - Governmental or Regulatory Activity
    - Educational Activity
  - Machine Activity

The ARTEMIS mediator can broker between privacy policies using ontology mappings linking organisational role ontologies with clinical concept ontologies. When a web service is invoked the mediator translates the role of the healthcare professional in the requesting organisation to the equivalent role in the providing organisation.

## 5 MEDIS (Medical Information System) Prototype Approach

MEDIS [9] prototype has been developed as EHR when integration from very beginning is needed. It has been implemented as a federated system.

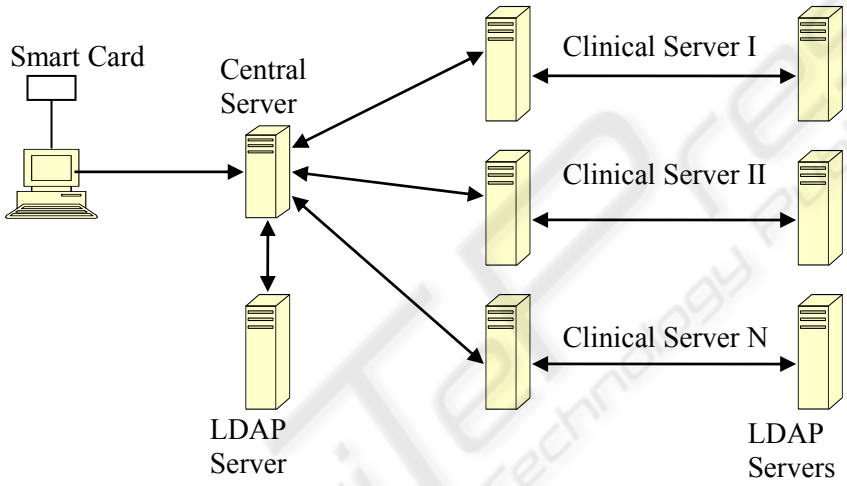


**Fig. 2.** MEDIS System architecture.

As MEDIS is multidomain system XACML should be applied for access control. There are master access control policies on central server and local policies on clinical servers. Architectural Components could be seen as resources or target and there are access control policies attached to them according to ISO 22 600. There are also subject attributes enveloped in attribute certificates [10]. We did step further: we supposed that attributes in policies are hierarchically organized (as in ARTEMIS) and measured performances.

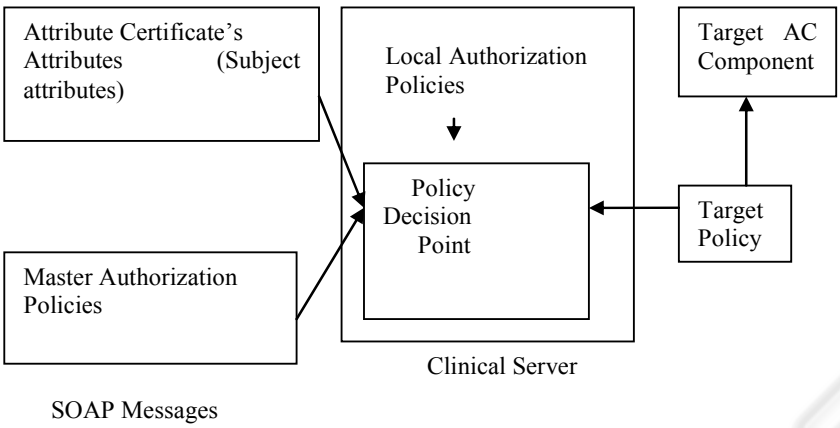


**Fig. 3.** Access control policies in MEDIS.

### 5.1 Performance Examination

Fig 4. presents forest attribute hierarchy, while Fig. 5 presents tree attribute hierarchy. Results have been obtained on 512 M RAM-a and 800 MHz CPU. Node 1 presents attribute which is possessed by subject, while node 2 represents attribute which is attached to resource. For all combinations of node attaced to subject and node attached to resource average access time in forest structure is 1702 ms, while Tmin = 1632ms and Tmax = 1783ms. In tree like access control policy Taverage=1698ms, while Tmin= 1622ms and Tmax=1942ms.

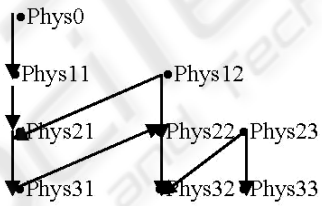


**Fig. 4.** Forest attribute hierarchy.

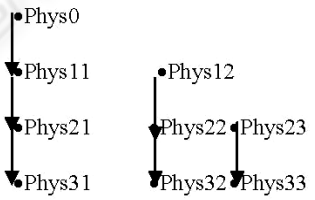


**Fig. 5.** Tree attribute hierarchy.

Table 2 contains access control time when two attributes (Healthcare Provider and Episode of Care) are involved. Node 3 is episode of care attached to subject, while Node4 is attribute attached to resource according to the policy represented at following figure.

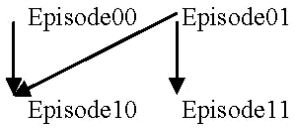


**Fig. 6.** Episode of care access control policy.

**Table 2.** Access Control time when two attributes policy is used.

| Node1 | Node2 | Node3/Node4 | Access time |
|---|---|---|---|
| 00 | 11 | 0010/0011/01 10/0111 | 1903/1792/1712/1823 |
| 00 | 21 | | 1732/1703/1812/1742 |
| 00 | 31 | | 1802/1712/1773/1762 |
| 00 | 12 | | 1782/1793/1782/1722 |
| 00 | 22 | | 1852/1732/1872/1723 |
| 00 | 23 | | 1723/1732/1722/1722 |
| 00 | 33 | | 1762/1682/1732/1712 |
| 00 | 32 | | 1713/1742/1693/1813 |
| 11 | 21 | | 1703/1742/1783/1713 |
| 11 | 31 | | 1793/1722/1723/1712 |
| 11 | 12 | | 1742/1722/1773/1742 |
| 11 | 22 | | 1793/1712/1722/1692 |
| 11 | 32 | | 1733/1782/1742/1773 |
| 11 | 23 | | 1793/1702/1822/1792 |
| 11 | 33 | | 1762/1742/1753/1793 |
| 21 | 31 | | 1712/1783/1783/1712 |
| 21 | 12 | | 1682/1783/1813/1692 |
| 21 | 22 | | 1672/1702/1793/1733 |
| 21 | 32 | | 1723/1712/1683/1702 |
| 21 | 23 | | 1772/1713/1713/1772 |
| 21 | 33 | | 1722/1792/1782/1722 |
| 31 | 12 | | 1782/1752/1672/1673 |
| 31 | 22 | | 1742/1763/1743/1723 |
| 31 | 32 | | 1733/1762/1682/1702 |
| 31 | 23 | | 1672/1772/1772/1762 |
| 31 | 33 | | 1782/1772/1723/1792 |
| 12 | 22 | | 1782/1792/1732/1733 |
| 12 | 32 | | 1723/1712/1783/1682 |
| 12 | 23 | | 1762/1753/1682/1722 |
| 12 | 33 | | 1723/1692/1722/1763 |
| 22 | 32 | | 1692/1782/1702/1753 |
| 22 | 23 | | 1693/1683/1793/1792 |
| 22 | 33 | | 1702/1662/1742/1732 |
| 32 | 23 | | 1712/1762/1713/1832 |
| 32 | 33 | | 1752/1702/1732/1702 |
| 23 | 33 | | 1722/1750/1742/1752 |
| | | | Taverage = 1742 |

## 6 Conclusions

In it's first paragraph this paper presents current work on use of OASIS's XACML in healthcare, while in the second current standards related to access control in EHR. The third paragraph deals with writing policies using XACML, while the forth presents a current project dealing with hierarchical attributes in access control policies. Finally, the fifth paragraph presents our approach, which includes XACML policies, current standards and hierarchical attributes. The paper aims at giving contribution to presentation of results of access control time using various SunXACML access control policies with hierarchical attributes. This results presents a fact that either we choose tree or forest architecture or if we write policy with several attributes in hierarchy, performances will not fall down.

## References

1. Bussiness Wire, accessed May 2008, http://www.allbusiness.com/technology/software-services-applications-information/8943546-1.html
2. CEN ENV 13 606 Extended Architecture, URL: http://www.centc251.org/WGI/N-documents/WGI-N04-24-prEN_13606-1_(E)preENQ.pdf, accessed June 2007.
3. ISO 22 600 "Access Control in Healthcare Information Systems"
4. Anderson A., A comparasion of Two Privacy Policy Languages: EPAL and XACML, http://research.sun.com/techrep/2005/smli_tr2005-147.pdf
5. Sun's XACML Implementation, http://sunxacml.sourceforge.net/
6. Artemis Project, http://www.srdc.metu.edu.tr/webpage/projects/
7. National Library of Medicine, Unified Medical Language System, Semantic Network, http://www.nlm.nih.gov/research/umls/meta3.html
8. National Library of Medicine, Unified Medical Language System, Metathesaurus, http://www.nlm.nih.gov/research/umls/meta2.html
9. S. Sucurovic, "Implementing security in a distributed web based EHCR", International Journal of Medical Informatics, May 2007, pp. 491-496, Elsevier
10. S. Sucurovic, Z. Jovanovic, Java Cryptography & Attribute Certificate Management, Dr. Dobb's Journal, Octobar 2006