Building a Trusted Location Service for Pervasive Computing Environments

Raquel Hill[†], Jalal Al-Muhtadi[*]
[†]*Indiana University, USA*
rahill@cs.indiana.edu
[*]*King Saud University, Saudi Arabia*
jalal@ccis.edu.sa

Abstract

Location- and context-awareness significantly enhance the functionality of pervasive computing services and applications, and enrich the way they interact with users and resources in the environment. Much of this functionality depends on validating context information and using it for granting access to data or resources. Since location and context services are usually composed of various distributed components spread throughout the pervasive environment, it is essential to develop a threat model that can be used as basis for developing a trustworthy platform for managing location and context information. In this paper we examine the various ways that trust can be incorporated and asserted in a location service for a pervasive environment. We specify the trust requirements for a non-forgeable location service for a pervasive computing space. Our approach is unique in that it is based on certification of behavior rather than a traditional reputation-based approach to trust. In our system, trust in a particular component is derived from its ability to attest that its actions meet the trust requirements.

1. Introduction

Pervasive computing has inspired the construction of smart, information-rich physical spaces that encompass large numbers of interconnected computing devices and embedded processors. These devices create a “dust” of computing machinery that allows users to interact seamlessly with the surrounding environment. This dust of computing machinery will provide new functionality, offer personalized services, and support omnipresent applications. Location awareness enables significant functionality for pervasive computing applications, users, resources and the ways they interact. It allows pervasive computing environments to tailor themselves according to users’ preferences and expectations, and reconfigure the available resources in the most efficient way to meet users’ demands and provide seamless interaction. For example, applications and data can follow users as they roam around, content can be customized based on users’ location, physical surroundings can be customized according to their inhabitants, and security services can be enhanced with accurate location detection.

In previous work [1] we aimed at providing enhanced levels of security and authentication through location and context. This included augmenting existing security technologies like authentication, confidentiality, and access control with location information, to provide an additional layer of security. Additionally, the system introduced new location-aware security enhancements that include location authentication, location-based access control models, and location-based encryption. A major barrier in the practical deployment of these technologies in any pervasive environment is the security and trustworthiness of the location data collected or sensed. Having a secure reliable location service is a cornerstone in any pervasive computing setup. While a great deal of work went into designing and implementing location services for pervasive environments [2-4], most solutions assumed the location detection techniques are trusted and ignored the consequences of mistrust or location forgery.

In this paper, we offer several contributions to this field. First, we develop a threat model that can be used as basis for developing a trustworthy platform for managing location and context information. Second, we examine the various ways that trust can be incorporated and asserted in a location service consisting of multiple autonomous components in a pervasive environment. Finally, we specify the trust requirements for a non-forgeable location service for a pervasive computing space and discuss the use of these technologies in our testbed. A distinguishing feature of our approach is our reliance on certification of behavior rather than a traditional reputation-based approach to trust.

The remainder of this paper is divided as follows. Section 2 introduces trust terminology and talks about related work. Section 3 explains the location service architecture. Section 4 describes our threat model. Section 5 addresses these threats. Finally, Section 6 concludes.

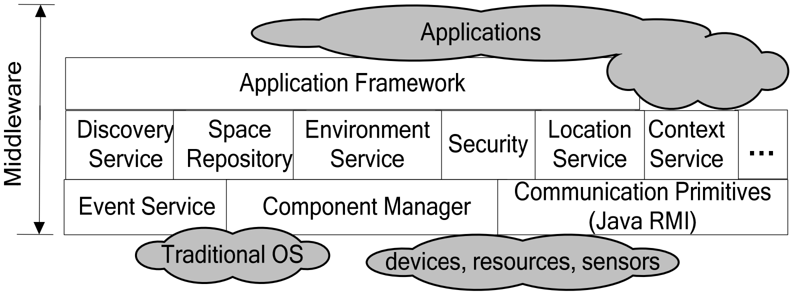

Figure 1: Moheet Architecture

2. Trust Terminology and Related Work

Trust is "the firm belief in the competence of an entity to act dependably, securely and reliably within a specified context" [5]. This type of behavioral trust may be derived from an entity's ability to prove that its behavior meets a trusted criteria. The Trusted Platform Module (TPM) [6] and its remote attestation function has been proposed to enable a device authenticate its behavior or function. The TPM is a chip that is embedded on the motherboard. Remote attestation creates a digitally signed summary of the software that is running on a computer. This summary allows a third party to determine whether the software has been changed. The TPM was developed with the assumption that any change means compromise. While this assumption may hold for systems that protect against pirated software and copyrighted materials, it does not hold for systems where updates to software are common and all changes are not necessarily malicious. Remote attestation, alone, does not address the problem of device authentication for pervasive computing environments and other distributed environments where the infrastructure must be trusted to perform functions in a secure and reliable manner.

Reputation systems [7] have also been used to determine belief in an entity to perform a function per some specific requirements. A reputation system uses the opinions of other entities in the system to formulate a rating or belief about the entity that is to be measured. Reputation systems are most useful in online communities where users often interact with unknown parties. The use of reputation systems is less feasible in environments where past and previous behavior cannot be correlated because of the presence malicious software.

3. Location Service Configuration

We briefly explain our testbed environment. Our approach for enabling pervasive computing consists of constructing a middleware that provides the necessary core functionality for constructing general-purpose pervasive environments, which we refer to as *smart spaces*. The *Moheet* architecture [8] is an extension of Gaia [9], that is lightweight, platform independent and features enhanced support for mobility. Moheet is fully implemented in Java and Java Micro Edition (J2ME), and utilizes lightweight Java RMI (remote method invocation) for the discovery, management, and communication among distributed objects. The use of J2ME enables Moheet services and applications to cater to mobile devices and mobile users. Moheet consists of a three-tier architecture, as illustrated in Figure 1. The 1st tier, consists of the core primitives that are needed in any transparent distributed system. This includes a lightweight event service that allows events to be communicated between distributed devices. The communication primitives provide facilities for communication between distributed components. The 2nd tier consists of the basic services and functionality that are needed by any pervasive computing environment, which includes resource discovery, storage, context capturing, location detection, and security primitives. The 3rd tier provides an interface for pervasive applications.

In this paper, we mainly concentrate on trust implications for the location service, which is vital for enabling location-aware security services. For this reason, we describe the location service in more detail. The location service is based on the design discussed in [10]. The location service architecture is illustrated in Figure 2. The service features a layered architecture for collecting sensor information, representing it in a spatial database, and providing an interface for location-aware applications. This layered architecture allows the incorporation of multiple location sensing technologies. The top layer of the service provides a high-level interface for applications and other services to query location information in a sensor-independent manner.

The middle layer consists of a spatial database that provides primitives for defining spatial regions. Spatial regions are represented in a hierarchical format. The representation could be symbolic, e.g., *Building-4/1st floor/Room-100*, (i.e., room 100, 1st floor, Building 4) or coordinate-based, e.g., *Building-4/1/(45,12), (45,40), (65,40), (65,12)* which represents a specific region in the 1st floor in Building-4. The spatial data-

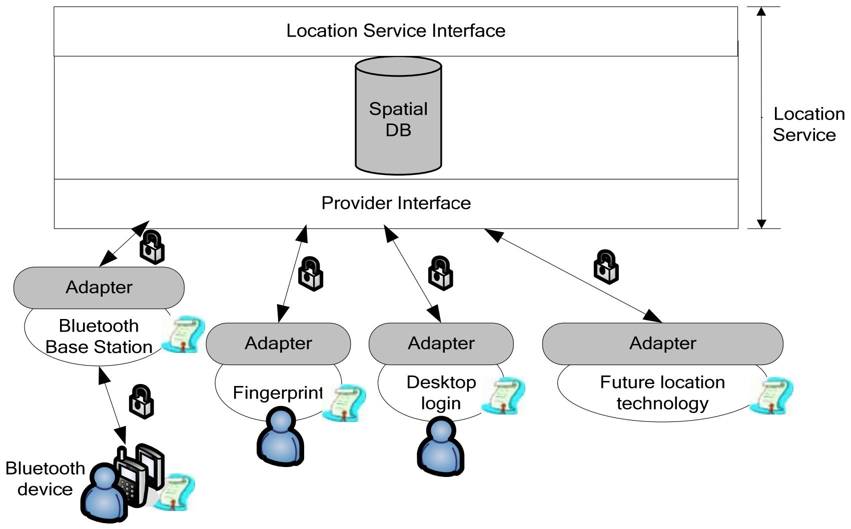

Figure 2: The Location Service Architecture

base is capable of handling spatial queries from applications.

The bottom layer of the service fuses location data from multiple sensors to get an approximation of an entity's location. Location sensors send information to the spatial database through the provider interface. In order to facilitate plug-and-play support for new location technologies, the location service defines an object called a *location adapter*. The location adapter is a Java RMI client wrapper for the specific location technology at hand. The adapter communicates natively to the interface exposed by the location sensor, and acts as a device driver that allows the location sensor to communicate with the location service seamlessly (through the provider interface). Upon installing a new location technology, a calibration process needs to be undertaken. This process involves using the characteristics and specifications of the location sensor to convert the location readings to symbolic and/or coordinate location information that matches the location model expected by the spatial database. Adapters map raw sensor information into a common representation to be stored in the spatial database. Adapters can be programmed to filter certain events or send information to the location service at varying rates (depending on the location sensing technology used). The location service features support to two types of location sensing technologies, as illustrated in Figure 2:

- Static location sensing devices: these devices are connected directly to the provider interface (through a secure channel). Examples include desktop logins, fingerprint devices, cameras capable of face recognition, etc. Such devices are often used for authentication purposes but require physical presence. We exploit this information to get short-lived but relatively accurate readings of a person's location. These devices do not transmit continuous signals when users are in the vicinity.
- Mobile location sensing devices: these technologies often involve the use of a base station to detect tokens or special hardware devices that users carry. For example, RFID tags, Bluetooth or UWB (ultra-wide bandwidth) devices. In this case, the special devices send location information to the base station. The base station then passes the location information to the provider interface of the location service.

As explained in the next section, the two different types of location sensing devices have different trust implications.

4. Threat Model

To provide a trusted location service, we must ensure that the nodes that are providing the location service are the expected nodes. In addition, we must ensure that the nodes are functioning in a secure and reliable manner. Finally, we must ensure that communication among the location service nodes is secure. These trust requirements require policies and mechanisms for identifying and authenticating users of the system as well as location sensing devices. Authentication of devices is two-fold. First, we must verify that the device is the expected device. Then we must verify that the device is functioning as expected.

Figure 3 depicts three types of threats which include: communication compromise via a man-in-the-middle attack; communication compromise via an amplified signal; and device compromise. These threats are common to both statically located and base station oriented location sensing technologies. As the figure illustrates, communication between the workstation and location service or the base station and location service can be compromised. Anyone who has physical access to the network can capture, replace and insert messages during communication between the location sensing devices and the location service. This type of threat is common to all electronic communication and has been widely researched. To counter such an attack, a secure channel must be used to exchange keys used to encrypt communications.

The problem of device compromise and trust has received less attention in current research. Addressing this problem requires mechanisms for determining whether the device or software that it running has changed. In addition, mechanisms that assess the change and determine whether the device will perform its function in a secure and reliable manner are also needed. In essence, the device compromise problem is a device authentication problem. Some means for certifying a device is needed to address the device authentication problem.

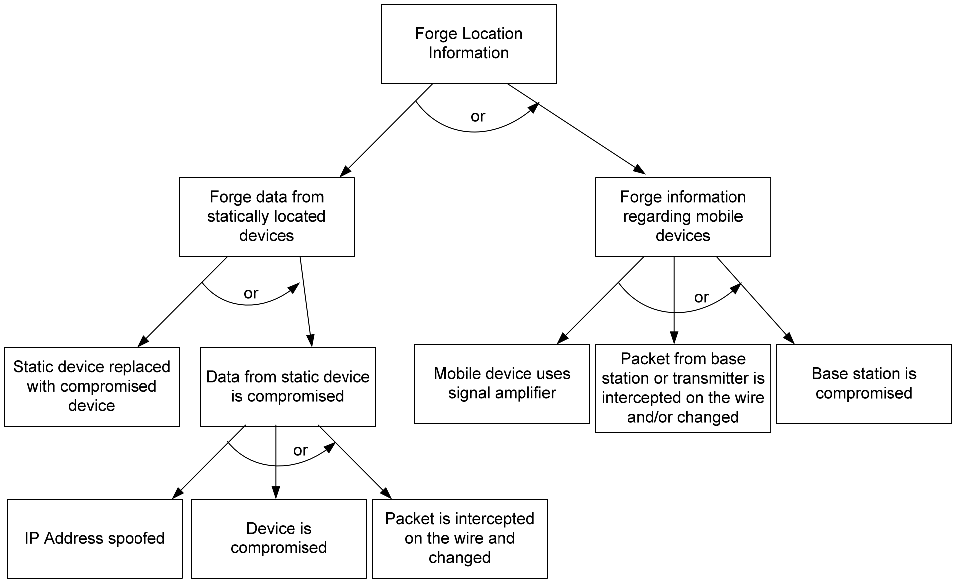
Forge Location Information
or
Forge data from statically located devices
Forge information regarding mobile devices
or
or
Static device replaced with compromised device
Data from static device is compromised
Mobile device uses signal amplifier
Packet from base station or transmitter is intercepted on the wire and/or changed
Base station is compromised
or
IP Address spoofed
Device is compromised
Packet is intercepted on the wire and changed

Figure 3: The Threat Model for the Location Service

5. Incorporating Trust into the Location Service

In order to utilize location and context awareness for enhancing security services, it is necessary to build a trustworthy location service. We briefly discuss the mechanisms that we used to protect against location forgery and to provide trustworthy location readings.

5.1. Setup

In our pervasive computing environment, we define a smart space as a physically-bounded space that features a plethora of devices and sensors that provide context-aware services and applications that aim to provide users with seamless interfaces to carry out their tasks. Each smart space is considered standalone and runs its own set of the core pervasive computing services, which include the location service, the context service, and the security service. Smart spaces are built around a trusted infrastructure. Sensitive services run on the trusted infrastructure, this includes the middle and top layers of the location section and the security services. However, location sensors themselves are either mobile devices assigned or owned by mobile users, or they are devices that have been set up in the room in public places, and as such are subject to tampering or forgery. We propose the use of trusted Certificate Authorities (CAs) in our pervasive computing environment that can provide a multi-level certification for various services and devices. Here we adapt a similar approach to how current operating systems certify their device drivers to assert that the hardware is compatible with the system and is expected to work properly.

5.2. Static Location-Sensing Devices

Static sensing devices have a direct connection to the location service, i.e., they do not involve the use of transmitters and receivers and do not involve wireless communication. However, these devices could be set up in publicly accessible location, which means they can be subject to physical attacks or attempts to replace them with rogue devices. In order to prevent this, we employ several safeguards:

1. The hardware devices are certified by the CA of the space. We use X.509-format certificates for this purpose. The certificates will store additional meta-data including the exact location of the device (since it is static), and an indication of its location-sensing accuracy.
2. Similar to the certified device drivers idea, the adapter that wraps the location sensing device is also certified to ensure: (a) it runs an authentic code, (b) it wraps the proper type of device, and (c) that it would parse the raw sensor data in a correct manner.
3. Periodic challenge-response mechanisms are initiated between the device and its adapter, and between the adapter and the location service. A secure communication channel can be established between the adapter and the location service to

prevent eavesdropping and to detect tampering possible tampering.

4. Some devices contain unique IDs that are difficult to forge (e.g. mobile phones' handset identity number (IMEI) or SIM's identity number (IMSI)), in such cases the adapter can be made to read these values to ensure that it is communicating with the proper hardware device.
5. Some hardware devices feature tamper-resistance capabilities at various levels [11]. For this purpose, we propose a certification system that supports multiple levels of trustworthiness (rather than providing a binary notion of trustworthiness). I.e., if the hardware is tamper resistant and features a unique, difficult-to-forge ID, then it can be assigned a higher level of trustworthiness (we refer to this as confidence value). The confidence value is then stored as part of the meta-data in the certificate.

The certified adapters map the raw sensor information into a common representation to be stored in the spatial database. The location data itself is sent in the form of certified predicates that can be evaluated by the database and stored along with the appropriate confidence level.

5.3. Mobile Location-Sensing Devices

These location-sensing devices consist of a pair of devices, one acts as a receiver (or base station) the other acts as a transmitter (like RFID tags and base stations). These technologies locate entities by sensing proximity or by detecting and calculating time of flight and/or angle of reflection etc. In this case, the location adapter wraps the base station. The base station can be certified as explained in Section 6.2. However, in these types of location-sensing technologies we reduce the confidence level of these location readings due to the fact that they are susceptible to more attacks, including the use of amplifiers to forge location or by sending forged packets to the base station. It is important in this case to have tamper resistant tags (or transmitters) to prevent forgery of these types. Ideally, the tags should be able to perform a challenge response protocol with the base station and to be able to transmit location data using certified predicates to ensure correctness. The confidence value associated with this type of location devices depend on (a) the level of tamper resistance of the transmitters, (b) the ability to participate in a challenge-response using cryptographic functions, and (c) the ability to provide unique, difficult to forge, hardware IDs.

Unfortunately, some widely-used systems for location tracking do not have these properties, for example, passive RFIDs, which can be forged easily. On the other hand, we find that using Bluetooth on PDAs can provide a relatively secure system with high trustworthiness, since it is possible to have the PDAs perform cryptographic challenge-response and it is possible to certify the PDAs and extract unique ID information, which can mitigate the chances of forging the devices.

6. Conclusion

Pervasive computing creates a computing atmosphere within which mobile users can interact with computers and carry out tasks seamlessly. Intrinsic to this notion of computing is context and location awareness in all applications and services. However, sensing location and context is plagued with security risks that result from improper trust assumptions and possible data forgery. In this paper, we touch on the main challenges in building a trusted location service. We develop a general threat model, and provide insights in addressing these threats and incorporating trust in the location service through certification of behavior with different confidence levels.

7. References

[1] *J. Al-Muhtadi, R. Hill, R. Campbell, and D. Mickunas, "Context and Location-Aware Encryption for Pervasive Computing Environments," in Third IEEE International Workshop on Pervasive Computing and Communication Security (PerSec), 2006.*

[2] *P. Castro and e. al., "A probabilistic room location service for wireless networked environments," in UbiComp, 2001.*

[3] *D. Graumann, W. Lara, J. Hightower, and G. Borriello, "Real-world implementation of the Location Stack: The Universal Location Framework," in 5th IEEE Workshop on Mobile Computing Systems & Applications, 2003.*

[4] *C. Jiang and P. Steenkiste, "A hybrid location model with a computable location identifier for ubiquitous computing," in Lecture Notes in Computer Science, 2498, UbiComp, 2002.*

[5] *Grandison and Sloman, "A Survey of Trust in Internet Applications," IEEE Communications Surveys and Tutorial, 2001.*

[6] *R. Sailer, X. Zhang, T. Jaeger, and L. V. Dorn, "Deisng and Implementation of a TCG-based Integrity Measurement Architecture," in 13 USENIX Security Symposium San Diego, CA, 2004.*

[7] *M. Gupta, P. Judge, and M. H. Ammar, "A reputation system for peer-to-peer networks," in NOSSDAV, 2003.*

[8] *J. Al-Muhtadi, "Moheet: A Distributed Middleware for Enabling Smart Spaces," in First National IT Symposium (NITS) Riyadh, Saudi Arabia, 2006.*

[9] *M. Roman, C. K. Hess, R. Cerqueira, R. H. Campbell, and K. Narhstedt, "Gaia: A Middleware Infrastructure to Enable Active Spaces," IEEE Pervasive Computing Magazine, vol. 1, pp. 74-83, October-December 2002.*

[10] *A. Ranganathan, J. Al-Muhtadi, S. Chetan, R. Campbell, and M. D. Mickunas, "MiddleWhere: A Middleware for Location Awareness in Ubiquitous Computing Applications," in 5th International Middleware Conference (Middleware 2004) (accepted), 2004.*

[11] S. Ravi, A. Raghunathan, and S. Chakradhar, "Tamper resistance mechanisms for secure embedded systems," in 17th International Conference on VLSI Design, 2004.