|  |  |
| --- | --- |
| Plagiarism Scan Report | |
|  | |
| **Summary** | |
| Report Genrated Date | 25 Jun, 2018 |
| Plagiarism Status | **91% Unique** |
| Total Words | 976 |
| Total Characters | 6808 |
| Any Ignore Url Used |  |

**Content Checked For Plagiarism:**

Consider a continuation of the online digital library example in which the library oﬀers a collaborative subscription to a group of small universities that cannot aﬀord an exclusive membership. In such a scenario, although access is available to students from the group of smaller subscribing universities, this access may be restricted based on either prespeciﬁed access slots or the access duration for each university that shares the infrastructure. The restriction arises from a need to achieve a balanced system load. Maintaining such context information, and constantly updating it, can be a challenging task in distributed environments, especially those with mobile users under- going domain transfers due to reasons such as handoﬀ. Correctly restoring valid connections becomes critical, and it is an issue our software architecture addresses.

Our framework allows content-based speciﬁcation at four levels: conceptual, schema, instance, and element. Grouping information content into concept clusters reduces the complexity of the speciﬁcation process and security administration.This approach uses a similarity-based function for content classiﬁcation. The similarity-based function analyzes the content-related meta information or schema information available in XML documents, then groups related XML schemas and their instances into a cluster. The classiﬁcation creates document clusters and assigns roles related to the concept to these clusters. The classiﬁcation process can organize such roles as a hierarchy that satisﬁes the aggregate relation.

As Figure 1 shows, a cluster can contain an arbitrary number of XML schemas, XML instances, or their elements and attributes. Once the classiﬁcation process has created document clusters, the system administrator generally speciﬁes additional ﬁne-grained access restrictions within valid XML document instances. Our approach, however, assumes that the administrator has not speciﬁed any negative permissions. Thus, once an administrator at a higher level has granted access, there is no need for an overriding policy speciﬁcation at a lower level. If a user with new credentials needs a predetermined role, the system might need to create a virtual cluster dynamically based on the new credential information.Our XML-based speciﬁcation language models the RBAC elements and incorporates the functional speciﬁcations according to RBAC standard.

Figure 1. XML document clustering and associated roles. (a) The schema and all of its XML document instances fall under the cluster; (b) the XML document falls under the cluster;(c) the XML document element falls under the cluster.

V. SOFTWARE ARCHITECTURE

The below ﬁgure, Figure 2 depicts a proposed software architecture for a single-enterprise

Web-service-enabled application that disseminates secure documents. The proposed architecture meets all the RBAC functional speciﬁcations of the RBAC standard.

1. Document Composition Module

The XML document composition module (XDCM) provides the main graphical interface for composing XML schemas for RBAC elements and policy administration documents. The same inter- face composes both sets of documents, which the policy base stores. This module provides all the administrative functions as part of RBAC functional speciﬁcations.

1. Access Control Module

The access control module (ACM), the architecture’s key component, interfaces with various other functional modules and information repositories to extract relevant information while making authorization decisions. The ACM extracts the policy information from the policy base and works closely with the XML instance generator (XIG) module to enforce authorization constraints.The XIG module gets information from the ACM about the access permissions allowed on XML documents associated with an access request and generates XML views accordingly in response to that request. The XML instance base (XIB) caches these views. XIG can simply be an extension of an XML document processor. Along with the session management module (SMM), the ACM manages the supporting system functions listed in the NIST RBAC functional speciﬁcations.

1. Session Management Module

The SMM monitors session activities, capturing relevant, dynamic context information that updates user credentials and thus might aﬀect future access control decisions. The system maintains this information in an XML session sheet (XSS) and communicates it to the ACM. The ACM then updates the user credential information in the policy base. ACM, XIG, and SMM together form the XML access control processor.SMM’s ﬂexible session management capability is particularly signiﬁcant.

1. Document Classiﬁcation/Clustering Module

The document classiﬁcation/clustering module (DCM) manages classiﬁcation and clustering of all documents. It organizes the concept clusters hierarchically. The role mapper associates roles with concepts and generates the XRSs for these roles and their hierarchy. This module provides functionality to add or delete clusters, as well as to create virtual clusters based on a new set of user credentials.Additionally, this module also handles the classiﬁcation of new documents entering the source. The module can assign a new document to an existing cluster based on its conformance to the schemas that compose the cluster. Previous work proposed similar approaches for document classiﬁcation.

1. Credential Evaluator Module

The credential evaluator module (CEM) evaluates the credentials the ACM presents. It also assigns the user to an existing credential type or creates a new credential type if the user credentials do not match any existing credential speciﬁcation. With the help of the role mapper, the CEM maps the credentials to a role using the assigned credential type. The context extractor evaluates the con- textual information the ACM provides and sends back relevant information for access decision after consulting the policy base.

Figure 2. Framework for a single enterprise Webservice-enabled application that disseminates secure documents. The numbered arrows show the steps in the authorization process.

1. Repositories

The referenced object base constitutes the physical objects present in the system from which the system administrator composes the XML documents. The XML schemas and instances contain actual XML sources to which the user will be requesting access. The XML

policy base contains all policy-related XML documents that XDCM composes. The system can retrieve the information content necessary for all review functions, as stipulated by the RBAC functional speciﬁcations, from the policy base, with support from SMM and role hierarchy components as necessary.The numbered arrows in Figure 2 ,show the steps involved in the authorization process.

Report generated by [smallseotools.com](http://smallseotools.com/)