Authentication Protocols using Mobile Agents

S.M.Chaware
Asst. Prof., Information Technology Department, DJSCOE, Vile-Parle (W), Mumbai – 400 056
Email: smchaware@gmail.com

Abstract- **Wired or wireless network are vulnerable to various attacks since they use shared medium. The overall security measures will not sufficient for ad-on security. The conventional authentication methods will not be sufficient, as they have many pitfalls. In this paper, I discussed the various authentication protocols used on a network and proposed authentication model using Mobile Agent. Using this system, the user's authentication will becomes so simple, easy and secure.**

Keywords- **Mobile Agent, MA; Security; Mobile Agent Platform; Authentication Server; Authentication Protocols**

I INTRODUCTION

Authentication is the process of proving identity of a station to another station or node. In the open system authentication, all stations are authenticated without any checking. There are many methods for authentication. In one of the method, a node A sends an authentication management frame (AMF) that contains the identity of himself, to other node B. Node B replies with a frame that indicates recognition, addressed to A. In the closed network architecture such as WLAN, the stations must know the SSID of the AP in order to connect to the AP. The shared key authentication uses a standard challenge and response along with a shared secret key. Most password-based protocols in use today rely on a hash of the password with a random challenge. The server issues a challenge, the client hashes that challenge with the password and forwards a response to the server, and the server validates that response against the user's password retrieved from its database. Legacy password protocols are easily subjected to eavesdropping and man-in-the-middle attacks. An eavesdropping attacker can easily mount a dictionary attack against such password protocols. A man-in-the-middle attacker can pass through the entire authentication, and then hijack the connection and act as the user [1].

II INTRODUCTION TO MOBILE AGENTS

Mobile Agents (MA) is an effective paradigm for distributed applications and is particularly attractive in a dynamic network environment involving partially connected computing elements [2]. MA is defined as a software component which is either a thread or a code carrying its execution state to perform the network function or an application [3]. MA can act as a middleware and perform network and other application related functions based on underlying infrastructure: fixed wire network, wireless cellular network or mobile ad hoc network [3]. MA paradigm is an emerging technology for developing applications in open, distributed and heterogeneous environment like the Internet. Agents have the ability to decide autonomously where to migrate to after they are dispatched; hence MA technology offers several advantages in many application areas, such as ecommerce, mobile computing, network management and information retrieval [3]. MAs are designed to execute locally on data at their destination, thus reducing network traffic and latency. Furthermore, MA asynchronous interaction can provide efficient solution in the case of unreliable and low bandwidth connection, to support mobile users that could disconnect while their agent still roam in the network. The proposed MA based authentication architecture will prevent the transmission of authentication information over the air eliminating the possibility of man-in-the-middle attack with rogue access point and eavesdropping since authentication information are carried by encrypted MAs. Furthermore, mobile clients requires authentication at the various access point to maintain their connection to the network, the proposed MA architecture is effective for moving dynamically both code and state during every authentication process since the operation performed are always the same. In this paper we explore the possibilities of using MAs to provide secure access control for next generation wireless infrastructure networks.

In particular, mobility is the most important feature of mobile agent for the following reasons [4]:

Efficiency: If an agent moves through the network to the node where resources reside, then the resulting traffic is reduced since it can pre-process data locally and decide what the most important information to transfer is. It is a crucial aspect for users who are connected by a link of lower bandwidth.
Persistence: When a mobile agent is launched, it is no longer connected to its creator node, and will not be affected if this node fails.

Peer to Peer Communication: A failure in the paradigm case of client or server is the inability of servers to communicate. Mobile agents are considered peer entities and, as such, can act as either client or server is like.

Fault tolerance: If the client/server, the transaction state is generally divided between the client and server. In case one server is down, the client can resume the situation and resynchronize with the server because the network connection is lost. However, since the mobile agent need not keep the connection permanently, in case of network failure it will continue to run in the node.

III AUTHETICATION PROTOCOLS: A SURVEY

There are many authentication protocols developed, which are based on password, shared symmetric key or public key. Some of the conventional protocols are discussed here:

A. Simple Method

In this kind of protocol, we assume that the two parties are well-known and trusted on each other. For authentication, both parties can send simple text messages without encryption, which are send to each other. The security flaws are, first the messages are in simple text, on which active attack is possible, second, there in no mutual authentication between the two, and third, there is possibility of reply attack [5].

B. Challenge-Response Method

To prevent replay attack, challenge response mechanism can be useful. As a challenge, one can send its nuance to the sender. The sender will send the hash of the received nuance along with his password to the destination as a response. The destination wills dehased the received message to authenticate the sender by looking at his own nuance [5].

C. Using Symmetric Key

For mutual authentication, we will assume that the symmetric key is generated by the destination B. The sender will send the simple text message along with his own nuance, R_A to the destination. The destination B will send his own nuance, R_B, along with encrypted form of sender's nuance and symmetric key K_{AB} to the sender. The sender will be able to decrypt the message to get K_{AB}. He will also verify the received R_A. The sender will reply with encrypted form of destination nuance along with same symmetric key. The destination will match the received R_B and K_{AB}. This way the entire procedure will give mutual authentication using symmetric key. But there may be chances of forging the nuances of each other. The overall protocol is as shown in figure 1 [5].

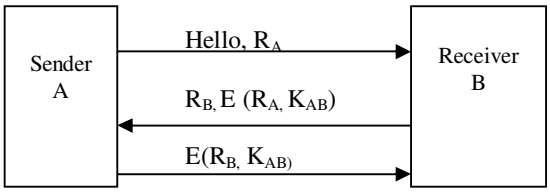

Figure 1: Symmetric Key Authentication Protocol

D. Using Public Key

1) Authentication with Public Key: In this method, one of the authenticator will send his encrypted nuance with public key of the receiver. The receiver will decrypt the nuance with his private key for verification.

2) Authentication with Digital Signature: In this method, the destination will send his nuance for the received request message. The sender will encrypt the nuance with his private key, thus applying the digital signature, which can be verified by the destination by applying the corresponding public key to decrypt the nuance. The destination will authenticate by verifying the same nuance he has send.

3) Authentication with Session Key: Here, the destination will encrypt the received nuance of the sender along with session key with sender's public key. The sender will decrypt the message to get the session key. He will increment the nuance by one and encrypt it with same session key by applying the public key of the receiver. The receiver will decrypt it with his private key to verify the session key. Thus mutual authentication can be done along with transmission with session key.

4) Mutual Authentication and Session Key (Signing and Encrypt): The figure 2 shows the protocol. The receiver will sign the nuance received from the sender along with session key by his private key. The entire message will be encrypted with sender's public key and transmitted the intended sender. The sender will get the session key by applying first his private key and then public key of the receiver. Thus authenticate the receiver. The sender also apply the signing to the increment value of nuance and the same session key by applying his private key and encrypted with public key of the receiver. The receiver will apply the same procedure to get back the same session key, thus authenticate the sender [5].

5) Mutual Authentication (Encrypt and Signing): This follows the reverse process of earlier protocol, encryption first with public key of the receiver followed by signing with private key of his own. The receiver will authenticate the sender by following reverse process [5].

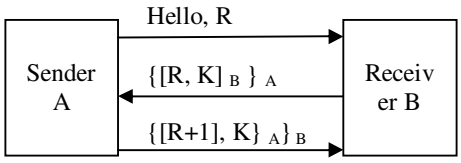

Figure 2: Mutual Authentication and Session Key

IV PROPOSED MA-BASED SYSTEM ARCHICTURE

Figure 3 shows the system architecture for proposed MA-based authentication protocol, which can be replaced the existing symmetric key protocol. The main stake holders are Mobile Agents, generated by users periodically, and migrated to the authentication platform for authentication. The terms are:

1) Mobile Agent (MA): When any user wishes to authentication with other user, he will generate Mobile Agent with necessary parameters and send it to the authentication platform for processing.
2) Authentication Server (MA-based Platform): This is centralized platform for Mobile Agents to communicate for authentication. This server will authenticate the mobile agents generated by the users.

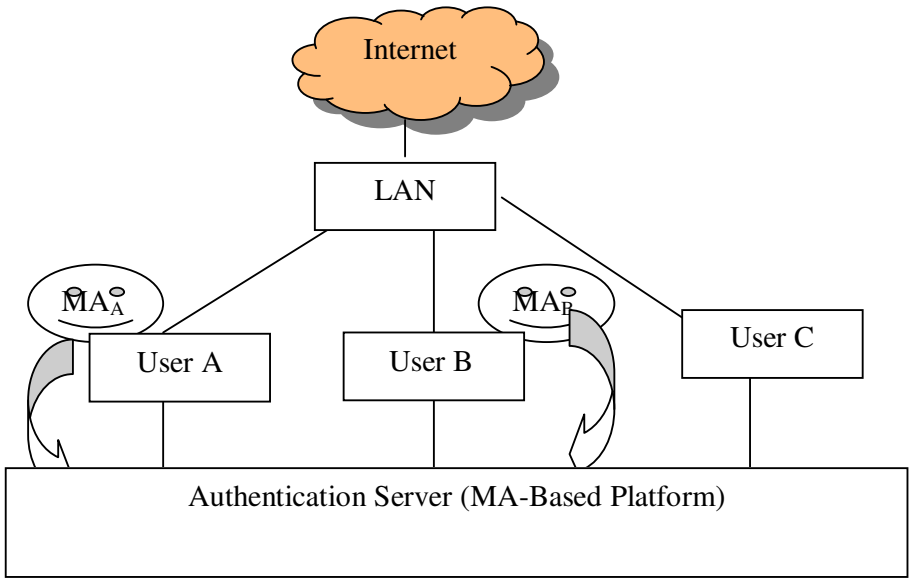

Figure 3: Proposed MA-based Authentication Protocol

A. Proposed MA-based Algorithm-A

The proposed algorithm for the authentication will be as follows: for example, user A and user B wishes mutual authentication. The steps will be:

1) User A will generate Mobile Agent MA_A (ID, A-B, Timestamp) and send it to the AS server.
2) AS will generate a small message meant for user B for authentication.
3) User B will generate mobile agent MA_B (ID, B-A, Timestamp) and send it to the AS server.
4) Now AS will have both the mobile agents for communication.
5) AS will authticate both by looking at the parameters.
6) AS will generate the symmetric key K_{AB} behalf of users, based on IDs of both the MAs.
7) Both mobile agents will take this K_{AB} from the server and send to each other.
8) Received K_{AB} will be compared with the server's K_{AB}. If that matches, both MAs will be authenticated, else the entire process will be repeated.
9) Finally, MAs will migrate from AS to their users.

B. Security Advantages of the Proposed System

First, there is no direct communication between the two users; this will make the network vulnerable to attack.

Secondly, the bandwidth will be utilized fully. Finally, Key will be secret as it will be generated by using MAs IDs. Hence, the proposed system will be more secure as there is no need to transmit the symmetric key.

C. Proposed MA-based Algorithm –B

1) User A will generate Mobile Agent MA_A (ID, A-B, Timestamp) and send it to the AS server.
2) AS will generate a small message meant for user B for authentication.
3) User B will generate mobile agent MA_B (ID, B-A, Timestamp) and send it to the AS server.
4) Now AS will have both the mobile agents for communication.
5) AS will authticate both by looking at the parameters.
6) AS will generate the session key K used by both MAs for authentication, which is based on MAs IDs.
7) We are assuming that public-private key pair will be maintained by AS for all users in a network.
8) MA_B will use his private key for signing and encrypt using public key of MA_A. This message will be sent to MA_A.
9) Similarly MAA will send the session key K to MAB.
10) Both will compare the session key from AS after decryption. If they are matches, then, both will be mutually authenticated.
11) Finally, MAs will migrate from AS to their users.

V. AGENT MIGRATION IN PROPOSED MA-BASED SYSTEM

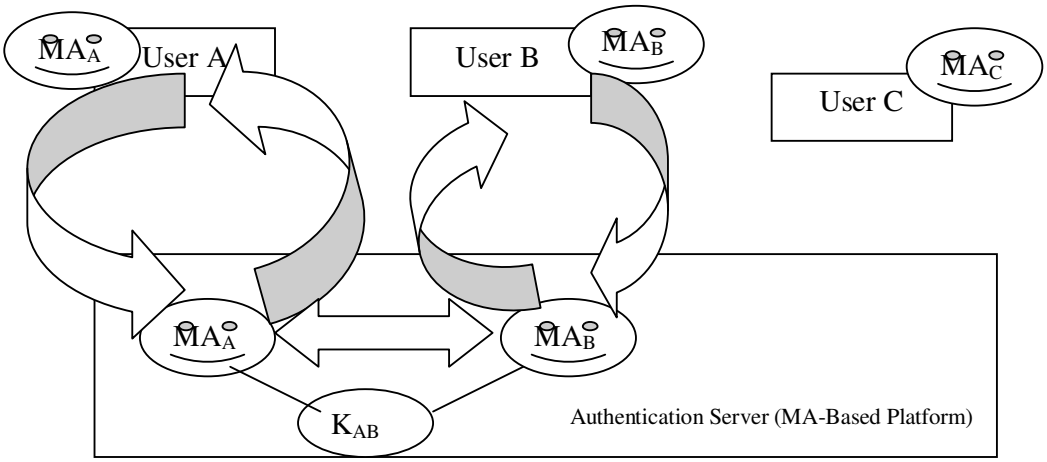

Figure 4: MAs Migration

D. Security Advantages of the Proposed System

First, each user should not maintain the public keys of other users; secondly, speed of applying public key algorithm will be increased due to simplicity of using the keys by MAs. Figure 4 shows the mobile agents migration system used for authentications. Whenever any user wishes to authenticate to other user, he will generate mobile agent. MA will be capable of identifying the environment to migrate from platform to another. With necessary parameters, the user will send the MAs to the authentication platform. The authentication platform will have many mobile agents from various users. Each mobile agent will be identified by his ID, message and timestamp. There will be communication between the two mobile agents, which belongs to different users by using either challenge-response or direct message passing within the platform. After all processing done at authentication platform, the corresponding mobile agents will be migrated to their user's platform with necessary data.

VI PROPOSED MA PLATFORMS

There are various environments for MA, but Aglet Software Development Kit is easy and simple to implement MA in Java. This platform supports the MA to execute, halt, dispatch to another host, and resume execution there. The aglet is capable of moving both the code as well as data. This is well suited for Internet environment. Some of the proposed mobile agent platforms are ASDK free software by IBM, implementation standards like MASIF and CORBA etc [1].

VII CONCLUSIONS AND FUTURE WORK

The explosive growth of network communication has made the provision of adequate and effective security challenging for such networks. User authentication security protocol is an effective technique for preventing unauthorized access to such networks. A major vulnerability is the shared medium. In this regard, we need strong and

secure access control and authentication system. In this paper, an attempt is made to propose MA-based authentication system, which can be used as substitute for using symmetric or public key techniques, which may lead to attack on the system.

In future, since MAs have strengths in a distributed and heterogeneous environment, the proposed system can be extended to support all the security goals, which protect the entire network broadly.

REFERENCES

[1] Olatunde O. Abiona and Yu Cheng, "Mobile Agent based Authentication for Wireless Network Security".
[2] L. Xia and J. Slay, "Securing Wireless ad-hoc Networks: towards Mobile Agents Security Architecture".
[3] S.P. Alampalayam and A. Kumar, "An Adaptive Security Model for Mobile Agents in Wireless Networks", in Proc. IEEE Global Telecommunication Conference, 2003, vol. 3, PP 1516-1521, Dec. 2003.
[4] J. Dale, "A mobile Agents Architecture to Support Distributed Resource Information Management", thesis, University of Southampton, 1998, 86p.
[5] Deven Shah, "Information and Network Security", a Book Published by Wiley publication.

BioData of Author

S. M. Chaware is working as Asst. Prof. in Information Technology Department at D.J.Sanghvi College of Engineering, Mumbai. His areas of interest are Mobile Agent Technology, Mobile Computing, Information Security and Multilingual Data Processing. He is currently pursuing PhD from NMIMS University.