From the Editor in Chief

Editor in Chief: M. Satyanarayanan ■ satya@cs.cmu.edu

Privacy: The Achilles Heel of Pervasive Computing?

M. Satyanarayanan

At the heart of the ubiquitous computing vision lies an inherent contradiction. On the one hand, a computing environment must be highly knowledgeable about a user to conform to his or her needs and desires without explicit interaction—almost reading the user's mind. On the other hand, a system that is truly ubiquitous will encompass numerous users, physical regions, and service providers. At such large scale, perfect trust among all parties is an unattainable ideal. Trust boundaries thus represent seams of discontinuity in the fabric of pervasive computing.

Privacy and security are already thorny problems in distributed systems. A variety of problems plague us, ranging from spam to identity theft. Pervasive computing provides many new avenues of attack. Mechanisms such as location tracking, smart spaces, and the use of surrogates require continuous monitoring of user actions. As a user becomes more dependent on a pervasive computing system, the system becomes more knowledgeable about that user's movements, behavior patterns, and habits. Exploiting this information is critical if the system is to be proactive and self-tuning. Yet this same build-up of detailed knowledge about a user represents a tempting target for the unscrupulous. Unless we can develop satisfactory solutions, the potential for serious loss of privacy might deter users from relying on a pervasive computing system.

Establishing trust is a two-way problem. Just as users must be confident of their computing environment's trustworthiness, the infrastructure must be confident of a user's identity and authorization level before responding to requests. It is difficult to establish this mutual trust in a manner that is minimally intrusive.

This will become a key requirement as pervasive computing moves from the lab to the real world. Without a reliable and accurate way to establish identity and authorization, service providers won't have incentives for deploying the infrastructures and services necessary for pervasive computing. At the same time, frequent demands for passwords or other proofs of authenticity from the user will destroy the essence of pervasive computing—namely, its ability to disappear into the user's subconscious. It is critical to develop techniques that balance these divergent requirements. I can think of at least three ways to begin the search for such techniques.

Perfect trust among all parties is an unattainable ideal. Trust boundaries thus represent seams of discontinuity in the fabric of pervasive computing.

INCREASING AWARENESS

A small step in the right direction would be to make users more aware of their current privacy exposure level. Just as a car's dashboard continuously provides information such as speed, fuel level, and outside temperatures, a handheld or wearable computer could unobtrusively indicate what information the system is exporting to its surroundings. For example, is the system sharing a user's identity or allowing its location to be tracked? Are network transmissions encrypted? A user uncomfortable with the current exposure level should able to temporarily restrict the export of certain kinds of information. This would undoubtedly hurt invisibility but it offers the user an explicit way to override an erroneous system choice.

Of course, systems must present this information unobtrusively, so it can be absorbed by a user's peripheral consciousness without distracting the user from a primary activity. The dashboard metaphor is thus a good fit: a driver should focus on the road ahead, only glancing at the dashboard from time to time.

MAINTAINING AN AUDIT TRAIL

The system should maintain an audit trail of privacy-related interactions. What service requested a particular piece of information, where and when? On what grounds was the information released? Who demanded user authentication and when? How was the request satisfied?

We can keep the computational overhead (in terms of disk space and CPU overhead) of this auditing relatively low because these events typically occur on

Published by the IEEE CS and IEEE Communications Society ■ 1536-1268/03/$17.00 © 2003 IEEE

the timescale of seconds rather than microseconds. Maintaining a comprehensive privacy audit trail should thus be feasible even on resource-limited mobile hardware. Although auditing cannot prevent privacy violations, it can help provide clues for forensic analysis of how a particular violation occurred. Additionally, the fear of being discovered can sometimes serve as a useful deterrent.

CREATING A "SIXTH SENSE"

A more ambitious goal would be to provide users with a "sixth sense" that alerts them to serious privacy threats. In the real world, such visceral mechanisms play a key role in survival. When you enter a bad neighborhood, your sixth sense warns you that danger lurks nearby, and your behavior changes: you quicken your pace, stay in well-lighted areas, and exit the area as quickly as possible. Can we provide a sixth sense for pervasive computing? How does a user's wearable or handheld computer trigger an alert, and how is that alert communicated viscerally to the user? What are the preventive actions that then become available to the user?

From a broader perspective, we must revisit many authentication and authorization questions in the context of pervasive computing. For example, what authentication techniques are best suited to pervasive computing? Are password-based challenge-response protocols such as Kerberos adequate, or are more exotic techniques such as biometric authentication necessary? What role, if any, can smart cards play in pervasive computing? As another example, how can we express generic identities in access control? How do we express security constraints, such as "Only the person currently using the projector in this room can set its lighting level" or "Only employees of our partner companies can negotiate QoS properties in this smart space"?

Clearly, the security and privacy challenges of pervasive computing will keep researchers, designers, and implementers busy for a long time! P