Security in Mobile Systems

Vineeth Subramanyam and Anupam Joshi
Department of Computer Engineering & Computer Science
University of Missouri, Columbia, MO 65211
e-mail: c697168@showme.missouri.edu, joshi@cecs.missouri.edu

1. Introduction

Mobile computing has become popular over the last few years. Users need to have continuous access to information even when they are mobile - e.g., a doctor may need to constantly monitor a patient's health or a stock broker may need periodic information about the stock market, etc. Communication in such cases is typically over wireless links and it becomes *critical* to ensure secure message exchange. The traditional goals of secure computing have been to achieve *confidentiality, integrity, availability, legitimacy* and *accountability*. Messages exchanged between two hosts are usually coded by using symmetric or asymmetric ciphers, which makes it difficult for them to be seen by an outsider.

This paper discusses issues concerned with mobility in a network and extensions of existing security schemes. The following section gives a description of some of these problems. Section 3 is a summary of related work in these areas and potential drawbacks. Section 4 details our proposed solution scheme. Section 5 provides concluding thoughts.

2. Characteristics of Mobility

In static environments, every host has an IP address that identifies it for transport-layer connections and is linked to its geographic location. In a mobile environment, however, this is no longer true because the host may change its point of attachment to the network at any time. The Mobile IP [6] protocol, an extension of IP, resolves this dilemma by using IP tunneling (IP over IP or IP over ATM) mechanisms. Baghwat, Perkins and Tripathi [7] discuss these issues in greater detail.

In a mobile environment, the MSS-MH model proposed by Badarinath & Imilienski states that a mobile client typically attaches itself to a fixed network through a Mobile Support Station (MSS), which acts as an intermediary for any connections with the server. Each MSS constantly sends beacon-signals to be picked up by mobile hosts in its domain. The mobile hosts then compare the strength of all such signals to deduce which domain they are attached to — the domain that sends the strongest beacon-signals. When a mobile host observes a substantial drop in the strength of the beacon-signal from its current MSS, it informs the MSS that it is possibly moving into a foreign domain. This is done with a handoff-takeover exchange with respect to the current and foreign MSS'. From the perspective of designing a scheme for secure communication in such environments, the challenges can be identified as three-fold:

- *Mobility of users and hosts.* In a mobile environment, users and/or hosts may move from one place to another attempting to connect to different servers as they move along. Most schemes conveniently assume that a user can authenticate himself/herself with a password on a his/her favorite hosts. However, when the user and/or host moves to another network, such authentication schemes do not work any more. To support user and/or host mobility, therefore, an authentication scheme should global in nature for the reason that the next destination of a mobile host is unknown at any given time. Therefore, entities in the network need to be able to communicate with one another to come up with a feasible authentication scheme for the mobile host irrespective of where it moves.
- *Resource-poor clients.* Mobile clients typically range from reasonably powerful laptops to small 3Com Palmpilots that run on limited battery power. Such machines cannot be expected to do complex computations (such as those involved in key generation) very often. As an example, it is undesirable for a mobile client to spend a substantial fraction of its lifetime in a network on computations used in authenticating itself. The server, on the other hand, is usually a fixed host and is much more powerful. This introduces a natural asymmetry between the capability of the server and the client. Further, mobile hosts are inherently "insecure" because they may be lost, damaged or even stolen in transit. A static member on

0-8186-9218-9/98 $10.00 © 1998 IEEE

the other hand, is free from such dangers sitting in a room under lock and key.

- *Presence of a wireless network.* It is equally important to recognize that a mobile network is typically made of wireless links. In a wireless network, the communication medium is RF-based and broadcast in nature. The obvious problems of such a medium are low bandwidth, high error rate and intermittent connections. Further, the presence of wireless links makes it feasible for intruders to eavesdrop and "listen" to the traffic across the link — this is similar to tapping a cordless phone by listening at the same frequency. The problem of eavesdropping is present even in wired networks, but it is possible to guard a physical wire against attacks by providing a shield around it or keeping watch or by other means. This is impossible to do in a wireless medium. Moreover, a mobile computer does not need a physical connection point. The damage due to such vulnerabilities may be severe (e.g., military applications). Therefore, it is critical to ensure that any communication be kept at a minimum by reducing the size and frequency of messages exchanged. In Mobile IP based systems, communication is typically initiated with agent advertisement messages issued by servers, followed by agent solicitation messages issued by clients. Many approaches consider the scenario of rogue clients attaching themselves to legitimate servers. The other possibility, which is equally dangerous, is the case of a rouge server sending fake beacon-signals to capture a mobile host. If the authentication scheme is not designed carefully, the rogue server could cause packets to be re-routed or even cause denial of service after executing a mandatory authentication protocol with the victim. The execution of the authentication protocol provides an illusion that the server may be legitimate in trying to authenticate the client.

3. Related Work

Several approaches have been presented in literature, most of which assume the existence of malicious host(s) in the network that are capable of presenting themselves as other hosts to gain unauthorized access or to restrict access to other hosts. Most of the approaches in literature have attempted to achieve authentication as a preliminary step towards eliminating security-related problems. In a network of wireless nature, it becomes necessary to have a trusted entity associated with a mobile host that can vouch for its identity. For a large-sized network, it is reasonable to have trusted entities (often called "agents") in *each domain* of the network. Protocols often make use of home agents and foreign agents corresponding to the home and foreign networks respectively. The ideas suggested in [1] and [17] represent some of these schemes. The protocol mentioned below is an abstraction of such schemes. The protocol provides authentication and anonymity. Anonymity has been defined as protection of relevant information from unintended parties, and not unrestrained anonymity as such.

The protocol runs as follows:

When the mobile host M shows up in a remote network R, it first calculates the key K_{MR} that is used for communication between M and R thereafter, unless changed later. M sends a message to R encrypted with this key. R does not know the key K_{MR}. Therefore, it sends the credentials to H. The home network H now evaluates the key K_{MR} and sends all relevant information to R. R can now assert the true identity of M with messages 4 and 5. The protocol also adds in nonces and time-stamps to provide a measure of the "freshness" for the messages being exchanged.

Entity–Entity	*Content of the message*
M–R	$\{M, R\}_{K\text{-}H}$, H, $\{message\}_{K\text{-}MR}$
R–H	$\{above\ message\}_{K\text{-}HR}$, R
H–R	$\{R, H, M, K_{MR}\}_{K\text{-}HR}$
R–M	$\{challenge\}_{K\text{-}MR}$
M–R	$\{response\}_{K\text{-}MR}$

This protocol provides anonymity because an onlooker can only deduce that an entity from H visited R but cannot determine the exact identity of M. Further, R and M can recall this authentication process and/or decide on a new key for communication between them, should the need arise. However, the following questions arise about this kind of protocol:

- The authentication procedure involves *five* steps that are executed for *every* mobile host. This is clearly undesirable keeping with the need for reduced message exchange. Also, the home agent is involved in two steps. This introduces side effects like overhead and slow communication as a result of the lengthy procedure.

- When the mobile host M arrives in the remote network R, the burden of generating the session key K_{MR} lies with the mobile host. The procedure of generating a unique session key is typically computationally intense. Moreover, this procedure needs to be repeated by the host *every* time it changes its point of attachment. Recalling our earlier observation, the mobile client is resource-poor and cannot be expected to run through these complex computations.
- A rouge server could send fake beacon signals to capture a mobile client. Indeed, in the above protocol, 'R' could be replaced with any rogue server and the authentication steps would still be followed to give the illusion of authenticating the client to the server. This is related to the problem of 'black-hole' routers and 'person-in-the-middle' attacks described by Cheung, Levitt and others at the University of California-Davis (more information is available at http://seclab.cs.ucdavis.edu/arpa/router_dns/welcome.html)

Another scheme proposed in the same paper states that every mobile host keeps a traveling alias, M_t, which could be revealed to foreign networks. Anonymity is preserved since the mapping between the alias and the actual name, is known only to the home network. This variation also suffers from most of the problems mentioned above. Most schemes use triangular-authentication (host to foreign agent, foreign agent to home agent and host to home agent) like the scheme mentioned above.

Several security schemes exist for a wired infrastructure. Some of them deal with security in low-layer communication like *IPSec* discussed in [8] that provides authentication, integrity and confidentiality for IP layer communication. Non-repudiation can also be provided under certain conditions. A mobile environment, however, needs special attention due to the aforementioned characteristics. The use of *IPSec* in a mobile environment [9] calls for strong authentication schemes and security association for message exchange as well as registration with home/foreign agents.

The CHARON project at the University of California-Berkeley [18] extends the Kerberos scheme to support mobility. This scheme uses a proxy-based architecture and attempts to make the client a part of the authentication procedure. The scheme still uses a Ticket Granting Service, which serves as a centralized authority to generate session keys between two entities that wish to communicate. We propose a scheme that removes the burden of authentication-related computations from the client with the use of a distributed architecture that requires that entities in the network talk to one another. Further, the objective of the CHARON project is to extend the Kerberos , a mechanism that attempts to authenticate any client to any arbitrary service in the network. Our scheme is more light-weight in attempting to achieve mutual authentication between a client and a server.

4. Proposed Solution Scheme

Noting the aforementioned asymmetry in the computational powers of the entities involved, we propose an authentication scheme where the burden of computing the session-keys lies with the *server* (MSS) as opposed to the resource-poor mobile client. Each MSS knows about its neighbors, all of which are assumed to be legitimate. Mobile clients compare the strengths all beacon-signals received to determine their current domain. When the beacon-signals from its current MSS start to drop in strength, it sends a message to its current MSS to initiate the process of computing keys. The MSS communicates with each neighboring MSS to (1) generate a potential session-key to be used by the mobile client if it moves there, and (2) generate a challenge that is essentially a message encoded with the session-key. The MSS then sends a *single* message to the mobile client with *all* the generated session-keys and their respective challenges.

Meanwhile, each MSS maintains a table of all generated session-keys and their associated challenge messages that it has received from its peer MSS. When the mobile host moves to a remote network, it picks the appropriate key from the list and decrypts the associated challenge, which it then sends to the foreign MSS. The MSS looks up its table and picks the associated session-key. It is now possible for the mobile client and the MSS to carry on further communication. An extension of this scheme is that the MSS' be given information (or means to estimate) about the direction of movement of the mobile. In this case, session-keys need to be generated only with a limited number of neighboring MSS'. Similar schemes have been proposed by Helal et. al. for Mobile database transaction models (Kangaroo transactions).

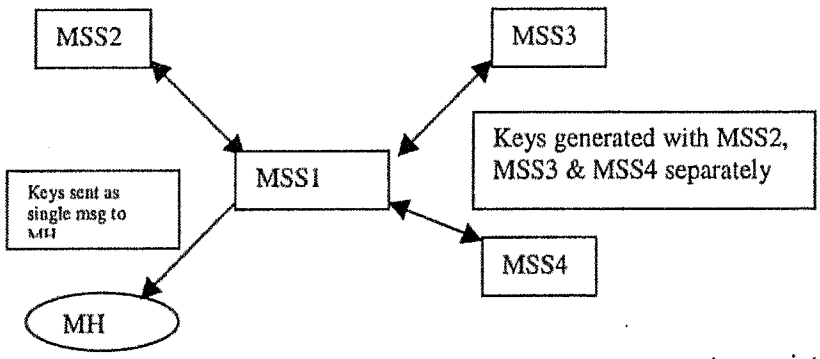

Figure 1. Mobile Host gets all keys as a single message from the MSS

Using a Client/Proxy/Server model (over the traditional Client/Server model) for *mobile* applications seems reasonable. The proxy essentially aims at making the mobile client's movement and other special constraints *transparent* to the server. A proxy based system has been used to support web browsing from mobile clients in [14, 15]. A proxy has also been used in the GloMop model described in [12] to perform 'distillation' of documents and in the Mowgli model described in [13] as one of two intermediaries

We observe that such a proxy-based scheme offers a number of advantages:

- All keys and challenges are exchanged as a single message with the mobile host and the home agent is seldom involved. This reduces the vulnerability due to eavesdropping, keeping with the goal of reduced message exchange. The overhead of having to communicate with the home agent each time is avoided. The home agent is involved only the mobile host is switched off and switched back on in a network away from more than one hop from its current MSS.
- The burden of generating session-keys is given to the MSS which works well for the asymmetry in the network since the computational power of the MSS is typically greater than that of the mobile host.
- The knowledge of the session-key alone is sufficient to authenticate the mobile host, since each mobile host is given a separate set of keys and challenges. This eliminates the need for complicated authentication schemes that can take more than three steps, introducing the overhead of traveling all the way back to the home-agent that further adds to problems of eavesdropping in each step.
- This scheme also solves the problem of a rouge server. A mobile client initially connects to its home network, which is assumed to be trustworthy. Thereafter, when the client wishes to move to a different network, it waits for a bunch of keys from its MSS. Meanwhile, the MSS contacts its neighbors and generates potential session-keys. It later sends a list to the client with keys for communication for all its trustworthy neighbors. When the client moves, it maps the new incoming beacon signal to a server name and checks if it occurs in its list of trustworthy servers. If there is a conflict, the client simply ignores the signal, assuming it to be a rouge server. This procedure is repeated when the client moves to a foreign network. Further trust follows by inductive reasoning — i.e., MSS-1 tells the client that MSS-2 can be trusted. MSS-2 can now vouch for MSS-3's trust and so on.
- This scheme does not include a centralized authority like a Ticket Granting Service unlike the Kerberos mechanism. Instead, the authentication scheme is of a distributed and global nature, where entities in the network talk to one another. Such a scheme can be extended with the use of 'monitors' in the network to handle attacks after they occur.

The only potential drawback of this scheme is the overhead of sending a large number of session-keys to the mobile host(s). However, this overhead can

be significantly reduced if the MSS is able to estimate the direction of movement of the mobile host. Alternatively, the mobile host could maintain a history of its own motion through the network to compute an estimate of its next destination. This estimate can then be given to the MSS.

This scheme was simulated with a server and a client program that were launched on different machines. The simulation also included rogue servers. Sample runs have shown that rogue servers in the network were indeed detected accurately. Further, the programs were designed such that a client would inform its previous MSS of the presence of a rogue server, which in turn would inform all its neighbors and so on. Some data was also collected for noting the time taken to transfer the set of keys and challenges over a wireless network. This was a 2.4 GHz FHSS based Proxim Range LAN System. It was observed that with even upto 35 neighbors, the time taken to transfer the single message to the mobile client is less than ½ second!

5. Concluding thoughts

This paper addresses some security issues in mobile computing. Existing approaches and their potential drawbacks have been discussed. A different scheme has been proposed that transfers the burden of generating session-keys onto the servers, recognizing the resource constraints of the client. As a potential improvement to this scheme, MSS' make a note of mobile clients that fail the challenge and increase their 'distrust level' for those clients. With active networks, it is possible for the MSS' to talk to each other and boycott any communication with a client who's 'distrust level' is higher than a pre-defined threshold. *Intrusion detection* techniques may also be applied to determine the occurrence of an attack. The COAST group at Purdue has proposed a scheme that uses pattern matching in [4]. This architecture essentially looks for faults and is a simplistic inter-agent scheme. We propose to extend this to a more complex architecture using learning and adaptation for agents that have been developed in the context of multiple information sources [11]. With such agents, it may be possible to take relevant actions like blocking data packets coming from rogue servers by communicating with routers in the network.

6. References

[1] N. Asokan, *Security Issues in Mobile Computing*, Department of Computer Science, University of Waterloo, Aug 1995.

[2] Andrew Myles, David B Johnson & Charles Perkins, *A Mobile Host Protocol Supporting Route Optimization and Authentication*, IEEE Journal on Selected Areas in Communications, 13(5); 839-849, June 1995.

[3] David B Johnson, *Routing in Ad Hoc Networks of Mobile Hosts*, Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, Dec 1994.

[4] Sandeep Kumar & Eugene H. Spafford, *A Pattern Matching Model for Misuse Intrusion Detection*, Proceedings of the National Computer Security Conference, pp. 11-21, Baltimore, MD; 1994.

[5] Andrew S. Tenanbaum, *Computer Networks*, 3-rd edition, Prentice Hall, 1996.

[6] Charles Perkins, *Mobile Networking through Mobile IP*, Sun Microsystems, Tutorial.

[7] P. Bhagwat, C. Perkins & S.K. Tripathi, *Network Layer Mobility: An Architecture and Survey*, IEEE Personal Comm., Vol. 3, No. 3, June 1996, pp. 54-64.

[8] R.Atkinson, *Security Architecture for the Internet Protocol*, RFC 1825, August 1995.

[9] John K. Zao & Matt Condell, *Use of IPSec in Mobile IP*, Mobile IP Internet Draft, November 1997.

[10] A. Fasbender, D. Kesodogan & O. Kubitz, *Analysis of Security and Privacy in Mobile IP*, Aachen Univeristy of Technology, Germany.

[11] A. Joshi, N. Ramakrishnan & E Houstis, *MultiAgent Systems to support Network Scientific Computing*, to appear in IEEE Internet Computing, Vol. 3, 1998.

[12] R. H. Katz, E. A. Brewer, E. Amir, H. Balakrishnan, A. Fox, S. Gribble, T. Hodes, D. Jiang, G.T. Nguyen, V. Padmanabhan & M. Stemm, *The Bay Area Research Wireless Access*

Network (BARWAN), Proceedings Springs COMPCON Conference 1996.

[13] M. Liljeberg, H. Helin, M. Kojo & K. Raatikainen, *Enhanced Services for World Wide Web in Mobile WAN Environment*, Report C-1996-28 April 1996.

[14] Harini Bharadvaj, Anupam Joshi & Sansanee Auephanwiriyakul, *An Active Transcoding Proxy to Support Mobile Web Access*, Technical Report #9802, Dept. of Computer Engg & Computer Science, University of Missouri, Columbia, 1998.

[15] Anupam Joshi, S. Weerawarana & E.N. Houstis, *On Disconnected Browsing of Distributed Information*, in Proc. IEEE RIDE '97 , pp. 101-107, 1997.

[16] V. Bharghavan & C.V. Ramamoorthy, *Security Issues in Mobile Communications*, International Symposium on Autonomous Decentralized Systems' 95.

[17] Vaduvur Bharghavan, *A Protocol for Authentication, Data and Location Privacy, and Accounting in Mobile Communications*, Center for Reliable and High Performance Computing, University of Illinois at Urbana-Champaign.

[18] Armando Fox & Steven D. Gribble, *Security on the Move: Indirect Authentication using Kerberos*, University of California at Berkeley.