A Look at XACML

Example

Policy Example

Assume that a corporation named Medi Corp (medico.com) has an access control policy that states: Any user with an e-mail address in the "medico.com" namespace is allowed to perform any action on any resource. In XACML, this policy is expressed as follows:

```
[p01]     <?xml version=1.0" encoding="UTF-8"?>
[p02]     <Policy xmlns="urn:oasis:names:tc:xacml:1.0:policy"
[p03]     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
[p04]     xsi:schemaLocation="urn:oasis:names:tc:xacml:1.0:policy
[p05]     http://www.oasis-open.org/tc/xacml/1.0/cs-xacml-schema-policy-01.xsd"
[p06]     PolicyId="identifier:example:SimplePolicy1"
[p07]     RuleCombiningAlgId="identifier:rule-combining-algorithm:deny-overrides">
     The header for the policy is in lines [p01]-[p07].
     The rule-combining algorithm used in the policy is specified in
     [p07].  This particular one provides that a "Deny" overides
     any "Permit".
[p08]     <Description>
[p09]     Medi Corp access control policy
[p10]     </Description>
[p11]     <Target>
[p12]         <Subjects>
[p13]             <AnySubject/>
[p14]         </Subjects>
[p15]         <Resources>
[p16]             <AnyResource/>
[p17]         </Resources>
[p18]         <Actions>
[p19]             <AnyAction/>
[p20]         </Actions>
[p21]     </Target>
     [p11]-[p21] describe the decision requests to which the policy applies.  In
     this case the policy is applicable to any decision request.  Remember,
     here we are defining that the policy applies to any request.
     Later we will be looking at targets related to a rule.
[p22]     <Rule
[p23]         RuleId= "urn:oasis:names:tc:xacml:1.0:example:SimpleRule1"
[p24]         Effect="Permit">
     This begins the definition of the only Rule. [p23] assigns a name to this
     rule.  Each rule name must be unique within the PDP.
[p25]         <Description>
[p26]             Any subject with an e-mail name in the medico.com domain
[p27]             can perform any action on any resource.
[p28]         </Description>
[p29]         <Target>
     Now we are defining the target of the rule.
[p30]             <Subjects>
[p31]                 <Subject>
[p32]                     <SubjectMatch MatchId="
              urn:oasis:names:tc:xacml:1.0:function:rfc822Name-match">
[p33]                         <SubjectAttributeDesignator
[p34]      AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
[p35]      DataType="urn:oasis:names:tc:xacml:1.0:data-type:rfc822Name"/>
[p36]                             <AttributeValue
[p37]      DataType="urn:oasis:names:tc:xacml:1.0:datatype:
               rfc822Name">medico.com
[p38]                             </AttributeValue>
[p39]                       </SubjectMatch>
[p40]                  </Subject>
[p41]              </Subjects>
[p42]              <Resources>
[p43]                  <AnyResource/>
[p44]              </Resources>
[p45]              <Actions>
[p46]                  <AnyAction/>
[p47]              </Actions>
[p48]         </Target>
     The target of the policy itself was <AnySubject>, but here we specify
     a specific value the subject in the decision request must match.  In
     [p32] we see "rfc822Name-match" and this refers to a
     function within the standard (line 4341) that handles the comparison.
[p49]     </Rule>
[p50] </xacml:Policy>
     The rule is now closed.  In this simple example, all the work was done
```

```
in the target of the rule.  This is not always the case in more complex
rules.
```