Security Model that prevents Data Leakage in Distributed Mobile Systems Using Surrogate Objects

R.Anitha
M.A.M. College of Engineering, Tiruchirappalli,India

S.Ravimaran
M.A.M. College of Engineering, Tiruchirappalli,India

P.Valarmathi
M.A.M. College of Engineering, Tiruchirappalli,India

ABSTRACT
Devices such as smart phones are considered to be the most common communication devices in current scenario. Recently, mobile phones are not only used for voice and message communication but also, sending and receiving important data such as social security numbers, bank account details and passwords, so it is important to provide proper security to the data's in the distributed mobile environment. Although there is lots of security mechanisms have been proposed already, a lot more needs to be done. We have proposed a new security model that prevents data leakage among object based transaction execution in distributed mobile environment. The model first secures the data on the object by authenticating the user and then secures the data during transaction execution and reconciliation. The proposed mechanism is efficient in securing the data's because communication has been reduced by making the object to act on behalf of the server. It also ensures confidentiality of transaction by creating an encrypted tunnel between the objects and database server. The entire process is simulated and results shows that our model is more secured and provide less communication overhead than the existing models in the distributed mobile environment.

Keywords
Security, Authentication, Surrogate object, Confidentiality.

1. INTRODUCTION
Wireless and mobile communication systems are very famous among the customers as well the operators and service providers. The services like online banking, e-payment, and e/mcommerce are already using the Internet. The financial institutions like banks and other organizations would like their customers to use online services through mobile devices keeping the wireless transaction as secure as possible from the security threats. Smart cards (e.g. SIM card) have been proposed for applications like secure access to services in GSM to authenticate users and secure payment in Visa and MasterCard [8]. Wireless transactions are facing several security challenges. Wireless data passing through air interface face almost the same security threats as the wired data. However, the limited wireless bandwidth, battery, computational power and memory of wireless devices add further limitations to the security mechanisms implementation [9]. The use of mobile communication in e/m-commerce has increased the importance of security. An efficient wireless communication infrastructure is required in every organization for secure voice/data communication and users authentication. Among the main objectives of an efficient infrastructure is to reduce the signaling overhead and reduce the number of updating *Home Location Register/Authentication Center* (HLR/AuC) while the *Mobile Station* (MS) changes its location frequently [9]. Although there are lots of security mechanisms have been proposed already, there are some flaws in performance. This paper aims at achieving object based security in distributed mobile environment. Our security model provides authentication and confidentiality of data's in the object. The proposed mechanism is efficient in securing the data's because use of object reduces the response time and network traffic[7].Along with this our model provide less communication overhead than the existing models in the distributed mobile environment.

The rest of the paper is organized as follows: Section 2 explains some of the related work for our model. In Section 3 we propose object based security model. Section 4 provides the performance evaluation and finally a conclusion is drawn in Section 5.

2. RELATED WORK
A significant amount of research was and continues to be devoted to mobile phone systems' security: integral components as well as complete systems are described and analyzed

F.Grecas, proposed brute-force approach [1] uses a public key-based cryptographic solution. The major problem using public keys in encryption/decryption is its complex algorithm and higher processing time. Recently, some published research papers proposed mobile key-based security solutions by modifying existing public-key algorithms. As known, most people still prefer to use asymmetric-key cryptographic techniques on mobile devices over symmetric-key cryptographic techniques. However, they must be customized and improved for the use on mobile devices. There are innovative approaches using a combination of new cryptographic algorithms based on data distribution, time distribution, and workload distribution. Uwe G. Wilhelm gives a clever alternative approach to secure the data exchanged among mobile systems in [2]. The system uses Cryptographically Protected Objects (CryPO) to control the security at the object-level. CryPO requires a Tamper-Proof Environment (TPE) on the mobile device, whose primary purpose is to store a private key, which is not known even for the mobile user. The advantage with this approach is that the Object User cannot decrypt the object due to the lack of private key. He downloads the object into the mobile device. Only the mobile device with the private key can decrypt the object. The main problem with CryPO is that it is nearly impossible to create a Tamper-Proof Environment on any device. If the device is lost, any user can decrypt a downloaded object.

As an alternative to RSA, DeviceForge.com proposed Elliptical Key Cryptography [3] is an implementation of asymmetric key cryptography, and is well suited for securing

mobile devices. ECC is an improvement over Discrete Logarithm cryptography. ECC is a typical result of improving the existing cryptographic algorithms for mobile device and access security.RSA requires considerably more bits for the key than ECC. With smaller keys, mobile device processors can perform arithmetic operations much faster, and consume less battery energy. ECC based ECIES is very efficient for key generation, and the disadvantage of its long encryption/decryption times is offset by its comprehensive functionality.

Since computing public key signatures is a CPU intensive operation, X. Ding proposed the server-aided signature which is an effective approach to offloading intensive security computations to a trusted server side [4]. Another related solution to offload intensive processing is called Offline/Online cryptography, or SignCryption which is proposed by Fangguo Zhang, Yi Mu, [5].

GSM networks provide a security enhancement over 1G by authenticating users and supporting confidentiality and anonymity features. However, the related algorithms initially weren't open for community review, which caused some serious flaws to be overlooked. Eventually GSM security algorithms leaked and their flaws were discovered [6]. GSM security model is based on a 128-bit shared secret key between the subscriber's SIM and the network – if that key is compromised, the entire account is compromised.

The surrogate object model elegantly solves a whole set of problems in distributed mobile systems: location management, mobile data access in client-server systems, disconnected operations etc. It also provides an ideal placeholder for host specific information, thus enabling application or host specific constraints to be enforced. This facilitates building customizable applications over distributed mobile systems [7]. Benefit of using the surrogate object model, is its response time and network traffic. But there is a security flaw in object model. To overcome this, our model provides authentication and confidentiality to the data's in the object in the efficient manner with low communication overhead.

3. PROPOSED OBJECT BASED SECURITY MODEL:

Fig.1 shows the overall architecture of the proposed object based security model.In our proposed model, first we provide security for the data being transmitted between the mobile devices in distributed environment. In our scheme even though the receiver is disconnected from the network during data transmission, the surrogate object will receive the data and sends it back after the receiver gets connected in a secure manner. In addition to that, we provide confidentiality to the data during transmission between the server and the object using SSL Handshake.

3.1 Authentication

Authentication is the process of verifying the validity of a claimed individual. Authentication is not limited to human beings; services, applications, and other entities may be required to authenticate also. There is lots of authentication process available, here we provide authentication using Certification Authority (CA). The authentication process used in our model is described as follows.

Certification Authority Configuration and Key generation: In this section, we set a Certification Authority which is able to manage the clients request and response. The Certification Authority is responsible for generating the keys for the clients newly added to the network. Client generates a pair of keys(public and private) and sends the public key to the CA.Then CA checks the validity of the client and issues the public key to the client. CA also stores the public key in the database.

Surrogate object (SO) creation and security: Create the surrogate in the static network (MSS) to act on behalf of each mobile device. One consequence of using the surrogate object model is that, mobile devices would be transparent to the instability of wireless communication. The surrogate object can remain active, maintaining information regarding the current state and plays an active role on behalf of the device. Each object is assigned with an ID that is SOID which is used as reference for the associated mobile device. In order to provide security for the data in the surrogate object, authentication process has to be done. Prior to perform authentication the SO and associated mobile device has to share a secret key (K_s) and the secret code (SC) in a secure manner.

Key Exchange Session*:* The shared secret key has been securely exchanged with the help of key exchange session in the proposed system. Diffie-Hellman algorithm is used to make agreement on a shared key that is used with AES-Rijndael to encrypt the secret code and exchange it with the object.

Mobile device can start the key exchange session immediately after receiving the message from the object. Mobile device can start the key exchange session by calculating the value of A,depending on the secretly generated value a, and the shared secret parameters g and p (that is, g and p are fixed by the mobile application).The value of A is calculated as g power a mod p.

Mobile device then sends the value of A, with request to start key exchange session. object can reject the session if it is not ready to go through the key exchange session steps. It can also accept the request; if so, the object must calculate the value of B depending on the secretly generated value b and the shared secret parameters g and p and send it back to the mobile device with accept message. The value of B is calculated as g power b mod p.

Mobile device will be able to calculate the value of K, once it receives the value of B from the object. Similarly object also will be able to calculate same key K, depending on the value of A, which has already been received from the mobile in the request message. Thus, mobile device and object will obtain the same secret key and then, they can use it for one time only to encrypt their secret code and exchange it.

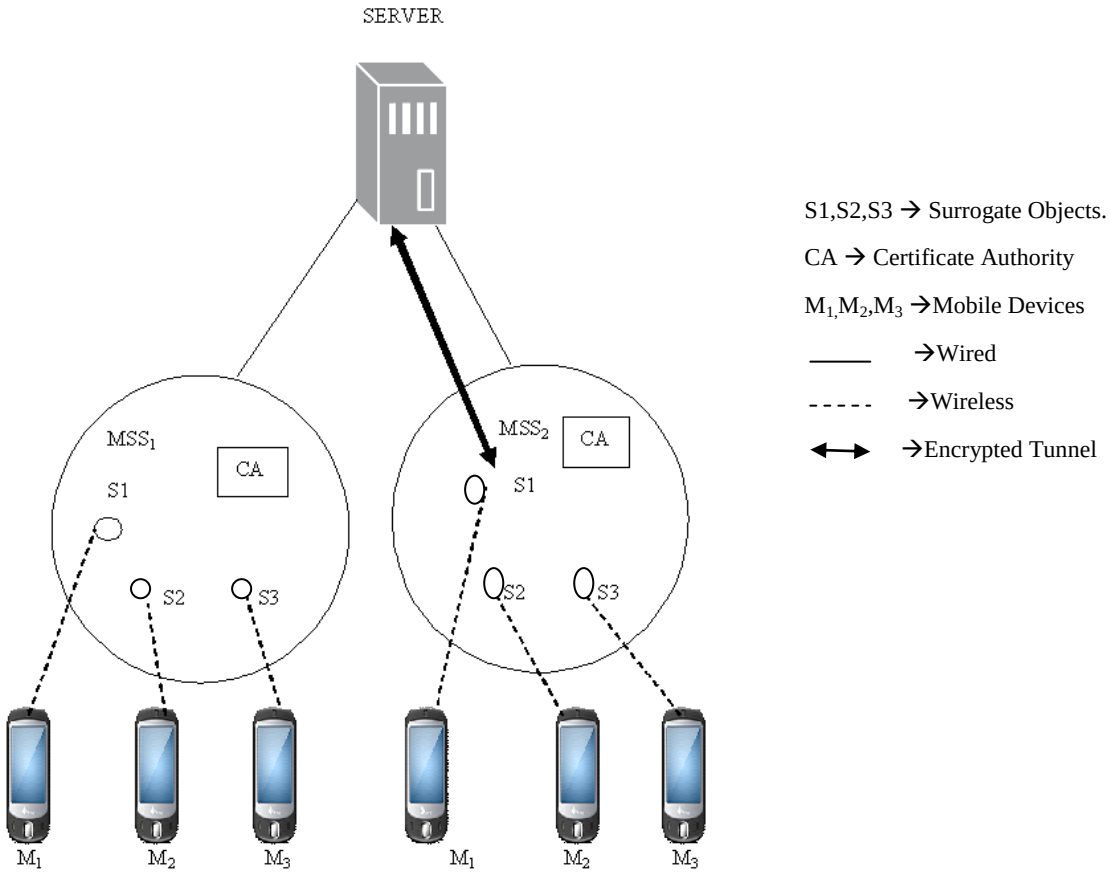

Fig.1 Proposed Architecture for Object Based Security Model

How the authentication works: To provide authentication, the mobile device send a secret code by encrypting it with K_s. Object decrypts the secret code and checks that both the shared secret code and decrypted secret code are same. If both are same then mobile device is authenticated.

Secured Sender – Surrogate – Receiver Transactions: This section discuss about the secured data transmission between the sender and receiver through the surrogate object. Here we are using the Elliptic curve crytography algorithm for encrypting the datas. It consists of two phases,

- Sender – Surrogate transaction management
- Surrogate – receiver transaction management

Sender – Surrogate transaction: Consider MH_s is the sender and MH_r is the receiver of the message. MH_s encrypts the message (M) using the public key of the receiver. Public key of the MH_r is got from the CA. The encrypted message $E_k(M)$ is sent to the object associated to MH_r.

Surrogate – receiver transaction: After getting the encryted message, SO sends a request to MH_r to authenticate itself. As described above, authentication process takes place. After performing the authentication, the SO encrypts the message again using the K_s that is $K_s(E_k(M))$ and sends to the MH_r. Finally, MH_r first decrpts the message with K_s and again decrypts the message with its private key to retrieve the message (M).

High level design:

After receiving the data, SO will check whether the intended receiver is available in MSS. If the receiver is within the MSS, SO performs the above said process immediately. If the receiver is not in the MSS currently, then the SO receives the message and hands it over to the receiver in a secure manner when it returns back to MSS.

The pseudo code for the authentication process is given below.

Authenticate()

BEGIN

Sender Gets the certificate from the certificate from the Certificate Authority.

Sender encrypts the message

Encrypt();

Sender sends the message to the receiver

If (receiver is available)

Receiver authenticates itself with SO using shared key.

If (flag==1)

Authentication success.

Else

Transaction is rejected.

Else

SO receives and stores the message.

SO hands over the message to the receiver on availability.

END

Encrypt()

BEGIN

Sender encrypts the message(M) with the public key of the receiver $E_k(M)$.

END

3.2 Confidentiality:

Along with providing security to data transmission between the mobile devices, our model also provides security to the data transmission between the server and the object by creating an encrypted tunnel using SSL handshake.

In order to establish the secure communication between the server and the object, a handshake must be established. This handshake is responsible for determining the SSL settings, exchanging public keys and the basis for the mutual authentication process. The handshake process is as follows:

1. An object contacts a remote server to start a secure session by using a digital X.509 ID certificate.

2. The object automatically sends the clients SSL version number, cipher settings, randomly generated data, and other information to the server.

3. The server responds, by sending the object digital certificate, along with the server's SSL version number, cipher settings, and so on.

4. The object examines the information contained in the server's certificate, and verifies that:

a. The server certificate is valid and has a valid date.

b. The CA that issued the server certificate has been signed by a trusted CA whose certificate is built into the object.

c. The issuing CA's public key, built into the object, validates the issuer's digital signature.

d. The domain name specified by the server certificate matches the server's actual domain name.

5. If the server can be successfully authenticated, the object generates a unique session key to Encrypt all communications with the server using asymmetric encryption.

6. The object encrypts the session key itself with the server's public key so that only the site can read the session key, and sends it to the server.

7. The server decrypts the session key using its own private key.

8. The object sends a message to the server informing it those future messages from the object will be encrypted with the session key. The server then sends a message to the object informing it, those future messages from the server will be encrypted with the session key.

9. An SSL-secured session is now established. SSL then uses symmetric encryption (which is much faster than asymmetric PKI encryption) to encrypt and decrypt messages within the SSL-secured *pipeline*.

10. Now one of the object have authenticated, the second object will now authenticate using the same process.

11. Once the session is complete, the session key is eliminated.

After performing the above said steps a secure tunnel will be created. Now the object can transmit the data to the server in a secure manner.

4. PERFORMANCE EVALUATION

The performance evaluation of our proposed model is based on the communication overhead and security. With the use of object, number of communication between the server and mobile has been reduced, as object takes the task of server. This improves the performance compared with the model providing communication between the server and the mobile.

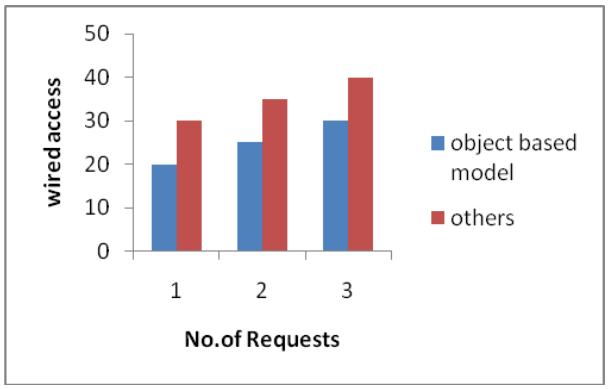

Fig.2 Performance of Object Based Model

The fig.2 compares the object based model with others based on the request and wired access. It thus shows that our object based model provides better performance due to the low communication in the wired network.

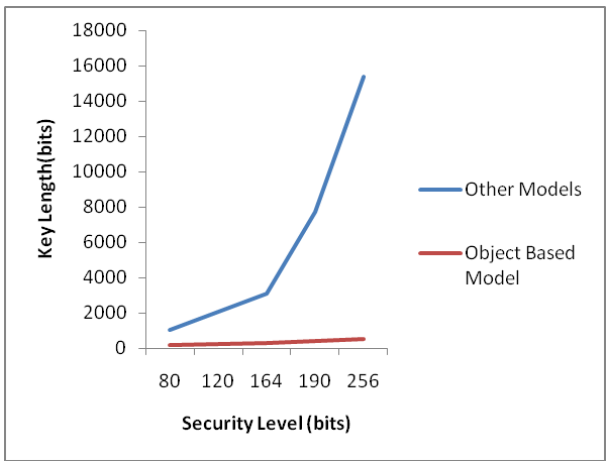

Fig.3 Comparison of Object Based Model with other models

The fig.3 compares the performance of proposed model with other models based on the security. Proposed algorithm uses shorter keys which lead to easier data management and lower hardware requirements. The main security parameter of a cryptosystem is the length of the key. Since the length of the key is shorter, the time needed to perform encryption/decryption is sufficiently small on the handheld devices.

5. CONCLUSION

The demand for mobile security has become increasingly important because of the increasing applications, built for mobile phone. There is lots of security mechanisms proposed but they did not use object based security method. This paper provides a new object based security model which reduces communication of mobile devices with server. This model uses surrogate object to store the data in absence of mobile device and returns the data to the mobile device in secured manner. If the data in the server has to be updated, it is done securely by creating an encrypted tunnel using SSL handshake. Our model is evaluated and it provides less communication overhead with more security.

6. REFERENCES

[1] Constantinos F. Grecas, Sotirios I. Maniatis and Iakovos S. Venieris, “Introduction of the Asymmetric Cryptography in GSM, GPRS, UMTS, and Its Public Key Infrastructure Integration”, Mobile Networks and Applications, Volume 8, 2003.

[2] Uwe G. Wilhelm,“Increasing Privacy in Mobile Communication Systems using Cryptographically Protected Objects”, Verläßliche IT-Systeme, 1997.

[3] DeviceForge “An IntroductioSurrogate Object Model: A New Paradigm for Distributed Mobile Systemsn to Elliptical Curve Cryptography”, July 20,2004.

[4] X. Ding, D Mazzochi, and G Tsudiatures”, ACM Transactions on Internet Technology (TOIT), 2007.

[5] Fangguo Zhang, Yi Mu, and Willy Susilo, "Reducing Security Overhead for Mobk, “Equipping Smart Devices with Public Key Signile Networks," The proceedings of 19th International Conference on Advanced Information Networking and Applications (AINA'05), Vol.1, 2005.

[6] Jagdish Bhatta and Lok Prakash Pandey “Performance Evaluation of RSA Variants and Elliptic Curve Cryptography on Handheld Devices” IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.11, November 2011.

[7] M.A. Maluk Mohamed, D. Janakiram and Mohit Chakraborty “Surrogate Object Model: A New Paradigm for Distributed Mobile Systems”

[8] N. T. Trask and M. V. Meyerstein, “Smart Cards in Electronic Commerce”, A SpringerLink journal on *BT* Technology, Vol. 17, No. 3, 2004, pp. 57-66.

[9] N T Trask and S A Jaweed, “Adapting Public Key Infrastructures to the Mobile Environment”, A SpringerLink journal on BT Technology, Vol. 19, No. 3, 2004, pp. 76-80.

[10] E Mohammed, A E Emarah, and K El-Shennawy, “Elliptic Curve Cryptosystk, “Equipping Smart Devices with Public Key Signems on Smart Cards”, Proceedings of IEEE International Carnahan Conference on Security Technology, 2001.

[11] A. S. Wander, N Gura, H Eberle, V Gupta, and S C Shantz, “Energy Analysis of Public-Key Cryptography for Wireless Sensor Networks”, Proceedings of 3rd IEEE International Conference on Pervasive Computing and Communications, 2005.

[12] Millan, W., Gauravaram, P., 2004. Improved Attack on the Cellular Authentication and Voice Encryption Algorithm. International Workshop on Cryptographic Algorithms and their Uses. Gold Coast, Australia, July 2004.

[13] Wingert, C., Naidu, M., 2002. CDMA 1xRTT Security Overview. Qualcomm Incorporated.

[14] Millan, W., Gauravaram, P., 2004. Cryptanalysis of the Cellular Authentication and Voice Encryption Algorithm. IEICE Electronics Exprss, Vol.1, No. 15.

[15] V. Gayoso Martínez, L. Hernández Encinas, and C. Sánchez Ávila “A Survey of the Elliptic Curve Integrated Encryption Scheme” Journal of computer science and engineering,Vol.2,issue2,August2010