Multimedia Access Control using RDF Metadata*

Naren Kodali[1], Csilla Farkas[3,4] and Duminda Wijesekera[1,2]
[2]Center for Secure Information Systems, [1]Dept of Info. and Software Eng.,
George Mason University, Fairfax, VA 22030–4444,
[3]Information Security Laboratory, [4]Dept of Computer Science and Eng.,
University of South Carolina,Columbia, SC-29208,
email:{nkodali|dwijesek}@gmu.edu,farkas@cse.sc.edu

Abstract. The Synchronized Multimedia Integration Language (SMIL) [Aya01] is an W3C [W3C03] specification for authoring multimedia documents. Although SMIL has XML like syntactic constructs, unlike XML, SMIL compositions have an intended interpretation stemming from intuitive notions of playing out many media streams relative to each other. Thus, more than one SMIL syntactic expression can represent a multimedia composition with the same intended semantics. In this work we propose a normal form for SMIL objects that allows to specify security policies that are independent of representational syntax. We also show how to represent access control and QoS polices applicable to multimedia compositions by decorating SMIL compositions with RDF [KC03] statements. Our RDF statements are based on an RDF structure tailored to represent known security paradigms such as Discretionary, Mandatory, and Role-Based Access Control. Once the security paradigm is chosen and the SMIL document is decorated with security and QoS specifications, we show how to create secure views of the SMIL document. We call these views secure normal forms. Next, we show how a secure multimedia server can use these views to provide secure runtime environment.

1 Introduction

SMIL [Aya01] is an XML-like language for authoring multimedia documents. Unlike XML for textual documents, SMIL constructs have an *intended meaning* that must be enforced by application runtimes. Therefore, any security policy specification has to respect that semantics. This paper proposes a framework to do so for a chosen fragment of SMIL. This fragment consists of SMIL specifications constructed using sequential (⟨seq⟩) and parallel (⟨par⟩) composition operators.

Our framework uses two techniques. The first is to transform a SMIL document to a syntactic form that preserves the runtime semantics and shows the semantic hierarchy of any SMIL specification. We call this syntactic form the *SMIL normal form (smilNF)* of the document, and is structurally similar to the disjunctive normal form of a formula in propositional logic. Consequently, we provide an algorithm to translate any formula to its SMIL normal form. We show that any arbitrary SMIL (syntax) tree does not accurately represent its complete semantic hierarchy as it exists today. We present a method to obtain the hierarchy from the normal form. It is our position that normal forms are necessary because security policies may depend on the object hierarchy and not necessarily on one of its syntactic representations.

We follow the specifications of the W3C in using the Resource Description Framework (RDF) [KC03,MM03] to define metadata for specifying security and QoS policies. In order to do so, we propose a preliminary form of an RDF structure to model security and QoS specifications for SMIL documents. Based on our structure, we propose some RDF decorations that can be superimposed on SMIL documents in their normal form so that security and QoS specifications can be enforced by security and QoS aware runtimes. We now introduce our first issue by an example.

* This work was partially supported by the National Science Foundation under grants CCS-0113515 and 0112874.

As described in more detail in Section 3, SMIL uses ⟨par⟩ and the ⟨seq⟩ to specify parallel and sequential playing of multimedia streams. In SMIL, basic objects are media intervals. A media interval begins at a specified time, plays out for a specified duration and consequently ends at a specified time. This constitutes a rudimentary semantics for media intervals such as (audio) A_1 ,A_2 and (video) V_1,V_2 in Figure 1. In this semantics two streams are connected by a ⟨par⟩ if they begin and end playout at the same time. Two streams are connected by a ⟨seq⟩ if the second begins when the first ends.

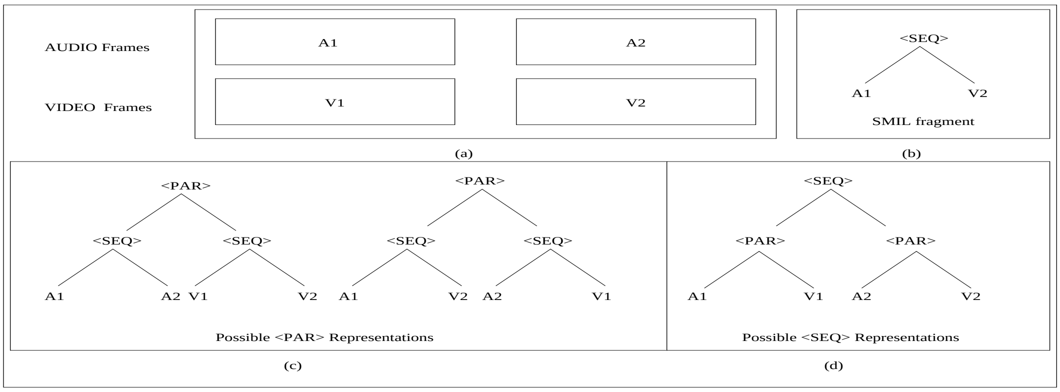

Fig. 1. Equivalence Class of the SMIL Constructs

Audio(A_1, A_2) and Video(V_1, V_2) frames as shown in part (a) of Figure 1, can be represented in SMIL in atmost three different ways using the ⟨par⟩ and ⟨seq⟩ constructs as shown in Figure 1 and explained below.

1. ⟨par⟩⟨seq⟩ A_1, A_2 ⟨/seq⟩ ⟨seq⟩ V_1, V_2 ⟨/seq⟩ ⟨/par⟩
2. ⟨par⟩ ⟨seq⟩A_1, V_2⟨/seq⟩ ⟨seq⟩ A_2, V_1 ⟨/seq⟩ ⟨/par⟩
3. ⟨seq⟩⟨par⟩A_1, V_1⟨/par⟩ ⟨par⟩ A_2, V_2 ⟨/par ⟩ ⟨/seq⟩
4. Because ⟨par⟩ is *commutative* ⟨par⟩ A_1, V_1 ⟨/par⟩ is the same as ⟨par⟩ V_1, A_1 ⟨/par⟩ and ⟨par⟩ A_2, V_2 ⟨/par⟩ is the same as ⟨par⟩ V_2, A_2 ⟨/par⟩.

Now consider the fragment ⟨seq⟩A_1, V_2 ⟨/seq⟩, as shown in part(b) is not a subtree of the given syntactic representations in part(d), but a sub-object of the SMIL tree. The identity of this *protection object* therefore is not a node in the XML tree, but an equivalence class, represented by the *normal form.*

Therefore, we propose that every SMIL specification is to be transformed to a sequence of parallel compositions that we call the *smil normal form (smilNF)* and show that all sub-objects of a SMIL object can be obtained as a subtree (created from) of this form. We also propose that security and QoS policies be specified on SMIL specifications in smilNF, and not on arbitrary syntax trees - because as shown, syntactic substructure does not coincide with semantic inheritance in SMIL.

Consequently, we present a nomenclature to specify security policies by appropriately decorating SMIL documents in smilNF. In order to do so, we have chosen the RDF [KC03,MM03] syntax. Because RDF syntax makes sense with respect to some RDF metadata, we propose meta structures and some metadata based on our metastructure for specifying access control and QoS policies applicable to multimedia compositions. Here again, we have chosen to represent limited features of access control polices. We show how some rudimentary discretionary, mandatory (also called multilevel secure (MLS)) and role-based access control policies can be specified using our nomenclature.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 describes the SMIL syntax. Section 4 defines the object identity and the SMIL normal form. Section 5 describes secure normal forms and give two algorithms for conversion for the secure normal forms. Section 6 describes the proposed RDF metastructure and . Section 7 shows how to decorate SMIL documents with RDF specifications. Section 8 describes how a secure run-time may communicate to obtain SMIL formatted data from a secure server. Section 9 concludes the paper.

2 Related Work

RDF is a W3C standard for representing metadata on the web. RDF provides syntax for representing entities, their properties and relationships. RDF Abstraction and Syntax [KC03], and RDF Primer [MM03] specify metainformation representation, and RDF Schema [BG03] is a general purpose schema language. Hayes et al. [Hay03] describes semantical aspects of RDF. We use the RDF vocabulary to specify our metastructure.

SMIL has a RDF based metainformation module [Mic01], but is insufficient to specify security policies. Independent of SMIL, Quality of Service (QoS) is an integral part of multimedia. Wijesekera et al. [WS96] proposed properties of quality metrics associated with continuous media and Gu et al. [GNY^{+}01] propose *HQML*, a language to negotiate some QoS parameters between clients and server.

We consider DAC, MLS and RBAC as security models governing the display and access to SMIL formatted multimedia. DAC(discretionary access control) is used to control access by restricting a subjects's access to an object. Sandhu et al [SS96], [SFK00] describe the principles and practices of RBAC systems. In RBAC the *role* that an user plays in the context of the application determines his access privileges. Multilevel security (MLS) systems provide controlled information flow based on the security classification of the protection objects (e.g., data items) and subjects of the MLS system (e.g., applications running in behalf of a user).

Damiani et al. [DdVPS00,DdVPS02] have proposed models for securing textual XML documents. In addition [DdV03] discuss feature protection of XML format images where the primary focus is controlled dissemination of sensitive data within an image. They propose an access control model with complex filtering conditions. This model uses SVG to render the map of a physical facility. This model has limitations when compared to flexibility and adaptability to issues, such as temporal and operational semantics. Bertino at al. [BHAE02] propose a security framework to model access control in video databases. Their objects are sequences of frames or identifiable objects within a frame. Their actions are viewing and editing. However they do not explain how objects with controlled accesses are released so that they do not lose their runtime semantics.

The main difference between SMIL and other XML documents are the temporal synchrony and continuity of the latter. The process of retrieval without losing the sense of continuity and synchronization needs better techniques and algorithms which all of the above models do not completely address. Kodali et al. [KW02,KWJ03,KFW03] propose three different models for enforcing different security paradigms. A release control for SMIL formatted multimedia objects for pay-per-view movies on the Internet that enforces DAC is described in [KW02]. The cinematic structure consisting of acts, scenes, frames of an actual movies are written as a SMIL document without losing the sense of a story. Here access is restricted to the granularity of an *act* in a movie. A secure and progressively updatable SMIL document [KWJ03] is used to enforce RBAC and respond to traffic emergencies. In an emergency response situation, different recipients of the live feeds have to be discriminated to people playing different roles. The paper describes a mechanism to enforce RBAC policies. [KFW03] describes an MLS application for secure surveillance of physical facilities where guards with different security classification in charge of the physical security of the building are provided live feeds matching their level in the MLS subject hierarchy.

3 SMIL: Synchronized Multimedia Integration Language

SMIL [Aya01] is an extension to XML developed by W3C to author multimedia presentations with audio, video, text and images to be integrated and synchronized. The distinguishing features of SMIL over XML are the syntactic constructs for timing and synchronizing live and stored media streams with qualitative requirements. In addition, SMIL provides a syntax for spatial layout including non-textual and non-image media and hyperlinks. We do not address the later aspects of SMIL in this paper. Consequently we explain those SMIL constructs that are relevant for our application.

SMIL constructs for synchronizing media are ⟨seq⟩, ⟨ excl ⟩ and ⟨par⟩. They are used to hierarchically specify synchronized multimedia compositions. The ⟨seq⟩ element plays its children one after another in sequence. ⟨ excl ⟩ specifies that its children are played one child at a time, but does not impose any order. The ⟨par⟩ plays all children elements as a group, allowing parallel play out. For example, the SMIL specification ⟨par⟩ video src=camera1 ⟩ ⟨audio src=microphone1⟩⟨/par⟩ specify that media sources camera1 and microphone1 are played in parallel.

In SMIL, the time period that a media clip is played out is referred to as its *active duration.* For parallel play to be meaningful, both sources must have equal active durations. When clips do not have equal active durations, SMIL provides many constructs to equate them. Some examples are begin (allows to begin components after a given amount of time), dur (controls the duration), end (specifies the ending time of the component with respect to the whole construct), *repeatCount* (allows a media clip to be repeated a maximum number of times). In addition, attributes such as *syncTolerance* and *syncMaster* controls runtime synchronization, where the former specifies the tolerable mis-synchronization (such as tolerable lip-synchronization delays) and the latter specifies a master-slave relationship between synchronized streams. In this paper we assume that children of ⟨ par ⟩ have active durations.

4 Object Identity in SMIL

For XML formatted textual documents [DdVPS00,DdVPS02] the *protection objects* are nodes of the XML tree. This may be acceptable for some forms of multimedia, such as movies [KW02]. But as shown in section 1 using Figure 1, this is problematic for multimedia in general. We therefore define the SMIL normal form in Definition 1.

Definition 1 (SMIL Normal Form) *We say that a SMIL specification(s) is in the SMIL Normal Form (smilNF) if it is of the following form ⟨seq⟩ ⟨par⟩ $C_{1,1}$(s) $C_{1,2}$(s) $C_{1,3}$ (s)... $C_{1,n}$(s) ⟨/par⟩ ... ⟨par⟩ $C_{m,1}$(s) $C_{1,2}$(s) $C_{1,3}$ (s)... $C_{m,n}$(s)⟨ /par ⟩ ⟨ /seq ⟩ where $C_{i,j}$ are audio or video media intervals.*

Figure 2 shows a more general representation of SMIL objects. In Representation 4, there are 4 sequentially arranged audio or video frames which in turn are time-sliced into three intervals. The boxes represented by $A_1, A_2 \ldots D_2, D_3$ could be either a audio or video frame. The right hand side of the representation shows how it is represented in the normal form according to Definition 1.
As stated, a sub object of a SMIL object does not have to be sub tree of one of its syntactic representation. In Representation 5, the sub-object we consider is shown by the enclosed area, and its normal form tree in shown on the right hand side.

4.1 Security Paradigms and Access Control Rules

In order to specify security policies the *subject* and the *protection object* need to be unambiguously identifiable. The subject may be granted an access permission in DAC, but in MLS and RBAC such granting is indirect and has to satisfy some constraints, usually expressed in the form of rules. This section formally defines the security paradigms we use and the associated constraints that are used to construct the access control lists.

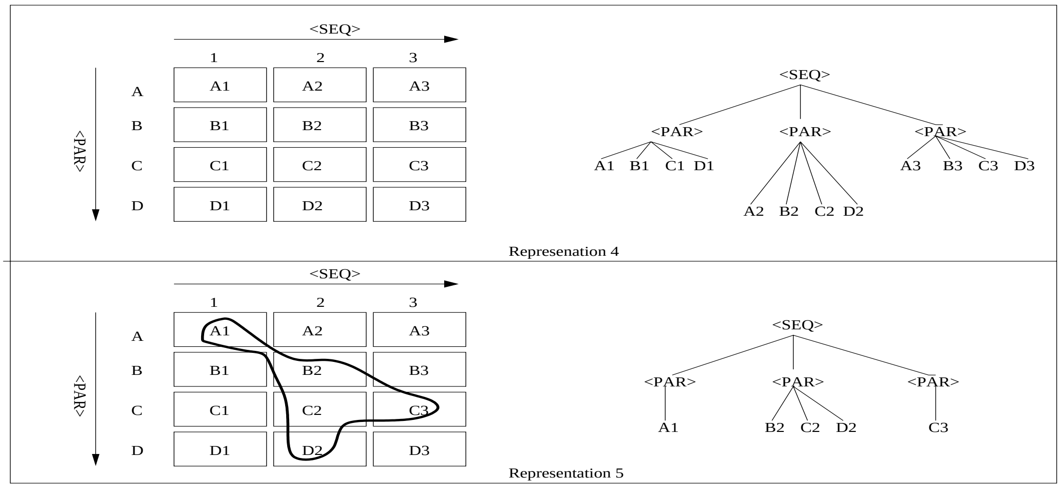

Fig. 2. A Generalized Representation based on the Normal Form

DAC (Discretionary Access Control) Discretionary Access Control defines access permissions a based on subjects s and objects o. Such a permission can be expressed by constructing an access control matrix containing appropriate triples (s,o,a).

RBAC (Role Based Access Control) The simplest Role-Based Access Control models has three entities roles, users, privileges, and two associations, subject-to-role and role-to-privilege assignments among them. A subject may activate any authorized roles, and by doing so obtains all privileges assigned to the activated role.

For each subject s let the set of active roles be given by $ActR(s)$, and $AuthR(s)$ be the set of roles permitted to be invoked by s. Then, the restriction that a user may activate only authorized roles can be stated as $ActR(s) \subseteq AuthR(s)$.

Privileges (access permissions) associated for each role are based on objects defined in the rbacNF. That is, a given specification S in rbacNF is organized in a manner that all objects permitted to a role R_i are represented together. Then, we can define the access permissions of each role r as rToPer(r_i), where rToPer(r_i) consists (object, action) pairs. Then (s,o,a) belongs to the access control matrix iff $ActR(s) \subseteq AuthR(s) \wedge \exists r \in ActR(s)(o,a) \in rToPer(r)$.

MLS (Multi Level Security) In Multi Level Security each access permission is guided by the security clearance of the subject and the security classification of the accessed object. Security labels form a lattice structure with the dominance relation among the labels. Information flow between the security labels is controlled based on the security objectives. In this paper we allow information flow from low security objects to high security objects, that is, from a dominated object to a dominating object. Assuming that our access permissions are “read” permissions, it means that a subject is allowed to access an object only if the subject’s security clearance dominates the security classification of the object.

Let $Class(s)$ denote the classification of subject s. L denotes the lattice structure and binary relation $dominates(l_1, l_2),\ l_1, l_2 \in L$ denotes that label l_1 dominates label l_2. To generate all labels dominated by the security classification of a subject ($Class(s)$), we generate the transitive closure of dominance relation as follows:

1. Let$Dominated(s) = Class(s)$
2. For all pairs $dominates(l_i, l_j)$, where l_i in $Dominated(s)$, $Dominated(s) = Dominated(s) \cup l_j$

To permit accesses for a subject to objects in mlsNF, we use the set *Dominated* to determine the appropriate data items. That is,

$\forall s$, if $Class(s)$ and $\{l_{i_1}, \ldots, l_{i_n}\} \in Dominated(s)$ and $o \in Cl_{i_k}$ $k = 1, \ldots, n$ then (s, o, a).
That is, a subject is granted the access a to an object o if the security clearance of the subject dominates the security classification of the object. Hence MLS could be stated as an (s,o,a) triple. In effect, the generalized access control rule in all three domains could be declared as a (s,o,a) triple.

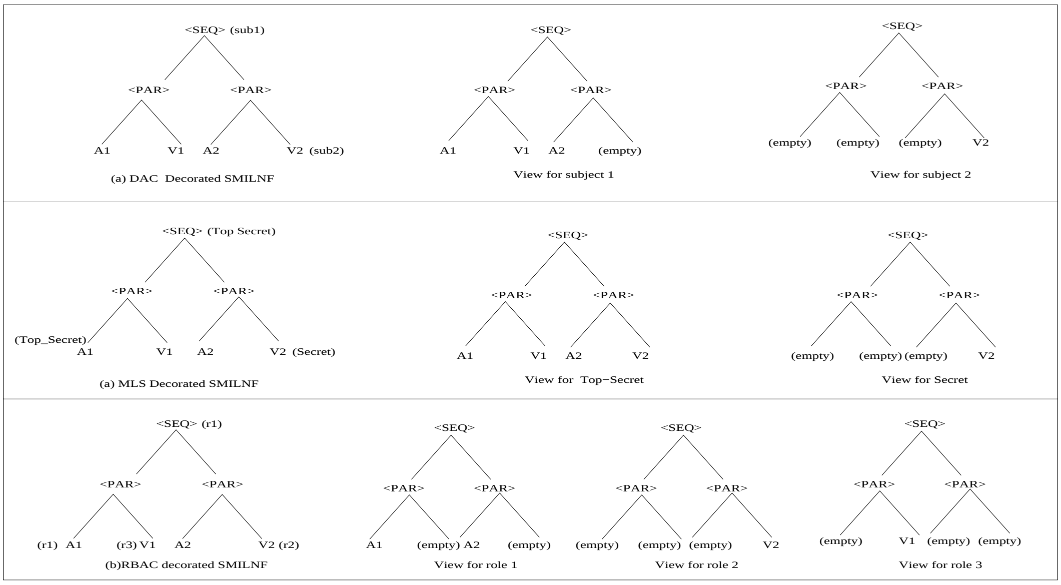

Fig. 3. Reduction to dacNF, mlsNF and rbacNF

5 Secure Normal Forms

As briefly described DAC, MLS and RBAC security policies can be reduced to (s,o,a) triples. However in RBAC the permissions are assigned primarily to roles and subject's permission (that is (o,a) pairs) could be derived depending on a subjects active roles. Similarly, in MLS permissions are assigned to security levels, and depending on the clearance of the subjects, subject's permission (that is (o,a) pairs) could be derived. Therefore, we alow SMIL documents in smilNF to be decorated to subjects, security levels and roles respectively. Then the final authorization triples (s,o,a) triples can be derived using appropriate rules.

The security decoration on the *protection object* is defined on the normal form. We allow any node of a SMIL tree in smilNF to be decorated as shown in the Figure 3. Given any such decoration, we can compute a view that is permitted for each subject, security level or a role. They are referred to as *security normal forms.* Security normal forms are formally defined in Definitions 2, 3, 4.

5.1 Normal Form for DAC

The DAC normal form is a parallel composition of permitted segments.The smilNF specification is decorated with the DAC metadata, and upon reduction, would group all permitted segments of a particular subject under a single ⟨par⟩ construct. Each of these ⟨ par ⟩ construct is the *view* of the associated subject.

Definition 2 (DAC Normal Form) *We say that a smilNF specification ($\tilde{s}$) is in the DAC Normal Form (dacNF) if it is of the form ⟨ seq ⟩ ⟨par⟩ $C_1(\tilde{s})$ ⟨/par⟩ ⟨par⟩ $C_2(\tilde{s})$ ⟨/par⟩ ⟨par⟩ C_3 ($\tilde{s}$)... $C_n(\tilde{s})$ ⟨ /par ⟩ ⟨ /seq ⟩ where $C_1, C_2, C_3 \ldots C_n$ are media intervals permitted to be accessible to security level.*

5.2 Normal Form for MLS

Definition 3 (MLS Normal Form) *We say that a smilNF specification ($\tilde{s}$) is in the mlsNF(MLS Normal Form) if it is of the form ⟨ seq ⟩ ⟨ par ⟩ $C_ts(\tilde{s})$⟨ /par⟩ ⟨par⟩ $C_s(\tilde{s})$⟨ /par⟩ ⟨ par ⟩ $C_u(\tilde{s})$ ⟨ /par ⟩ ⟨ /seq ⟩ where all Security classifications in C_ts ($\tilde{s}$), $C_s(\tilde{s})$, C_u ($\tilde{s}$)are respectively Top-Secret, Secret and Unclassified.*

As stated in Definition 3, a Normal Form in mlsNF is one that is a parallel composition of at most three documents, where each document belongs to one security class, that are said to be the views corresponding to the respective security classes.

5.3 Normal Form for RBAC

Definition 4 (RBAC Normal Form) *We say that a smilNF specification ($\tilde{s}$) is in the rbacNF (RBAC Normal Form) if it is of the form ⟨ seq ⟩ ⟨ par ⟩ $C_{r_1}(\tilde{s})$ ⟨ /par⟩ ⟨par⟩ $C_{r_2}(\tilde{s})$⟨ /par⟩ ⟨par⟩ $C_{r_3}(\tilde{s})$... $C_{r_n}(\tilde{s})$ ⟨ /par ⟩ ⟨ /seq ⟩ where the Role attributes in $C_{r_1}(\tilde{s})$, $C_{r_2}(\tilde{s})$, $C_{r_3}(\tilde{s})$ ... C_{r_n} are respectively $role_1, role_2, role_3 \ldots role_n$.*

As stated in Definition 4, a Normal Form in rbacNF is one that is parallel composition of at one or more role specifications, where each specification belongs to a particular role assignment, and is said to be the view corresponding to the assigned role.

5.4 Algorithms for conversion into Secure Normal Forms

This section gives the algorithms for the reduction of the smilNF to the appropriate secure normal forms, based on the security paradigm that we are using. When we try to reduce a smilNF to a secure normal form we encounter different time containers, some of which are nested. We give below the algorithms for conversion to the mlsNF and rbacNF. They represent the actions necessary when to facilitate reduction under all possible circumstances.

During the rewrite, some of the nodes are represented as ⟨ empty ⟩. This representation is used to establish an audio or video *silence* in the playout. When grouping elements that satisfy a particular access control rule, there is a need to eliminate those that do not qualify. Normally, a silent audio segment or a blank video segment are used to during playout to maintain continuity without losing synchronization.

Algorithm 1 details the mechanics of conversion from smilNF to mlsNF. It details how the rewrite should be done when we encounter different time containers, some of which are nested. The generated output would have atmost three parallel compositions each corresponding to a unique security level. The MLS paradigm has an unique property which allows subjects with a higher classification access to the view of the lower classified subjects. This algorithm takes this property into consideration when generating smilNF.

Algorithm 1 TOmlsNF (Conversion to MLS Normal form)

```
INPUT : Security Classification decorated smilNF, possible classifications Top-Secret, Secret, Unclassified.
OUTPUT : mlsNF
$(\tilde{s})$ is a smilNF specification (as described in Definition 1 ) with a possible Security classification
if $(\tilde{s})$ is $\langle$ seq $\rangle$ $s_1 s_2$ $\langle$ /seq $\rangle$ then
    $C_ts\ (\tilde{s}) = \langle \text{ seq } \rangle \langle \text{ par } \rangle C_ts(s_1) \langle \text{ /par } \rangle \langle \text{ par } \rangle C_ts(s_2) \langle \text{ /par } \rangle \langle \text{ /seq } \rangle$
    $C_s\ (\tilde{s}) = \langle \text{ seq } \rangle \langle \text{ par } \rangle C_s(s_1) \langle \text{ /par } \rangle \langle \text{ par } \rangle C_s(s_2) \langle \text{ /par } \rangle \langle \text{ /seq } \rangle$
    $C_u\ (\tilde{s}) = \langle \text{ seq } \rangle \langle \text{ par } \rangle C_u(s_1) \langle \text{ /par } \rangle \langle \text{ par } \rangle C_u(s_2) \langle \text{ /par } \rangle \langle \text{ /seq } \rangle$
else if $(\tilde{s})$ is $\langle$ par $\rangle$ $s_1\ s_2$ $\langle$ /par $\rangle$ then
    $C_ts\ (\tilde{s}) = \langle \text{ par } \rangle C_ts(s_1) \langle \text{ /par } \rangle \langle \text{ par } \rangle C_ts(s_2) \langle \text{ /par } \rangle$
    $C_s\ (\tilde{s}) = \langle \text{ par } \rangle C_s(s_1) \langle \text{ /par } \rangle \langle \text{ par } \rangle C_s(s_2) \langle \text{ /par } \rangle$
    $C_u\ (\tilde{s}) = \langle \text{ par } \rangle C_u(s_1) \langle \text{ /par } \rangle \langle \text{ par } \rangle C_u(s_2) \langle \text{ /par } \rangle$
end if
if either of $C_x(s_i)$ are empty for some x $\in$ {TS,S,U} and i $\in${1,2} then
    $C_x(s_i)$ in the right hand sides above must be substituted by $\phi\ (S_i)$ where $\phi\ (s_i)$ is defined as $\langle$ audio or video src = empty $\rangle$
end if
If Security classification =Top-Secret, then $C_{ts}\ (\tilde{s}) = (\tilde{s})$
If Security classification =Secret, then $C_ts(\tilde{s}) = \phi$ ,$C_s\ (\tilde{s}) = (\tilde{s})$
If Security classification=Unclassified, then $C_ts\ (\tilde{s}) = \phi$ , $C_s(\tilde{s}) = \phi$ , and $C_u\ (\tilde{s})= (\tilde{s})$.
Then let mlsNF $(\tilde{s}) = \langle \text{ seq } \rangle \langle \text{ par } \rangle C_ts \langle \text{ /par } \rangle \langle \text{ par } \rangle (\tilde{s})\ C_s \langle \text{ /par } \rangle \langle \text{ par } \rangle (\tilde{s})\ C_u\ (\tilde{s}) \langle \text{ /par } \rangle \langle \text{ /seq } \rangle$ .
```

Algorithm 2 details the conversion from smilNF to rbacNF. The generated output would have as many parallel compositions as the number of roles involved. The view granted to a subject, is one of these parallel compositions depending on the association with that role. A subject could be given access to multiple views, equalling the number of roles it is associated with.

In Figure 3 we have three examples of decorated smilNF. The security classification could be done at three levels the primary time container, the nested time container and at the frame level. The Figure 3 shows the schematic reduction after applying the algorithms listed in previous sections.In our DAC example subject sub_1 is permitted access to the whole tree, where as subject sub_2 is granted access only to video frame V_2. The reduction uses the $\langle$ empty $\rangle$ to denote an element that is disallowed. The views corresponding to sub_1 and sub_2 that when combined form the dacNF after the application of the algorithm is shown on the right hand side. The first composition denotes the *view* of subject sub_1 and the second composition the *view* of subject sub_2. In the MLS example the $\langle$ par $\rangle$ is classified as Top-Secret and audio frame A_1 is also classified as Top-Secret. The video frame V_2 is classified as secret. The algorithm TOmlsNF is applied and the resulting views for Top-Secret and Secret are shown. The resulting mlsNF is a parallel composition of two security classifications, and the Top-Secret(higher classification) is allowed access to the Secret(lower) classification by the virtue its position in the classification hierarchy. Similarly a RBAC decorated smilNF with three roles r_1, r_2, r_3 and its reduced rbacNF is also shown, but role hierarchy and superiority in roles is not discussed.

6 Metastructure

Metadata is needed for specifying access control policies for multimedia because the current specification of SMIL [Aya01] does not have constructs for security and minimal constructs for QoS. The SMIL metamodule [Mic01] claims that RDF could be used to declare metadata to be used within a SMIL document, but does not provide sufficient detail on how to effectively use RDF to state our

Algorithm 2 TOrbacNF (Conversion to RBAC Normal form)

```
  INPUT : Role decorated smilNF, possible roles $r_1, r_2, r_3, \ldots r_n$
  OUTPUT : rbacNF
Ensure: $(\tilde{s})$ is a smilNF specification (as described in Definition 1 ) with a possible Role attribute
  if $(\tilde{s})$ is ⟨ seq ⟩ $s_1 s_2$ ⟨ /seq ⟩ then
    $C_{r_1}$ $(\tilde{s})$ = ⟨ seq ⟩ ⟨ par ⟩ $C_{r_1}(s_1)$ ⟨ /par ⟩ ⟨ par ⟩ $C_{r_1}(s_2)$ ⟨ /par ⟩ ⟨ /seq ⟩
    $C_{r_2}$ $(\tilde{s})$ = ⟨ seq ⟩ ⟨ par ⟩ $C_{r_2}(s_1)$ ⟨ /par ⟩ ⟨ par ⟩ $C_{r_2}(s_2)$ ⟨ /par ⟩ ⟨ /seq ⟩
    $C_{r_3}$ $(\tilde{s})$ = ⟨ seq ⟩ ⟨ par ⟩ $C_{r_3}(s_1)$ ⟨ /par ⟩ ⟨ par ⟩ $C_{r_3}(s_2)$ ⟨ /par ⟩ ⟨ /seq ⟩
    $C_{r_n}$ $(\tilde{s})$ = ⟨ seq ⟩ ⟨ par ⟩ $C_{r_n}(s_1)$ ⟨ /par ⟩ ⟨ par ⟩ $C_{r_n}(s_2)$ ⟨ /par ⟩ ⟨ /seq ⟩
  else if $(\tilde{s})$ is ⟨ par ⟩ $S_1 S_2$ ⟨ /par ⟩ then
    $C_{r_1}$ $(\tilde{s})$ = ⟨ par ⟩ $C_{r_1}(s_1)$ ⟨ /par ⟩ ⟨ par ⟩ $C_{r_1}(s_2)$ ⟨ /par ⟩
    $C_{r_2}$ $(\tilde{s})$ = ⟨ par ⟩ $C_{r_2}(s_1)$ ⟨ /par ⟩ ⟨ par ⟩ $C_{r_2}(s_2)$ ⟨ /par ⟩
    $C_{r_3}$ $(\tilde{s})$ = ⟨ par ⟩ $C_{r_3}(s_1)$ ⟨ /par ⟩ ⟨ par ⟩ $C_{r_3}(s_2)$ ⟨ /par ⟩
    $\ldots C_{r_n}$ $(\tilde{s})$ = ⟨ par ⟩ $C_{r_n}(s_1)$ ⟨ /par ⟩ ⟨ par ⟩ $C_{r_n}(s_2)$ ⟨ /par ⟩
  end if
  if  either of $C_x(S_i)$ are empty for some x $\in \{r_1, r_2, r_3 \ldots r_n\}$ and i $\in \{1,2\}$ then
    $C_x(s_i)$ in the right hand sides above must be substituted by $\phi$ $(s_i)$ where $\phi$ $(s_i)$ is defined as ⟨ audio/video
    src = empty⟩
  end if
  If Role Attribute = $r_1$, then $C_{r_1}$ $(\tilde{s})$ = $(\tilde{s})$, $C_{r_2}(\tilde{s})$= $\phi$ and $C_{r_3} \ldots C_{r_n}(\tilde{s})$= $\phi$
  If Role Attribute = $r_2$, then $C_{r_2}$ $(\tilde{s})$ = $(\tilde{s})$, $C_{r_1}(\tilde{s})$ = $\phi$, and $C_{r_3} \ldots C_{r_n}$ $(\tilde{s})$ = $\phi$
  If Role Attribute = $r_n$, then $C_{r_n}$ $(\tilde{s})$= $(\tilde{s})$, $C_{r_1}$ $(\tilde{s})$ $\ldots$ $C_{r_{n-1}}$ = $\phi$
  Then let rbacNF $(\tilde{s})$=⟨ seq ⟩ ⟨ par ⟩ $C_{r_1}$ ⟨ /par ⟩ ⟨ par ⟩ $(\tilde{s})$ ⟨ /par ⟩ ⟨ par ⟩ $C_{r_2}$ $(\tilde{s}) C_{r_3} \ldots C_{r_n}(\tilde{s})$⟨ /par
  ⟩⟨ /seq ⟩
```

needs. The RDF[KC03] and RDFS[BG03] enable defining metadata but not the interpretation or anticipated meaning applicable to multimedia. Consequently, we design a structure for metadata to enforce security related to various paradigms.

6.1 Resource Description Framework

RDF (Resource Description framework) is a language for representing information about resources that can be identified on the web. The URI (Uniform Resource Identifiers) with optional fragment identifiers are used to describe subjects objects and predicates in statements, and relationships between URI-identifiable entities. This representation primarily uses RDF/XML, but because our focus is synchronized multimedia the representation is in RDF/SMIL. In this section we describe the a RDF metastructure for secure multimedia using RDF-Schema. Our vocabulary is defined in a namespace identified by the URI reference http://www.w3.org/2000/01/rdfschema/♯ . In the following structure the prefix *rdfs* is used to refer to that namespace.

As stated using the RDF/XML [MM03] we define the *xmlns*(XML namespace) for the metadata and call it *smilmetadata*. We refer to *smilmetadata* in order to use any metadata we define. The description *smilmetadata*:MLS is useful in identifying permissible media elements within a SMIL-formatted document when our security paradigm is MLS (Multi-Level-Security).
Figure 4 represents the class hierarchy of the metadata we define in RDF for specifying security and QoS in a SMIL formatted multimedia document. Figure 4 represents those components necessary to represent security and QoS parameters chosen for this study.

The metastructure we define is based on a schema and represents metadata for our chosen security and QoS parameters. In the context of security we need to define metadata to effectively represent the security paradigm with respect to DAC, MLS and RBAC. The MLS class, consists of Top-Secret, Secret and Unclassified as sub-classes. RBAC and DAC have subjects and roles defines as sub-classes. Our QoS metric consists of two parameters: `delay` and `rate of display` under the

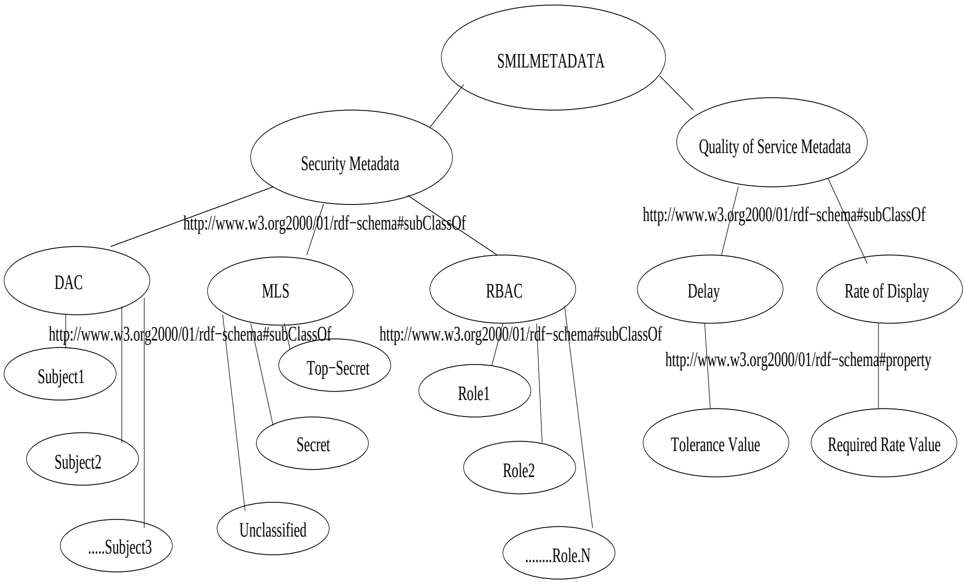
SMILMETADATA
Security Metadata
Quality of Service Metadata
http://www.w3.org2000/01/rdf-schema#subClassOf
http://www.w3.org2000/01/rdf-schema#subClassOf
DAC
MLS
RBAC
Delay
Rate of Display
http://www.w3.org2000/01/rdf-schema#subClassOf
http://www.w3.org2000/01/rdf-schema#subClassOf
Subject1
Top-Secret
Role1
http://www.w3.org2000/01/rdf-schema#property
Subject2
Secret
Role2
Tolerance Value
Required Rate Value
Unclassified
.....Subject3
........Role.N

Fig. 4. Class Hierarchy of the Metastructure

class Run-time-QoS. These attributes take values `toleranceValue` and `requiredRateValue` respectively. The namespace for the metastructure is http://svp.gmu.edu/smil-ns♯ and is referred to as *smilmetadata*.

The `subClassOf` in RDF Schema is a special subset/set relationship between two classes. In the metastructure, Top-Secret, Secret and Unclassified are sub-classes of Class: MLS. The `rdf:subClassOf` property is transitive, implying that resources that are instances of `subClass` are implicit instances of the Class. The `rdf:domain` and `rdf:range` attributes available in RDF are used to define the scope of the members of a container with respect to a property of a class.

1. `MLS` is used to define the security level of a particular media element. The actual levels of Security [Top-Secret, Secret, Unclassified] are sub-classes of the class MLS.
2. `RBAC` defines the role of the current role assigned to the object.
3. `Run-time-QoS` regulates different parameters for maintaining good service during the delivery of the media.

   As specified by W3C interpretation of the metadata is entirely the responsibility of the application that uses them.

The security metadata used to decorate a SMIL specification is supposed to reflect the security paradigm used. The DAC is enforced through subjects and the MLS through security classifications, Top-Secret, Secret and Unclassified and the RBAC through role attributes. For e.g ⟨ *smilmetadata* :MLS ⟩ enclosing ⟨ *smilmetadata* :Top-Secret ⟩ means we refer to the Top-Secret `rdf:subclass` of the *smilmetadata*: MLS class to define the security attribute of an element. The metastructure needs to describe QoS parameters. Among a choice of many, we only consider *delay* and *rateOfDisplay* as the minimum negotiable application level parameters and are described as stated above. The `toleranceValue` and `requiredRateValue` are properties used to specify the requirements to the clients during delivery of the media.

```
      xmlns: smilmetadata = http://svp.gmu.edu/AudioVideo/.../ smilmetadata ♯ ⟩
      ⟨ ?xml version=1.0? ⟩
      ⟨ rdf:RDF xml:lang=xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-ns♯
      xmlns:rdfs=http://www.w3.org/TR/2003/WD-rdf-schema-20030123/♯
                ⟨ rdfs:Class rdf:ID= ”DAC” ⟩
                ⟨ rdfs:Class rdf:ID= ”MLS” ⟩
                ⟨ rdfs:Class rdf:ID= ”RBAC ⟩
                ⟨ rdfs:Class rdf:ID= ”Run-time-QoS”/ ⟩
        ⟨ rdfs:Class rdf:ID= ”Subject1” ⟩
                ⟨ rdfs:subClassOf rdf:resource=” ♯ DAC ”/ ⟩
        ⟨ /rdfs:Class ⟩
// Other Subjects
        ⟨ rdfs:Class rdf:ID= ”Top-Secret”⟩
                ⟨ rdfs:subClassOf rdf:resource=♯ MLS ⟩
        ⟨ /rdfs:Class⟩
        ⟨ rdfs:Class rdf:ID= ”Secret”⟩
                ⟨ rdfs:subClassOf rdf:resource=♯ MLS ⟩
        ⟨ /rdfs:Class⟩
        ⟨ rdfs:Class rdf:ID= ”Unclassified”⟩
                ⟨ rdfs:subClassOf rdf:resource=♯ MLS ⟩
        ⟨ /rdfs:Class⟩
        ⟨ rdfs:Class rdf:ID=”role1” ⟩
                ⟨ rdfs:subClassOf rdf:resource=” ♯ RBAC ”/ ⟩
        ⟨ /rdfs:Class ⟩
//Other Roles
```

7 Metadata in SMIL

This section describes how the designed metastructure is to be used in a SMIL specification. As stated earlier, the document on which we use the designed metadata must be in smilNF. The Dublin Core metadata [BMB02]is also used for describe the document. The *smilmetadata* structure that has been defined earlier is utilized for the RDF-metadata for namespace references.

```
⟨ body ⟩
        ⟨ smilmetadata :MLS ⟩
        ⟨ par id=”shot3” smilmetadata : Top-Secret ⟩
                    ⟨ video src=”shot3.mpg” / ⟩
                    ⟨ audio src=”shot3.au” / ⟩
        ⟨ /par ⟩
        ⟨ par id=”shot4” ⟩
                    ⟨ video src=”shot4.mpg” / ⟩
                    ⟨ audio src=”shot4.au” smilmetadata :Unclassified / ⟩
        ⟨ /par ⟩
        ⟨ /smilmetadata :MLS ⟩
⟨ /body ⟩
```

The example above shows a MLS decorated smilNF. The ⟨par⟩ in shot 3 is Top-Secret and the audio frame of shot 4 is Unclassified. The evaluation of the SMIL document in runtime requires a semantic query model and an efficient interpreter to understand and interpret the RDF metadata used to declare security and QoS parameters.

The security decoration in the following SMIL specification belongs to the RBAC security paradigm. The video frame of shot 1 is allowed for $role_1$ and the entire parallel composition in shot 4 is allowed for $role_3$.

```
⟨ body ⟩
        ⟨ smilmetadata :RBAC ⟩
        ⟨ par id=”shot1” ⟩
                ⟨ video src=”shot1.mpg” smilmetadata: role1 ⟩
                ⟨ audio src=”shot1.au” / ⟩
        ⟨ /par ⟩
        ⟨ par id=”shot4” smilmetadata:role3 ⟩
                ⟨ video src=”shot4.mpg” / ⟩
                ⟨ audio src=”shot4.au” / ⟩
        ⟨ /par ⟩
        ⟨ smilmetadata :RBAC ⟩
⟨ /body⟩
```

8 Operational Semantics

Our metastructure can be used by a multimedia client that seeks to obtain SMIL documents with proposed RDF decorations. Our client must use an RDF based query system for this purpose to generate views for DAC, MLS and RBAC. The RDF Query [MS98] uses a declarative syntax for selecting RDF resources that meet specified criteria. For example, for RBAC retrieval, we show how to construct a RDF query to retrieve the view for a given role. Similarly, we show an example query to retrieve all objects corresponding to particular security classification.

An RDF-interpreter is necessary to understand and assemble a SMIL view from a RDF decorated SMIL document that is to be interpreted by a SMIL player at the client. Although we do not provide such an interpreter, our client need to have two interacting interpreters, where the SMIL-Interpreter calls the RDF-interpreter to interpret RDF decorations.

As stated in Section 4.1, all DAC, MLS and RBAC can be reduced to the access control rule could be stated as a simple (s: subject, o: object, a:access). Therefore the access control rule is defined as a 4 tuple (c,o,d,a) where C is a condition expressed in RDF Query, o is the security object(Normal Form), d is the decision to grant or deny and a is the action to be performed when this rule is activated.

An example of RDF Query [MS98] for the RBAC and MLS security paradigms are discussed in Section 8.1 and 8.2. The conditions use SQL keywords such as *select*, *from* etc. Complex and nested queries could be formulated with the use of boolean expressions.

8.1 An RBAC Query

This query represented below retrieves the view pertaining to a single role ($role_1$) from the rbacNF. The scope of the RBAC query is the RBAC Normal form. The structure of rbacNF guarantees that media components associated with the particular role is grouped together, and the retrieval could be based on the metadata used to define the particular role assignment. The RBAC query in section 8.1 would `select` components associated with *smilmetadata*: $role_1$ `from` the specified URI for the location of the rbacNF.

```
⟨ rdfq:rdfquery ⟩
        ⟨ rdfq:From eachResource=”http://svp.gmu.edu/AudioVideo/smil-ns ♯ rbacNF ”/ ⟩
            ⟨ rdfq:Select ⟩
            ⟨ rdfq:Property name=” $role_1$ ”/ ⟩
            ⟨ /rdfq:Select ⟩
        ⟨ /rdfq:From ⟩
 ⟨ /rdfq:rdfquery ⟩
```

The query below retrieves the view pertaining to a specified security classification within a MLS Normal Form. The scope of the MLS query is the mlsNF represented by the appropriate URI. The MLS query in section 8.2 would `select` components associated with *smilmetadata* :Top-Secret `from` the specified URI that denotes the location of the mlsNF.

8.2 MLS Query

```
⟨ rdfq:rdfquery ⟩
        ⟨ rdfq:From eachResource=”http://svp.gmu.edu/AudioVideo/smil-ns ♯mlsNF”/ ⟩
            ⟨ rdfq:Select ⟩
            ⟨ rdf:ID ⟩ Top-Secret ⟨ /rdf:ID ⟩
            ⟨ /rdfq:Select ⟩
        ⟨ /rdfq:From ⟩
⟨ /rdfq:rdfquery ⟩
```

A The run-time algorithm describes the retrieval of a secure SMIL document. During the first stage, the algorithm negotiates the QoS parameters. A failure of available QoS would result in the termination of the media transfer. Once the query answer is obtained, the access control policy is evaluated. If access is granted the associated action is initiated. Views could be encrypted to enforce integrity and unwanted stream acquisition and guarantee unforgability. Several encryption techniques can be used, such as the ones suggested in [KWJ03,KW02].

9 Conclusions

We showed that syntactic trees used in textual XML documents to specify access control policies are insufficient to specify access control policies for SMIL formatted multimedia documents. As a solution, we proposed that SMIL documents to be translated to a normal form similar to the DNF representation of propositional formulas.

Having resolved the issues of objects and their identity in SMIL, we presented a RDF metastructure to specify accedes control policies for multimedia documents. We have shown via examples the applicability of the structure for DAC, MAC, and RBAC. Our security normal forms are similar to secure views computed for XML and other textual documents. We present algorithms to compute normal forms. We showed a straw-man's design of a run-time that uses RDF and SMIL queries to securely retrieve documents decorated as specified by us.

Results presented here only consider limited aspects of security models with a fragments of SMIL syntax. Our ongoing work addresses these limitations and provide comprehensive security models. In addition, we are incorporation advanced Semantic Web technologies, like DAML+OIL [CHH01], OWL [DC03] and RuleML [BTW01].

References

[Aya01] Jeff Ayars. *Synchronized Multimedia Integration Language.* W3C Recommendation, 2001. http://www.w3.org/TR/2001/REC-smil20-20010807.

[BG03] Dan Brickley and R.V. Guha. *RDF Vocabulary Description Language 1.0:RDF Schema.* W3C Working Draft, January 23 2003. http://www.w3.org/TR/2003/WD-rdf-schema-20030123.

[BHAE02] Elisa Bertino, Moustafa Hammad, Walid Aref, and Ahmed Elmagarmid. An access control model for video database systems. In *Conferece on Information and Knowledge Management*, 2002.

[BMB02] Dave Beckett, Eric Miller, and Dan Brickley. *Expressing simple Dublin Core in RDF/XML.* Dublin Core Metadata Initiative, July 21 2002.

[BTW01] Harold Boley, Said Tabet, and Gerd Wagner. Design rationale of ruleml: A markup language for semantic web rules. In *SWWS, Stanford*, 2001.

[CHH01] Dan Connoly, Frank Harmelen, and Ian Horrocks. *DAML+OIL Reference Description.* W3C Note, 2001. http://www.w3.org/TR/daml+oil-reference.

[DC03] Mike Dean and Dan Connolly. *OWL Web Ontology Language Overview*, 31st March 2003.

[DdV03] Ernesto Damiani and Sabrina De Capitani di Vimercati. Securing xml based multimedia content. In *18th IFIP International Information Security Conference*, 2003.

[DdVPS00] Ernesto Damiani, Sabrina De Capitani di Vimercati, Stefano Paraboschi, and Pierangela Samarati. Securing XML documents. *Lecture Notes in Computer Science*, 1777:121–122, 2000.

[DdVPS02] Ernesto Damiani, Sabrina De Capitani di Vimercati, Stefano Paraboschi, and Pierangela Samarati. A fine grained access control system for xml documents. *ACM Transactions on Information and System Security*, 5, 2002.

[GNY$^+$01] Xiaohui Gu, Klara Nahrstedt, Wanghong Yuan, Duangdao Wichadakul, and Dongyan Xu. An xml-based quality of service enabling language for the web, 2001.

[Hay03] Patrick Hayes. *RDF Semantics.* W3C Working Draft, January 23 2003. http://www.w3.org/TR/2003/WD-rdf-mt-20030123.

[KC03] Graham Klyne and Jeremy Carroll. *Resource Description Framework(RDF) Concepts and Abstract Syntax.* W3C Working Draft, January 23 2003. http://www.w3.org/TR/2003/WD-rdf-concepts-20030123.

[KFW03] Naren Kodali, Csilla Farkas, and Duminda Wijesekera. Enforcing integrity in multimedia surveillance. In *IFIP 11.5 Working Conference on Integrity and Internal Control in Information Systems*, 2003.

[KW02] Naren Kodali and Duminda Wijesekera. Regulating access to smil formatted pay-per-view movies. In *2002 ACM Workshop on XML Security*, 2002.

[KWJ03] Naren Kodali, Duminda Wijesekera, and J.B.Michael. Sputers: A secure traffic surveillance and emergency response architecture. In *submission to the Journal of Intelligent Transportation Systems*, 2003.

[Mic01] Thierry Michel. *The SMIL 2.0 MetaInformation Module.* W3C Recommendation, 2001. http://www.w3.org/TR/2003/WD-rdf-mt-20030123.

[MM03] Frank Manola and Eric Miller. *RDF Primer.* W3C Working Draft, January 23 2003. http://www.w3.org/TR/2003/WD-rdf-primer-20030123.

[MS98] Ashok Malhotra and Neel Sundaresan. *RDF Query Specification.* W3C Specification, December 03 1998. http://www.w3.org/TR/2003/WD-rdf-primer-20030123.

[SFK00] Ravi Sandhu, David Ferraiolo, and Richard Kuhn. The NISI model for role-based access control: Towards a unified standard. In *ACM RBAC 2000*, pages 47–64, 2000.

[SS96] Ravi Sandhu and Pierangela Samarati. Access control: Principles and practices. *IEEE Communications*, 29(2):38–47, 1996.

[W3C03] *World-Wide-Web Consortium*, 31st July 2003.

[WS96] Duminda Wijesekera and Jaideep Srivastava. Quality of service (qos) metrics for continuous media. *Multimedia Tools and Applications*, 3(2):127–166, 1996.