On the use of Policy Based Management for Pervasive m-Government Services

Steven Davy, Keara Barrett, Brendan Jennings, Sven van der Meer

Telecommunications Software & Systems Group, Waterford Institute Technology
Cork Road, Waterford, Ireland
e-mail: {kbarrett,sdavy,bjennings,vdmeer}@tssg.org, web page: http://www.tssg.org

Abstract: *This paper discusses some of the challenges encountered when using policy-based management to manage pervasive m-Government services. Users within a pervasive computing environment can take advantage of pervasive m-Government services if management of these services is developed and integrated into the environment's management system. The mobility of the user is a key feature of pervasive computing environments. Adapting the management system to account for user's mobility is a challenging and highly active research area. Application of policy based management techniques appears to have the potential to successfully manage the provision of services across multiple management domains, however this potential will only be realised if solutions to a number of challenging research issues are realised. In particular, current policy based management techniques do not fully support user or service mobility across management domains. Thus we argue that research into specific areas, including dynamic policy refinement, dynamic policy conflict detection and resolution, policy interoperability among domains, and inter-domain policy negotiation, must be carried out.*
Keywords: Pervasive m-Gov services, Policy Based Management.

1. Introduction

The principal ambition of Mobile Government (m-Gov) is to make government services easier, faster and more accessible to the general public by means of the established and highly reliable mobile communications infrastructure. This seems like the most appropriate step for e-government to take, if one considers the high penetration rate of mobile phone technology across Europe. For example, a survey published in December of 2004 in Ireland (Púca, 2004), confirmed that while 42% of the Irish population have access to an Internet connected computer, over 80% have a mobile phones and 48% of these mobile phone owners were interested in using their phone to access public sector services. The roll out of Third Generation (3G) wireless communication technology, such as Universal Mobile Telecommunications System (UMTS) with a data rate up to 2Mbps, offers a great opportunity to provide a diverse range of high-quality multimedia m-Gov services. Research into Fourth Generation (4G) networks is also underway; these networks will facilitate the interaction between wireless technologies with varying capabilities and characteristics (such as UMTS, GSM, Wireless 802.11 LANs and Bluetooth), to make available 'always-on' connectivity and make possible pervasive m-Gov services.

A pervasive service is one that is seamlessly available anywhere, anytime and in any format. A pervasive m-Gov service will possibly make public sector services more accessible and easier to use by delivering the service in a format determined by a user's preference and by the functionality of the user's mobile access terminal, be it a mobile phone or a Personal Digital Assistant (PDA). The seamless and continuous access to a pervasive m-Gov services while on the move, even at high speeds, through densely and sparsely populated areas should also assist in making m-Gov services more accessible and convenient. However, such service can only exist if there is a supporting pervasive infrastructure.

The advantages of constant accessibility to relevant customised services will arguably be rendered futile without the assurance of a proficient management framework. The management of security, of instance, is critical for the success of pervasive services (Jiang et al., 2002). The security breaches experienced with today's computer infrastructure are a mere sample of what is expected with the proliferation of wireless networks and mobile terminals and the increased reliance on pervasive m-Gov services from individuals, households and businesses alike. Features of pervasive services will produce new opportunities for

accidental and deliberate security violations, frequent and random configuration changes and varying Quality of Service (QoS) requirements, thus managing these services will not be a trivial task. This suggests that the development of novel methods that consider these features and assists in the management of pervasive services is essential (Kagal et al., 2001).

Traditional management systems, that control somewhat static systems, require a human operator to handle operations at a fine-grained or in-depth level. Working at such a detailed level is becoming increasingly expensive and impractical as identifying how management decisions made, at this level, impact other aspects of the system is progressively difficult (Misra, 2001). The major reasons, according to Misra, for this are:

(i) The highly dynamic and changeable system and user requirements, which require fast and proactive response;

(ii) The interdependencies between user behaviour and service performance;

(iii) The vast number of heterogeneous entities involved; and

(iv) The transparency or invisibility requirement central to pervasive services.

Policy Based Management (PBM) is a flexible and adaptive approach for managing networks, systems and services and for this reason it is viewed as an appropriate technique for managing pervasive services in a pervasive computing environment. The remainder of this paper takes the following form. In section 2 the notion of a policy is defined and the basic structure of a PBM system with its four principal components is described. Section 3 includes an overview of pervasive environments (focusing specifically on smart spaces and Managed Zones), inter and intra-domain management and the benefits of using PBM in these environments. Section 4 delineates what we view as the key research challenges posed when implementing PBM in pervasive computing environments, namely: policy interoperability across management domains; dynamic policy conflict detection and resolution; inter-domain policy negotiation; and policy refinement. Section 5 presents some concluding remarks.

2. Policy Based Management

The use of PBM enables specification, control, administration, maintenance and enforcement of desired system behaviour in a flexible and dynamic way. Specification of management functionality at different levels, achievable through PBM, allows for better control and allocation of resources, increases automation, provides the rapid response beneficial for dynamic environments and simplifies management operations. In addition, PBM system management improves scalability and flexibility. Scalability is improved by uniformly applying the same policy to large sets of entities, while flexibility is obtained by separating the policy from the implementation of the managed system. Policy can be changed dynamically, thus changing the behaviour and strategy of the system, without modifying the implementation or interrupting the system's operation making it an attractive alternative to other human resource-intensive management techniques, such as SNMP and Command Line Interface (CLI) (Cakic, 2003).

The focal point in the area of PBM is the idea of policy as a means of driving management procedures (Damianou, 2001). The notion of a policy is arduous to define since policies can be specified at varying levels of abstraction from corporate-level business goals (high-level policies) down to system-level enforceable policy rules (low-level policies). This has lead to many definitions with many contrasts. In essence, policies are persistent directives that define choices in behaviour in terms of the condition under which operation can be invoked to generate a specific state (Damianou et al., 2002). High-level business policies are abstract statements written in natural language that are meant for human interpretation, for example Service Level Agreements (SLAs). Low-level policies, in contrast, are commands or configurations enforced on devices, such as access control entries and firewall rules enforced on a router. It is vital that low-level policies have high-level policies as a reference otherwise the policies may be ineffective and restrictive. Typically, low-level policies are implementations of the requirements of high-

level policies and for this reason the concatenation of many low-level policies may satisfy a single high-level policy (Zhi Fu et al., 2001).

2.1. Policy Architecture

The IETF's Policy Framework (POLICY) Working Group defined a Policy-based management architecture, see Figure 1, which is being used as the foundation for other policy architectures and is the generally accepted architecture for PBM systems. This architecture, defined in the now expired Internet Draft (Stevens et al., 1999), is composed of the following four elements:

(i) The Policy Management Tool

(ii) The Policy Repository

(iii) The Policy Decision Point (PDP)

(iv) The Policy Enforcement Point (PEP)

The system/network administrator may use the policy management tool to specify and update the policies that are active in the network by means of a graphical user interface (GUI) or Application Programming Interface (API). The policy management tool taking high-level policies as input and converts them to a much more precise low-level policy description that can be applied to specific devices in the network. The generated policies are then stored in the Policy Repository and the policy management tool can optionally monitor the deployment of the policies onto the entities in the network. To ensure interoperability across products from various vendors, information stored in the repository must correspond to an information model specified by the Policy Framework Working group, for example the LDAP directory service. The PDP, also known as the policy server, is responsible for retrieving and interpreting the policy rules stored in the repository and communicating them to the PEP. The PEP, or policy client, is an agent running on or within a device, for example a gateway router that can apply and execute the different polices rules retrieved from the PDP. The PEP and PDP may be on a single device or different physical devices.

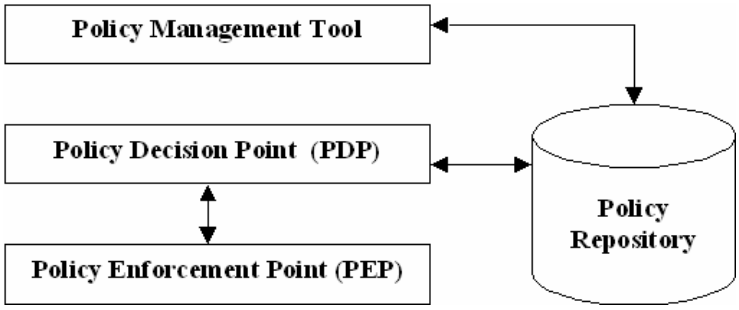

Figure 1: IETF Policy Management Architecture

Different protocols are used for and between the different elements of the architecture. For instance, the COPS protocol may be used for communication between the PDP and the PEP and LDAP may be applied to structure the policies in the repository. The COPS protocol is an adaptable protocol as it allows for a number of different client types, which govern the structure and storage of the policy information to be exchanged. The two major client types, both defined by the IETF, are the:

(i) Outsourcing mode – using the COPS-RSVP protocol

For the outsourcing mode the PEP makes the decision request to the PDP and waits for a policy rule to be returned from the PDP that outlining the actions to be taken.

(ii) Provisioning mode – using the COPS-PR protocol

For the provisioning mode the PDP makes a decision, communicates the decision to the PEP and can optionally wait for an unsolicited notification from the PEP to confirm the actions undertaken.

3. PBM of an m-Gov pervasive service within an M-Zone

The main concept behind the pervasive computing initiative is to intertwine our physical surroundings with computing environments to provide non-intrusive, transparent computing ability anytime, anywhere, thus enabling access to m-Gov services ubiquitously, for example, continuous access to a customised a License Renewal m-Gov service via a PDA while travelling in a car. Research into pervasive computing, otherwise known as ubiquitous computing or ubicomp, aims to make computing capabilities so omnipresent and seamless that it is beneath the level of human attention (Meyers, 2001). Smart Spaces are environments that embody the ideals of pervasive computing. A Smart Space is a physical space rich in devices and services that is capable of interacting unobtrusively with users to bring tangible benefits in support of users' tasks. A Managed Zone (M-Zone) represents management domains of Smart Spaces, between which users and devices may roam and dynamic service provisioning and adaptable services are realised. A very basic hierarchy, illustrated in Figure 2, can be identified in which an administrative domain or M-Zone contains (one or more) smart spaces. For example, a college M-Zone may contain a canteen smart space, a library smart space, and a lecture hall smart space. When considering pervasive computing environments there are two types of roaming that may be identified and must be managed. Firstly, there is the situation where mobile service users move between separate and distinct smart spaces within one M-Zone, termed *intra*-zone roaming. Secondly, roaming between M-Zones called *inter*-zone roaming is also managed.

Advances in fields, such as distributed and mobile network, sensors, embedded system, middleware and power consumption have made pervasive computing environments a reasonable aspiration and many testbeds and prototype exist today. While these advances bring the vision a step closer, their collaboration, essential for pervasive environments, creates new management challenges. The universal adoption of pervasive environments will not materialise without an innovative management framework, despite this such management gets little attention. Policy Based Management (PBM) is an inherently flexible and adaptive management technique, which makes it an attractive and appropriate technique for managing pervasive services in a pervasive computing environment.

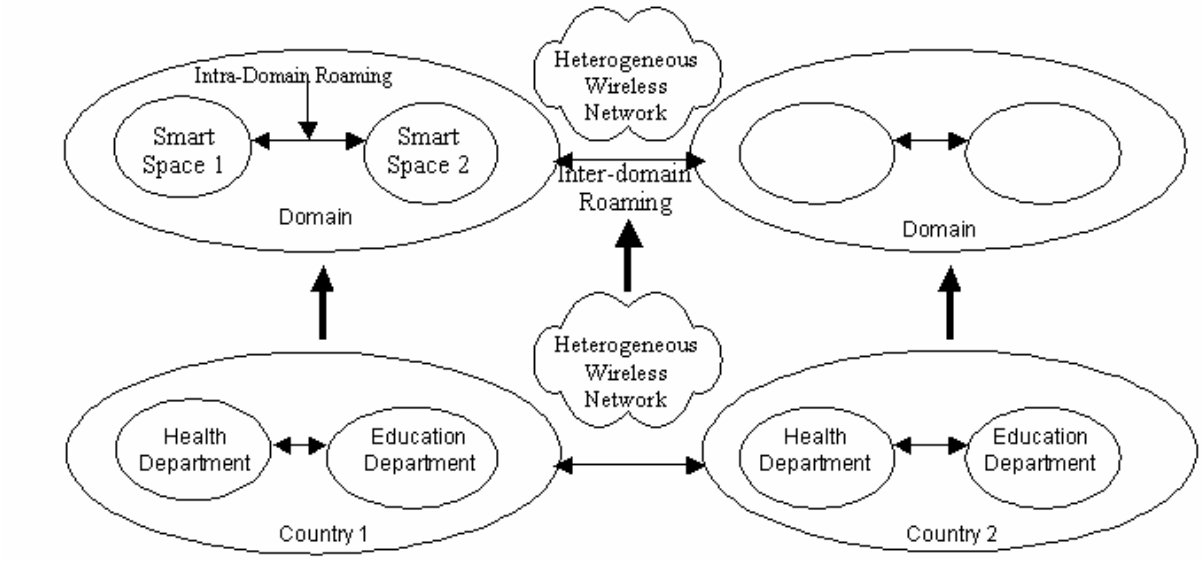

Figure 2: Pervasive domain architecture

4. PBM Challenges for mobile pervasive services

The key to managing users / services in a pervasive computing environment such as described above is to effectively and efficiently manage the user's interaction with the services provided to him, while he is on the move. This essentially resolves to managing users across smart space management domains. Managing policies across management domains is a difficult and challenging research issue, which is especially difficult when coupled with the responsibility of taking the user's preferences into account.

The main research issues in the area are:

1. Policy interoperability across management domains, so that policies specified in a user's home domain can be interpreted when the user roams to another domain;
2. Dynamic policy conflict detection and resolution, so that policies defined by management entities and/or users which dictate contradictory outcomes can be identified and appropriately prioritised and enforced;
3. Inter-domain policy negotiation, so that users can carry policy specified in a home domain to any foreign domain and it can be understood and deployed; and
4. Policy refinement, so that policies can be enforced across different domains.

4.1.Policy interoperability

When a policy is defined by a user or a system administrator, they are limited as to what can be specified in their policies. Policies are described in reference to the particular system they are intended to be deployed on, primarily taking into account services, resources, and users. The standard way of doing this is to create a model of the target system. This model can be described in an arbitrary number of ways. The description of the model outlines the meaning of every system component, and their relationships with each other. This model description has to be formatted to be machine-readable, and again there are a many ways in which this can be achieved.

The design of the model is created with a particular purpose in mind that suits the system in question. The problem is, when a user specifies a policy, or system administrator against a model of the system then the resulting policy specification is bound to that system via the system model. Subsequently, the defined policy cannot be used against another system to provide the same behaviour, as the other system has no knowledge of the previous system's model and therefore cannot understand the context for which the policy applies.

A pervasive computing environment is split up into many independently managed domains called M-Zones. These management domains may have specified unique system models and therefore their policies will be defined uniquely to their system.

As the user roams in a pervasive computing environment, he will provide, to the many management domains he encounters, his policies. The many management domains must be able to understand these policies despite the vast array of differences that can exist amongst the models.

There is a need for a process or method of modelling that will eliminate the need for defining incompatible models. If such a process existed then management domains would not have to deal with the complexities associated with model incompatibility, and they can focus on managing their own services and current users.

There are a number of modelling languages for representing the different systems, including CIM − the core information model created within the DMTF (DMTF, 1999). This model has been mapped to a number of repository types. LDAP (Light-weight directory access protocol) is a very common repository type being used by computer systems at present. Models defined using the CIM notation can be mapped to LDAP

directories, as defined in a number of RFCs. However, even if management domains use the same repository type does not imply that their models are compatible.

Systems can also be modelled using the unified modelling language (UML), which is a domain independent modelling language. However, the same issues are still present when the model is defined, in that user-specified policies defined in reference to one model may not be enforceable in a system with an incompatible model.

Work towards mapping semantic representations of system models using ontologies is described in (van der Meer, 2005). In this paper the authors argue that system models can represent the same concept in a number of different ways; and thus they attempt to create common semantic representations of system components, so that policies can eventually be mapped between the systems. This approach may potentially provide an abstraction layer above information models, where prior incompatible models may become interoperable. Policies can then be transformed into a model independent form and re-specified in that destination models format, thus allowing for policy portability. The problem arises when a mapping from every model is made to an abstract representation, which provides its own complications.

4.2. Dynamic policy conflict detection and resolution

Policy-based management of a pervasive computing environment is a highly complicated system. Policy-based management of multiple diverse management domains deals with managing objects located across domain boundaries. The problem here is that multiple policies may apply to the one object. As the objects have multiple sources of policy, there may be conflicts among these policies.

A conflict among two or more policies occurs when the objectives of the policies cannot be simultaneously met. There a two broad categories of conflicts, that of static policy conflict and dynamic policy conflict. The difference being static policy conflict can be recognised through offline methods such as syntax analysis, and dynamic policy conflict is more unpredictable and can occur spontaneously. Static policy conflict has been explored in (Lupu, 1997) and initial work towards the more complex process of dynamic policy conflict has been described in (Dunlop, 2002; Dunlop, 2003). For the latter the authors highlight the need for a highly efficient process for policy conflict detection and resolution, and try to develop a dynamic policy model to deal with this. Their work is carried out in the back ground of open and dynamic distributed systems, and in principle could be adapted to be deployed in a pervasive computing environment. Other methods of policy conflict detection are logical analysis, suggested in (Li et al., 2003), and Meta policies defining policies among policies (Lupu, 1999).

There has also been some work in this area from the eBiquity project (Lalana et al., 2003), focussing on the introduction of precedence and priority into policy specification. However, these methods may not work in an environment where policy subjects and targets may be initially unknown, and so a precedence hierarchy cannot be established for policy conflicts. Therefore, other more dynamic policy systems that do not rely on domain dependant knowledge need to be used.

Interesting advances in the area of dynamic policy conflict detection and resolution includes investigating user-specified policies. (White 2004) discusses the use of using user profiles to adapt security settings in pervasive computing environments. Introducing user profiles, or user-specified policies provide the management system of a pervasive computing environment with more information about users so that better informed decisions can be made. Another advantage is that more personalised services can be provided to the user. Such policies will bring a more dynamic element into pervasive computing environments and will open interesting complexities in the dynamic process of conflict detection and resolution. In this circumstance, the resolution will need to incorporate different processes that can communicate amongst diverse management domains in order to resolve policy conflicts.

4.3.Inter-domain policy negotiation

When a user is accessing pervasive m-Gov services, he may roam across many management domains. These management domains need to communicate in some form in order to help maintain service provision for the users. This will involve inter-domain policy negotiation, so that the domain currently providing the service can gain full knowledge of the previous actions and decisions of the user to which it is providing the service.

When a user roams from one management domain to another, then there must be a protocol to allow these domains to communicate with each other to resolve policy issues and share information about the specific user. When mobile pervasive services are taken into account, a user may utilise the same service from multiple domains, and be able to roam within and across these domains as they see fit. This seamless mobility of service access provided to the user requires that the separate management domains communicate with each other to provide a better quality of service delivery to the user. Inter-domain policy negotiation attempts to address this by providing protocols and interfaces that help with the management of users across multiple management domain boundaries.

Negotiation among domains is needed because when a user moves out from one management domain and into another, there may be very different policies governing the provision of the services he uses. The new domain needs to know how best to provide the service and how to deal with the user. In essence the new domain needs to discover certain information about the user, such as his preferences and also some domain dependant information that only the previous domain may have knowledge of. This information may be for example, the most recent mobility patterns, some trust levels that the previous domain had for the user, and also some service specific information such as QoS and security requirements. The user is not expected to maintain all this information, as this would make the roaming process heavy and will demand a larger responsibility from the user. The new domain may discover this information from the user's previous domains. To compound this problem the user will have specified through his policies that certain information must be kept private and some information may be shared across domains, in this case the new domain must negotiate with the previous domains to agree on what information about the user is to be shared. Upon receiving the user's previous domain information the new domain can now offer the user consistent mobile services derived from other domains information.

Inter-domain negotiation can also be used to spread information not directly associated with the user, such as traffic patterns, and other public information. Trust information gathered from the users usage patterns of services, to which the user cannot control may also be shared throughout independent domains.

This work is similar to work in the W3C in P3P (Platform for Privacy Preferences) (W3C, 2002), where a user can specify a personal privacy policy and the user can be guaranteed that a P3P compliant website will follow its privacy policy. This work is similar to work in (Zheng, 2004) that documents the current work going on in QoS policy control for mobile networks. In this paper, they describe an inter-domain policy framework based on the Diameter protocol (Calhoun, 2001), using roaming profiles to realise inter-domain polices. Although this work is interesting, it isn't geared towards pervasive computer, where the policy system is more dynamic and policies not only have to negotiate across domain boundaries but must be portable across domains boundaries.

There has been a lot of work done in the area of inter-domain negotiation in terms of QoS, routing information, and security (Barrere, 2003) but not in the area of pervasive service management.

4.4.Policy Refinement

Mont et al (1999) argue the need for policy refinement as follows: *"A policy-based management system is only really useful if it allows not only high level description of abstract policy, but also enables such policy to be refined and eventually mapped into an appropriate configuration for controlling devices in the managed system"*. This is particularly true when PBM is employed for managing pervasive services across a heterogeneous wireless networks. Since pervasive services are highly adaptive and the infrastructure

through which they are offered is unpredictable it would be impossible for the service provider to identify and enforce all the device level configuration changes for a given situation without assistance. The enormous amounts of infrastructure and the concealment of the underlying management activities, needed for the realisation of pervasive services, exacerbates this challenge. Consequently, manually adjusting low-level enforceable policies on the fly to generate the most optimal behaviour for a particular situation and to accommodate the mobility and changing demands of users and the system would be impractical. A mechanism that can automatically generate low-level policies that implement the goals (high-level policies specified by the service provider) in relation to the current system state is required.

For example a mechanism that uses situational information, otherwise known as context information, to automatically trigger, by way of refinements, the removal of existing low-level policies and to create new policies, in an effort to support management procedures would be highly beneficial. There is a significant gap between business goals and device specific configuration rules that must be bridged automatically through iterative refinement along a policy hierarchy or policy continuum with the aid of an information model. Policy refinement has been defined formally by Bandara et al (2003) as follows: *"If there exists a set of policies Prs: p1, p2, .. pn, such that the enforcement of a combination of these policies results in a system behaving in an identical manner to a system that is enforcing some base policy Pb, it can be said that Prs is a refinement of Pb. The set of policies Prs: p1, p2, .. pn is referred to as the refined policy set"*. Refinement may be executed for at each levels of the hierarchy producing progressively more concrete, enforceable policies.

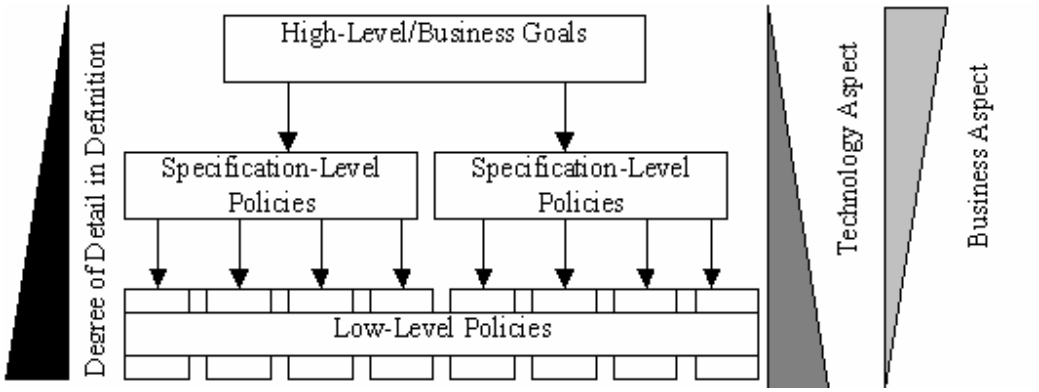

Figure 3: Policy Hierarchy

The policies produced after the final refinement iteration should be enforceable on the system and meet the requirements of the high-level policy. This should be verified through analysis for (i) correctness – an accurate representation of the business goal (high level policy) at a different viewpoint, (ii) Consistency – no conflicts, static or dynamic and (iii) validity – the refined policies can be executed on a tangible managed object in the environment (Darimont and Lamsweerde, 1996). The PBM system should exploit these refinement and analysis capabilities when events trigger policy decisions. These events can come from three sources: changes in business goals (m-Gov service objectives), changes in environmental context (wireless infrastructure) and direct user interactions with the pervasive service. Decisions relating to events take into account relevant user policies, but only to the extent that actions generated are not in conflict with the service provider's policies. Generally, these actions result in re-configuration of policies deployed in one or more smart spaces, whose enforcement affects the manner in which users can interact with services.

5. Conclusion

Future m-Government services may be realised through pervasive services where the mobility of the user's service access and policies across management domains can be managed by the use of policy-based management. There are specific research challenges in the domain of policy-based management that arise

when dealing with roaming users and services. Policy based management systems, and therefore the users and services within, may be developed independently for specific domains and so may not be interoperable. This needs to be addressed for users to be able to use multiple domains to access and use services across a pervasive computing environment. Management domains or smart spaces, may change depending on the locations or mobility patterns of users, and so may need to re-evaluate their low-level policy behaviours; therefore, dynamic policy refinement is needed to take account of these events. When a user roams across management domains, be it intra-domain movement or inter-domain movement, then the services he accesses and uses may be altered. For the current domain to be able to provide the services to the user as best it can, it must be able to negotiate with any of the other domains the user has been in contact with so that it can gain a better understanding of the user intentions. This involves a form of inter-domain negotiation among policy based management systems. A user's policy, when roaming into a new domain, may conflict with the domain policies in a number of ways. The complexity arises due to the unpredictability of new policies. There must be a way of dynamically detecting and resolving the conflicts in these scenarios.

References

Bandara, Arosha K, Lupu, Emil C & Alessandra Russo, 2003, " Using Event Calculus to Formalise Policy Specification and Analysis", Proceedings 4th IEEE Workshop on Policies for Distributed Systems and Networks (Policy 2003), Lake Como, Italy

Barrere F., Benzeki A., Grasset F., Laborde R., Nasser B., 2003, "Inter-domain policy negotiation" IEEE 4th International Workshop on Policies for Distributed Systems and Networks

Cakic, Jovan, 2003, "A high-level framework for Policy-based management of distributed systems", PhD thesis, University of Kent, Canterbury, UK

Calhoun, P.; Akhtar, H.; Arkko, E.; Guttman, E. and Rubens A., (2001) "Diameter Base Protocol", draft-ieft-aaa-diameter-08-alpha02.txt, IETF work in progress, Available 2005-03-14

Damianou, N., 2001, "A policy Framework for Management of Distributed Systems", PhD Thesis, Imperial College, University of London.

Damianou, N., Bandara K. A., Sloman M., Lupu E., 2002, "A Survey of Policy Specification Approaches", Imperial College, University of London. Available (20/4/2005): http://www.doc.ic.ac.uk/~mss/Papers/PolicySurvey.pdf

Darimont, R., van Lamsweerde, A., 1996, "Formal Refinement Patterns for Goal-Driven Requirements Elaboration", 4th ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE4), 179-190

Distributed Management Task Force (DMTF), Inc., 1999 "Common Information Model (CIM) Specification, version 2.2"

Dunlop N., Indulska, J., and Redmond K., 2002, "Dynamic Conflict Detection in Policy-Based Management Systems", In Proceedings of the Sixth International Enterprise Distributed Object Computing Conference (EDOC' 02)

Dunlop N., Indulska, J., and Redmond K., 2003, "Methods of Conflict Resolution in Policy-Based Management Systems", In Proceedings of the Seventh International Enterprise Distributed Object Computing Conference (EDOC' 03)

Jiang and Landay, James A., 2002, "Modelling Privacy Control in Context-Aware Systems", University of California, Berkeley, IEEE Pervasive Computing, Vol.1, No. 3.

Lalana K., Finin T., and Anupam J., 2001, "Trust based security for pervasive computing environments", University of

Maryland, Baltimore County, IEEE Communications Volume 34 , Issue 12 pp.154-157, December 2001

Lalana K., Finin T., Anupam J., 2003, "A Policy Language for a Pervasive Computing Environment" In Proceedings of the 4th International Workshop on Policies for Distributed Systems and Networks (POLICY'03)

Li N., Grosof B., Feigenbaum J., 2003, "Delegation Logic: A Logic-based Approach to Distributed Authorization" ACM Transactions on Information and System Security, Vol 6, No. 1, Pages 128-171

Lupu E., Sloman M., 1997, "Conflict Analysis for Management Policies" In Proceedings of the 5th International Symposium on Integrated Network Management IM'97

Lupu E., Sloman M., 1999, "Conflicts in Policy-based Distributed Systems Management" IEEE Transactions on Software Engineering Volume 25, Issue 6 , pp.852-869

Misra A., 2001, "Autoconfiguration, Registration and Mobility Management for Pervasive Computing", IEEE Personal Communications Systems Magazine, vol.8, August pp.24-31

Púca, 2004, "Púca mGovernment Survey Indicates Strong Demand in Ireland for Services via Mobile Phone"

Stevens M., Weiss W., Mahon H., Moore B., Strassner J., Waters G., Westerinen A., and Wheeler J., 1999 "PolicyFramework," Policy Framework working group, IETF, 13 Sept 99, Available (20/4/2005): http://www.netsys.com/ietf/1999/4380.html

van der Meer S., O'Sullivan D., Lewis D., Agoulmine N., 2005, "Ontology Based Policy Mobility for Pervasive Computing", accepted for publication in Proceedings of the 9th IFIP/IEEE International Symposium on Integrated Network Management, IM2005

White M., Jennings B, 2004 "Adapting Access Rights to the changing profile of user sets present in a Ubiquitous Computing Environment", in proceedings 2nd International Workshop on Managing Ubiquitous Communications and Services, MUCS 2004

W3C, 2002, The Platform for Privacy Preferences 1.0 (P3P1.0) Specification, Available (20/4/2005): http://www.w3.org/TR/P3P/

Zheng, H. and Greis, M. 2004, "Ongoing Research on QoS Policy Control Schemes in Mobile Networks", Mobile Networks and Applications 9,pp.235-241.

Zhi Fu, Wu, S. Flex, 2001,"Automatic Generation of IPSec/VPN Security Policies In an Intra-Domain Environment", In Proceedings of IEEE 12th International Workshop on Distributed Systems: Operations and Management (DSOM 2001), Available: http://www.loria.fr/~festor/DSOM2001/proceedings/S8-3.pdf

Steven Davy was awarded a B.A. in Computer Science from Trinity College Dublin in 2003 and is currently a PhD candidate with the Telecommunications Software & Systems Group (TSSG) in Waterford Institute of Technology. His areas of interest include user-centric, distributed policy-based management systems, specifically looking at dynamic policy conflict detection and inter-domain management through user-specified policies. He has published two publications on security performance in pervasive computing environments.

Keara Barrett was awarded a BSc in Computer Networking and Software Development, from Cork Institute of Technology in 2002. On completing the course Keara embarked on Postgraduate research in the Telecommunications Software & Systems Group (TSSG), at Waterford Institute of Technology. She is now a PhD student working on novel management approaches for the HEA-funded M-Zones programme. Her primary research interests are, policy based management, refinement techniques and pervasive computing environments. She has published papers at MUCS2004, ITSRS2003, ISICT2003, on a range of topics including user-centric management of pervasive environments, mobility management for pervasive environments and context-aware adaptive services.

Brendan Jennings was awarded a PhD in Telecommunications from Dublin City University in 2001 and a BEng in Electronic Engineering from Dublin City University in 1993. Since 2003 he has been a Senior Investigator with the Telecommunications Software & Systems Group (TSSG) in Waterford Institute of Technology. He is a lead researcher in the HEA-funded M-Zones programme, addressing issues relating to the design, configuration and management of pervasive computing environments. His research interests are in the areas of accounting systems for dynamically composed services; user-centric, distributed policy-based management systems; performance management; and agent technology. He has published over twenty peer-reviewed publications in international journals and conference proceedings, two book chapters and has authored contributions to standards bodies ETSI and FIPA.

Sven van der Meer received his Diploma in computer science (M.Sc. in computer science) and his Dr.-Ing. (PhD) from Technical University Berlin (TUB), Germany, in 1996 and 2003. Since 2003, Sven is Senior Investigator at the Telecommunication Software and Systems Group (TSSG) at the Waterford Institute of Technology (WIT) in Waterford, Ireland. He is project leader of M-Zones, a four year research programme developing concepts for managing smart spaces. Additionally, he is working on a number of FP6 proposals, supervising a number of postgraduate students and spends much of his time to further improve the TSSG. He has published more then 20 peer refereed papers, one book chapter and several tutorials.