Policy Based Access for Home Contents and Services *

Mohammad M. R. Chowdhury
University Graduate Center
Pb 70, N-2027 Kjeller
Norway
mohammad@unik.no

Sarfraz Alam
University Graduate Center
Pb 70, N-2027 Kjeller
Norway
sarfraz@unik.no

Josef Noll
University Graduate Center
Pb 70, N-2027 Kjeller
Norway
josef@unik.no

ABSTRACT

Policies are familiar approach to preserve security and privacy of the Web contents and services. This paper is going to address the policy based constrained access to the home contents and services but in the context of distributed but connected devices. A community concept equipped with trust metric is introduced to facilitate such restricted access. The community structure is maintained through a knowledge base exploiting Web Ontology Language and the policies are formulated using Semantic Web Rule Language. Reasoner then executes the policies to derive the access authorization results. In this paper, we provide a prototypical implementation of the whole scenario where a community member can download videos from the owner's home devices through a Web application. Besides, this paper critically investigates several challenges of the proposed approach with regard to various implementation issues.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Security

Keywords

Authorization, policy, rule, reasoning

1. INTRODUCTION

The explosion of digital devices, contents and services escalates users' access to contents and services manifold. Ubiquitous computing environment [18], [19] and capabilities make such access provisions more appealing. However in many cases, the characteristics of contents and services (e.g. age, trust restriction) require strict security and privacy assurance while accessing them. Contents like videos and musics are stored into different devices, and some of the services (e.g. mobile operator's services) can be accessed only from specific devices. The later we call in this paper as *device hosted service*. In addition, there are many Internet services (e.g. Internet Banking) which can be accessed using certificate/digital keys stored in devices. In this paper, we are concerned with the devices at home premises, most of which have wireless/wired connectivity.

With such infrastructures and service provisions in place, contents and services can be shared with other users through connected devices. In connection with this, Microsoft[1] and Cisco[2] initiated the concept of 'Connected Home'. Besides the need for intelligent network, it requires personalized control of access, a phenomenon often mentioned as the ability of 'empowered' users. To meet the access personalization demand and strict security and privacy assurance, we are proposing to build an environment that includes the notion of community. Such a community environment along with access authorization policies can ensure the privacy through controlling access to home contents and services. In this work, set-top box (STB) having wireless connectivity is assumed to be sitting at the center and managing access to the contents and services. It contains the community structure and authorization policies are maintained in a connected external server. Access by anonymous users may require importing their profiles & preferences from the external knowledge sources. But in this paper, we only consider providing access only to the registered users of the community. Figure 1 illustrates a scenario of this kind where all the devices are connected with the STB which contains an intelligent privacy layer.

The paper is organized as follows: section 2 introduces challenges for accessing home contents and services. Section 3 describes the policy design principles to deal with the proposed access scenario and the functional architecture of the solution is illustrated in section 4 with brief descriptions of each components. Section 5 presents the implementation of

*This work was supported in part by the Norwegian Research Council and ITEA through the SWACOM and WellCom project respectively.

CSTST 2008 October 27-31, 2008, Cergy-Pontoise, France

[1]Gates expands Microsoft's digital home plan, http://www.macworld.com/article/54725/2007/01/gates.html [accessed on July 15, 2008]

[2]Cisco's vision for the connected home, http://newsroom.cisco.com/dlls/2007/hd_010907.html [accessed on July 15, 2008]

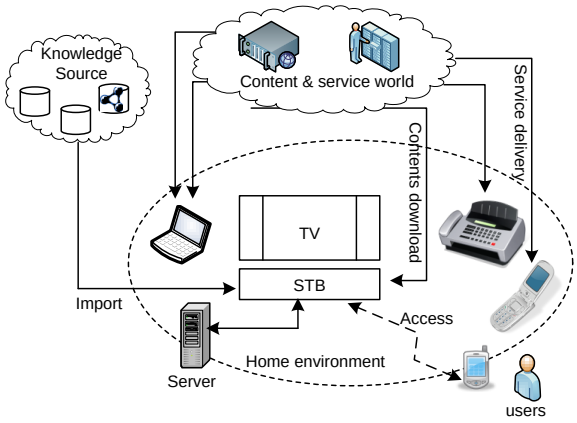

Figure 1: Home content and service access scenario.

these components and it is followed by analytical discussion of the proposed solution in section 6. Finally, in section 7 we conclude with key points of the paper and idea of future work.

2. CHALLENGES FOR ACCESS

The scenario described in the last section introduces crucial challenges in the content and service access. The deficiencies of capabilities in home infrastructure, the management of distributed content and service access, and the ways of creating community are the most critical challenges.

2.1 Home Infrastructure

One of the characteristics of the distributed content management systems is administering it from a single point of access[3]. STB is placed at the heart of the home environment from connectivity and management point of view. It has wireless connectivity through three different technologies: Bluetooth, WiFi and NFC (Near Field Communication). Advanced STBs[4] are equipped with processors, memory, middleware (application manager, virtual machine, interactive engine, libraries and databases), software applications (APIs, Application Servers etc.) and hard drives. However, its processing power and memory capacity are insufficient to manage and maintain the complexities related to the proposed content and service access. Moreover, STB has limited development tool support and the development is quite complex. These inadequacies also limit the number of devices it can connect simultaneously.

2.2 Distributed Content and Service Environment

Contents and services are distributed over the network and various network enabled devices (including STB). In case of sharing device hosted services, users need to access the hosted device from the devices they are using. Ensuring privacy in accessing distributed contents and services is a major concern. Controlling access through authentication and authorization deal with this issue but designing required access authorization policies is not a trivial job. Formulation and execution of complex but consistent policies are always challenging. The advent of mobile devices also demand personalized service access which requires inputs from users' profiles & preferences.

2.3 Creating Community

Community is a group of interacting individuals who share same characteristics or values. The notion of social capital [14] strengthens the concepts of modern-day community. The concept can also facilitate restricted access [12]. There are number of ways to categorize communities, figure 2 illustrates one such breakdown. Communities are created based on individuals' profile & preference inputs (e.g. location, interested, membership of groups etc.). These are either explicitly user defined or automatically generated from the contexts. The home environment described in figure 1 can be an ideal setting from where a community can be created based on the presence information (within the vicinity of STB). However, building communities based on location or presence information is somewhat challenging.

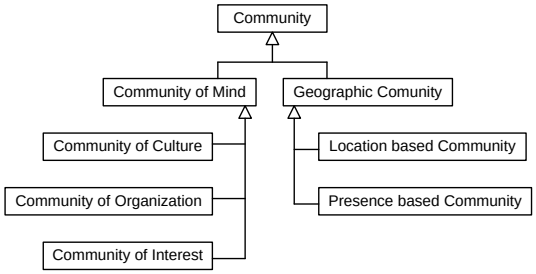

Figure 2: An example breakdown of community.

In this paper, the lack of capabilities of STBs are addressed by distributing the access management mechanism over two different places: STB and external server (detail description in sec.4). Access authorization and personalization aspects are realized through policies supported by the notion of community. Content owner explicitly creates a community based on their relationships and announces the available contents and services. Dynamic service discovery is not considered here. The details of the infrastructure and connectivity set up and management are also beyond the scope of this paper.

3. POLICY DESIGN PRINCIPLES

Security and privacy of the contents/services are ensured through access control mechanism. It contains two components: authentication and authorization. In this paper, we are concerned with the authorization only as there exist various established means of authentication. Access authorization determines a) what an user can do in a system (access to contents/services) and b) with which privileges? (e.g. download and/or streaming privileges for recorded videos). It is achieved through policies containing user defined restrictions. By definition, policy specifies who is allowed to perform which action on which object depending on properties of the requester and of the object [4].

Granularity of restrictions in policies is one of the main features of the design principles. The restrictions are added according to the user's requirements. Few services can be

[3]Distributed Content Management, http://www.content-management-junction.com/ [accessed on July 05, 2008]

[4]Set-top Boxes, The Interactive Television Institute, http://itvdictionary.com/set-top_box.html [accessed on July 05, 2008]

accessed only by the family members. While some other services need more restrictions through age or trust and these two may even be included in conjunction. The definitions of the policies used in this work are as follows,

Definition 1: Access to video sharing service -

Definition 1.1: *Everyone whose age is more than 15 and who is more trust worthy can access videos with privilege download and/or streaming.*

Definition 1.2: *Everyone whose age is less than 15 can access those videos which are restricted to less than or equal to 15 years of age with chosen privileges (download and/or streaming).*

Definition 1.3: *There are some videos which can be shared with those who are more than 15 but who are less trust worthy with streaming privilege only.*

Definition 2: Access to voice call service - *Only the members of the family can access the voice call service hosted to one of the devices (IP phone).*

Definition 3: Access to Internet service - *Everyone whose age is more than 15 and who is more trust worthy can access all the digital library resources (ACM digital library) with chosen privileges (download and/or read).*

Definition 4: Delegation of video download privilege - *Administrator can delegate download privilege of some videos to those who are more than 15 but who are less trustworthy (these persons otherwise had only streaming privilege of these videos).*

Only the *Definition 4* outlines the delegation policy of a specific privilege. Delegation is used to furnish the temporary transfer of access rights to somebody else acting on behalf of someone. Temporary delegation requires explicitly adding time or duration of transfer which we have not considered here. In this work, the rights would be delegated until the administrator revoked that specific delegation. Here the administrator has the right to choose the privileges held by each role and needed by each content/service.

4. FUNCTIONAL ARCHITECTURE

This section provides functional architecture, a brief description of each of its components and how they interact with each other. Figure 3 depicts an architectural overview of the proposed framework. The core components of the framework consist of (i) Enforcement Point, (ii) Home Knowledge Base, and (iii) Semantic Knowledge Server, where the first two are located in STB.

Enforcement Point The enforcement point is sub-divided into three components:

Authentication Handler User authentication is managed through authentication handler. By incorporating an external authentication engine, the framework ensures the flexibility of using different authentication methods ranging from simple username/password to SIM-based authentication. Besides, the authentication handler sends login failure response to the user if user is not authenticated.

Query Handler When a user is authenticated, it constructs the outbound query expression based on user attributes and original HTTP or SOAP request and acts as a delegator for the requester to send query to the knowledge base. It has a SPARQL [13] (a query language for RDF) front end to generate and process SPARQL queries.

Enforcement Module It works as proxy that enforces authorization decisions after processing the SPARQL query results and formulating the appropriate non-semantic HTTP or SOAP response for the user.

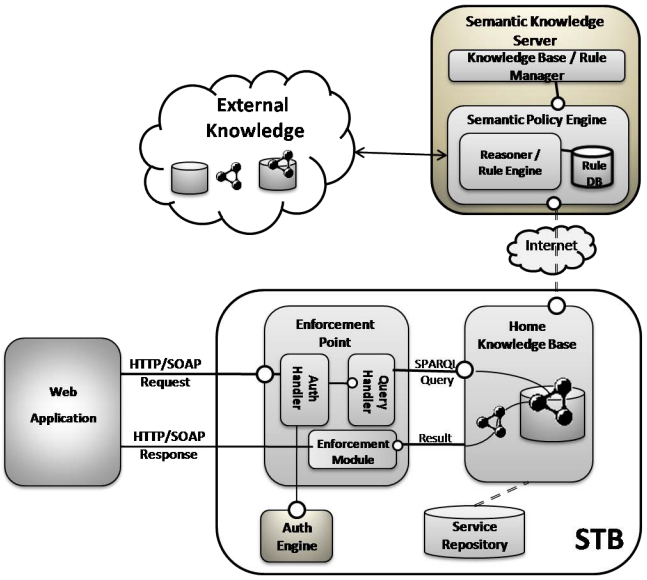

Figure 3: Proposed functional architecture.

Home Knowledge Base It is the formal representation of various actors involved in a community (detail description in 5.1). In practice, it is stored in RDF data store but in order to boost the performance and to overcome the resource limitation of STB, we converted the ontology to RDF/XML serialization and stored as a file. The code snippet of the serialization is provided in figure of sec. 5.3. Semantic Knowledge Server updates the knowledge here with the results derived through periodical or event-based reasoning.

Semantic Knowledge Server It consists of (i) Knowledge Base/Rule Manager, and (ii) Semantic Policy Engine. The Knowledge Base/Rule manager provides an interface to the administrator to edit/update the knowledge base and access rules.

Semantic Policy Engine Semantic Policy engine (SPE) along with Home Knowledge Base can prevent anonymous user from accessing home content/services regardless of authentication methods used. Instead of real time reasoning the framework uses an event based reasoning through rule execution approach. These events are the rule addition, rule modification, knowledge base modification etc. As an example, with the addition of a new rule, the engine starts executing the rule and inferred facts are supplied to the home knowledge base (i.e, on STB). While executing, the engine also imports existing knowledge from the knowledge base (optionally from external knowledge sources, if it requires), as the rules are expressed in terms of knowledge elements. To implement SPE we use Pellet inference/rule engine that takes the SWRL rules from rule database.

5. IMPLEMENTATION AND THE RESULTS

5.1 Home Knowledge Base Representation

The proposed knowledge base comprises of a community structure containing identities of users (by unique name only), roles, their attributes (age and trust level), privileges of roles, registered contents or services and the relationships

Table 1: The list of properties, their domains and ranges.

Property	Domain	Range	Example
hasRole	Identity	Role	$P\,(Anne, Child\_Anne)$
hasAge	Identity	{10..60}	$P\,(Anne, 14)$
hasAccessTo	Identity	Service	$P\,(InferredInstances)$
canDelegateTo	Identity	Identity	$P\,(InferredInstances)$
hasprivilege	Role	Privilege	$P\,(Child\_Anne, Download)$
hasTrustLevel	Role	{0.1..1.0}	$P\,(Anne, 0.9)$
needPrivilege	Service	Privilege	$P\,(Video1, Download)$
restrictedToUnder	Service	{10..60}	$P\,(Video1, 15)$
subscribedBy	IS	Identity	$P\,(ACM\_DL, Liz)$
hostedToDevice	DHS	Device	$P\,(Video1, DID21\_HDD)$

between all these entities. It is formally represented through ontology using the Web Ontology Language (OWL)[15]. While designing the ontology, we followed nearly the similar conceptual understanding of [3] and we used Description Logic[5] syntax to describe it. An ontology is a set of *classes* C, *properties* P and *instances* i. In ontology, the key concepts of the domain (content sharing through *community* environment) are defined through *classes*. In this work, concepts hold the *subClassOf* relation among them which is defined as, $SC\,(C_1) \subseteq SC\,(C_2)$, the semantic scope of C_1 is narrower than that of C_2, where C_1 and C_2 are two classes.

Figure 4 illustrates the main classes and properties of the ontology. *Role*, *Device_Hosted_Service (DHS)* and *Internet_Service (IS)* are further divided into the following subclasses (VSS = *VideoSharingService*, VCS = *VoiceCallService*, $ODLS$ = *OnlineDigitalLibraryService*),
$\{SC\,(Family), SC\,(Relative), SC\,(Friend)\} \subseteq SC\,(Role)$;
$\{SC\,(Parents), SC\,(Children)\} \subseteq SC\,(Family)$;
$\{SC\,(VSS), SC\,(VCS)\} \subseteq SC\,(DHS)$;
$SC\,(ODLS) \subseteq SC\,(IS)$

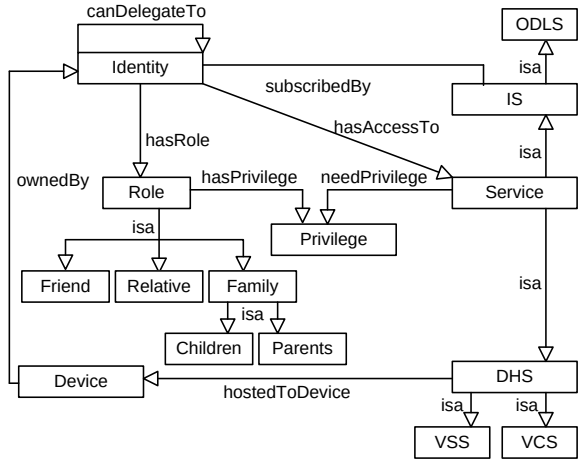

Figure 4: Formalization of knowledge base through classes and properties.

Here roles represent the relationships and services are classified based on their types. We included various service types in the knowledge base for future extension of the proposed solution. The real actors (or individuals) of a practical use case scenario are defined through the instances and they belong to the classes ($\{i_1, i_2, ..., i_n\} : SC\,(C)$). The instances of the ontology and which classes (or subclasses) they belong to are as follows,
$\{Erik, Liz, ...Josef, Najeeb\} : SC\,(Identity)$;
$\{Father\_Erik, Mother\_Liz\} : SC\,(Parents)$;
$\{Child\_Anne, Child\_Paul\} : SC\,(Children)$;
$\{Relative\_Josef, Relative\_George\} : SC\,(Relative)$;
$\{Friend\_Najeeb, Friend\_Sarfraz\} : SC\,(Friend)$;
$\{Video1, Video2, Video3\} : SC\,(VSS)$;
$\{Phoning\} : SC\,(VCS)$
$\{ACM\_DL\} : SC\,(ODLS)$
$\{Admin, Download, Messaging, Streaming, Read, Call, ..\} : SC\,(Privilege)$
$\{DID11\_Mobile, DID12\_IPPhone, DID21\_HDD..\} : SC\,(Device)$

In the ontologies, a *property* relates two instances of the classes. Property, $P\,(i_1, i_2)$ states that i_1 relates with i_2 through the property P. Properties have a *domain* and *range*. Syntactically, *domain* links a property to a class and *range* links a property to either a class or a data range [15]. A property relates instances from the *domain* with the instances from the *range*. Table 1 lists the properties used in the ontologies. The relationships through *hasAccessTo* and *canDelegateTo* properties are inferred upon reasoning. *hasAge*, *hasTrustLevel* and *restrictedToUnder* are the datatype properties which take the data (integer or float) as range. The remaining properties are object type which take classes as range.

The level of trust is included through trust metric [11]. Trust metric is a measure of how an individual trusts his friends. In this paper, it is realized through the values carried by *hasTrustLevel* as range (0.1..1.0). Trust is a subjective quality and the measure of trust metric depends on various characteristics. To avoid additional complexities, we assume that the system administrator or the resource owner sets the metric values simply based on faith and confidence on others.

5.2 Formulation of Policies

The access authorization and delegation policies are realized through rules. These rules are formulated exploiting the Semantic Web Rule Language (SWRL)[6]. The rules are written using the classes, properties and instances of the knowledge base, and are defined as a set of antecedent and consequent parts containing conjunctions of atoms. If all the atoms in the antecedent are true, then the consequent must also be true. Within rules, we also use SWRL built-ins

[5]Description logics (DL) are a family of knowledge representation languages which can be used to represent the terminological knowledge of an application domain in a structured and formally well-understood way.

***Rule 1.1*:**
Identity(?ID) ∧ hasRole(?ID, ?R) ∧ Role(?R) ∧ hasAge(?ID, ?y) ∧ swrlb:greaterThan(?y, 15) ∧ hasTrustLevel(?R, ?x) ∧ swrlb:greaterThan(?x, 0.7) ∧ hasPrivilege(?R, ?Y) ∧ Video_Sharing_Services(?Z) ∧ needPrivilege(?Z, ?Y) → hasAccessTo(?ID, ?Z)
***Rule 1.2*:**
Identity(?ID) ∧ hasRole(?ID, ?R) ∧ Role(?R) ∧ hasAge(?ID, ?y) ∧ swrlb:lessThan(?y, 15) ∧ hasPrivilege(?R, ?Y) ∧ Video_Sharing_Services(?Z) ∧ restrictedToUnder(?Z, ?x) ∧ swrlb:lessThanOrEqual(?x, 12) ∧ needPrivilege(?Z, ?Y) → hasAccessTo(?ID, ?Z)
Rule 1.3:
Identity(?ID) ∧ hasRole(?ID, ?R) ∧ Role(?R) ∧ hasAge(?ID, ?y) ∧ swrlb:greaterThan(?y, 15) ∧ hasTrustLevel(?R, ?x) ∧ swrlb:lessThan(?x, 0.7) ∧ hasPrivilege(?R, ?Y) ∧ Video_Sharing_Services(?Z) ∧ needPrivilege(?Z, ?Y) → hasAccessTo(?ID, ?Z)
Rule 2:
Identity(?ID) ∧ hasRole(?ID, ?R) ∧ Family(?R) ∧ hasPrivilege(?R, ?Y) ∧ Voice_Call_Services(?Z) ∧ needPrivilege(?Z, ?Y) → hasAccessTo(?ID, ?Z)
Rule 3:
Identity(?ID) ∧ hasRole(?ID, ?R) ∧ Role(?R) ∧ hasAge(?ID, ?y) ∧ swrlb:greaterThan(?y, 15) ∧ hasTrustLevel(?R, ?x) ∧ swrlb:greaterThan(?x, 0.7) ∧ hasPrivilege(?R, ?Y) ∧ Online_Digital_Library(?Z) ∧ needPrivilege(?Z, ?Y) → hasAccessTo(?ID, ?Z)
Rule 4:
Friends(?FR) ∧ hasRole(?ID, ?FR) ∧ hasAge(?ID, ?y) ∧ swrlb:greaterThan(?y, 15) ∧ hasTrustLevel(?FR, ?x) ∧ swrlb:lessThan(?x, 0.7) ∧ Video_Sharing_Services(?S) ∧ Parents(?PR) ∧ hasRole(?I, ?PR) → canDelegateRoleTo(?I, ?ID) ∧ accessWithPrivilege(?S, Download)

Figure 5: Rules representing access authorization and delegation policies.

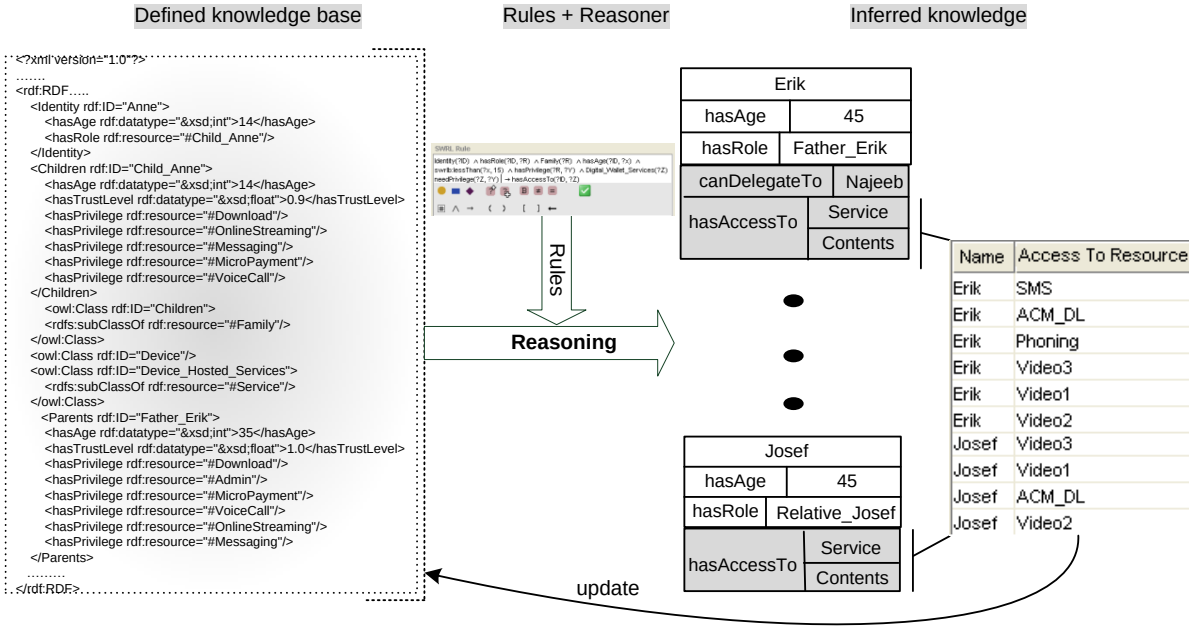

Figure 6: The complete process of reasoning and authorization results.

(swrlb:greaterThan, swrlb:lessThan, swrlb:lessThanOrEqual) which includes the basic mathematical operations. Apart from generating the authorization and delegation policy decisions, the environment can also check the consistency and validity of the knowledge base and the policies. Rule editor platform (SWRL plug in) can check the validity of the policies while editing them. With the addition of (new) instances if the execution of rule fails to generate expected results, the consistency of the ontology is in question. It requires complete review of the ontology. The rules representing our policy principles (Sec. 3) are defined in figure 5. Trustworthiness threshold has been set at 0.7. The value is randomly chosen (any of the values greater than 0.5) and also somewhat based on intuition. Rule 1.1 to 3 are designed according to the corresponding access authorization policy definitions and rule 4 stands for the delegation policy definition. The privileges associated with the roles and service/contents are already selected by the administrator within the knowledge base. As the community is small in size and there are not many policies involved, we are not considering conflicting policies and strategies to avoid the conflicts in this paper.

5.3 Results of the Reasoning

The reasoner updates the knowledge base with the results derived from executing the rules. The results infer the instances associated with the properties, *hasAccessTo* and *canDelegateTo*. They indicate 'which contents or services an user can access' and 'who can delegate a specific privilege of accessing contents to whom' respectively. The inferred instances are exported back to the knowledge base because the SPARQL queries sent by the enforcement point would require these answers. As we have not considered the duration of delegation, to revoke the delegation administrator has to delete the *canDelegateTo* relationship explicitly from the knowledge base. Figure 6 illustrates the complete reasoning process.

5.4 User Interface

We are currently working to develop a user interface of the application through which one can access the home con-

tents/services using the Web. Figure 7 illustrates an early prototype of such user interface where we only tried to share videos (download) with Erik. It shows the links of the videos which can be downloaded when Erik has been authenticated. In response to the HTTP/SOAP request from the application, Enforcement Point sends SPARQL queries and processes the results. We used Exhibit API[6] to process these results. The code snippet of constructed query to answer, *What resources/services are accessible by user and to whom a user can delegate its access right?*, is given below:

```
PREFIX  SemID:  <http://myhomecontent.com/SemID#>
SELECT  ?ID  ?hasAccessTo  ?canDelegateRoleTo
FROM  http://myhomecontent.com/SemID.owl
WHERE
{
?Identity SemID:ID ?ID
?Identity SemID:hasAccessTo ?hasAccessTo
?Identity SemID:canDelegateRoleTo ?canDelegateRoleTo
}
```

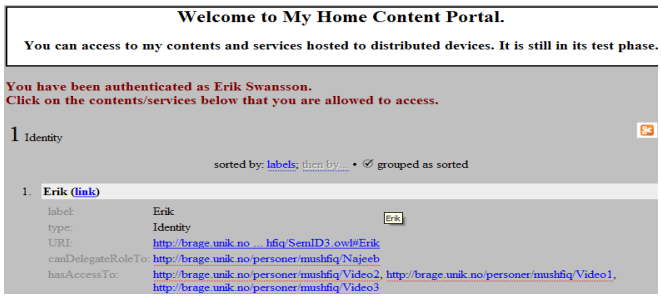

Figure 7: Prototypical user interface.

6. DISCUSSION

This section introduces analytical discussions on some performance issues related to the proposed architecture. It also evaluates the suggested policy design principles and methodologies, and compares them with other existing policy language features.

Due to the resource limitations (sec.2), STB is not capable to perform the reasoning process to answer the queries it receives. That is why, the reasoner has been placed at an external server (Semantic Knowledge Server). The STB holds only the Home Knowledge Base which is updated with the results of the reasoning process. Figure 8 shows the reasons why STB is incapable of doing reasoning. In this figure, we include the time required for computing inferred instances and for updating the knowledge base with the increase of content/service instances. We also measured the time for single as well as multiple simultaneous queries. All the measurements were done in a desktop PC based on Windows XP with a P4 2.0 GHz processor and 1 GB RAM which is much more powerful than the STBs.

Latency[7] is one of the QoS requirements for Web services [8]. Apart from network delays, latency includes the request processing time. According to Cisco, latency for streaming-video should be no more than 4 to 5 seconds [16]. The figure shows that for fairly small number of contents/services, only the reasoning and update processes are taking quite a significant amount of time. The situation aggravates for multiple simultaneous queries. The reasoning process is impractical with current hardwares in STBs. Considering the reasoning performance, the real time reasoning with the reasoner located in server is still a challenging issue. Therefore, to reduce the latency in service delivery, we suggest here periodical or event-based reasoning.

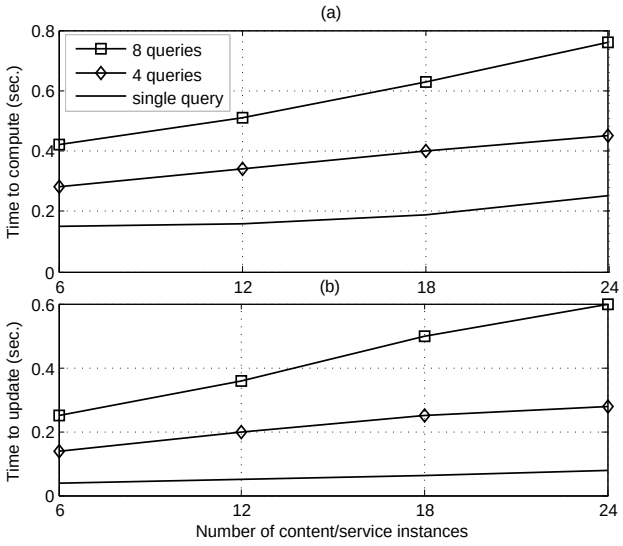

Figure 8: (a) Time to compute inferred instances, (b) Time to update knowledge base.

A concept of trust was brought in [2], [5] to provide access to community resources and privacy solutions. Massa in his paper [11] suggested the use of trust metric to represent the closeness among individuals. In this work, we included this through trust metric while designing the authorization policies. Though the metrics are static, the values can be dynamically updated from the distributed trust management approach [5] stored somewhere in the external knowledge source.

Policies are familiar approach to protect security and privacy of users in distributed systems [4] and has been the subject of extensive research in recent years [20]. XACML[10], WSPL[1], KAoS[17] and Rei[7] are some of the most prominent policy languages. In [9], the authors suggested expressing the access control policies based on OWL and SWRL citing the lack of formal semantics in XACML. The proposed solution used OWL with description logics (DL) which provides the required expressivity (for decision support) through automated reasoning. Whereas, Rei is based on OWL Lite which is less expressive than OWL DL. KAoS and Rei do not support execution of action but here the proposed policies allow the designers to specify actions within the policies. A basic delegation support is included within this work, KAoS and XACML do not support delegation of rights. Policy specifications with OWL and SWRL also allow the user to adapt the policies to his current needs, such extensibility is one of the policy design criteria [4].

Most of the policy languages are intended for controlling access to Web services. In this paper, we have proposed a solution that integrates policies destined to support constrained access not only to services but also to contents

[6]Exhibit 2.0, http://simile.mit.edu/exhibit/ [accessed on July 05, 2008]

[7]Latency is the round-trip delay between sending a request and receiving the response.

stored in distributed devices. Distributed content management solution also alleviate the need for centralized content repositories.

7. CONCLUSION

This paper suggests a policy based approach to protect home contents and services through access authorization provision. The area of its application is particularly important because we have addressed the constrained access situations in the context of distributed but connected devices. The set-top box handles the connectivity, and manages the access of contents and services hosted in devices. As the owners intend to share the contents and services with family, friends and relatives, we brought in the concept of community to design the policies. Along with trust metric it can enhance the granularity of the policies to meet the increased security requirements. This paper provides a prototypical implementation of the whole scenario where a community member can download videos from the owner's home devices through a Web application.

Due to the resource limitations of STBs, the policies and the reasoning processes are maintained in an external server. Even though the real time reasoning is a challenging task. As a future work, we are planning to extend the proposed architecture to meet this goal.

8. REFERENCES

[1] A. H. Anderson. An introduction to the Web Services Policy Language (WSPL). In *5th IEEE International Workshop on Policies for Distributed Systems and Networks.*, pages 189–192, June 2004.

[2] H.-C. Choi, S. R. Kruk, S. Grzonkowski, K. Stankiewicz, B. Davis, and J. G. Breslin. Trust models for community-aware identity management. *Identity, Reference and the Web IRW2006, WWW2006 Workshop.*, May 2006.

[3] M. M. R. Chowdhury, J. Chamizo, J. Noll, and J. M. Gómez. Capturing semantics for information security and privacy assurance. *UIC 2008, LNCS 5061, Springer 2008.*, pages 105–118, June 2008.

[4] J. L. D. Coi and D. Olmedilla. A review of trust management, security and privacy policy languages. In *International Conference on Security and Cryptography (SECRYPT 2008) Proceedings.*, July 2008.

[5] T. Finin and A. Joshi. Agents, trust, and information access on the Semantic Web. *ACM SIGMOD, Special Issue: Special section on semantic web and data management.*, 31(4):30–35, December 2002.

[6] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean. SWRL: A Semantic Web Rule Language combining OWL and RuleML. *W3C Member Submission*, May 2004.

[7] L. Kagal, T. Finin, and A. Joshi. A policy language for a pervasive computing environment. *Proceedings of the 4th IEEE International Workshop on Policies for Distributed Systems and Networks.*, pages 63–74, June 2003.

[8] K. Lee, J. Jeon, W. Lee, S.-H. Jeong, and S.-W. Park. QoS for Web services: requirements and possible approaches. *W3C Working Group Submission.*, November 2003.

[9] H. Li, X. Zhang, H. Wu, and Y. Qu. Design and application of rule based access control policies. In *4th International Semantic Web Conference (ISWC 2005) Proceedings.*, pages 35–41, November 2005.

[10] M. Lorch, S. Proctor, R. Lepro, D. Kafura, and S. Shah. First experiences using XACML for access control in distributed systems. *Proceedings of the 2003 ACM workshop on XML security.*, pages 25–37, 2003.

[11] P. Massa and P. Avesani. Trust-aware collaborative filtering for recommender systems. *Proceeding of ACM Recommender Systems Conference.*, pages 492–508, 2007.

[12] L. Pearlman, C. Kesselman, V. Welch, I. Foster, and S. Tuecke. The community authorization service: Status and future. *Proceedings of the Conference for Computing in High Energy and Nuclear Physics (CHEP03).*, pages 24–28, July 2003.

[13] E. Prudhommeaux and A. Seaborne. SPARQL query language for RDF. *W3C Recommendation.*, January 2008.

[14] R. D. Putnam. Bowling alone: America's declining social capital. *Journal of Democracy 6.1.*, pages 65–78, Jan 1995.

[15] M. K. Smith, C. Welty, and D. L. McGuinness. OWL Web Ontology Language guide. *W3C Recommendation.*, February 2004.

[16] T. Szigeti and C. Hattingh. Quality of service design overview. *Cisco press, http://www.ciscopress.com/articles/article.asp?p=357102&seqNum=2. [accessed on July 05, 2008]*, December 2004.

[17] A. Uszok, J. M. Bradshaw, R. Jeffers, N. Suri, P. Hayes, M. Breedy, L. Bunch, M. Johnson, S. Kulkarni, and J. Lott. KAoS policy and domain services: Toward a description-logic approach to policy representation, deconfliction, and enforcement. *Proceedings of the 4th IEEE International Workshop on Policies for Distributed Systems and Networks.*, pages 93–96, June 2003.

[18] M. Weiser. Hot topics: Ubiquitous computing. *IEEE Computer.*, pages 71–72, October 1993.

[19] M. Weiser. The computer for the 21st century. *ACM SIGMOBILE Mobile Computing and Communications Review.*, 3(3):3–11, July 1999.

[20] S. Wright, R. Chadha, and G. L. (eds.). Special Issue on policy based networking. *IEEE Network.*, 16(2):8–56, 2002.