Modelling Context-Aware Security for Electronic Health Records

Pravin Shetty
Monash University, Australia

Seng Loke
La Trobe University, Australia

INTRODUCTION

The Internet has proven to be the most convenient and demanding facility for various types of businesses and transactions for the past few years. In recent years, business information systems have expanded into networks, encompassing partners, suppliers, and customers. There has been a global availability (Anderson, 2001; BSI Global, 2003) of resources over the Internet to satisfy different needs in various fields. The availability factor has called for various security challenges in fields where information is very valuable and not meant for all. Potential threats to information and system security come from a variety of sources. These threats may result in violations to confidentiality, interruptions in information integrity, and possible disruption in the delivery of services. So it is essential to manage the flow of information over the network with the required level of security. There are many security technologies and models that have been introduced which are capable of realizing the functions and objectives of information system security.

This article first gives a brief overview of what we term basic security policies of an integrated security model. Then it suggests context-based security policies for a health organization scenario using contextual graphs augmented with details about specific security actions, which relate to the security policies enumerated in the integrated security model.

The plan of the article is as follows. We first overview the three concepts in detail and briefly describe the concept of contextual (meta-policy) graphs. We then develop a context-based security meta-policy for securing patient records based on the security policies overviewed and discuss related work, before concluding the paper.

BASIC SECURITY POLICIES

Mobile ambients were first proposed by Cardelli and Gordon (1998a, 1998b) and then further extended by Bugliesi, Castagna, and Crafa (2004); and Braghin et al. (2002) were very efficient in modeling multilevel security issues. These three notions are very effective in modeling a foolproof security solution in a computing scenario by stating various security steps to be taken in the corresponding scenario. On this basis we have five cases that form the basic security policies in this article which we note can be concisely and precisely modeled using the mobile ambients formalism, though we omit such details of the formalism here and only describe the policies in plain language. The article uses them in appropriate scenarios depending on the context. Thus, the combined use of these five policies and a contextual graph representing the contexts of use of these policies provides a context-based security solution for pervasive environments. This section briefly describes the five policies using ambient (representing a boundary of security restrictions) notions.

Policy 1: Authenticate Returning Mobile Agent

When a privileged process (agent or person) leaves the parent ambient (e.g., a host institution) to execute some external independent activities, it relinquishes its local privileges and authority within its bounding parent ambient and ambient community. It exits the parent and might later return to the parent ambient. At this point an *authentication mechanism* is needed to check the authenticity of the returning original process. Cardelli and Gordon (1998a, 1998b, 1999) suggest that these high-level privileges must not be automatically

restored to the returning agents/processes without first verifying their identity. This is to preserve the security and integrity of the ambient as well as the services and resources contained within it.

Policy 2: Firewall Access

If any agent/process has to enter an ambient, it has to know the name of the ambient and also possess the capability to enter it. The functionality of firewall is achieved with the help of restriction primitives and with the help of anonymity of the ambient name. Thus without knowing the ambient name, no process or agent can exit or enter the parent ambient. This helps in achieving protection of the resources from unwanted agents. The ambient name could be interpreted as a secret password.

Policy 3: Encryption Using Shared Keys to Secure the Data While Communicating

Cardelli and Gordon (1998a, 1998b, 1999) also put forth the encryption primitives to communicate between two ambients or between an ambient and a remote agent. These primitives helped in maintaining the *confidentiality* of the message or data. Consider a Plaintext message *M*. The encryption of the plaintext message is done with the help of the encryption key *k*. A name can represent a shared key, as long as it is kept secret and shared only by certain parties. A shared key can be reused multiple times, for example, to encrypt a stream of messages. A message encrypted under a key *k* can be represented as a folder that contains the message and whose label is *k* (Cardelli & Gordon, 1998a, 1998b).

Policy 4: Security Across Multiple Levels

In general, an enclosed ambient environment would typically contain numerous subambients as well as active processes, agents, and information resources. These groups of subambients within an ambient may be arbitrarily nested and organized in a hierarchical structure. Ambients and processes at the higher level of the nested structure are responsible for managing resources that are more vital and important than those at a lower level. In such multilevel environments, it is necessary to restrict the access to the flow of information depending upon the need and the security levels. Information can only flow from lower levels of security to higher levels and not conversely. A policy for this assigns levels to users and restricts information flow among the users.

Policy 5: Movement of Data and Entities Through Different Communities

The multilevel security policy mandatory access control security in the boxed ambients provided restricted access to information based on the various security levels in the hierarchical levels. The access is defined by the level at which the agents are which are predetermined based on their needs. But Braghin et al. (2002) were of the view that the implementation of mandatory access control security is complex, as agents and processes may move from one security level to another. The agents themselves may be confidential or may be carrying secure/confidential information. Thus there is no way of ensuring the agents will not be illegally attacked, accessed, or executed by untrustworthy entities at the lower security levels. The *security boundary* concept put forth by Braghin et al. (2002) guarantees absence of information leakage.

According to this concept, every high-level data or process should be encapsulated in a boundary ambient. A boundary ambient can be opened only when it is nested into another pre-specified boundary ambient. A policy for this states that the protected information cannot be read without being contained within some safety boundary (e.g., physically, an item cannot be viewed in the absence of a bodyguard).

CONTEXTUAL (META-POLICY) GRAPHS

Contextual meta-policy graphs are derived from contextual graphs (Braghin et al., 2002; Bugliesi et al., 2001a, 2001b). We replace the security actions in contextual graphs by security policies, which in turn, represent the security actions accordingly. By virtue of this embedding of policies (such as the five mentioned above) into a contextual graph, the graph becomes a meta-policy construct. The contextual meta-policy graphs are very general and can be used to depict security architecture in any scenario. The use of such graphs is to provide a high-level picture of the security framework, thereby avoiding lower details (security