Security in Pervasive Computing: Current Status and Open Issues

Munirul Haque and Sheikh Iqbal Ahamed
(Corresponding author: Sheikh Iqbal Ahamed)

Department of Mathematics, Statistics and Computer Science,Marquette University
1313 West Wisconsin Avenue, Milwaukee, WI 53233, USA
(Email: {mhaque, iq}@mscs.mu.edu)
(Invited Paper)

Abstract

Million of wireless device users are ever on the move, becoming more dependent on their PDAs, smart phones, and other handheld devices. With the advancement of pervasive computing, new and unique capabilities are available to aid mobile societies. The wireless nature of these devices has fostered a new era of mobility. Thousands of pervasive devices are able to arbitrarily join and leave a network, creating a nomadic environment known as a pervasive ad hoc network. However, mobile devices have vulnerabilities, and some are proving to be challenging. Security in pervasive computing is the most critical challenge. Security is needed to ensure exact and accurate confidentiality, integrity, authentication, and access control, to name a few. Security for mobile devices, though still in its infancy, has drawn the attention of various researchers. As pervasive devices become incorporated in our day-to-day lives, security will increasingly becoming a common concern for all users - - though for most it will be an afterthought, like many other computing functions. The usability and expansion of pervasive computing applications depends greatly on the security and reliability provided by the applications. At this critical juncture, security research is growing. This paper examines the recent trends and forward thinking investigation in several fields of security, along with a brief history of previous accomplishments in the corresponding areas. Some open issues have been discussed for further investigation.

Keywords: Pervasive security, privacy, security, and trust

1 Introduction

The importance of security has been supported with thousands of recent surveys, and it is far beyond an afterthought nowadays. Network security has topped the priority list of 47% respondents in the Networking Report Card survey by SearchNetworking.com [65]. Closely related to security are issues of corporate reputation, competitive position, and monetary gain. A study by eMarketer indicates an average loss of $10 billion per year due to infractions in computer security [66]. Microsoft has defined security as “The protection of information assets through the use of technology, processes, and training” [63]. Wikipedia defines security as a “... platform, designed so that agents (users or programs) can only perform actions that have been allowed. This involves specifying and implementing a security policy” [64]. CIA (Confidentiality, Integrity, Availability) is the term commonly used to describe the required characteristics of security. Confidentiality ensures information is not exposed to any unauthorized user. Integrity indicates information has not been altered or falsified by an unauthorized user. Availability denotes information is readily available when required.

Security in pervasive computing has been termed pervasive security. Though pervasive security includes all the characteristics and requirements of computer security, it introduces some novel vulnerabilities and security rifts due to a few unique characteristics of pervasive computing. Pervasive computing has been defined as “Numerous, casually accessible, often invisible computing devices, frequently mobile or embedded in the environment, connected to an increasingly ubiquitous network infrastructure composed of a wired core and wireless edges” [67]. Pervasive computing is the brain child of Weiser [69]. This vision embeds computation into the environment and ensures transparent interaction of these computational devices with the users. It can be considered the opposite of virtual reality.

Pervasive computing is proving its usability and scope in almost every aspect nowadays. The availability of, and tremendous improvement in, pervasive devices including PDAs, smart phones, tiny sensors, etc., have made this next generation of computing technology suitable for many situations in places like the home, hospital, or battlefield. Recent surveys like [26] indicate 50% of physicians used PDAs in 2002, [1] and they were used by approximately 50% of people in the U.S., indicating the tremendous growth in the use handheld computers and pervasive devices. To overcome several constraints related to capability, pervasive devices actually form a collaborative space where devices are highly inter-connected and mutually cooperative; this becomes the key to success and leads to sharing of resources and information. The downside is that this provides opportunities for theft and hacking. The characteristics of pervasive scenarios sometimes seem to provide an open invitation for active and passive eavesdroppers. In order to increase the usability and spectrum range of scenarios that can benefit from this technique, pervasive computing has yet to prove it is up to solving the security challenges.

As devices can join and leave completely arbitrarily, they form an extremely volatile ad hoc network which is changing from time to time. Thus, we cannot rely on a static, permanent security model. A solution is to invoke a dynamic security system with sufficient intelligence to prevent security breaches. In order to grow this intelligence, a system should avail itself of several types of private information. Some contextual information that carries importance in pervasive security are:

1) **Security policy of the host:** Each user has some personalized rules for access control of varying resources as well as the visibility.

2) **Security policy of the resource/system:** This includes the details of policies under which a resource can be accessed in a specific scenario.

3) **Location:** The situational information for both the user and resource is very important. A resource may declare some specific trusted locations as a precondition for getting the service. A resource may be accessed by remote users whereas others might have access restricted to their own domain.

4) **Methodology for connection:** This contextual information include a connection mechanism and its security, bandwidth, packet routing information, etc. It also includes information about the devices being used in communication between the user and the pervasive environment.

5) **Interaction methodology of the host:** A host may be able to access a specific resource in one situation while fail to access the same resource in other scenario, due to change in his/her participation in a different activity or role.

Security in pervasive computing actually spreads over a broad sphere, encompassing a large number of issues including security in authentication, authorization, and access control. We divided these issues into several sections and tried to provide a brief history along with the present trend in each corresponding field. We have tried to cover all recent research in related to security. The open issues section has been attached to provide further research directions.

Section 2 describes several security models, some of which have incorporated agents. As agents are taking the place of actors in many scenarios with perfection these models have become attractive. Authentication and authorization perform the key roles in ensuring security, which has been described in Section 3. After authorization, the next question is the right to access a specific facility. Focus of Section 4 is on access control which deals with access right, feedback, etc. A user may employ any pervasive application to get some kind of service which is generally provided through the discovery of resources. This issue has been discussed in Section 5. The issue of trust depicted in Section 6 is actually inseparably related to all subsections of security. Section 7 puts forward some open discussions and issues, which are followed by our conclusions in Section 8.

2 Security Model

Several works exist where agent-based applications have proved to be promising. Some projects have come up with different security issues in Mobile Agent System. In order to prevent malicious use [21], it is suggested that agents should communicate only with trusted and authenticated nodes. Hence several trust models appear which we discuss later. A scenario is described [50] where the credibility of a node will vary depending on the agents' interaction with that node. [37] describes a method to defend against several types of attacks and to restrict an agent from occupying a specific resource for a long time.

In a recent interesting study [20], the researchers proposed a security model named 'QED' (Quarantine, Examination and Decontamination). QED was designed to provide several aspects of security which are well known for fixed infrastructures within the realm of a pervasive computing environment – virus scan, firewall, intrusion detection, and update and patch management. As part of an examination phase, the QED model incorporates a fixed infrastructure based security nodes which can provide updated virus scanners and patches. These nodes are seeking permission to enter in the network, and QED can push the nodes to receive the updated information as a precondition for entrance. The Quarantine phase performs the isolation of clients to ensure that they meet the local integrity constraints. On the other hand, the device can also decide not to access some of the available services of the network due to conflict with its own access policy. Clients are checked for potential vulnerabilities and malicious code in the Examination period. The probable investigations include virus scans and memory scans. During an active examination, clients need to go through all the defined investigations, whereas in passive investigation the system acknowledges a digital certificate that ensures that the corresponding client have passed similar checks in the previous environment. The Decontamination phase deals with removing vulnerabilities from the examined clients. Several tools can be used for this phase.

At present, we have observed many agent-based pervasive computing applications. [46] discusses several dimensions of security for a specific environment named Multi Agent System (MAS). It also represents a security model named Buddy where a security feature has been distributed among all the nodes and each node tries to safeguard its neighbor. In contrast with many others, here the authors proposed a non-hierarchical implementation and mentioned that hierarchical models are more likely to be attacked by a malefic force. According to the authors, if a specific agent or group of agents maintain the security features in hierarchical architecture, it is much easier to locate and penetrate them. In the Buddy model, each agent records the presence of its closest neighbor or buddy through a token passing mechanism. When facing danger, each agent will seek help from its buddy. Each agent acts once as 'Token Sender' and once as 'Token Receiver'. When an agent receives a token in a predefined time limit, it gets the idea that its buddy is in good shape. Otherwise it senses a problem and broadcasts a global message to identify the problem. Each agent in the topology gets a chance to periodically broadcast. TokenSender and TokenReceiver classes of Java have been used to implement the scenario.

The researchers in [5, 7] have proposed a new architecture named Ubiquitous Mobile Agent System (UbiMAS) where they have shown some unique security characteristics. Here the agent contains all the personal information and can request a service on behalf of the user. Finally, it actually accompanies the corresponding mobile user as he travels across different environments and domains. The system incorporates two types of agents, user agent and service agent. The system architecture operates in the following manner: A Message Delivery Engine deals with message delivery to and from agents. If the message is destined to an agent that resides in the current node then the message is passed to the PoBox. PoBox maintains a PoBox Adder and a queue for each agent connected with the agent node. PoBox Adder maintains the communication link between an agent and agent node. Using PoBox Adder and the queue, the system can ensure security in message transfer and protect itself from several attacks including DoS attack. The agent node can issue a timer that can dynamically change the length of the queue which facilitates the security features. If the Message Delivery Engine finds that the destined agent is not in the current node, it then forwards the message to the required node. An agent can migrate from one node to another using the Migration Engine and Serializer / Deserializer component. While sending messages from agents or nodes, the Message Delivery Engine performs packet formatting, inclusion of header, and other required security operations. Acknowledgement has been introduced to guarantee against packet loss. All the messages are sent in encrypted form. The security features enable the system to protect the agents from hosts as well as both agents and hosts from malicious agents. UbiMAS has been developed in Java as a service that runs on top of a middleware named Autonomic Middleware for Ubiquitous eNvironments (AMUN) [61].

3 Authentication and Authorization

Authentication and authorization have long been discussed in pervasive computing. Both features are needed, in order to restrict any malicious user from entering the network or to prove one's own identity. These issues have gained paramount importance over several aspects of security. Starting from Active Badge System [8] in 1994 several authentication systems were proposed for different ad hoc situations based on identity [16, 68], proximity [6, 19], private-public key combination [6, 17], reputation [57], trust [44], etc.

Authors in [73] present an authorization mechanism based on context awareness. Several other applications can cope with this mechanism without any difficulty as it incorporates GSSAPI (Generic Security Services Application Program Interface-RFC 2743) through the well known technology Kerberos. Here the authors used contextual information of different roles in generating a simpler access control policy. The whole architecture includes authentication service, authorization service, dynamic context services provided through dynamic context service manager (DCSM), dynamic context update mechanism and event update mechanism. Kerberos, LDAP and XML have been thoroughly used in building the authentication architecture. These open protocols increase the operability of the architecture over several platforms. This architecture can also function on LINUX. Kerberos protocol provides the cryptographic technique required for authentication and LDAP supports the required storage. Two types of roles, standard role and task, have been specified in the authorization service. Dynamic Context Service Manager (DCSM) deals with the responsibility of activating and deactivating these roles based on the contextual constraints. Dynamic Context Service (DCS) is responsible for collecting context data and providing the information to DCSM based on a predefined policy. This policy specifies the frequency for collecting contextual information, threshold values for various events that indicate when to apprize DCSM about those events, etc. Dynamic Context Update Mechanism modifies and updates a current contextual information object when it receives such a request from DCS or authorization services as a result of activation/deactivation.

In a research paper from Dartmouth College [40], researchers present a context aware authorization mechanism based on rules and facts. This rule-based authorization differs from others in that it does not need any central server or certificate authority (CA) which will be trusted by all and will store all the contextual information. In this method, when a user wants to acquire a service from a resource/server, that server issues a logical authentication query and sends it to the host of the resource. Each host has a knowledge domain with which it attempts to prove the authorization query. If it fails, it distributes several portions of the proof to multiple hosts. Through this distribution, the computational overhead is actually reduced. After getting the sub-proofs from co-hosts, the host of the resource can declare the result of the query to be TRUE or FALSE, thus indicating grant or denial of access. By design, this approach facilitates confidentiality, integrity and scalability. In the architecture of the host, the 'Query Handler' deals with the remote request and ensures personal confidentiality rules. The 'Query Issuer' takes the responsibility of passing request for sub-proofs to other hosts and enforces personal integrity issues. The 'Interference Engine' attempts to compose the proof tree based on the rules and facts available in its knowledge domain. XProloga of Java has been used in building the prototype that evaluates the authorization query.

In the Gaia Authentication [11] the authors incorporates a number of authentication means where each authentication mechanism attains a specific value known as a 'confidence value'. This value ranges from 0 to 1 depending on the device and protocol used in the authentication process. In order to increase the confidence value, a specific authentication mechanism may include any number of authentication processes. Reasoning technique is used to formulate the net confidence value from the partial confidence values. This authentication provides a unique feature which decouples the authentication procedures and authentication devices into two sections. The Authentication Mechanism Module (AMM) encompasses all the authentication procedures available like challenge-response, Kerberos, SESAME [35] etc. The Authentication Device Module (ADM) incorporates a module for each authentication device like PDA, smart badge, etc., and these modules are device dependent. This decoupling facilitates the incorporation of a new protocol in the AMM section or a new module in the ADM section for a new authentication device without interacting with the other section. In order to ensure lightweight CORBA services, universally Interoperable Core (UIC) has been used.

Style of authentication sometimes varies based on the situation. Authors in [6] have presented an authentication mechanism based on the proximity of the user. This protocol has been implemented in a hospital scenario using several components. In this context aware system, a JavaCard has been used to contain the identity information of the user including an id, password and a pair of secret and public keys. In order to incorporate context awareness, the system encompasses context monitors and context servers. The context aware infrastructure mainly provides location information. If both the identification and context aware system fail for some unknown reason, the entire system returns to the manual username/password system to ensure security. The authenticity of the entire system depends critically on the accuracy of the location identification. The authors also analyze several types of passive and active attacks and their impact on the system.

In [2] authors have gone through several authentication mechanisms and a number of categorizations of the systems have been provided based on different classification criterions.

4 Access Control

Many projects and frameworks have dealt with the mechanism of access control and related security issues. In 1996, a protocol named PolicyMaker [9] was implemented with options for setting policies and providing access right queries. Then a new model Role Based Access Control (RBAC) [38, 39] gained popularity that defined access based on the role of the user. Though this model tries to secure the system from unauthorized users based on this theme, sometimes it becomes very difficult to define roles for every user. Later several researchers proposed a central knowledge base for access control mechanism in their projects [13, 15, 41].

In pervasive ad hoc scenarios, information are collected and stored in different ways through different devices in different environments. This becomes nearly impossible if the owner of the information has to grant separate access based on client, situation, category of information, etc. There have already been addressed to reduce the number of access right permissions like RBAC (Role Based Access Control), sharing of access right strategies over multiple domains, etc. Here the authors focus on information relationship and place this as a new axis for limiting the issue of access rights. The information relationship has been classified in to three categories: 1. Bundling based, 2. Combination based, and 3. Granularity based. Access rights are stored as SPKI/SDSI digital certificates [18] in the corresponding client rather than storing all access rights in a central server, thus ensuring the required distributed approach. Whenever a client receives an access right, it stores the right and corresponding information relationship. Later when the client seeks to access a different resource, the stored access rights and information relationships are used to build a proof for that access, and permission will be granted if he succeeds in building the proof. Thus this methodology will reduce the interaction with the owner of the information in issuing access rights. The conditions needed for access rights have been formalized. Java has been used in building the framework. The facility of proving access rights have been incorporated in the framework provided by Howell and Kotz [30]. Access right proofs have been built as Java classes. This protocol implementation has used the CSI (Contextual Service Interface) [32] of the well-known Aura project [23] as a test bed.

If a user is denied access to any event, the user should receive specific feedback information based on the refusal. The pervasive computing environment involves thousands of scenarios as well as a dynamic access control policy that changes based on various contextual information such as role of the user, activity, and time. As a result, the same user might be initially granted access to a particular service and then be refused at other times thus causing confusion for the user. Consequently, merely showing a simple message 'Access Denied' is not enough from the users' perspective. Being inspired by this scenario, some researcher in Urbana-Champaign have proposed a feedback model named 'Know' [36] that provides an optimal alternative solution. When access to a particular service is made available, it simultaneously ensures that system's security and access control policy is not being disclosed. The feedback information certainly increases the usability and reliability of the system but there has to be a trade off between quality and quantity of feedback and disclosure of access control policies. First, OBDD (Ordered Binary Decision Diagram) is used to construct a graph structure. The goal is to start from the root and reach a leaf node marked as TRUE. Each edge denotes a condition. A cost function which is based on activities, roles and meta policies is defined to identify a weight for a specific edge. Then a shortest path algorithm is used to find a path from the root to a TRUE leaf node that consumes the minimum cost. In order to protect meta data, the corresponding edges are assigned an infinite value. As a result these paths will never be chosen as a solution and the feedback information will not contain any information about the access control policy. The portal provides the feedback information only if it can calculate an alternative solution within predefined constraints of time and space. As a first step OBDD is generated based on several access conditions. BuDDy [45] library is used to optimize the initial OBDD. This prototype has been implemented in the Gaia project [53].

5 Secure Resource Discovery

Service or resource discovery is one of the main features of pervasive applications. If security is not employed with certainty when discovering and achieving services, active and passive intruders can enjoy unauthorized services and there is a possibility of even corrupting the service provider. In 1999 the Ninja project [24, 25] was implemented in UC Berkeley. It developed the concept of secure identification of service through Secure Service Discovery Service (SSDS). Here Certificate Authority (CA) deals with issuing valid certificates and Capability Manager plays an important role in enforcing security where capabilities indicate the access permission of a user to a set of resources. The service providers can also mention the required conditions (capabilities) that a user needs to obtain in order to discover a particular service. Some of the service discovery projects enforced an encryption technique whereas some imposed a simplified version of Public Key Infrastructure (PKI) [62].

In [14] the authors provided a survey on the available security issues in some of the well-known Service Oriented Architectures (SOA) along with some required issues in designing secure Service Oriented Architectures. Literature on the security aspects of several standard protocols like UPnP (Universal Plug and Play), Jini, Bluetooth, Salutation architecture, Service Location Protocol (SLP) have been analyzed along with some recent architectures like Ninja project [4], Splendor [74] etc. Besides some general aspects like authentication or authorization, researchers have proposed four service oriented issues that are needed to be taken care of from the point of view of security.

1) Service Registration and Deregistration has to be done before indicating that a specific service is available. At the same time, registered services have to maintain their integrity.

2) Secure Discovery indicates that there has to be a security mechanism which ensures that services will be available only to valid clients. While a client is requesting a service, the system may need to regulate the flow of information including type of service, owner of the service, etc.

3) At this step 'Secure Delivery' ensures that the service requested by the user will be provided to him in the required manner. That means the service, on the way to the client, should be secure from any counterfeiting or tampering.

4) Availability indicates that the system should always have an updated list of available registered services, which will be provided if an authorized user requests for that service maintaining all the rules and restrictions.

Smith [60] focuses on a context aware discovery of resources and how to access resources in a secure and unobtrusive manner. In a pervasive computing environment, rules and limitations imposed by the user, the system, and the collaborative activity scenario have to be combined dynamically at runtime. Here the researchers have defined a namespace related to each user and domain. These namespaces include resources, services and activities. The binding protocol defines the association of a user to a specific resource in the space. This protocol will dynamically adapt itself based on the contextual information of the user including the location, activity, and role, to name a few. A descriptor is associated with each namespace that encompasses functional attributes represented in WSDL (Web Services Description Language) and RDF (Resource Description Framework), conditions for security, and policies for binding protocol. The binding protocol specifies whether the binding of a resource is 'shared' or 'private' and whether the binding is 'permanent' or 'context-based'. In the architecture, the 'context manager' provides the necessary contexts to the 'view manager' which is responsible for updating the 'view' that will be visible to the user based on this contextual information. Along with the context aware security model, the research provides a role based model to specify different activities in a pervasive computing environment.

As an extension of the project Centaurus [33, 34], researchers at the University of Maryland in Baltimore have presented Centaurus2 [62] that provides a secure mechanism for service discovery. It also enables the users to access services across heterogeneous network domains. In order to achieve the required security features like authentication and authorization with minimum interaction with the user, the system incorporates a modified simple version of Public Key Infrastructure (PKI). Here each entity has to be registered in the system. The Certificate Authority (CA) issues a certificate to each identified and verified entity. Smart Cards are used to store the digital certificates. It also contains the private key of the clients thus providing security into the functionality of the clients. PKCS #11 is used to contain the private keys of computing components like service managers and capability managers. The design of Centaurus2 encompasses five vital computational components:

1) The Certificate Authority is responsible for issuing digital certificates and for providing replies to the queries asking for validation of digital certificates.

2) The Communication Manager deals with the communication and interaction between clients and services. The communication is independent of protocol and medium.

3) The group membership(s) is maintained and stored by the Capability Manager. It maintains a database and ensures dynamic update of the file containing group membership information.

4) Each client is registered to a specific Service Manager that ensures security and access rights, and acts as the communicator between user client and service client.

Other jobs of Service Managers are to provide a list of services, and to send updated messages to all registered clients when a state is changed, etc. This project facilitates users with the access right of services in other domains by establishing a bridge between the root Service Managers of different networks. The researchers are now working to provide dynamic access rights rather than incorporating static access rights where a user will be able to access a service for which he does not have access rights through other granted users under certain conditions. The project has been implemented using Java and XML.

6 Trust

Trust is inseparably related to every aspect of authentication and authorization, and can be considered another face of the security coin. Several trust models have been defined starting from the middle of the last decade. Abdul Rahman and Hailes [48] introduced the notion 'distributed trust model' in 1998. Here each node has to maintain a storage of the trust value of other devices where the trust value spreads over -1 (complete distrust) to 4 (complete trust). Using master/slave pairing, Stajano [55] proposed a hierarchical model for trust requiring pre-configuration known as Resurrecting Duckling Model. In 2004 researches proposed a distributed trust model [47] based on an 'effort/return' mechanism. Upon introducing agents in the nodes that are responsible for collecting necessary information, it focuses on direct monitoring and devaluation of perceived trust for malicious devices. Here the trust value ranges from -1 to 1 and this model specifically deals with 'pure' ad hoc scenarios. The reputed game theory and distributed algorithm are the pillars of the trust model proposed by Sun [56]. Another example of a decentralized trust management model is 'PTM' [3] which provides data exchange features based on a recommendation protocol. Here each node is responsible for its personal security and maintains a chart that includes the identity of trusted and distrusted nodes, corresponding trust values, and other related information. The total trust formation and evolution contains two phases known as 'Belief Space' and 'Evidence Space'.

Very recently two researchers from Imperial College, London, proposed a dynamic model [31] for determining the trustworthiness of a context provider. Each Context Provider (CP) registers with the Service Directory (SD) and describes the attributes of the contexts according to their capabilities. When an application requires a specific context, it passes that query with a utility function to the SD; a value is then determined for that function by inputting the described attributes and trust value of a CP that can provide the required contextual information. This process is continued for all the CPs capable of serving that specific context. The CP with highest utility value achieves the responsibility of providing the context value. The trust value of a CP is a dynamically calculated value based on the binary feedbacks of other consumers of that service, and a value is generated by cross checking with other CPs that provide the same service. A Bayesian probability density function is used to calculate the trust value by taking the above stated facts as input. In order to calculate trust dynamically only the last n feedback values are used.

Researchers in [71] present a trust model which can perform in both 'pure' and 'managed' pervasive environment network structure with equal efficiency. This model identifies some essential attributes including absence of central authority and pre- configuration, distributed nature, miniature footprint, flexibility to customization, dynamic recalculation of trust value, etc. Based on some specific criterions, nodes have been classified into managerial nodes, independent nodes, and dependent nodes. The architecture encompassed the following main components:

1) User interface which is used by the administrator for customization.

2) rValue (Recommendation Value) Manager deals with the recommendation values.

3) Request/Response Handler handles query request and response.

4) Application Request Handler is responsible for handling queries about trust initiated by pervasive applications.

5) Trust Calculation handles the overall manipulation of trust.

Based on monitored data, recommendation values and recommenders' rValue, an overall trust value is calculated which has the range from -1 (complete distrust) to 1 (complete trust). This trust value calculation is performed periodically to ensure dynamicity. The framework has been implemented using C# and Compact .NET framework.

As part of SSRD (Simple and Secure Resource Discovery) researchers proposed a trust model [54]. Here a trust value of 1.0 represents complete trustworthiness and 0.0 represents complete untrustworthiness. A new node with no prior history of interaction receives a trust value of .5 indicating neither trust nor distrust. Here the attribute trust is service dependent and has the properties of reflexivity and transitivity. Each owner or manager of a device retains a table which indicates the security level required by each of the available services. The security level required by a service varies from 1 to 10. The resource managers also conserve another list named 'Service-trust', which describes the trust related to each available service for all the neighboring nodes. Whenever a request for service is received, this table is used as a look up table.

Risk is an important factor that we need to characterize in maintaining security. This is especially critical for the scenario when a decision has to be made about granting a privilege to an unknown entity who does not have proper recommendation; the associated probable risk needs to be calculated. Here risk has been considered as the probability that an interaction will lead to a catastrophic situation. This paper [12] has proposed a risk assessment model which will be embedded in the SECURE framework [10]. A general risk assessment formula $R = F(x_1, x_2, x_3, \ldots, x_m) + Z$ has been used where R is the probability of risk $x_1, x_2, x_3, \ldots, x_m$ denotes the components of feature vector, and Z is the random disturbance factor. The risk estimator first extracts features which are then clustered and the Average Loss Rate (ALR) is calculated. Mahalanobis distance is used in formulating the similarity between vectors. Later we find the Risk Probability (RP) of an interaction. This model differs from other static models because it has the capability to dynamically calculate the risk factor for a new interaction.

7 Open Issues

In the pervasive computing environment, we need a security policy that will simultaneously be an unobtrusive mechanism to the user as well as have the ability to discover the services available for the user in a transparent manner. The system needs a dynamic security policy which is flexible enough to update and modify on the fly. Both the user and the system need a secure access control and authorization mechanism that will act as a middleman and negotiate with both the parties to find a best possible service within the limitations imposed by both the participants. The augmentation of contexts in access control is enhancing the static security features towards dynamic security.

Heterogeneity

Due to the distributed and ad hoc nature of the pervasive computing environment, this system is open to several unique vulnerabilities and suffers from quite a number of well-known problems whose reputed solutions are not applicable here. Along with this, the capability of pervasive devices varies widely in terms of memory storage, battery power, computational capability, etc. For example, a RFID tag contains some hundred bits of information where as a latest PDA has the speed up to 400 MHz with 80 GB memory capacity. Again pervasive devices can appear from different domains with differing topologies, thus creating a thorny heterogeneous scenario which involves a complete unique set of vulnerability and susceptibility. Again as this scenario is facilitated by mobility of the user, a device can frequently change its domain thus moving from one network topology to other. There is no central administrative backbone that can provide the required characteristics of security with responsibility. As a result, the only option left is to make the small, tiny pervasive devices more responsible for their own security. But the burden of the security features may be too large for them due to their limitations in battery power, memory storage and computational capability. In a recent paper [49] author mentioned five obstacles in security and included barriers like privacy and trustworthiness of the devices as security issues.

Location detection

In this surrounding and because the number of devices can be really huge, it is very difficult to detect the physical device with which I am interacting. For this we need a secure communication channel along with device authentication. Again the request for establishing this trust channel is flowing through the shared, unreliable wireless channel. As an approach to solve this problem, the author in [49] has mentioned GPS and other location tracking systems for detecting the precise location of the interacting device. But we know that GPS doesn't work inside buildings and a feasible location tracking system which is applicable for tiny pervasive devices is still in the phase of research. A researcher from MIT has shown [70] the utility of using the Learning Parity with Noise (LPN) algorithm [28, 29] in the authentication mechanism of RFID where RFID was taken as a representative of tiny pervasive computing devices.

Access control

In case of access control, the system is based on the role and identity of the user. Again this privilege of accessing system resources and services is a variable which depends on the time, situation and other contextual information. Here the user needs to trust the pervasive computing environment including the resources and services available. At the same time the system needs to ensure the identity and access rights of the user. Though several access control mechanisms have been developed for several specific scenarios, we need a common framework which works in all scenarios with equal efficiency.

Privacy

In case of privacy two issues come up with equal priority:

1) Is privacy of the user being maintained?

2) Is privacy of the data being maintained?

Unlike distributed computing, pervasive computing likes to take user information and consider it as important contextual information. Though this contextual information plays a vital role in updating the system dynamically, it sometimes poses serious threat to the privacy of the user, especially in situations where people do not want to disclose their identity or location.

In a research [42, 43, 52] in IMSS General Hospital, Mexico the researchers formulated an ad hoc contextual information based hospital system which was very useful from the perspective of doctors, nurses, resident doctors and other medical staffs within the boundary of the hospital. But the availability of user information and displaying information in public created a negative impact on some users. As a result, the researchers had to make some changes in the design and put abstraction in the private information [22, 58, 59]. In Castro Valley, California, nurses of the medical center refused to incorporate the location tracking system as they believed that this would hamper their privacy [51]. As part of the famous Gaia project, developers have shown a privacy preserving hop by hop routing methodology [48] that carries information about the residing place of the user but it does not reveal the exact location or identity of the user. From these projects it is very evident that the privacy level and willingness of disclosure of personal information varies depending on information type, collection method, time and other concerns. In some scenarios users are reluctant to disclose identity information but don't care much about location information. The situation might be reversed for some others, and there are scenarios where users are reluctant to reveal both. An intelligent system which can identify these issues and can dynamically adopt a mechanism in relation with other contextual variables can be a project of attention.

Data communication

Privacy of the data encompasses two aspects. First, it has to ensure that data being shared or communicated is not being hacked by any active or passive eavesdroppers. As an initial thought, we consider several encryption and decryption techniques. But simultaneously we need to think about the other side of the coin which reminds us about the memory, battery power and other limitations. Along with that, the users in pervasive computing environment have much more flexibility and independence in mobility. This includes a large variety of domains ranging from well secured environments to totally open unsecured situations which makes the data security issue worse. Secondly, how can it be guaranteed that the user data which is being collected almost transparently will not be used maliciously? Or how we can ensure with certainty that the sophisticated data is not being manipulated by any unauthorized user?

Trust

In order to overcome several constraints, mutual cooperation, interconnectedness and inter dependability have been exposed as the obvious characteristics of pervasive computing environment. Along with these occurs the issue of trust. If data is shared with an unwarranted device, the probability of data security reduces automatically. Several trust models have been developed addressing various issues of trust [3, 47, 55]. Again, what should be the exact criteria for proving trust? This is one of the questions yet to be resolved.

Feedback

One of the characteristics of pervasive security is to minimize interaction with the user. As a result, the information being collected about the user remains almost transparent from users' point of view. In order to increase psychological satisfaction for the user, feedback can play a vital role. It will inform the user about the manipulation of the data and to whom and how it will be used. On one hand we need to minimize user involvement in security assurance mechanisms; on the other, pervasive devices are needed to be fed by several contextual information including location, identity, situation, time, etc., where in some cases user involvement is needed. The issue of balancing security with user interaction is always there.

8 Conclusions

In this paper, we have presented the current status of pervasive security area. The feedback model presented in the access control section is going to motivate many researchers as this is the first model in this issue, to the best of our knowledge. Risk is another issue that is inseparably related with trust, though it is not a heavily discussed issue in pervasive computing. This factor can play an important role in defining threshold values in trust. A discussion of this kind has been placed in the trust section. Overall, we tried to provide a complete summary of pervasive security with some diversified recent research and open issues. As a pervasive computing environment can come in different formats such as static (e.g. sensor network) or mobile (MANET), and pure (where administrator has no prior information about the ad hoc network) or managed (where administrator has some prior knowledge about the network), the security requirements also take different shapes. Combining all these concerns, security in pervasive computing has become a most complex issue. These concerns have to be resolved in every aspect to ensure this latest computing technology will flourish.

Acknowledgement

The authors would like to thank Paula Stroud for her valuable assistance in improving the paper.

References

[1] ACP-ASIM press release, American College of Physicians, (electronic citation), 9-3-2002.

[2] N. Aboudagga, M. T. Refaei, M. Eltoweissy, L. A. DaSilva, and J. Quisquater, "Wireless network security II: Authentication protocols for ad hoc networks: taxonomy and research issues," in *Proceedings of the 1st ACM International Workshop on Quality of Service & Security in Wireless and Mobile Networks Q2SWinet'05*, pp. 96–104, Oct. 2005.

[3] F. Almenarez, A. Marin, C. Campo, and C. Garcia, "PTM: A pervasive trust management model for dynamic open environments," in *Proceedings of International Conference on Pervasive Security, Privacy, and Trust (PSPT 2004)*, Massachusetts, 2004.

[4] T. D. Hodes, S. E. Czerwinski, B. Y. Zhao, A. D. Joseph, and R. H. Katz, "An architecture for secure wide-area service discovery," *ACM Wireless Networks Journal, special issue*, vol. 8, no. 2/3, pp. 213-230, 2002.

[5] F. Bagci, J. Petzold, W. Trumler, and T. Ungerer, "Ubiquitous mobile agent system in a P2P- Network," in *UbiSys-Workshop at the Fifth Annual Conference on Ubiquitous Computing*, Seattle, Oct. 2003.

[6] J. E. Bardram, R. E. Kjær, and M. Ø. Pedersen, "Context-aware user authentication - Supporting proximity-based login in pervasive computing," in *Proceedings of Ubicomp 2003: Ubiquitous Computing*, LNCS 2864, pp. 107-123, Seattle, Washington, USA, Springer Verlag, Oct. 2003.

[7] F. Bagci, H. Schick, J. Petzold, W. Trumler, and T. Ungerer, "Communication and security extensions for a ubiquitous mobile agent system (UbiMAS)," in *Proceedings of the 2nd Conference on Computing Frontiers*, pp. 246–251, May 2005.

[8] F. Bennett, T. Richardson, and A. Harter, "Teleporting - Making applications mobile," in *Proceedings of the IEEE Workshop on Mobile Computer Systems and Applications*, pp. 82-84, Los Alamitos, CA, USA, IEEE CS Press, 1994.

[9] M. Blaze, J. Feigenbaum and J. Lacy, "Decentralized trust management," in *Proceedings of the 17th IEEE Symposium on Security and Privacy*, pp. 164–173, May 1996.

[10] V. Cahill, B. Shand, E. Gray, N. Dimmock, A. Twigg, J. Bacon, C. English, W. Wagealla, S. Terzis, P. Nixon, C. Bryce, G. Di Marzo Serugendo, J.-M. Seigneur, M. Carbone, K. Krukow, C. Jensen, Y. Chen, and M. Nielsen, "Using Trust for Secure Collaboration in Uncertain Environments," *IEEE Pervasive Computing Magazine, special issue Dealing with Uncertainty*, vol. 2, no. 3, pp. 52-61, Jul-Sep 2003.

[11] R. H. Campbell, J. Al-Muhtadi, P. Naldurg, G. Sampemane, and M. D. Mickunas, "Towards security and privacy for pervasive computing," in *Proceedings of Theories and Systems, Mext-NSF-JSPS International Sympsoium, ISSS 2002*, pp. 1-15, Tokyo, Japan, Nov. 2002.

[12] Y. Chen, C. D. Jensen, E. Gray, V. Cahill and J. M. Seigneur, *A General Risk Assessment of Security in Pervasive Computing*, Technical Report: TCD-CS-2003-45, 6 Nov. 2003, https://www.cs.tcd.ie/publications/tech-reports/reports.03/TCD-CS-2003-45.pdf

[13] H. Chen, T. Finin, and A. Joshi, "Semantic web in the context broker architecture," in *Proceedings of 2nd IEEE International Conference on Pervasive Computing and Communications (PerCom 2004)*, pp. 277-286, Mar. 2004.

[14] D. Cotroneo, A. Graziano, S. Russo, "Security requirements in service oriented architectures for ubiquitous computing," in *Proceedings of the 2nd Workshop on Middleware for Pervasive and Ad-hoc Computing*, pp. 172–177, Oct. 2004.

[15] M. J. Covington, P. Fogla, Z. Zhan, and M. Ahamad, "A context-aware security architecture for emerging applications," in *Proceedings of 18th Annual Computer Security Applications Conference (ACSAC 2002)*, pp. 249–258, Dec. 2002.

[16] H. Deng, A. Mukherjee, D. P. Agrawal, "Threshold and identity-based key management and authentication for wireless ad hoc networks," in *International Conference on Information Technology: Coding and Computing (ITCC'04)*, pp. 107–111, 2004.

[17] W. Du, R. Wang, and P. Ning, "An efficient scheme for authenticating public keys in sensor networks," in *Proceedings of 6th ACM International Symposium on Mobile Ad Hoc Network and Computing (MobiHoc)*, pp. 58–67, 2005.

[18] C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, and T. Ylonen, *SPKI Certificate Theory*, RFC 2693, Sept. 1999.

[19] *Ensure Technologies*, http://www.ensuretech.com.

[20] K. Eustice, L. Kleinrock, S. Markstrum, G. Popek, V. Ramakrishna, and P. Reiher, "Securing nomads: The case for quarantine, examination, and decontamination," in *Proceedings of the 2003 Workshop on New Security Paradigms*, pp. 123–128, Aug. 2003.

[21] W. Farmer, J. Guttmann, and V. Swarup, "Security for mobile agents: Authentication and state appraisal," in *Proceedings of the Europea Symposium on Research in Computer Security (ESORICS)*, LNCS 1146, pp. 118-130, Springer-Verlag, 1996.

[22] J. Favela, M. Rodriguez, A. Preciado, and V. Gonzalez, "Integrating context-aware public displays into a mobile hospital information system," *IEEE Transactions on Information Technology in Biomedicine*, vol. 8. no. 3, pp. 279–286, Sept. 2004.

[23] D. Garlan, D. Siewiorek, A. Smailagic, and P. Steenkiste, "Project aura: Towards distraction-free pervasive computing," *IEEE Pervasive Computing*, vol. 1, no. 2, pp. 22-31, Apr-Jun 2002.

[24] I. Goldberg, S. D. Gribble, D. Wagner and E. A. Brewer, "The Ninja jukebox," in *Proceedings of the 2nd USENIX Symposium on Internet Technologies and Systems (USITS-99)*, pp. 37-46, Berkeley, CA, USENIX Association, 11-14 Oct. 1999.

[25] S. D. Gribble, M. Welsh, R. von Behren, E. A. Brewer, D. Culler, N. Borisov, S. Czerwinski, R. Gummadi, J. Hill, A. Joseph, R. H. Katz, Z. M. Mao, S. Ross, and B. Zhao, "The Ninja architecture for robust Internet-scale systems and services," *Computer Networks*, vol. 35, no. 4, pp. 473-497, Mar. 2001.

[26] Harris interactive poll results, Harris Poll (2002): Physician's use of hand-helds increases from 15% in 1999 to 26%, in 2001, (electronic citation), 24 Aug. 2002.

[27] U. Hengartner, and P. Steenkiste, "Exploiting information relationships for access control," in *Third IEEE International Conference on Pervasive Computing and Communications, PerCom 2005*, pp. 269-278, 8-12 Mar. 2005.

[28] N. Hopper and M. Blum, *A Secure Human-Computer Authentication Scheme*, Technical Report: CMU-CS-00-139, Carnegie Mellon University, 2000.

[29] N. J. Hopper and M. Blum, "Secure human identification protocols," in *Advances in Cryptology - ASIACRYPT 2001*, LNCS 2248, pp. 52-66, Springer-Verlag, 2001.

[30] J. Howell and D. Kotz, "End-to-end authorization", in *Proceedings of 4th Symposium on Operating System Design & Implementation (OSDI 2000)*, pp. 151-164, Oct. 2000.

[31] M. C. Huebscher, and J. A. McCann, "A learning model for trustworthiness of context-awareness services," in *Third IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom 2005*, pp. 120-124, Mar. 2005.

[32] G. Judd and P. Steenkiste, "Providing contextual information to ubiquitous computing applications," in *Proceedings of IEEE International Conference on Pervasive Computing and Communications (PerCom 2003)*, pp. 133-142, Mar. 2003.

[33] L. Kagal, V. Korolev, S. Avancha, A. Joshi, T. Finin and Y. Yesha, *Highly Adaptable Infrastructure for Service Discovery and Managementin Ubiquitous Computing*, Technical Report: TR CS-01-06, Department of Computer Science and Electrical Engineering, University of Maryland Baltimore County, Baltimore, MD, 2001.

[34] L. Kagal, V. Korolev, H. Chen, A. Joshi and T. Finin, "Centaurus: a framework for intelligent services in a mobile environment," in *Proceedings of International Workshop on Smart Appliances and Wearable Computing IWSAWC, in the 21st International Conference on Distributed Computing Systems (ICDCS-21)*, pp. 195–201, Department of Computer Science and Electrical Engineering, University of Maryland Baltimore County, Baltimore, MD, Apr, 2001.

[35] P. Kaijser, T. Parker, and D. Pinkas, "SESAME: The solution to security for open distributed systems," *Computer Communications*, vol. 17, pp. 501-518, 1994.

[36] A. Kapadia, G. Sampemane, and R. H. Campbell, "Access control: KNOW Why your access was denied: Regulating feedvack for usable security," in *Proceedings of the 11th ACM Conference on Computer and Communications Security*, pp. 52–61, Oct. 2004.

[37] G. Karjoth, D. B. Lange, and M. Oshirma, "A security model for Aglets," *IEEE Internet Computing*, pp. 68–77, July-Aug 1997.

[38] E. C. Lupu, D. A. Marriott, M. S. Sloman and N. Yialelis, "A policy based role framework for access control," in *Proceedings of the First ACM Workshop on Role-Based Access Control*, pp. 11, 1995.

[39] E. Lupu and M. Sloman, "A policy based role object model," in *First International Enterprise Distributed Object Computing Workshop (EDOC'97)*, pp. 36-47, Oct. 1997.

[40] K. Minami, and D. Kotz, "Secure context-sensitive authorization," in *Third IEEE International Conference on Pervasive Computing and Communications, PerCom 2005*, pp. 257-268, Mar. 2005,

[41] J. A. Muhtadi, A. Ranganathan, R. Campbell, and M. D. Mickunas., "Cerberus: A context-aware security scheme for smart spaces," in *Proceedings of IEEE International Conference on Pervasive Computing and Communications (PerCom 2003)*, pp. 489-496, Mar. 2003.

[42] M. Muñoz, V. Gonzalez, M. Rodríguez, and J. Favela, "Supporting context-aware collaboration in a hospital: An ethnographic informed design," *Proceedings of Workshop on Artificial Intelligence, Information Access, and Mobile Computing 9th International Workshop on Groupware*, LNCS 2806. pp. 330-344, Springer-Verlag, Sept. 2003.

[43] M. A. Muñoz, M. Rodriguez, J. Favela, A. I. M. Garcia, and V. Gonzalez, "Context-aware mobile communication in hospitals," *IEEE Computer*, vol. 36, no. 9, pp. 38-46, 2003.

[44] E. C. H. Ngai and M. R. Lyu, "Trust- and clustering-based authentication services in mobile ad hoc networks," in *Proceedings of 24th International Conference on Distributed Computing Systems Workshops - W4: MDC (ICDCSW'04)*, pp. 582–587, 2004.

[45] J. L. Nielsen, *BuDDy - A Binary Decision Diagram Package*, Technical Report: IT-TR: 1999-028, Technical University of Denmark, 1999.

[46] J. Page, A. Zaslavsky, and M. Indrawan, "A buddy model of security for mobile agent communities operating in pervasive scenarios," in *Proceedings of the Second Workshop on Australasian Information Security, Data Mining and Web Intelligence, and Software Internationalisation - CRPIT'04*, vol. 32, pp. 17–25, Jan. 2004.

[47] A. Pirzada and C. McDonald, "Establishing trust in pure ad-hoc networks," in *Proceedings of the 27th Conference on Australasian Computer Science*, vol. 26, pp. 47–54, 2004.

[48] A. A. Rahman and S. Hailes, "A distributed trust model," *Proceedings of the 1997 Workshop on New Security Paradigms*, pp. 48–60, 1998.

[49] K. Ranganathan, "Trustworthy pervasive computing: The hard security problems", in *Proceeding of 2nd IEEE Conference on Pervasive Computing and Communications Workshops (PerCom 2004 Workshops)*, pp. 117-121, Orlando, USA, 14-17 Mar. 2004.

[50] L. Rasmusson and S. Jansson. "Simulated social control for secure internet commerce," in *New Security Paradigms '96*, pp. 18–26, ACM Press, 1996.

[51] P. Reang," Dozens of nurses in Castro Valley balk at wearing locators,"in *The Mercury News*, 2002.

[52] M. Rodriguez, J. Favela, V. Gonzalez, and M.A. Muñoz, "Agent-based mobile collaboration and information access in a helathcare environment," in *eHealth: Application of Computing Science in Medicine and Healthcare*, pp. 133-148, Published by the Instituto Politecnico Nacional. Mexico, ISNB 970-36-0118-9, 2003.

[53] M. Rom'an, C. K. Hess, R. Cerqueira, A. Ranganathan, R. H. Campbell, and N. Klara, "GaiaOS: A middleware infrastructure to enable Active Spaces," *IEEE Pervasive Computing*, pp. 74-83, Oct.-Dec. 2002.

[54] M. Sharmin, S. Ahmed, and S. I. Ahamed, "An adaptive lightweight trust reliant secure resource discovery for pervasive computing environments," in *Fourth Annual IEEE International Conference on Pervasive Computer and Communications (PerCom 2006)*, pp. 258-263, Pisa, Italy, Mar. 2006.

[55] F. Stajano and R. Anderson, "The resurrecting duckling: Security issues for ad-hoc wireless networks," in *Proceedings of the 7th International Workshop on Security Protocols*, pp. 172–194, 1999.

[56] H. Sun and J. Song, "Strategyproof trust management in wireless ad hoc network," in *Proceedings of the IEEE Canadian Conference on Computer and Electrical Engineering*, vol. 3, pp. 1593–1596, 2004.

[57] M. T. Refaei, V. Srivastava, L. DaSilva, and M. Eltoweissy, "A reputation-based mechanism for isolating selfish nodes in ad hoc networks," in *Proceedings of the IEEE Mobiquitous*, pp. 3–11, San Diego, CA, 2005.

[58] M. Tentori, J. Favela, V. Gonzalez, and M. Rodríguez, "Supporting quality of privacy (QoP) in pervasive computing," in *Proceedings of Encuentro Internacional de Ciencias de la Computación (ENC)*, pp. 58–67, IEEE Computer Press, Puebla, Mexico, 2005.

[59] M. Tentori, J. Favela, and V. González, "Designing for privacy in pervasive hospital environments", in *Prooceedings of UCAMI*, Granada, España, 2005.

[60] A. Tripathi, T. Ahmed, D. Kulkarni, R. Kumar, and K. Kashiramka, "Context-based secure resource access in pervasive computing environments," in *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp. 159-163, 14-17 Mar, 2004.

[61] W. Trumler, F. Bagci, J. Petzold, and T. Ungerer, "AMUN - autonomic middleware for ubiquitious environments applied to the smart doorplate project," in *International Conference on Autonomic Computing (ICAC-04)*, pp. 274-275, New York, May 2004.

[62] J. Undercoffer, F. Perich, A. Cedilnik, L. Kagal, and A. Joshi, "A secure infrastructure for service discovery and access in pervasive computing," *Mobile Networks and Applications*, vol. 8, Issue 2, pp. 113–125, Apr. 2003.

[63] Web link, www.microsoft.com/security/glossary.mspx

[64] Web link, en.wikipedia.org/wiki/Computer-security

[65] Web link, http://searchnetworking.techtarget.com/originalContent/0,289142,sid7-gci936766,00.html

[66] Web link, http://www.cisco.com/en/US/netsol/networking-solutions-networking-basic09186a00800a3549.html

[67] Web link, http://www.nist.gov/pc2001/about-pervasive.html

[68] A. Weirmerskirch and D. Westhoff, "Identity certified authentication for ad-hoc networks," in *Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks*, pp. 33–40, VA USA, 2003.

[69] M. Weiser, "Some computer science problems in ubiquitous computing," *Communications of the ACM*, vol. 36, no. 7, pp. 75-84, July 1993.

[70] S. A. Weis, "Security parallels between people and pervasive devices," *Pervasive Computing and Communications Workshops, 2005, Third IEEE International Conference*, pp. 105-109, 2005.

[71] S. T. Wolfe, S. I. Ahamed, and M. Zulkernine, "A trust framework for pervasive computing environments," in *The 4th ACS/IEEE International Conference on Computer Systems and Applications*

(AICCSA-06), IEEE CS Press, pp. 312-319, Dubai, UAE, Mar. 2006.

[72] T. Woo, "Dynamic security in pervasive computing," *ECE750 presentation*, University of Waterloo, Apr. 2003.

[73] C. Wullems, M. Looi, and A. Clark, "Towards context-aware security: an authorization architecture for intranet environments," in *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp. 132-137, Mar. 2004.

[74] F. Zhu, "Splendor: A secure, private, and location-aware service discovery protocol supporting mobile services," in *Proceedings of the 2003 IEEE Annual Conference on Pervasive Computing and Communications (Percom 2003)*, pp. 235–242, Mar. 2003.

Munirul Haque is a graduate student in the department of Math., Stat. and Computer Science at Marquette University, USA. Haque received his B.Sc. in computer science and engineering from the Bangladesh University of Engineering and Technology, Bangladesh in 2004. His research interests are security in pervasive computing and middleware for ubiquitous/pervasive computing. He can be contacted at md.haque@mu.edu; http://www.mscs.mu.edu/ mhaque

Sheikh Iqbal Ahamed is an assistant professor in the department of Math., Stat. and Computer Science at Marquette University, USA. He is a member of the IEEE, ACM, and the IEEE Computer Society. Dr. Ahamed received the B.Sc. in computer science and engineering from the Bangladesh University of Engineering and Technology, Bangladesh in 1995. He completed his Ph.D in Computer Science from Arizona State University, USA in 2003. His research interests are security in ad hoc networks, middleware for ubiquitous/pervasive computing, sensor networks, and component-based software development. He serves regularly on international conference program committees in software engineering and pervasive computing such as COMPSAC 04, COMPSAC 05, COMPSAC 06, and ITCC 05. He is the Workshop Program Co-Chair of International Workshop on Security, Privacy, and Trust for Pervasive Computing (SPTPA 06). He also directs the Ubicomp research lab (www.mscs.mu.edu/ ubicomp) in the the department of Math., Stat. and Computer Science at Marquette University, USA.. Dr. Ahamed can be contacted at iq@mscs.mu.edu; http://www.mscs.mu.edu/ iq.