CONTEXT DRIVEN, USER-CENTRIC ACCESS CONTROL FOR SMART SPACES

M White, B Jennings, V Osmani and S van der Meer

Telecommunications Software & Services Group
Waterford Institute of Technology, Ireland
{mwhite, bjennings, vosmani, vdmeer}@tssg.org

ABSTRACT

In this paper, we describe a user-centric access control process for devices and services in smart space environments. The M-Zones Access Control (MAC) Process is driven by context information relating to the activities of the users present in a smart space, and by user-defined policies reflecting these users' preferences. As well as dynamically assigning access rights in response to context changes, the process provides for automatic reconfiguration of resources in order to protect a user's privacy as other users enter/leave his/her vicinity. To illustrate the implementation of the process we discuss its realisation in a test bed emulating an office-based smart space.

1 INTRODUCTION

Ubiquitous computing environments are characterised by the use of a wide range of user interfaces, for example, communal displays, voice-based command interfaces, and gesture recognition systems. The presence of such interfaces heightens users' awareness of, and requirements for, privacy protection measures. When using such interfaces, users may desire that the system automatically reacts to the environment in which they are present, in particular so that their privacy concerns are addressed. The presence of individual or groups of users in an environment constitutes a key element of the environment context. In many cases the process of defining what actions need to be taken will depend on the relationships between a user and these other individuals. Furthermore, a user's access rights should be determined based on ongoing analysis of the environment in which they are present, including analysis of the access rights and activities of other individuals present in his/her vicinity.

In this paper we investigate how a management system for a ubiquitous computing environment can control access rights in a manner that adapts to the preferences of the user set present, or active, within a physical space, but within the constraints imposed by system-wide management policies. In particular, we focus on the use of context information as a key trigger for the reconfiguration of services and resources in order to adapt user access rights and protect user privacy. Users are themselves afforded the opportunity to define policies embodying their preferences for actions to take place based on changes in the environment they are present in. A user policy could specify a preference for particular actions to take place if certain events occur in the presence of specific individuals, individuals with specified roles, or individuals possessing specified access rights.

The paper is structured as follows. First, we briefly review previous work on access control approaches in ubiquitous computing environments, focussing in particular on utilisation of context information as a means of defining this behaviour. We then describe the operation of a Ubiquitous Management Architecture (UMA) with particular emphasis on the use of a Context Information Manager (CIM) as a provider of context information. The M-Zones Access Control (MAC) process is then presented, and its integration with a policy-based management system to allow for the reconfiguration of services and resources to protect user privacy is described. We outline the implementation of the CIM and MAC process in a test bed emulating an office-based environment, and describe its operation in the context of a specific use case scenario. Finally, we summarise the benefits of the proposed approach and outline future work.

2 ACCESS CONTROL IN UBIQUITOUS COMPUTING ENVIRONMENTS

Many systems employ role-based access control, in which users are assigned one or more roles, typically mapping to functions in an organisational hierarchy, with each role being associated with a defined profile of access permissions. Whilst offering flexibility and relatively low management overhead, it can be argued that this approach does not provide the fine-grained control required in many ubiquitous computing applications. The ability to assign access rights based solely on a user's organisational role is insufficient in many ubiquitous computing environments where other attributes are likely to be more relevant when assigning access rights. Typical attributes that might be desirable as conditions in defining access rights include a user's current activity (for example, meeting chairperson), the resources he/she is currently accessing, and other contextual information, such as his/her project affiliation, or the presence of unauthorised users in the environment. To overcome these drawbacks, access control decisions can instead be based on specific attributes associated with a user, rather than on the user's identity or assigned organisational role(s). These attributes can relate to user profile values, the current activity of the user or outline information about the environment specifically. This attribute based approach is at the core of the eXtensible Access Control Markup Language (XACML) (1), which has been used to provide access control in systems such as PRIMA (2) and Cardea (3).

Ideally, access control solutions would automatically adjust access rights based on the changing context in which users are requesting access to resources; this is known as context-based access control. From an access control perspective, user location, presence of a group of users in a location, the relationship between the users within such groups, set of services currently available and the particular activities users are engaged in, are likely to be of most relevance. Basing access control decisions on this kind of information is an important research topic presently; for example, Corradi at al. (4) have developed UbiCOSM, a context-centric access control middleware that assigns access rights taking into account context, user profiles and system/user-level authorisation policies. Also, Sampemane at al. (5) address aspects of context-based access control for ubiquitous computing environments, describing a system that changes access rights depending on the set of users and the activity being undertaken in a physical space.

Based on the above observations we conclude that access control systems for ubiquitous computing environments should be:

- user-centric: allow users the freedom to adjust access rights as their needs evolve;
- attribute-based: access control decisions should be based on evaluation of appropriate user attributes, not rigidly on their identity or pre-assigned role(s);
- context-driven: access rights should be dynamically assigned based on analysis of context information provided by the environment.

To realise these properties we employ a policy-based management approach in which access rights assignment, as well as resource configuration based on access rights, is achieved through context-driven analysis of system and user-specific policies.

3 UBIQUITOUS MANAGEMENT ARCHITECTURE

The access control solution we propose has been realised in the context of a "Ubiquitous Management Architecture" (UMA) (6), developed as part of the M-Zones research programme (7). The UMA approaches management of ubiquitous computing environments, specifically smart spaces, by introducing the concept of "Managed Zones" (M-Zones) corresponding to administrative domains encompassing one or more distinct smart spaces. The UMA adopts a policy-based management (PBM) approach to facilitate intra- and inter- smart space management. The concept of PBM is being widely adopted in the Telecommunications and Internet Management area, for example the IETF have developed a PBM standard for access control (8). In the UMA Policy Decision Points (PDPs) (where policy rules are triggered in response to events and appropriate re-configuration operations generated) are organised in two levels, following the hierarchical approach described by Ghamri-Doudane et al. (9). The PDP at the upper (M-Zone) level is responsible for all high level policies relating to the administration of the smart spaces. When a policy decision is required at the upper (M-Zone) level the PDP interprets the relevant policies and sends the decision in the form of modified low-level policies to the smart space(s) in question. The PDP at the smart space level is then responsible for enforcing the low level policies, generating policy decisions in the form of configuration operations that are expedited by Policy Enforcement Points (PEPs). The smart space PDPs and PEPs also control the discovery and execution of services. Applying this approach to access control we have ongoing decision making relating to access rights occurring at the M-Zone level (via the MAC Process), with access rights being communicated to individual smart spaces in the form of access control lists (in effect low-level policies), which are enforced by the local PDP and PEPs.

The MAC Process forms part of the M-Zones PDP and is responsible for reaching access control policy decisions and for collating relevant information from other UMA components needed to inform these decisions. There are two other UMA components involved in access control at the M-Zone level: the CIM and Personal Information Managers (PIMs) associated with users. The CIM is an integral part of the architecture and is further explained in §3.1; it is responsible for gathering, aggregating and semantically enhancing context information subscribed to by the MAC Process and for notifying the MAC Process when context changes pertaining to the environment and users occur. Each user has associated with him/her a PIM, which stores their user profiles, preferences and policies, and also acts as their interface to the system. The MAC Process interacts with the CIM and PIM in order to assign access rights based on policies relating to the smart spaces themselves – “system” policies, and “user” policies (retrieved from the PIMs of users currently present in the smart space), in response to context change events notified by the CIM and/or events outlining changes in a users profile including changes to a users personal preference policies notified by the PIM.

3.1 Context Information Manager Operation

The CIM is responsible for collecting context information provided by diverse sources, then aggregating and processing this information, so that is structured in a manner amenable for use by context-aware entities. The operation of the CIM is illustrated in Figure 1 below. It makes use of the smart space repository, which contains a listing of all available services and their associated context information, so that it gets a consistent view of the environmental context, maintaining and updating this view whenever a change is detected. The CIM responds to two types of requests for context information: ad-hoc and subscription-based. Ad-hoc requests are simpler in nature and result in a single piece of context information being delivered to the requesting entity. In certain cases CIM may need to perform the conversion of the data into the format appropriate for use by the requesting entity (for example converting spatial coordinates into physical locations). Subscription-based requests are more complicated, since context-aware applications specify a set of conditions that are monitored actively by the CIM, so the response is sent only when the specified conditions hold true. The MAC process makes use of the subscription based requests.

The CIM provides context information that can be sensed directly from the environment (for example, temperature or user location). However, a more challenging task is to provide context information that cannot be sensed directly, for example, a meeting taking place between a group of users. Since such events cannot be observed directly, the CIM uses a set of rules that govern the outcome of an inference process, which, in conjunction with information retrieved from the environment, can be applied to deduce useful results for exploitation by context-aware applications. In the existing implementation the only entity requesting context information from the CIM is the M-Zone PDP’s MAC process, which configures CIM to pass on context notifications expressing them using a set of ontological terms pre-agreed between the CIM and the MAC process.

In terms of communications, the CIM employs a publish/subscribe mechanism. The MAC process initially sends a request to the CIM with the context information it is interested in. This request specifies a set of rules that are of relevance, so that notifications will be sent when those rules are assessed to be true. The main interface to the CIM is the component that listens for context requests from other entities. As soon as the request is made, the process of negotiating a common ontology begins. Once the CIM and the requesting entity have agreed on the ontology to use, the request can be parsed. Since the CIM responds to two types of requests, ad-hoc and subscription-based, the parsing process determines the type of request. Ad-hoc requests will typically involve retrieving data from one context source, in contrast with the more complex subscription-based requests that requires the aggregation of data from multiple context sources. As the CIM collects, processes and monitors context information from the environment it analyses whether it matches with any of the rules specified by its subscribers. When a match occurs, the CIM notifies the subscribing component, in this case the MAC process, the event description and additional information deemed relevant to the subscriber with regards to that particular event.

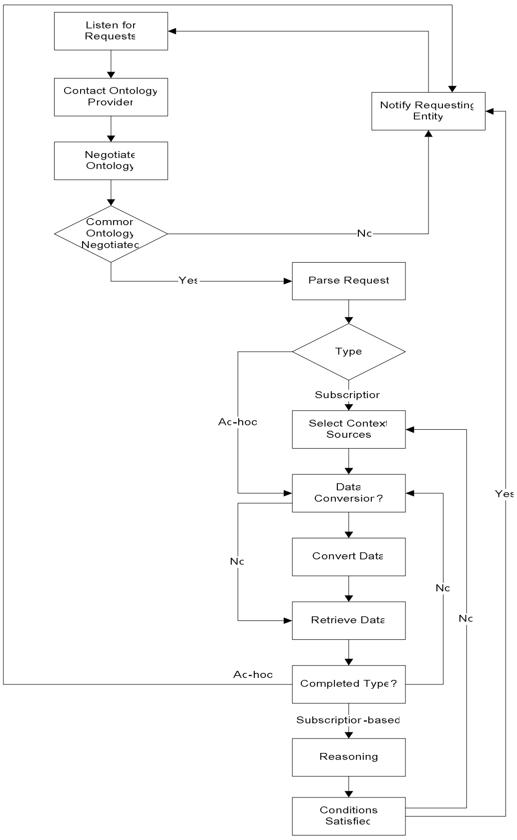


Figure 1: CIM process flowchart

3.2 MAC Process Operation

The MAC process is realised via a XACML policy engine (10), which allows for reaching policy decisions on the basis of the value of arbitrary user attributes. In our case these attribute values are stored in user PIMs, or are values of attributes of the environment itself (the latter being notified by CIM context events). As well as re-configuring access rights, the MAC process analyses whether user policies indicate that specified actions be requested as a result of the notified event. For example, users may desire that configuration actions be taken to protect their privacy in the presence of users unknown to them. Once the process has been completed PDPs at the smart space level are forwarded new access control lists to be enforced for users currently in the space, and configuration actions generated from policy analysis.

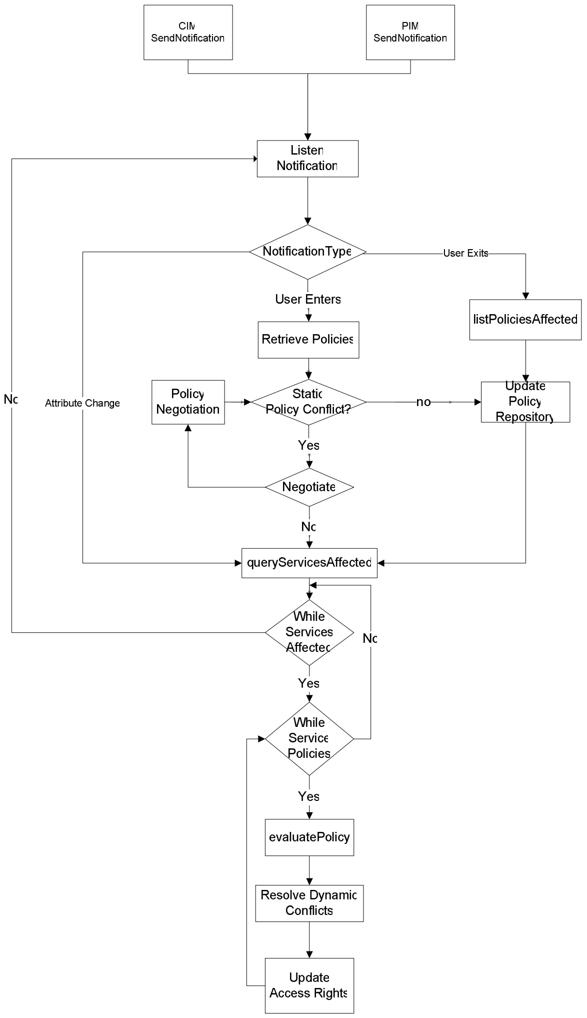


Figure 2: MAC process flow chart.

Figure 2 above outlines the flow concerning the access control decision mechanism implemented by the MAC process. The MAC process subscribes to the relevant components (PIM/CIM) in order to be notified should an event occur that requires access control decisions. Events that would typically be monitored include the entry/exit of users, or the changing of an attribute value that was considered as a condition of an active policy. When a user enters the smart space in question the MAC process retrieves the relevant user policies from the respective PIM. Static policy conflict detection is then carried out with respect to system and user policies in order to enforce the appropriate set of policies (see below for a description of this part of the process). The MAC process then evaluates the respective access rights for the Smart Space resources. This is achieved by iterating through the applicable policies governing access to each Smart Space resource. Should the event the MAC Process be notified of a change in an attribute value that forms the condition of an active policy, this again necessitates the identification of the resources affected and the re-evaluation of the various resource polices in order to reconfigure the relevant access rights. Central to the success of the MAC process in providing an adaptive access control is its ability to detect and resolve conflicts between various user and system policies that may occur in certain operational contexts. Conflict detection involves the identification of actual or potential policy conflicts. In general two approaches to conflict resolution are possible: revoke, re-specify and re-deploy offending policies, or let the system assign precedence levels that dictate which of the conflicting policies are actually invoked, with the latter approach being more practical. Precedence can be assigned based on, for example, specific policies overriding general policies; newer policies override older policies; or policies specified by a higher authority overriding those specified by lower authorities. As described in (11), we have enhanced the XACML policy engine's policy conflict/detection resolution facilities to realise configurable policy precedence schemes, by allowing specification of sets of attributes based on which precedence can be evaluated. Thus, how a scheme is implemented will be environment-specific, but will be based on appropriate attributes, for example security levels, or date/time. Our implementation targets an office environment (see §4.1), and ranks policies based on the resource in question, the policy author's organisational role, the policy author's project affiliation, the policy author's current activity, and the project context in which resource is being accessed.

4 IMPLEMENTATION AND CASE STUDY

Figure 3 below illustrates the test bed in which the MAC process has been realised. For the test bed, the Ubiquitous Management Architecture has been partially deployed onto the TSSG/O_2 Home of the Future ubiquitous computing environment (12). Access to resources and services is managed through the policy-based management system described in (9), which is based on the COPS-SD protocol.

The test bed consists of a number of PEPs controlling household/office devices/services by means of UPnP, Jini and proprietary approaches to service discovery and resource access. PEPs have COPS-SD wrappers to allow them communicate with their smart space PDP. Access to devices in the test-bed is ultimately controlled by the M-Zone PDP's MAC process, which uses COPS-SD to inform the smart space PDP of access rights to enforce. To test the operation of the MAC process a COPS-SD wrapped PEP was developed; this PEP controls a HP projector – the device used to realise the office-based use case scenario as described in §4.1.

The PIM has been implemented as a web service to host the user profiles. When a user initially enters an M-Zone they provide a link to their PIM, which will then be queried for the credentials required for authentication. The PIM also acts as a repository for user policies, including those dictating desired actions in the presence of other users. The PIM provides notifications to the M-Zone PDP if user attributes or policies are modified by a PIM, as, in many cases, these modifications will necessitate reconfiguration of access policies, or generation of new configuration action requests.

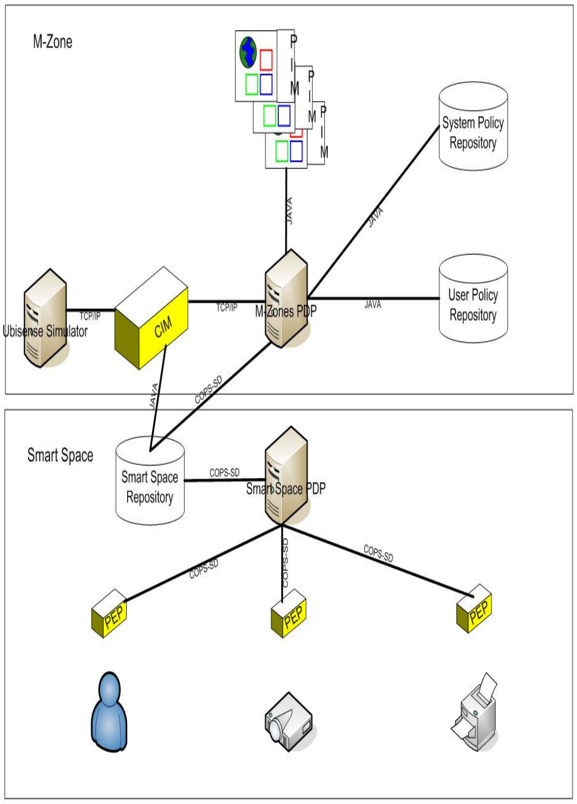


Figure 3: Test bed Architecture.

Context notifications, specifically user location and presence information are generated using the Ubisense simulator software (13), which has been used to model the movement of individuals along predefined paths through an office environment equipped with ultra-wideband (UWB) location detection. The current CIM implementation passes on context notifications as requested by the M-Zone PDP and in this instance the MAC Process in particular, expressing them in ontological terms understood by the MAC process. However MAC process and CIM may not necessarily utilise a shared ontology. Therefore prior to any communication taking place, the process of ontology negotiation has to be completed. The ontology negotiation is critical since it allows the different entities to understand mutual concepts, provided the negotiation is successful. Ontology negotiation is realised through an ontology server. In cases when concepts between two entities do not match, the ontology server is used to resolve the ambiguity. Essentially the ontology server includes the mappings between similar concepts. Once the concepts have been resolved and the entities have agreed on the common notation, further communication can take place.

4.1 Office-based Access Control Case Study

We now discuss a use case scenario in order to illustrate the MAC process functionality, in particular, how access rights adapt to the changing user set present and how policy conflicts are handled. The scenario concerns a meeting room owned by Company X, in which a projector service is used by meeting participants. This scenario highlights the role of context information as a means of defining expressive policies embodying the adaptive behaviour of the solution proposed while also highlighting how context information can be harnessed as a means of assigning policy precedence as a means of policy conflict resolution. In particular, it explores how policy conflicts between individuals at similar levels in an organisational role hierarchy (for example Alice and Bill in Figure 4) are resolved by harnessing context information such as smart space roles and the "project context" in which the meeting is taken place.

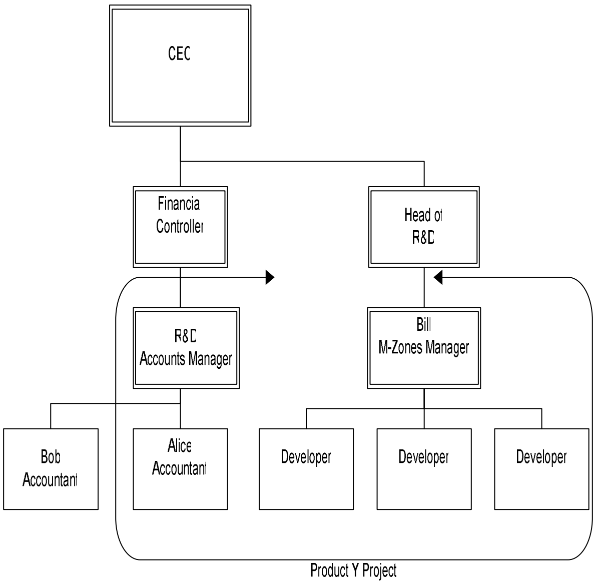
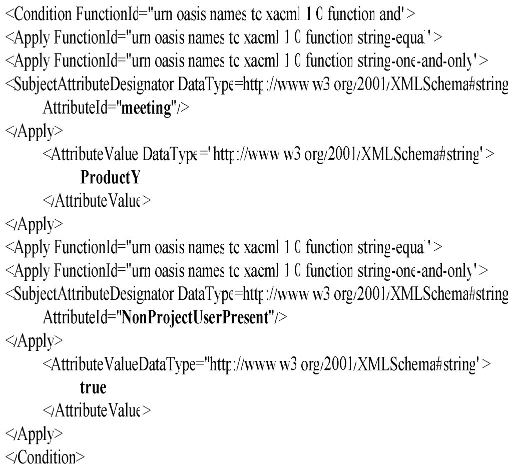


Figure 4: Company X's Organisational Chart

In the scenario Bill is a project manager and Alice is an accountant. Both are assigned to the Product Y project, and are conducting a meeting together, in which Alice is presenting the project financial accounts using the projector service. Both have been previously authenticated and authorised regarding the various services, including the projector service, available in the meeting room smart space. Bill as project manager is responsible for assigning access rights in the meeting room routinely used by the project, and as such has defined a policy which permits all project members to use all resources at all times. On the other hand, Alice has specified a policy that denies all users, herself included, access to the projector service should a non project member enter the room (a fragment of this policy is shown in Figure 5). This is to prevent unauthorised users seeing sensitive financial information relating to the project, so, if a guest enters the meeting room her access to the projector is revoked, and her presentation will be immediately removed from the projector.



```
<Condition FunctionId="urn oasis names tc xacml 1 0 function and'>
<Apply FunctionId="urn oasis names tc xacml 1 0 function string-equal'>
<Apply FunctionId="urn oasis names tc xacml 1 0 function string-one-and-only'>
<SubjectAttributeDesignator DataType=http://www w3 org/2001/XMLSchema#string
        AttributeId="meeting"/>
</Apply>
        <AttributeValue DataType='http://www w3 org/2001/XMLSchema#string'>
                ProductY
        </AttributeValue>
</Apply>
<Apply FunctionId="urn oasis names tc xacml 1 0 function string-equal'>
<Apply FunctionId="urn oasis names tc xacml 1 0 function string-one-and-only'>
<SubjectAttributeDesignator DataType=http://www w3 org/2001/XMLSchema#string
        AttributeId="NonProjectUserPresent"/>
</Apply>
        <AttributeValueDataType="http://www w3 org/2001/XMLSchema#string'>
                true
        </AttributeValue>
</Apply>
</Condition>
```

Figure 5: Fragment of Alice's Projector Policy

Clearly, Alice and Bill's policies conflict with each other, but only should an unauthorised user be present in the room. Both policies will be deployed as they won't always conflict with each other. This conflict necessitates dynamic policy conflict detection and resolution. Typical "Higher Authority Overrides Lower Authority" based approaches assign precedence to those users higher in the company hierarchy over those specified by users lower in the hierarchy (which would favour Bill in this scenario). However the precedence scheme for the meeting room smart space is configured such that policies specified by a user currently presenting and thus a presenter smart space role and accountant organisational role always have precedence over policies specified by a user that has a manager organisational role and an audience smart space role, thus, in this case, Alice's policy is enforced.

Bob, a guest, now enters the meeting room. The MAC Process receives notification via the subscription/ notification agreement it has with the CIM. The notification triggers the assignment of access rights to the new entrant based on system policies and any relevant policies of other users in the space. In this case the MAC Process examines the relevant policies governing access to the projector service. This leads to a conflict between Bill and Alice's policies. The precedence scheme employed favours Alice's policy as outlined above; thus her own access to the projector service is revoked, and the projector is blanked before Bill has a chance to see the sensitive information contained in Alice's presentation.

5 SUMMARY AND FUTURE WORK

This paper has outlined an approach to access control in ubiquitous computing environments that realises user-centric, attribute based access control, based on analysis of both system and user-defined policies. It harnesses context information relating to the user set present in a physical space and the context in which resources are accessed, as inputs into the access right configuration process. Furthermore, based on preferences specified in user policies, it supports automatic configuration of resources in response to changes in the profile of this user set. The approach allows the organisation administering the ubiquitous computing infrastructure to set policies governing default access rights associated with users, but also gives users themselves scope to dynamically affect, within the constraints imposed by system policies, the access rights of others and to ensure that resources are automatically configured to ensure their privacy is protected. Management functionality is therefore partially the responsibility of the user, resulting in a more user-centric system that adapts to changing user needs, but not in a manner that violates system-wide policies.

The MAC process currently provides for allocation of access rights to devices and services that are managed by the organisation controlling the ubiquitous computing infrastructure. It does not address the scenario in which users themselves make devices or services available to other users in their vicinity. In such a scenario policies defined by the user would be expected to take precedence over policies defined by other users, or organisations controlling the infrastructure. Future work will focus on how the MAC process can be generalised to address the complexities of access rights assignment where services and devices are controlled by multiple parties.

The CIM implementation at present is heavily focused on location and presence information with further context sources to be integrated as soon as they become available. The knowledge engineering process presently is rule based and we are currently investigating techniques to allow dynamic evolution of these rules based on the observations from the environment that would enable further deductions of indirect phenomena.

6 REFERENCES

1. OASIS Consortium, 2003, "eXtensible Access Control Markup Language (XACML) Version 1.0", eds. Godik S., Moses T., available (29/4/2005): http://www.oasis-open.org;
2. Lorch, M., Adams, D., Kafura, D., Koneni, M., Rathi, A. and Shah, S. 2003, "The PRIMA System for Privilege Management, Authorization and Enforcement in Grid Environments", Proc. 4th Ind.

Workshop on Grid Computing (Grid 2003), 109-116;

3. Lepro, R. 2003, “Cardea: Dynamic Access Control in Distributed Systems”, NAS Technical Report NAS-03-020;
4. Corradi, A., Montanari, R. and Tibaldi, D. 2004, “Context-based Access Control Management in Ubiquitous Environments”, Proc. Third IEEE Int’l Symp. on Network Computing and Applications (NCA’04), 253-260;
5. Sampemane, G., Naldurg, P. and Campbell, R. 2002, “Access Control for Active Spaces”, Proc. 18th Annual Computer Security Applications Conference, 343-352;
6. Barrett, K., Carroll, R., Osmani, V. and van der Meer, S. 2004, “User-centric Management of Ubiquitous Environments: Challenges and Initial Solutions”, Proc. 2nd Int’l Workshop on Managing Ubiquitous Communications and Services (MUCS 2004);
7. M-Zones research programme, information available (29/4/2005): http://www.m-zones.org;
8. Internet Engineering Taskforce (IETF) 2000, “RFC2753: Framework for Policy-Based Access Control”, eds. Yavatkar, R.,Pendarakis, D. and Guerin R. A., available (29/4/2005): http://www/ieft.org;
9. Ghamri-Doudane, S., van der Meer, S., O’Connor, R., Ghamri-Doudane, Y. and Agoulmine, N. 2004, “Resources Discovery and Management Using Policies in Smart Spaces”, Proc. Workshop of the 2004 HP OpenView University Association (HPOVUA 2004);
10. Sun Microsystems 2005, “SunXACML Implementation”, information available (29/4/2005): http://sunxacml.sourceforge.net;
11. White, M., Jennings, B. and van der Meer, S. 2005, “User-centric Adaptive Access Control and Resource Configuration for Ubiquitous Computing Environments”, Proc. 7th Int’l Conf. on Enterprise Information Systems (ICEIS), to appear;
12. “TSSG/O_2 Home of the Future Smart Home Demonstration”, information available (29/4/2005): http://www.o2home.com;
13. Ubisense Ltd. 2005, “Ubisense Product Description: Simulator Module”, Information available (29/4/2005): http://ubisense.net/Software/Simulate%20environments.htm.