White Paper:

XACML Policy Enforcement System

Identity Management and XML Directory Services Solution

Background

The ViewDS Directory Server's development efforts began in 1987 as an implementation of the X.500 Directory standards. Support for the X.500 standards provided the Directory with fundamental service capabilities, including:

- Hierarchical Database and Information Model
- Ability to support replicated and distributed modes of operation
- High performance searching with approximate matching capabilities
- Security capabilities includes PKI based authentication and Fine Grained Access Control

The Directory has progressed beyond the X.500 standards to adapt to the moving trends of the computing industry. Due to the rapid uptake of the internet, the Directory provided support for LDAP, the Lightweight Directory Access Protocol. This provided the directory with the ability to be used within environments that made use of Internet based technology such as CP/IP.

LDAP also reduced the level of complexity required from client applications, by simplifying the way in which information was represented. Boasting support for both X.500 and LDAP allowed ViewDS to offer a well-adapted directory service that leveraged its sophisticated searching capabilities and fundamental underlying X.500 architecture.

The adoption of XML prompted ViewDS to include support for the XML Enabled Directory (XED) specifications. Considering that ViewDS Directory Server could provide efficient data storage, search and retrieval capabilities, it made sense to extend these capabilities to include XML.

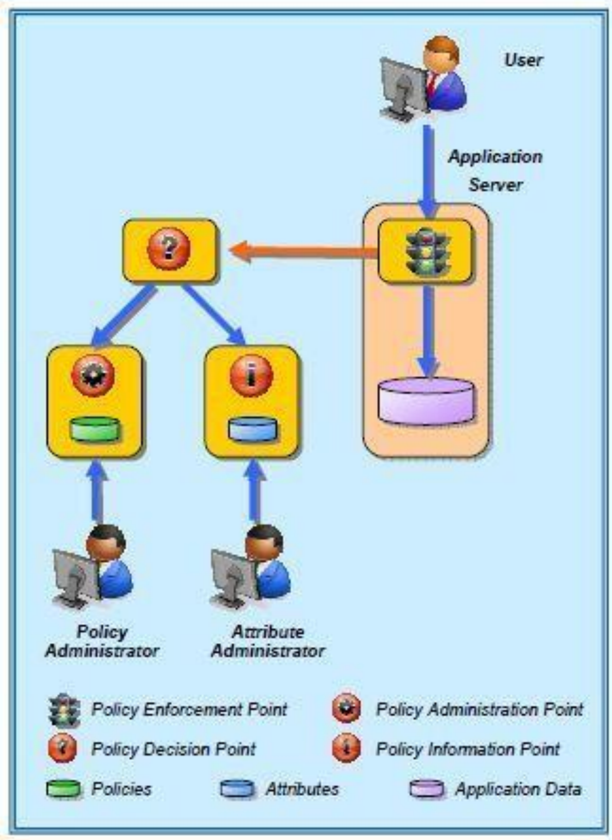
Support for XED allows ViewDS to:

- Extend its supported syntaxes at runtime to include support for new XML Schema to allow it to validate, store and search XML data
- Search XML data by querying inner components / elements of an XML Document
- Provide services over SOAP / HTTP
- Provide an XML based directory access protocol that is not only XML based, but offers support for XML directory data

Support for XML within ViewDS (on top of its support for X.500 and LDAP) allows it to be leveraged by applications that are XML based or need a repository for XML information.

How the XACML Policy Enforcement System Works

The eXtensible Access Control Markup Language provides a standardized way to unify access control policy across a variety of enterprise applications by externalizing access control decisions. The figure on below illustrates the main components of the XACML model. The PEP (Policy Enforcement Point) allows or disallows access requests based on authorization policy. The PEP however does not make the decision itself though and will defer this to the Policy Decision Point (PDP). The PDP makes authorization decisions based on XACML policies.

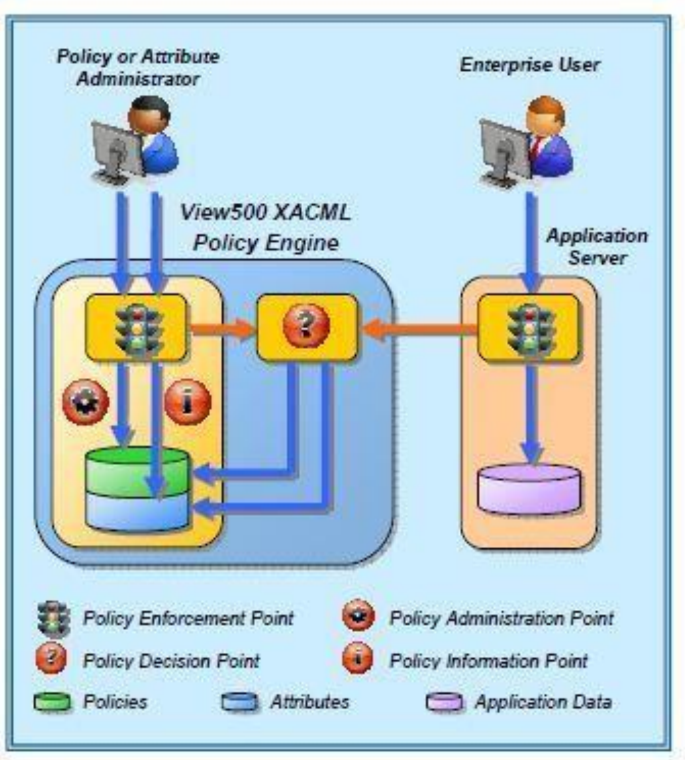


These policies are held and managed by a Policy Administration Point (PAP). The PDP requests the XACML policy documents that it needs from the PAP. A PDP may need additional data in order to decide whether to grant or deny a request for access, specifically, properties of the subjects (users) requesting access and properties of the resources to which access is being requested. These properties are called attributes in the XACML model and are assumed to be held by a Policy Information Point (PIP).

While the depicted model can certainly provide the functionality of an XACML policy enforcement system, there are architectural aspects that make the system less efficient and impose additional overheads.

- The PDP will need to communicate to the PAP and the PIP. The PAP/PIP will need to control the PDP's access to their data, which requires access control mechanisms that are external to the XACML system itself. This means that in order to provide a centralized access control, the system will introduce additional access control points, which is counter to the original intent of the system.
- A consequence of the PAP and PIP being separate systems means that the policy needs to be maintained in a separate system from the enterprise's existing identity data (which is held in the PIP). A combined PIP and PAP would allow easier information access and maintenance.
- The usage of discrete PEP, PDP, PIP and PAP components requires secure communication between each of them to uphold the integrity of the system.

ViewDS XACML Policy Enforcement



These simplifications, along with the elimination of external message passing between the PDP, PAP and PIP, will significantly improve performance. The ViewDS XACML Policy Engine will reduce the administrative overhead of managing the policy enforcement because there are fewer discrete components to configure.

The ViewDS XACML Policy Engine is able to fully realize the goal of unified authorization policy because requests for access to policies and other attributes are handled by the same PDP that handles requests for access to other enterprise data, and the policy that the PDP uses can express authorization policy for accessing the policy itself. The other attributes include the user identity data that the underlying ViewDS Directory Server stores when being used as an identity management identity store. This data, as well as all the access protocols (e.g., LDAP) that the Directory Server provides, are brought under the control of the unified enterprise policy.

With all access requests passing through the ViewDS XACML Policy Engine, it is possible for to provide comprehensive access logging for auditing and compliance reporting.

Developments to allow the directory to offer a robust and efficient XACML Policy Server are made possible by leveraging the underlying Directory and XML capabilities