

MC^{SE}: A Multimedia Context-based Security Engine

Bechara AL BOUNA

LE2I Laboratory UMR-CNRS
University of Bourgogne
21078 Dijon Cedex
France

bechara.albouna@u-bourgogne.fr

Richard CHBEIR

LE2I Laboratory UMR-CNRS
University of Bourgogne
21078 Dijon Cedex
France

richard.chbeir@u-bourgogne.fr

ABSTRACT

In this paper, we describe our Multimedia Context based Security Engine (MC^{SE}) which is a Java Based Prototype able to integrate multimedia context in order to enforce access control policies. The prototype provides supervised access to a database containing sensitive viral images.

Categories and Subject Descriptors

K.6.5 [Management of Computing and Information Systems]: Security and Protection – *Authentication, Insurance, Invasive software (e.g., viruses, worms, Trojan horses), Physical security, Unauthorized access (e.g., hacking, phreaking).*

H.2.0 [Database Management]: General – *Security, integrity, and protection.*

General Terms

Security.

Keywords

Access control – Conditions - Context– Multimedia Objects

1. INTRODUCTION

The increasing advances in information systems make the process of securing data a serious issue to consider effectively. In several application domains such as healthcare departments, insurance companies, governmental departments, and many others, information systems are typically vital and hence require enforced authentication and access control models. The work done so far by the security research community to address these requirements has brought the definition of a number of access control models, including DAC [8], MAC [8], RBAC (Role Based Access Control Model) [14, 15] and others [11, 12, 20, 23]. These models deal with subjects on which the access is controlled in different ways. To facilitate associating authorization and access policies to users, several access models suggest assembling users into groups (administrators, managers, etc.) and providing links between them. On the other hand, models such as [4, 7, 21, 22, 25, 26, 27] have incorporated context as key issue for authorization control to integrate additional information aiming at guarantying safe access and prevent information disclosure. In essence, the context varies

on the basis of information describing users and their surroundings. It is effectively related to the environment characterizing for instance the spot and the location in which the user access is to be controlled. Contextual information has been largely discussed in the literature and can be used to enforce access control across heterogeneous domains. On the other hand, multimedia objects describing users and related context (user surrounding snapshot, her moves and gesture, etc.) reveal interesting information to be studied while accessing data. These multimedia-based information and descriptions are of complementary importance to the textual-based ones and should be considered to enforce the protection of information systems.

Let us consider a pharmaceutical research laboratory with 2 different departments: an information department which has critical data (concerning viral bacteria) and thus requires a high level of security, while the other contains an experiment section which can be dangerous if employees enter without appropriate protection and consequently requires some sort of preventive action. The laboratory employs the RBAC model to easily manage user access for the hundreds of employees. It is also equipped with surveillance cameras installed on each access door and in the departments. Lab staff is divided into several groups depending on the role they occupy and the department they belong to, for instance:

- *Lab technician* handling bio samples must wear white suits and grey caps. They have access to storing areas and cold rooms.
- *Researchers* responsible for testing viruses must wear yellow suits with intoxication masks. They are allowed to enter testing areas and cold rooms.

In order to enforce the dress code and to facilitate the management of users regarding the roles they occupy as well as access permissions based on identity, it would be interesting for the *authorization manager* to define roles on the basis of multimedia objects (along with traditional credentials) and to associate appropriate permissions to any combination of multimedia objects and credentials. Furthermore, the mainframe of the archival department gives access to a database of images representing critical viral bacteria information. Only authorized personnel having the role “Researchers” are able to access the room and to use the mainframe to browse the critical images. Nevertheless, a researcher Bob with the same role is suspected to be inferring information to an outside source. The *authorization manager* does not have the right to prevent Bob from accessing the mainframe (in purpose of fully completing the investigations and revealing related researchers). He should deny him from accessing and viewing the newly updated sensitive images and

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. EDBT'08, March 25-30, 2008, Nantes, France. Copyright 2008 ACM 978-1-59593-926-5/08/0003...\$5.00.

protect information disclosure. Moreover, he should deny whoever is working on the mainframe, in the presence of Bob, from accessing new sensitive images without making Bob suspicious.

To the best of our knowledge, none has considered the issue of supervising user's context using multimedia objects in order to enforce access control decision. This uncovers the need of a full-fledged easy-to-manage access control language able to handle information disclosure and deny unauthorized access. This access control language needs to be flexible enough in order to integrate and analyse multimedia objects describing users' context.

In this demonstration paper, we present MC^{SE} : *Multimedia Context based Security Engine*, a Java based prototype designed to control user access. MC^{SE} is a prototype in which we integrate multimedia objects into an access control system in order to provide flexibility to subjects' definition and enforced access control models. Our prototype facilitates the management of access control policies and enforces these policies with complex and multi-criteria conditions. In this paper, policies are evaluated in an Browser Engine where the system browses images based on authenticated users' environment.

2. PRELIMINARIES

Here we present several definitions needed to understand MC^{SE} 's functional specification.

Definition 1 - Multimedia Function (f): is used to handle the comparison¹ and feature extraction between a set of predefined multimedia objects and another set of input multimedia objects. Numerous types of multimedia functions are provided in several commercial tools and in the literature through various forms. For instance, several are provided in DBMSs SQL-operators such as Oracle and DB2 [9, 24], while others are accessible *via* API functions [13, 19] and web services [3] for multimedia data processing.

Definition 2 – Filter Function (μ): is used to aggregate a set of values (or facts) in order to select or compute one relevant value to facilitate decision-making. The importance of filter functions is security-dependant due to the use of similarity functions that may lead to uncertain decisions, potentially granting access to unauthorized users or denying it to legitimate ones. In many applications, the risk of false positives (likewise false negatives) is fully acceptable. In the scenario describe above, a (partial) violation of the dress code (i.e. because someone wears a white rather than a grey cap), would never lead to a serious security breach. In other cases where security breaches are crucial, rules can be used in conjunction with ordinary credentials to increase access verification [5, 6]. A filter function can be defined by any probabilistic function such as the combination rule of *Dempster and Shafer theory of evidence (DS)* [2, 17], *Bayesian Decision theory* [10], the average, the minimum, the maximum, and so on.

Definition 3 - Multimedia Predicate (P): allows the authorization manager to handle the analysis of a set of inputs using multimedia functions and a filter function. It is 'valid' *iff* the

¹ When handling rules which conditions include multimedia objects, traditional logical operators such as 'equality', 'greater than' or others are not applicable due to the complex structure of multimedia objects and must be extended with *similarity functions*.

result returned by a filter function is satisfied within the Boolean operator (AND, OR, etc.) and the overall threshold defined by the authorization manager.

Definition 4 – Provisional Action (pa): has been effectively used to grant the authorization manager more flexibility when specifying authorization permissions. A provisional action represents a set of predefined actions assigned to a multimedia predicate. Such actions are only executed when the corresponding multimedia predicate is considered 'invalid'.

Definition 5 – Multimedia Identifier (M_{id}): allows assigning users to a set of predefined multimedia-based clusters using multimedia objects that identify them in order to facilitate their permissions management. This is similar to traditional approaches (particularly RBAC) where users can be assigned to a set of groups or roles using their characteristics (most often their job functions). A user, assigned to a specified *Multimedia Identifier*, activates it when the objects identifying the user satisfy its multimedia predicate.

Definition 6 - Multimedia Context Condition (Mcc): represents a set of conditions related to the surrounding environment of a given user (or else) and described using multimedia criteria. An Mcc is considered valid *iff* the multimedia objects describing the user's environment satisfy its multimedia predicate.

Definition 7 – Permissions: are used to grant the possibility to perform an action upon a specified object. In our case, permissions are assigned to predefined Multimedia Identifiers, and are granted to whoever is making the request when the condition attached to it is considered 'valid'.

In the following we give a brief overview of MC^{SE} 's Architecture;

3. MC^{SE} : MULTIMEDIA CONTEXT BASED SECURITY ENGINE ARCHITECTURE

MC^{SE} provides the authorization manager the possibility to easily integrate multimedia context (as represented in the scenario) using an access control language, in order to enforce decision making. In essence, images to which access should be controlled are stored in a database along with an identifier and a textual description. The architecture described in **Figure 1** is divided into 2 different components:

1. Administration Toolkit: contains the different administration components used to create Multimedia Identifiers, Multimedia Context Conditions and Permissions.
2. User Toolkit: contains the Browser application where users are able to retrieve and browse stored images.

3.1 Administration Toolkit

Here, an authorization manager creates a set of predefined Multimedia Identifiers to which he assigns the corresponding permissions. In essence, each defined permission grants users (those who are able to activate related Multimedia Identifiers) the possibility to activate an action upon a set of images stored in the Images DB *iff* associated Multimedia Context Conditions are valid. Once created, Multimedia Identifiers, Multimedia Context Conditions, and Permissions are exported to XML documents and stored in a local XML Repository (Figure 2). In the following subsections we briefly present the MC^{SE} 's administration components.

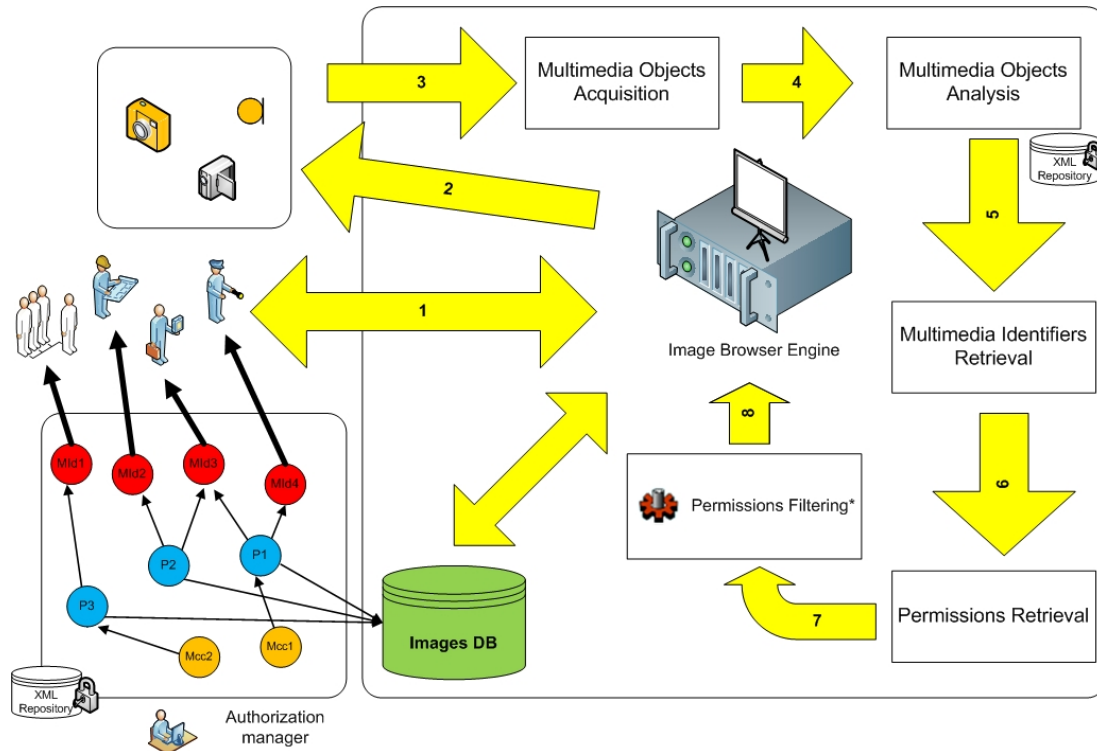


Figure 1: MC^{SE} Architecture

3.1.1 Multimedia Identifier Specification Component

The Multimedia Identifier Specification Component is used to define Multimedia Identifiers. It is divided into two different parts; one for specifying the simple characteristics of *Multimedia Identifiers*, formally defined as $(M_{id}, \text{Description}, \text{and Status})$. The other is used to define related multimedia predicates. The *authorization manager* defines the description and the status of the *Multimedia Identifier* to be specified. Thereafter, he uses the grid shown in the main interface (see Figure 3) to add multimedia predicates. For these multimedia predicates, the *authorization manager* is able to specify:

- The filter function (and its related uncertainty threshold)
- The logical operator (>, <, =, etc.)
- The overall threshold to which the result should be compared to
- The set of multimedia functions and their predefined multimedia objects.

The set of predefined multimedia objects can be acquired using a simple browsing window or using snapshots from live broadcasting Webcams.

After creating the *Multimedia Identifier*, its related properties can be exported into XML format.

3.1.2 Multimedia Context Condition Specification Component

The Multimedia Context Condition Specification component is similar to the Multimedia Identification Specification Component providing the additional possibility to define provisional actions and several multimedia predicates (location_identification, object_identification, user_identification, etc.) via the Multimedia Predicate Interface.

3.1.3 Permission Specification Component

The Permission Specification component is used to grant access to users, identified by one or several Multimedia Identifiers, to a set of images while associating a Multimedia Context Condition. In essence, the module allows to load images from the database, thus providing the *authorization manager* the ability to specify the authorized set of images and actions (view, delete, etc.). Once the permission is defined, it becomes possible to assign it a Multimedia Context Condition and consequently associate it to a set of Multimedia Identifiers. We give in Figure 2 an XML representation of a defined permission. In Figure 2 (a), the sample XML document describes the content of a given permission which grants the view action to the set of images identified by the ids (1, 5, 11, 10). In Figure 2 (b), the document shows the Mcc with id (290154776) attributed to the defined permission. Whereas in Figure 2 (c), assigned permissions for two different Multimedia Identifiers are described.

```

<?xml version="1.0" ?>
<!-- Permissions -->
- <Permission>
  <Id>53590</Id>
  <Desc>Permission1</Desc>
  <Actions>
    <action>View</action>
  </Actions>
- <MObjects>
  <mo>1</mo>
  <mo>5</mo>
  <mo>11</mo>
  <mo>10</mo>
</MObjects>
</Permission>

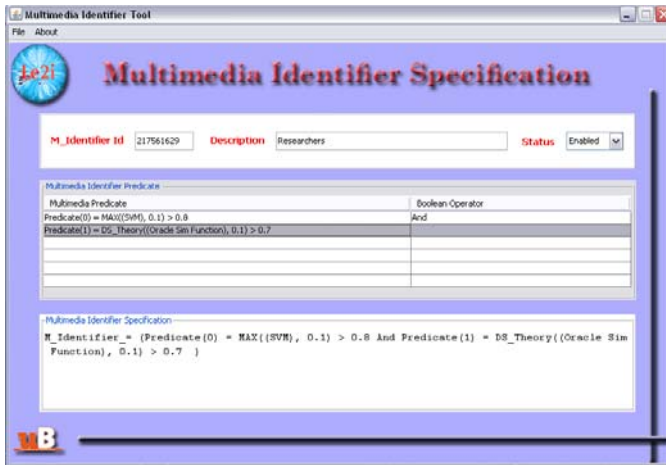
<?xml version="1.0" ?>
<!-- Mcc To Permission Assignment -->
- <Mcc_to_Perm>
  <Permission Id="53590">
    <Mcc>290154776</Mcc>
  </Permission>
</Mcc_to_Perm>

<?xml version="1.0" ?>
<!-- Permission To MId Assignment -->
- <Perm_to_MId>
  <Multimedia_Identifier Id="695851526">
    <Permission>83641</Permission>
    <Permission>53590</Permission>
    <Permission>78117</Permission>
  </Multimedia_Identifier>
- <Multimedia_Identifier Id="187540464">
  <Permission>53590</Permission>
</Multimedia_Identifier>
</Perm_to_MId>

```

(a) Permission (b) Mcc Assignment (c) M_{id} Assignment

Figure 2 : Permission Specification And Permission Assignment



(a) Main Interface



(b) Multimedia Predicates Interface

Figure 3: Multimedia Identifier Specification Module

3.2 User Toolkit

The user toolkit contains the Image Browser Engine, which is an application interface used to search for viral bacteria images either by using textual search or a query by image search requests. Results are filtered to the defined Multimedia Context Conditions. In fact, when a user wishes to access the application for the first time, she sends a request (1) to the Browser Engine. The Browser Engine saves the request and asks multimedia devices² to capture the user's environment (2). The devices send the captured images (4) to the analysis module in order to retrieve (5) user related Multimedia Identifiers which they are stored in protected directory along with Multimedia Context Conditions and Permissions. Images are analyzed using the predefined multimedia functions for each Multimedia Identifier. After, retrieving Multimedia Identifiers, the assigned permissions are retrieved (6) and forwarded to the Permission Filtering Engine (7). The Permission Filtering Engine stores the permissions and gets the attributed Mccs to check whether these are valid or not in order to send to the Browser Engine the list of allowed Images. This list is continuously updated depending on the Mccs validation state.

Now, when the user invokes a search task, she executes a query which is forwarded by the Browser Engine to the Images DB and the returned result is filtered. A snapshot of our application is shown in Figure 4.

² we used a simple Webcam to capture Images

In our implementation, we utilized two different multimedia functions:

- The first function is based on color object recognition and an SVM classifier. It computes decisions based on a set of classes representing the trained images (See [16] for more details).
- The second one is related to the InterMedia Oracle module [24] and is used for image similarity.

These functions are used to detect user assigned Multimedia Identifiers and check whether Multimedia Context Conditions are valid or not.

4. DEMONSTRATION

The demonstration will show the following features:

Multimedia Identifiers and Multimedia Context Conditions Specification: we will give an overview on how we create a set of multimedia identifiers and multimedia conditions based on user multimedia contexts.

Permissions Specification: we will show how we associate the defined Multimedia Context Conditions and Multimedia Identifiers to permissions specified upon a set of images.

Image Browsing: we will show how the system interacts when an end user connects in a real-time environment and we will demonstrate how multimedia context can affect system decision for image browsing once integrated in the specified permissions.

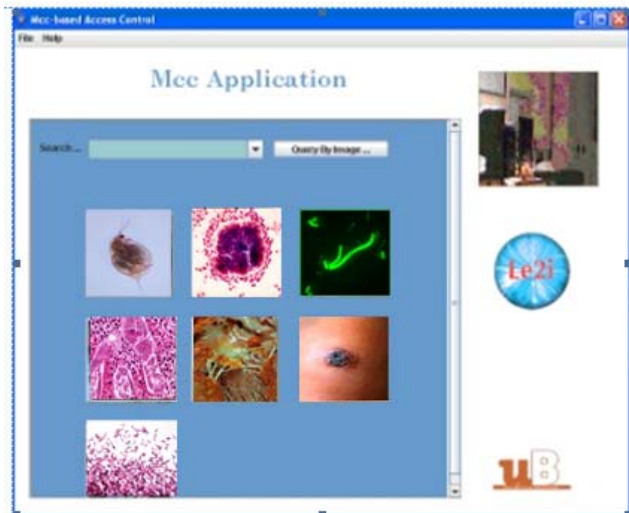


Figure 4: Image Browsing Application

5. REFERENCES

- [1] A. K. Jain, A. Ross, S. Prabhakar: "An introduction to biometric recognition". *IEEE Trans. Circuits Syst. Video Techn.* 14(1) 2004: pp. 4-20
- [2] A. P. Dempster. "A generalization of the Bayesian inference". *Journal of Royal Statistical Society*, 1968, Society 30: pp. 205-447.
- [3] ASP Alliance, http://aspalliance.com/404_Image_Web_Service, (22/10/2006)
- [4] A. Toninelli, R. Montanari, L. Kagal, O. Lassila: "A Semantic Context-Aware Access Control Framework for Secure Collaborations in Pervasive Computing Environments". *International Semantic Web Conference 2006*: pp. 473-486
- [5] B. Al Bouna, R. Chbeir and J. Miteran, MCA²CM: Multimedia Context Aware Access Control Model, Intelligence and Security Informatics, IEEE International Conference on Intelligence and Security Informatics, ISI 2007, New Jersey, USA, May 23-24, 2007
- [6] C. Agostino Ardagna, E. Damiani, Sabrina De Capitani di Vimercati, Pierangela Samarati: Towards Privacy-Enhanced Authorization Policies and Languages. *DBSec 2005*: 16-27
- [7] C. A. Ardagna, M. Cremonini, E. Damiani, S. De Capitani di Vimercati, P. Samarati: "Supporting location-based conditions in access control policies". *ASIACCS 2006*: pp. 212-222
- [8] C. E. Landwehr, "Formal models of computer security," *ACM Comput. Surv.*, Volume 13, Issue 3, Pages: 247 - 278, Sept. 1981.
- [9] DB2 Image Extenders, QBIC, <http://www.qbic.almaden.ibm.com/> (27/12/2006)
- [10] D. Poole. Logic, knowledge representation, and Bayesian decision theory. In *Proceedings CL-2000*, 2000 pp. 70-86
- [11] E. Damiani, Sabrina De Capitani di Vimercati, Stefano Paraboschi, Pierangela Samarati: Securing XML Documents. *EDBT 2000*: 121-135
- [12] E. Damiani, Sabrina De Capitani di Vimercati, Eduardo Fernández-Medina, Pierangela Samarati: Access Control of SVG Documents. *DBSec 2002*: 219-230
- [13] Efg2 Computer labs, <http://www.efg2.com/Lab/Library/ImageProcessing/Software Packages.htm> (05/10/2006)
- [14] Ferraiolo, D. F., Barkley, J. F., Kuhn, D. R., A role-based access control model and reference implementation within a corporate intranet, *ACM Transactions on Information and System Security*, February 1999, 2(1): 34-64.
- [15] Ferraiolo, D. F., et al., "Proposed NIST Standard for Role-Based Access Control," *ACM Trans. Information and System Security (TISSEC)*, vol. 4, no. 3, ACM Press, 2001, pp. 224-274.
- [16] F. Smach, C. Lemaitre, J. Miteran, J. Gauthier, M. Atri, "Colour Object recognition combining Motion Descriptors, Zernike Moments and Support Vector Machine", *Proceedings of IECON'06, IEEE, Paris-CNAM, France*, 2006, pp. 3238-3242
- [17] G. Shafer. "A Mathematical Theory of Evidence". Princeton University Press, 1976.
- [18] H. Ferhatosmanoglu, E. Tuncel, D. Agrawal, and A. El Abbadi. Approximate nearest neighbor searching in multimedia databases. In *Proc of 17th IEEE Int. Conf. on Data Engineering (ICDE)*, pages 503--511, Heidelberg, Germany, April 2001.
- [19] Java Community Press, <http://jcp.org/en/jsr/detail?id=135>, (12/11/2006)
- [20] J. Joshi, R. Bhatti, E. Bertino, A. Ghafoor: Access-Control Language for Multidomain Environments. *IEEE Internet Computing* 8(6): 40-50 (2004)
- [21] M. J. Covington, P. Fogla, Z. Zhan, M. Ahamad: "A Context-Aware Security Architecture for Emerging Applications". *ACSAC 2002*: pp. 249-260
- [22] M. J. Covington, W. Long, S. Srinivasan, A. K. Dey, M. Ahamad, G. D. Abowd, "Securing context-aware applications using environment roles". *SACMAT 2001*: pp.10-20
- [23] N.R. Adam, V. Atluri, E. Bertino and E. Ferrari, A Content-based Authorization Model for Digital Libraries' *IEEE Transactions Knowledge and Data Engineering*, Volume 14, Number 2, 2002, pages 296-315.
- [24] Oracle Technology Network, <http://www.oracle.com/technology/products/intermedia/index.html>. (20/09/2006)
- [25] P.E. Keller, D.L. McMakin, D.M. Sheen, A.D. McKinnon, Summet, J.W., "Privacy Algorithm for Cylindrical Holographic weapons surveillance system", *Aerospace and Electronic Systems Magazine, IEEE*, Feb 2000, pp. 17-24
- [26] R. Bhatti, E. Bertino, A. Ghafoor: "A Trust-Based Context-Aware Access Control Model for Web-Services". *Distributed and Parallel Databases* 18(1) 2005, pp. 83-105
- [27] R. Wolf, M. Schneider: "Context-Dependent Access Control for Web-Based Collaboration Environments with Role-based Approach". *MMM-ACNS 2003*: pp. 267-27