Challenges in Securing Pervasive Computing Applications

Indrakshi Ray
Computer Science Department
Colorado State University
Fort Collins, CO 80523-1873
iray@cs.colostate.edu

Abstract

Pervasive computing technology has the potential to benefit applications in the military, financial, and health care domain. Although pervasive computing technology looks promising, one critical challenge needs to be addressed before it can be widely deployed – security. The problem is serious because pervasive computing applications involve interactions between a large number of entities that can span different organizational boundaries. Unlike traditional applications, these applications do not usually have well-defined security perimeters and are dynamic in nature. Moreover, these applications use knowledge of surrounding physical spaces. This requires security policies to use contextual information that, in turn, must be adequately protected from security breaches. Uncontrolled disclosure of information or unconstrained interactions among entities can lead to very serious consequences. Traditional access control policies and mechanisms rarely address these issues and are thus inadequate for securing pervasive computing applications.

1 Motivation

With the growth of mobile and sensor devices, embedded systems, and communication technologies, we are moving towards an era of pervasive computing. Pervasive computing uses numerous, casually accessible, often invisible computing and sensor devices, that are frequently mobile or embedded in the environment and that are inter-connected to each other with wireless or wired technology. This allows pervasive computing to provide services and functionalities that use the knowledge of surrounding physical places. Although this technology looks promising, security issues must be addressed before this technology can be widely deployed. Designing secure pervasive applications requires one to understand what resources an entity has access to, which entities it should interact with, what information can be released to an entity, how to protect the information used or produced by an entity, which entities can be trusted and to what extent, and how these trust relationships change over time.

Consider a potential use of such technology: offering real-time health care for patients. A cardiac patient lives independently in a smart home equipped with sensors and wireless controllers. The patient’s movements are tracked by sensors and wireless controllers send this information to a monitoring service that oversees the patient’s condition and initiates appropriate action, such as, alerting emergency services. To perform, the monitoring service needs access to the patient’s medical history maintained by a health care provider. In an emergency, the records must also be shared with the admitting hospital which will perform examinations, such as X-rays and ECGs. The hospital may have to consult experts unfamiliar with the patient or search for similar reports to interpret the patient’s case.

Security and privacy are a major concern for such applications. Preventing data transmission to the monitoring service or sending false data may be fatal. Sending too many false alarms can cripple emergency services. Disclosing the patient’s health data to prospective employers may cause financial hardship and
disclosing the data to unapproved doctors causes breach of privacy. Comparing a patient’s report to unauthen-
tic reports of other patients may result in incorrect diagnosis. These severe consequences motivate the
need to consider security issues when designing secure pervasive computing applications.

Security policies and mechanisms developed for traditional applications are inadequate for pervasive
computing applications. Unlike traditional applications, pervasive computing applications have no definite
security perimeters – the entities an application will interact with or the resources that will be accessed may
not be known in advance. These applications are also dynamic in nature – the accessing entities may change,
resources requiring protection may be created or modified, and an entity’s access to resources may change
during the course of the application. Protecting resources during application execution remains challenging.
Moreover, pervasive computing applications use the knowledge of surrounding physical spaces to provide
services which requires security policies to use contextual information. For instance, access to a resource
may be contingent upon the location of the user and time of day. This contextual information can be used
to infer the activities of the user and cause a privacy breach. Contextual information must, therefore, be
protected by security and privacy policies.

Researchers are working on augmenting traditional security policies and mechanisms to accommodate
new applications. Examples include the development of new access control models and technologies [17,
18, 21, 29, 39, 34, 35, 37, 38], formalizing the notion of trust [1, 3, 7, 12, 15, 16, 20, 22, 23, 24, 25, 26, 30,
31, 32, 33, 36, 40, 41, 42, 43, 44, 45, 46], trust management and trust negotiation strategies [4, 10, 8, 9, 14,
5, 6, 11, 47, 48]. Some researchers [2, 13, 19, 27, 28] have also addressed security, privacy and trust issues
of pervasive computing environments. However, we do not yet have a comprehensive solution describing
the security and privacy policies necessary for pervasive computing applications.

2 Directions for Future Research

In the following, we briefly describe some research issues that must be addressed for controlling the access to
pervasive computing applications. In traditional models, namely Discretionary Access Control (DAC), Bell-
LaPadula (BLP), and Role-Based Access Control (RBAC), the identity of the user, the security clearance of
the user, and the role of the user respectively determine what objects the user can access. Since the set of
users involved in a pervasive computing application may not be known in advance, it may not be possible
to determine the values of these specific attributes. We need some generalized attribute that will allow
pervasive applications to interact with both known and unknown users. We propose formalizing the concept
of trust and using this to determine the access privileges of the various entities. Trust is a relationship
between entities and it depends on multiple factors. Quantifying this relationship is important because the
nature of interaction will depend on the level of trust among the participating entities. This quantification
will enable one to reason about and compare the trustworthiness of different entities. In cases, where trust
relationship does not exist between participating entities, trust negotiation must be performed to establish
trust. Performing trust negotiation in an environment where some entities have limited resources remains a
challenge.

In traditional models, access control depends on specific attributes of the users and the objects. In perva-
sive computing applications, attributes of the users and objects are important, but so are other environmental
contexts, such as, location and time. These contexts will determine where and when certain access is al-
lowed. For example, night time nurse has the permission to administer emergency drugs to a patient in his
home during the night, but may not do so at day time. In addition to the environmental contexts location
and time, there are application dependent contexts which may have to be taken into account for deciding
access. For instance, in an emergency situation, a patient’s health record may be shared with out of network
physicians, but this type of access is prohibited when the patient is feeling well. The access control model
should be able to formalize and reason about these different types of contexts and make informed decisions.
Pervasive computing applications are dynamic in nature; the entities participating in an application and the types of interactions are subject to change. The application context may change, which, in turn, may change the rules governing access. Such changes may occur while the access control policies are deployed. We need to identify and formalize the changes that will be permitted so as to ensure that access control policies are not violated. We also need to propagate the changes to the policies and enforce them in a timely manner so as to ensure that security breaches do not occur.

The complexity of the pervasive computing applications and the access control models protecting them necessitates rigorous analysis to ensure that applications are adequately protected. The analysis will proceed at two levels. First, we need to verify that the access control models developed for pervasive computing applications are consistent and the interaction of their various features do not produce any conflicts. Second, we need to investigate how well a given set of access control policies, conforming to some model, protect a pervasive computing application. Pervasive computing applications will typically be modeled as dynamic workflows. We need to study the interactions of access control policies with workflow dependencies, and provide assurance that the workflow is able to complete without violating any policy. We need to investigate what types of verification and validation techniques are useful for doing such types of analysis.

Since a pervasive computing application involves multiple domains, we also need to formalize the concept of secure interoperation. Secure interoperation must allow controlled sharing of resources. Since each domain will have its own policy and an entity may belong to multiple domains at a given time, we also need to understand how to combine these policies, what types of conflicts occur when such policies are combined and how do we resolve such conflicts. We also need to analyze what effect the combined policies have on secure interoperation.

References


