[246] RCP Thomas Hardjono and Jennifer Seberry, Information security in mobile databases, *Sixth Australasian Database Conference*, (R Sacks and J Zobel, (Eds.)), 17(2), conference held at Glenelg, South Australia, February, 1995.

Information Security in Mobile Databases

(Extended Abstract)

Thomas Hardjono[1] *Jennifer Seberry*

Centre for Computer Security Research
University of Wollongong
Wollongong, NSW 2522
Australia
thomas/jennie@cs.uow.edu.au

Abstract

During the last decade the decrease in the size of computing machinery, coupled with the increase in their computing power have lend to the development of the concept of mobile computing. Effects of this new vision can be seen currently in the flourishing numbers of mobile telephones and portable computing units. In the current work we investigate some issues concerning the security of databases and database systems in the mobile computing environment. We present a scheme for end-to-end secure data transfer between one mobile computing element to another based on a recent protocol for key distribution. Secure data transfer is crucial as a foundation to providing other more complex interactions between the various mobile computing elements. Following this we also present an extension to the scheme, applicable for the storage of data in a mobile database, allowing for remote accessibility of the data in a secure manner.

1 Introduction

The field of computing has entered a new era with the birth of a new area of interest referred to loosely as "mobile computing." The concept of mobile computing developed from the realization that as computing machinery decrease in size and increase in computing power users will demand these machinery to be part of their everyday life, accompanying them in the carrying-out of their everyday tasks. Researchers in this new field envisage that mobile computing units, such as today's laptops and palmtops, in the future will be communicating with each other, whilst providing location transparency to the user. This notion of transparency is carried-over from that in distributed computing, in which the user is unaware of the remote physical location of the resources being used by the distributed computing system. In the case of mobile computing, however, several differences emerge. The major difference – and in contrast to distributed computing – is precisely the mobility or the non-fixed positioning of some of the computing elements. This difference in itself presents various new challenges to researchers in the field. These challenges lie not only in the underlying area of mobile communications, but also in the the area of computing strategies and algorithms for non-fixed computing units.

In this work we discuss some issues concerning the security of databases and database systems in the mobile computing environment. In the same way that distributed databases presented challenges to the broader area of distributed computing, mobile database poses some new and particular problems in the wider concept of mobile computing. We recognize that security must be integrated into the design of new solutions to mobile computing. The second aim of this paper is to present a scheme for secure data transfer between one mobile computing element to another. This ability is crucial as a foundation to providing other more complex interactions between the various mobile computing elements. Following this we also present an extension to the scheme, applicable for the storage of data in a mobile database, allowing for remote accessibility of the data in a secure manner.

In the next section we briefly discuss the background and motivations of the current work. This is followed by a general discussion on some security issues in mobile computing, focusing on mobile databases. Section 4 discusses a scheme that provides secure end-to-end data transfer between mobile computing elements. Section 5 applies the scheme to the storage of data in a mobile database, concentrating on the accessibility of the stored date by remote users. Finally, Section 6 closes the paper with some remarks and conclusion.

[1] The author is also at the Department of Computing & Information Systems, University of Western Sydney - Macarthur, Campbelltown, NSW 2560, Australia

2 Background and Motivation

In order to realize the revolutionary concept of mobile computing, a wireless network architecture is required that will support the mobile computing environment in the near and distant future, and that will accommodate further technological developments in the area. Such a basic network architecture has been suggested in [1] (Figure 1). Three of its main components are the *Mobile Units* (MU), which are the users and their portable computing units; the *Mobile Support Stations* (MSS) which are the stations which maintain communications with the mobile units; the *Fixed Host* which are the other nodes connected to the mobile support stations.

A further component is the *Location Servers* (LS) which are employed to maintain data on the position and actions of each mobile unit. Hence, a given location server holds a database containing information about a mobile unit registered as having a "home-base" in the area or zone managed by the location server and data on those "visitors" in the area (cell). These location server databases are used to manage and operate the mobile wireless network [2], and as such, they are not directly accessible to the mobile units. The data stored in these database are used and inter-exchanged only among location servers, and between location servers and mobile support stations. The location servers typically correspond to mobile switching offices, with approximately 60 to 100 mobile support stations being catered for by one location server [3] In the current paper we assume that each mobile unit incorporates a multiuser database system which is accessible by the owner of the mobile unit and by the other (mobile) users from remote sites.

From the point of security, one immediate problem becomes evident in mobile and nomadic computing environment, namely the issue of global user identification and authentication. This issue is further complicated by the fact that mobile units have limited resources which must be used sparingly. Hence, overheads due to security which were tolerable in traditional fixed distributed systems are here required to be minimized or eliminated. With respect to security, one of the major sources of overheads is the security (cryptographic) parameters that need to be exchanged between communicating parties before any interactions commence. This requirement of exchanging parameters is also present in mobile computing systems and is of major concern for researchers who are developing new cryptographic algorithms for mobile computing systems.

Coupled with the problem of communications overheads is the problem of the management of cryptographic parameters, particularly in the mobile unit carried by the user. Bearing in mind that mobile units may often carry sensitive data (eg. corporate sales data) and may connect to remote fixed hosts, the safe and secure storage of the cryptographic parameters that provide access to the data is of paramount importance.

The later of the two related problem can be solved using current technology in the form of smartcards. Increasingly, smartcards are used for user authentication is various non-mobile environments and and are recognized to be a suitable medium for the storage of a limited amount of cryptographic parameters [4, 5].

In this paper we turn to the ILM protocol [6, 7] in the hope of providing a solution for the first of the two problems. The protocol provides a method for two parties to commence communications without previously communicating with each other. The protocol does away with the need of exchanging cryptographic parameters before the parties communicate. The problem of the achieving a common cryptographic parameter (eg. session key) is shifted into that of creating suitable and matching parameters which are stored in the users' smartcards, as will be discussed in Section 4.

In the following section we continue to discuss other security issues within mobile computing, and in particular in mobile database systems. The aim of the discussion is illustrate different problems which were previously unforeseen, and identify those which are immediately relevant in the near future.

3 Mobile Databases: Some Security Issues

The nomadic nature of some mobile computing elements have introduced new problems which were non-existent in the traditional areas of computing. New solutions and solutions derived from traditional computing are in demand in order to transform the dreams of mobile computing into a reality.

In the notable work of [8], four categories for future developments in mobile wireless computing have been proposed, namely *mobility*, *disconnection*, *data access modes* and *scale* of operation. These areas present challenges to the traditional approach to data management and database systems, which until recently have not included effects of mobility into their design and applications. In this paper we would like to propose *security* as being the fifth category which intersects the first four categories. In the following, we briefly discuss some security issues within the framework of two of the areas specified by [8].

3.1 Mobility and Security

The fact that both users and the data that they carry have become a mobile component in computing has in itself introduced a different set of security problems. In the traditional case of fixed (non-mobile) computing physical protection could easily

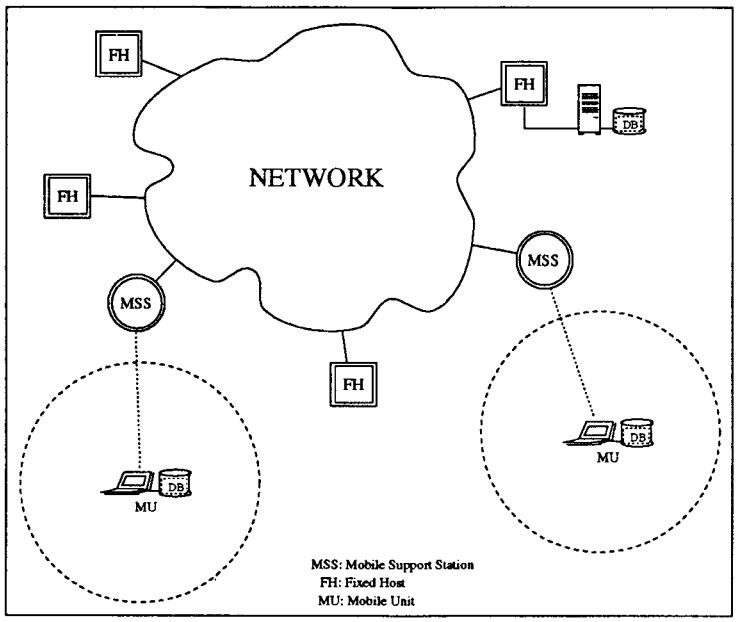

Figure 1: The Mobile Computing Network (after [1])

be afforded by making a computer and database system physically isolated from the other components in the environment. In such a configuration it was possible to make the system self-sufficient, without any need to communicate with the external world.

In mobile computing this form of isolation and self-sufficiency is difficult to achieve due the relatively limited resources available to a mobile unit, thereby necessitating it to communicate with the mobile support station. The mobility of users and the data that they carry introduces security problems from the point of view of the *existence* and *location* of a user (which is deemed to be data in themselves) and the secrecy and authenticity of the data exchanged between users and between a user and a fixed host.

More specifically, a user on a mobile wireless network may choose to have the information concerning his or her existence treated as being confidential. That is, a user may choose to remain anonymous to the majority of other users on the network, with the exception of a select number with whom the user often interacts. This problem of user anonymity is related to a more difficult problem of the *trust* level afforded by each node in the network and the problem of the security of location data concerning a user when the location data is stored or transferred between nodes as the user moves in a nomadic fashion. These nodes must provide some assurance to the user about his or her anonymity, independent of the differing levels of trust that may exist for each node. This requirement is of particular importance in the case of a user that crosses between two zones which are under two nodes respectively, each having a different trust level. Equally important is the secure transfer of data between databases at nodes which hold location data and other information or parameters in the user profile. Here all traffic internal to the network and transparent to the nomadic user must be maintained secure and authentic.

Other problems include the possible inference attacks on the databases and the provision of performance transparency [3] for the nomadic user. Thus, as the user roams across zones, the user must not experience a degradation in the access and latency times. Again, security must be considered in the context of replication, both from the trust level of the mobile support stations and fixed hosts and from the point of view of data leakage. In general, as sensitive data is replicated across several sites, the security risks are also increased due to the multiplication of the points of attack [9].

3.2 Disconnections and Security

Another major issue in mobile computing which arises from mobility and power (battery) restrictions is disconnection. Disconnection of a mobile unit from a mobile support station is necessary for the conservation of power of the mobile unit. A mobile unit can typically find itself running on a temporary form of power supply (eg. spare battery) while its main power source is being recovered (ie. recharged). In this situation differing levels of disconnection may be introduced, ranging from the normal connection to connections using low bandwidth channels.

A crucial aspect of disconnection is the *elective* or *non-elective* nature of a disconnection [8]. The non-elective disconnection refers to the cases

when a mobile unit disconnects due to an unforeseen event, such as system crashes or total communications break-down when moving into certain geographic regions. The elective disconnection refers to the planned and desired disconnection on the part of the owner of the mobile unit who wishes to temporarily limit remote access to his or her mobile unit. This type of disconnection typically covers instances where power availability is degrading, where the owner wishes to employ the maximum resources (eg. CPU) available for a short task, or where the owner simply desires the mobile unit to be put into a "sleep" state. In both types of disconnections, a number of potential security loopholes may be introduced. The transition from one level of disconnection to another may present an opportunity for an attacker to masquerade either the mobile unit or the mobile support station.

One possible solution to this problem is for the mobile unit and the mobile support station to agree upon a *secret* before any transition in the levels of elective disconnections. When at a later time the mobile unit wishes to upgrade the connection level with the mobile support station, both parties can start by exchanging the shared secret in a secure manner. A good candidate for a solution to this problem is zero-knowledge protocols, in which the mobile unit and the mobile support station can convince each other that they hold the shared secret without having to transmit the full secret [10].

Another potential security problem related to disconnections is the leaking of information [11, 12] through the use of inappropriate concurrency control techniques in the database of the mobile unit. Related to this issue is the problem of incomplete transactions caused by elective and non-elective disconnections. A number of scenarios present themselves in this context. Security and integrity problems may occur in the case when hand-offs occur between two mobile support stations as the mobile unit crosses zones (cells). Other security problem may occur when a mobile unit deposits a "timed" transaction at a fixed-node, which begins to execute when certain conditions are met (eg. time of day, availability of raw data) and which transmits the results back to the nomadic mobile unit. These scenarios represent only a few potential security problems among many others in the context of levels of disconnections and transaction management in mobile databases.

There is considerable research work to be done in the area of mobile computing systems, and in particular, in the area of secure mobile databases. As the area of research matures, more will be understood concerning the security of data in mobile computing and in the mobile wireless network. In the next section we will discuss the issue of how to provide a secure channel for data exchange, which will be the foundation upon which other more complex interactions will be based.

4 Secure Data Transfers

End-to-end security and authenticity between components within a mobile computing systems represents an important foundation upon which other operations must be based. The scheme presented in the following is based on the Improved Leighton-Micali (ILM) protocol [6], which establishes a secure end-to-end channel for data transfer [2].

Following the requirements of [6, 7], we assume that tamper-proof VLSI chips are readily available. These are incorporated into the computing equipment that handle secure data transfer, such as the Mobile Units (MU), the Mobile Support Stations (MSS) and the Fixed Hosts (FH) that hold cryptographic information. For simplicity, in the following discussions we will only refer to the MUs and the MSSs, although the intent is clear that other entities in the network that require secure data transfer must have tamper-proof chips (eg. memory). Furthermore, we will employ the terms "user" and "mobile unit" interchangeably, denoting the computing element that is actually mobile or non-fixed. We assume that a publicly known one-way hash function h exists (which may also be replaced with a cryptographically strong pseudo-random function). All data transfer instances are assumed to be protected against the replay attacks through the use of timestamps, nonces or other suitable mechanisms.

In the initialization stage, each user owning a mobile unit will be assigned cryptographic parameters by the *Trusted Authority* in the network. This is achieved through the use of smartcards issued to each user and to the administrator at each MSS. The parameters are not accessible to the users or the administrators, and are used to generate a session key for secret and authentic data transfer. Before using the MU a user (administrator) must insert his or her smartcard into the tamper-proof compartment of each MU (MSS) in order for the parameters to be loaded into the secure memory of the MU (MSS) [4].

4.1 Registration

In the ILM protocol it is assumed that the Trusted Authority holds M secret keys $(X_1, \ldots, X_M)$. Each secret key is chosen uniformly at random by the Trusted Authority and is of length k-bits. For each user i who is enrolled into the system, the Trusted Authority assigns an identity I_i and selects

[2]The original Leighton-Micali protocol of [6] had an inherent flaw which in our context allowed an attacker to compromise data when it is in transit. This flaw has subsequently been solved and the protocol improved by the work in [7].

M random integers (α_1, ..., α_M) from the interval $[1, L]$, where L is an integer.

The Trusted Authority then employs h to compute $Y_m = h^{\alpha_m}(X_m)$ for all $m = 1, \ldots, M$, where $h^s(X)$ indicates applying consecutively the function h on an input X for s times. That is,

$$h^s(X) = \overbrace{h(\cdots h(h(}^{s \text{ times}} X)) \cdots).$$

Here (α_1, ..., α_M) becomes the public key of user i. The corresponding secret key (Y_1, ..., Y_M) is then placed by the Trusted Authority into the tamper-proof chip of the user's smartcard. The Trusted Authority also injects a secret *database key* D_i into the chip (its use will be evident in the following sections). The smartcard is only delivered to the user after the secret key has been inserted, and hence no person has access to the secret keys or the other secret parameters within that user's tamper-proof chip.

4.2 Secure Channels

When a user i at a mobile unit (MU) desires to exchange data with another user j, such as for the purpose of retrieving data, the calling user i must first establish a secure channel with the controlling MSS, and then request the MSS to further create a secure channel with the destination MU of user j.

In the following we will apply the ILM protocol to establish a session key between user i and the MSS *without* them previously communicating. This session key will be denoted as $S_{i,mss}$. The aim of the scheme is for user i and the MSS to arrive independently at keys $K_{i,mss}$ and $K_{mss,i}$, where $K_{i,mss} = K_{mss,i}$. The scheme is described as follows [7], employing I_i as the identity of user i and I_{mss} as the identity of the MSS.

1. The MU of user i must obtain the public key (β_1, ..., β_M) of the MSS. This public key can be resident in a list within each MU or it can be read from a publicly readable file at one of the MSSs.

2. After user i inserts his or her smartcard into his or her MU, the MU must provide the smartcard with the public key of the MSS. The tamper-proof chip within the smartcard of user i then computes the common key $K_{i,mss}$ as:

$$K_{i,mss} = \begin{cases} h(h^{\delta_1}(X_1)|| \cdots \\ h(h^{\delta_1}(X_1)|| \cdots \end{cases}$$

$$\begin{matrix} \cdots ||h^{\delta_M}(X_M)||I_i||I_{mss}), & I_i \le I_{mss}, \\ \cdots ||h^{\delta_M}(X_M)||I_{mss}||I_i), & I_i > I_{mss}. \end{matrix} \quad (1)$$

where $\delta_m = \max(\alpha_m, \beta_m)$, $m = 1, \ldots, M$, and $||$ denotes concatenation.

Note that the tamper-proof chip can easily compute $h^{\delta_m}(X_m) = h^{|\delta_m - \alpha_m|}(Y_m)$ $(m = 1, \ldots, M)$ and thus $K_{i,mss}$ because it has available the values $Y_m = h^{\alpha_m}(X_m)$ residing in its internal memory. The tamper-proof chip of the MSS then performs symmetric procedures, obtaining the key $K_{mss,i}$, which is in fact identical to $K_{i,mss}$ [7].

In order to aid our subsequent discussions we will simplify Equation (1) into

$$K_{i,mss} = \begin{cases} h(X||I_i||I_{mss}), & I_i \le I_{mss}, \\ h(X||I_{mss}||I_i), & I_i > I_{mss}. \end{cases} \quad (2)$$

As before, the k-bit value X is chosen randomly by the Trusted Authority where k should be sufficiently large, say $k \ge 100$, in order to prevent it from an exhaustive search attack [7]. The value X is kept secret by the Trusted Authority, and during the enrollment of users the Trusted Authority also injects a copy of X into the chip of the smartcard belonging to the user and into that belonging to the administrator at each MSS and Fixed Host. Hence, in fact, the value X is common to all parties in the system.

4.3 Session Keys

After the user i and the MSS have established a secure channel via keys $K_{i,mss} = K_{mss,i}$, the MSS proceeds to establish another secure channel with the MU of the destination user j using the previous method. Here, the secure channel between the MSS and user j will be via keys $K_{mss,j} = K_{j,mss}$

At this point two general options are available for the selection of the session keys. First, the MSS can be given the authority to generate two session keys, namely key $S_{i,mss}$ to be shared between user i and the MSS, and key $S_{mss,j}$ to be shared between the MSS and user j. Alternatively, each of the users can choose randomly and uniformly a key that they will share with the MSS.

The first option, where the MSS has the authority to choose randomly and uniformly the keys $S_{i,mss}$ and $S_{mss,j}$, is advantageous from the point of view of resource consumption at the MUs of both users. However, the second option can remain attractive if some part of the computation could be performed off-line, with the MU carrying the partly completed result of the computation. Other methods such as real-time off-loading to a remote computation server can also be employed.

Assuming the first option is adopted, two approaches are available, namely for the MSS to choose two distinct keys $S_{i,mss}$ and $S_{mss,j}$, or for the MSS to select one key which it will share with both users (ie. $S_{i,mss} = S_{mss,j}$). From the point of view of security, the usage of a single session key across a number of communications segments may increase the risk from compromises. In this case, the all parties within end-to-end communications

must have the same level of trust, something which is difficult to assure since the mobile unit may be prone to physical attacks as compared to the MSS.

The use of a distinct session key for each communications segment may also be beneficial from the point of view of data transfer between MSSs over land-based transmission medium, such as fibre optics. More specifically, two MSSs that exchange a large amount of traffic may already have a secure channel established between them via a given session key. They may then prefer to add (multiplexing) new communications instances under the existing session key with the other traffic, rather than to create a separate secure channel.

In addition, this approach has an advantage in terms of the control by the MSS over a user's MU when the user is not active. That is, when a user sets his or her MU to "sleep" mode in which the MU can be "awaken" remotely by another MU, the MSS can be assigned authority by the user as to which other MUs are allowed to "wake-up" the user's MU. This delegation of authority from the MU to the MSS is a possible solution due to the fact that any calling MU must first deal with the controlling MSS before interacting any other MU. This is of particular importance in cases where the user's MU carries a database (eg. sales data) which can be interrogated remotely by another user (eg. a company executive).

5 Mobile Storage: Security

In this section we briefly discuss a method of storage for data in the database of a mobile unit (MU), providing for accessibility to other MUs and fixed hosts (FH). We focus particularly on small databases which are located in MUs with small computing power, such as laptop and palmtop computers. Hence, even though such databases may offer access in the same manner as ordinary DBMSs, we assume that the number of users that access the database is limited and that as far as possible all CPU-consuming operations should be performed by the calling party. In the future, one cannot rule-out the possibility of having larger DBMSs running on MUs with a higher level of power. One example would be the placement of a permanent computing unit and DBMS in a user's vehicle (eg. car) with a larger power supply, which is updated periodically by the user's smaller computing unit (eg. laptop or palmtop). Such a database should then be accessible concurrently by a number of remote users in the traditional sense.

One important requirement in all databases is that the data stored in it should be available to only the specified set of authorized users. In the situation of a database within a mobile unit, the user that carries the unit has practically complete control over the accessibility of the data. Data in a mobile unit should remain in an encrypted state when not in use, since the unit may be subject to theft and other mishaps. Due to the need of sharing of data with other users, data in the mobile database must also be accessible by remote users, both when the owner of the unit is running the database (ie. concurrent access) and when the unit is in "sleep" mode. The limited power supply of a mobile unit commands that only a small number of remote users should be able to access the database. They should also be geared towards retrieving data, rather than requesting operations to be performed on the portable computing unit. Furthermore, the internal battery of the mobile unit should provide a signal when its battery power is low, leading to the rejection of further access by remote users whilst maintaining the correctness and integrity of the database.

One technique that can be used to protect data from illegal theft and to provide secure access to remote users is by way of encipherment. More specifically, each data item (eg. record) R_i is enciphered under a key K_{R_i}. Several copies of this key are created corresponding to the number of remote users that have been registered to access the database. For a given remote user j, the key K_{R_i} is stored enciphered as $\{K_{R_i}\}_{KE_{i,j}}$ under the key

$$KE_{i,j} = \begin{cases} h(X||D_i||I_i||I_j), & I_i \leq I_j, \\ h(X||D_i||I_j||I_i), & I_i > I_j. \end{cases} \quad (3)$$

where D_i is the secret database key uniformly chosen by the Trusted Authority and injected into the smartcards of users i and j at registration time. Here D_i is common to all remote users that have registered to access the mobile database of user i.

When an authentic remote user j requests access to record R_i stored in the mobile database of user i, the pair

$$\{K_{R_i}\}_{KE_{i,j}}, \{R_i\}_{K_{R_i}}$$

is sent over the air by the mobile unit of user i. (Here the symbol "$\{\}_K$" denotes encipherment using key K). Upon receiving the pair, the tamper-proof chip of the remote user j's smartcard – lodged inside his or her mobile unit – must generate $KE_{i,j}$ and decipher $\{K_{R_i}\}_{KE_{i,j}}$ to obtain K_{R_i}. It then uses K_{R_i} to decipher $\{R_i\}_{K_{R_i}}$ in order to recover the plaintext R_i. After this, the chip discards K_{R_i}, thereby making the value always inaccessible directly by mobile units other than that belonging to user i.

To reduce traffic on the air from the mobile unit of user i carrying the database, an alternative approach is to store the entries $\{K_{R_i}\}_{KE_{i,j}}$ for each registered user j at the home site of user i. Since this information is only decipherable by the

tamper-proof chip within the mobile units of users i and j respectively, it can be safely replicated to other sites depending on the location strategy used by the network. In this case the entry can then be delivered over the air by one of the MSSs, thereby reducing the unnecessary power consumption at the mobile unit of user i.

6 Remarks and Conclusion

There is still a long way for research to proceed before mobile computing will become a daily reality in society. Although considerable effort is being addressed towards research in mobile computing, much of it is concentrating on the performance and availability of mobile computing, with comparatively little attention being given to the security issues in such an environment.

In this paper we have proposed *security* – in addition to the existing categories developed in [8] – to be a major category for future developments in mobile computing. We have discussed briefly the issues of security in the context of mobility and disconnection, presenting a number of potential problems in the security of a mobile computing environment.

This was followed by the presentation of a scheme for secure data transfers between elements within the mobile computing environment, such as mobile units and a mobile support stations. The scheme is based on the Improved Leighton-Micali (ILM) protocol [6, 7], providing a means for the establishment of a secure channel without the parties needing to communicate previously. The basic scheme was then applied for the storage of data in a mobile database, which allowed for data to be accessible to registered remote users at other mobile units or other fixed hosts.

The mobile computing environment and its security presents a new ground for further research, with some problems which are non-existent in the traditional non-mobile computing environment. Future work on the security of mobile computing and mobile databases will concentrate on solving problems pertaining to the security of information within the three sub-areas of the mobile environment:

- The security of information residing in the mobile units and the correctness and integrity of databases in these mobile units.
- The security of information as it travels "over the air" between mobile units and mobile support stations. An important consideration in this area is the power consumption of the algorithms and schemes that implement this secure data transfer. New data storage schemes and data organization techniques will be required to facilitate rapid searching and transfer of data to and from mobile databases.
- The security of information within the mobile wireless network. This includes the security of databases holding control data used for the operations and management of the mobile wireless network.

These three sub-areas of research will be crucial if mobile computing is to be a reality in the future.

Acknowledgments

We thank our colleague Dr Yuliang Zheng who first brought the Leighton-Micali protocol to our attention. This work has been supported in part by the Australian Research Council (ARC) under the reference number A49232172 and the University of Wollongong *Computer Security Technical and Social Issues* research program. The second author has received additional funding from the ARC under the reference numbers A49130102 and A49131885. We also thank the referees for their insights and suggestions.

References

[1] T. Imielinski and B. R. Badrinath, "Mobile wireless computing: Solutions and challenges in data management," Technical Report DCS -TR-296/WINLAB-TR-49, Department of Computer Science, Rutgers University, NJ, 1992.

[2] W. van den Broek and E. Buitenwerf, "Distributed databases for third generation mobile systems," in *Proceedings of the International Council for Computer Communication Intelligent Networks Conference* (P. W. Bayliss, ed.), (Tampa, Florida), pp. 333–347, IOS Press, 1992.

[3] B. R. Badrinath and T. Imielinski, "Replication and mobility," in *Proceedings of the 2nd IEEE Workshop on Management of Replicated Data*, pp. 9–12, IEEE, November 1992.

[4] D. Chaum and I. Schaümuller-Bichl, eds., *Smart Card 2000: The Future of IC Cards.* Berlin: Springer-Verlag, 1987.

[5] M. Abadi, M. Burrows, C. Kaufman, and B. Lampson, "Authentication and delegation with smart-cards," Technical Report 67, Digital Systems Research Center, October 1990.

[6] T. Leighton and S. Micali, "New approaches to secret-key exchange," in *Advances in Cryptology – Proceedings of Crypto '93,* Lecture Notes in Computer Science, Springer-Verlag, 1993. (to appear).

[7] Y. Zheng, "Amending Leighton and Micali's key distribution protocol," Technical Report Preprint 93-17, Centre for Computer Security Research, Computer Science Department, University of Wollongong, September 1993.

[8] T. Imielinski and B. R. Badrinath, "Data management for mobile computing," *SIGMOD RECORD*, vol. 22, no. 1, pp. 34–39, 1993.

[9] L. Gong, "Increasing availability and security of an authentication service," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 5, pp. 657–662, 1993.

[10] J. Seberry and J. Pieprzyk, *Cryptography: An Introduction to Computer Security.* Sydney: Prentice Hall, 1989.

[11] T. F. Keefe, W. T. Tsai, and J. Srivastava, "Multilevel secure database concurrency control," Technical Report TR 89-45, University of Minnesota, July 1989.

[12] S. Jajodia and B. Kogan, "Transaction processing in multilevel-secure databases using replicated architecture," in *Proceedings of the 1990 IEEE Symposium on Security and Privacy*, (Oakland, CA), pp. 360–368, IEEE Computer Society, 1990.