Chapter 4 Testing and implementation strategy for the proposed protocol

Proposed protocol carries no value if not evaluated and proven to be working properly. In this chapter testing and evaluating the proposed protocol are described. We discuss the complete setup chosen for implementing the protocol, reasons for choosing a typical implementation strategy and other components like language and suit of protocols to implement and test along with the proposed protocol. The authors have also chosen the most suitable of applications which demands a secure protocol.

This chapter begins with description of different requirements for testing the protocol. It then moves on to present the setup used during simulations. The chapter also gives the simulation details such as the need for the particular setup and justifies usage of each component in the setup. Then mobile based payment system application where protocol can be applied is presented in detail. The chapter ends with conclusion.

4.1 Requirements for Testing

Any research work is not considered complete until its worth is proven objectively. Keeping this argument in mind, the authors need to prove that the modified key exchange protocol proposed by them in this thesis is more efficient in comparison to other variations of the protocol with providing same level of security as other variants. There are two methods that can be utilized to test a protocol: 1) by mathematical model [94] [95] and 2) by testing the protocol through simulation techniques [96] [97].

There are parameters such as the number of operations, battery power usage, network bandwidth available and key exchange time that affect the performance of the protocol during execution. As there is no clear relationship between parameters defined, it is difficult to formulate a mathematical model relating different parameters affecting protocol performance during execution. In fact, one method is to use those protocols in real world applications, collect user’s responses and decide which protocol is most suitable. This is not applicable in most cases including ours. The only other way left is to use some kind of simulation in a controlled environment. We need to test the worth of the modified protocols by implementing them in a specifically chosen modeled real world application and comparing execution performance of different well known key exchange protocols against the proposed protocol.
The authors made several assumptions about the setup of experiment to test the protocol. They are as follows:

1. Role of a wireless device is to send and receive information to and from the web service regarding the shared secret key and to and from server regarding critical information.
2. Web service is available and accessible from client and server devices.
3. The role of server would be accepting the communication from client and web service and co-operate in establishing a secure connection.
4. During the experiment network connection is stable and no data loss takes place.
5. All the devices that are required for communication are online and working properly.
6. The devices and server are at geographically separate locations, and an Internet connection is required for communication.
7. Internet speed with different bandwidth affects performance of the protocol.

As the proposed protocol is designed with focus on wireless devices, we need them for experimentation. The proposed protocol uses public key cryptography technique as discussed in chapter two. The server computers will be used as a trusted third party and web application host. The internet connection is required for communication between wireless devices and the server. Mobile application or website will be used to simulate a secure connection provided by the authors’ protocol. A web service will allow the wireless devices to securely request the shared secret key which will then be used for subsequent communication over insecure network. Ubuntu Linux system is used as an open source alternative to proprietary operating system, however, our experimentation does not have any assumption about the operating system and thus this choice does not actually affect the results. Therefore, the produced results are generalized across operating systems unless it is a quite typical and specific type of OS.

The authors have designed the system with three parts. These parts are 1) client; 2) server and 3) web service. The client provides input and receives output. The web service processes the request of client, generates and sends the shared secret key to client. The client then forwards the secret key to server. In this protocol key generation is done on a remote computer. It can be argued that key generation on client's side is faster than generating key on a remote computer and getting it to client. However, in our scenario there are two computers which are unknown to each other. As these computers cannot be assured of authenticity of each other, these computers are
not in a position to trust message coming from each other unless some third party introduces them, which our protocol does. Hence, we need a trusted third party to introduce both sender and receiver so they can communicate further. It is for this reason that the key needs to be generated on a trusted third party which will encapsulate the key in a ticket and then send to A. A will have a ticket as a message for B, and since this message is in form of a ticket which cannot be altered by A, B will trust this message establishing trust relationship between A and B and solving trust relationship problem. As the ticket is encrypted by a shared secret key between B and the server, it is not possible for A to generate a new message as well. This strengthens the security.

The wireless device in our protocol acts as a client. The client initiates communication as it needs to exchange critical data with server over insecure channel. This critical data needs to be securely transmitted such that only intended recipients can access that. Our protocol is specially designed for wireless devices to perform the key exchange function that is crucial for establishment of a secure communication channel.

The web service is used as a trusted third party as the devices communicating with each other are unknown to each other. The web service acts as a trusted third party that both sides trust. Hence, web service is crucial to function of our protocol.

Server computer is used as a recipient for communication to which the wireless device sends information on an insecure channel. The wireless device receives the key from the web service and sends it to server along with the ticket to establish a trust based relationship.

Figure 4.1 shows components that are used for testing of the proposed protocol. Following is the description of components and their roles in the system used for testing the protocol.

**Figure 4.1 System Components**
• Wireless Device: Wireless device is required to work as an initiator of communication in our system. As the devices communicating are unknown to each other, there is a trust gap. This trust gap is filled by web service acting as a trusted third party. The device initiates communication and gets the key and a ticket from a trusted third party in response. The device then forwards the key and ticket received from trusted third party to server.

• Server Computer: The server computer is used as a recipient of communication from client's side in our system. The server computer does not trust the message coming from client's wireless device unless it is coming along with the ticket from a trusted third party.

• Internet Connection: An Internet connection is required for the communication between clients and the trusted third party server. This Internet connection is insecure and information that is passing through this channel can be easily eavesdropped and recorded. The protocol overcomes this limitation of Internet channel by sending messages encrypted with previously exchanged key between two parties. Internet connection is used for testing efficiency of the protocol on different bandwidth. Following different types of bandwidth are used to test the proposed protocol: 1) 2g; 2) 3g and 3) Wi-Fi (2 Mbps).

• Mobile Application or Website: A website will be working as an initiator of communication in our system. It will work with third party to establish a trust based relationship with the recipient. A website or mobile application will work as a front end
for users wanting to secure their communication when exchanging information with the server. This website or mobile application will be clients from where communication will be initiated.

- **Web service**: A web service will work as a trusted third party for our system. As the two parties are unknown to each other before communication they have a trust deficit. This web service is used in this system to overcome this issue. It is used by initiator to request the key and the ticket. Web service will be responsible for generating the key, assembling packet with the key, ticket and other information and encrypting it before sending it back to the initiator.

- **Android**: Android operating system is an open source operating system for mobile phones. It is widely available on various smart phones and tablet’s configurations. As android is an open source system, application deployment in android is easier compared to other systems available. In this system the authors were able to deploy the client app directly without going through the android app store. In Apple’s iOS and Windows phone systems the app can only be deployed by downloading it through the official app store.

### 4.2 Test Setup

Following setup was used to test the protocol. The proposed protocol by authors is targeted at mobile devices and wireless computers. Therefore, it is imperative that devices of these categories are included in the experiment. Hence, the authors include three device categories: 1) Laptop computer; 2) Mobile phone and 3) Tablet computer.

The authors have used the latest hardware configuration available to them at the time of simulation. The authors have used open source software for the development of application and for the testing. The authors expect that due to this choice of open source software, even if the hardware is changed, the software will still remain the same hence there could be slight difference in overall performance. However, this difference will not be significant for the protocol.

<table>
<thead>
<tr>
<th></th>
<th>Computer</th>
<th>Mobile Phone</th>
<th>Tablet Computer</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Table 4.1</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prepared By: Pranav Vyas (11DRMCA004)  53  Guided By: Dr. Bhushan Trivedi
The laptop computer here is acting as a server hosting the application for testing the proposed protocol as well as the trusted third party. The laptop computer is connected to Internet through a 2 Mbps Wi-Fi connection. The laptop computer has the PHP web service hosted on laptop with live IP. The mobile phone and tablet computer have a mobile app installed for testing. Mobile phone and tablet can communicate with the server and trusted third party through various internet bandwidths through GPRs and Wi-Fi connection. The setup is unbiased in nature as it can be used to test other public key cryptography system protocols.

4.3 Application of the Protocol
There are various applications that require secure connection for communication over the insecure network. Key exchange protocols play an important part in this scenario by providing secure technique to exchange the secret key between various parties involved in communication. Although the core requirements for key exchange protocols remain the same for all the applications that seek to have a secure connection, there are many application specific requirements. The authors studied following applications as they will have a large impact on Indian citizens in future [98] [99] [100].

<table>
<thead>
<tr>
<th>Manufacturer</th>
<th>Dell</th>
<th>Motorola</th>
<th>Samsung</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>XPS 15</td>
<td>Moto G 2nd Gen</td>
<td>Galaxy Tab</td>
</tr>
<tr>
<td>Processor</td>
<td>Intel Core i5 2.4 GHz</td>
<td>Cortex 1.2 GHz</td>
<td>Cortex 1.2 GHz</td>
</tr>
<tr>
<td>Main Memory</td>
<td>4 GB</td>
<td>1 GB</td>
<td>1.5 GB</td>
</tr>
<tr>
<td>Operating System</td>
<td>Ubuntu Linux</td>
<td>Android</td>
<td>Android</td>
</tr>
<tr>
<td>Web Service for Trusted Third Party</td>
<td>PHP Web Service Hosted on Apache Web Server</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Internet Connection</td>
<td>2g, 3g and Wi-Fi (2 Mbps)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
According to sources in news, the election commission of India is planning to give voting rights to NRIs and other Indians who do not wish to visit polling station [98]. As per another news source, the health ministry of government of India is planning to start e-health services that will enable remote diagnostics and faster and more efficient health service to more than a billion of Indian citizens [100]. According to a study [99], e-commerce market places are growing at more than 20% a year in India, enabling Indian to be third largest e-commerce market in the world. As per the review done by authors in [101], the authors studied security requirements of different mobile application systems. The proposed protocol aides in establishing a trust based relationship when two computers are unknown to each other. However, protocol needs to be augmented to include security requirements specific to certain systems.

The authors concluded in this review that the mobile based payment system’s requirements can be fulfilled by our protocol, hence, the authors selected this system as a test system.

4.3.1 Electronic Voting System

For an electronic voting system, the protocol needs to have properties like: 1) authentication; 2) uniqueness; 3) integrity; 4) verifiability; 5) auditability; 6) reliability and 7) secrecy [101].

The proposed protocol is a three party protocol. It uses a combination of public and private keys. The receiver’s private key is only known to a trusted third party and receiver himself. With this assumption, if any request is received from any machine and is encrypted with the private key of receiver, it is determined that the message is coming from a trusted source as it is originally sent to sender from a trusted third party.

Uniqueness in the protocol is achieved by using timestamps with each key preventing the multiplicity attack and maintaining freshness of the key. Authentication is achieved by usage of nonce.

In the protocol integrity of message is guaranteed by encrypting the key with a private key of the receiver by a trusted third party. As no one other than intended receiver and trusted third party has access to the key, only receiver can decrypt the message and read contents of it.
The protocol does not have features that let user verify his or her vote from other votes casted. However, the protocol can be modified in future to provide this functionality.

The protocol does not have a feature to authenticate election records. It does not allow verification of ballots and vote counting.

The protocol shows reliability by being able to execute faster than other similar protocol. This is achieved by shortening the number of steps required for key exchange. It is also resistant to attacks resulting from exchange of stale key. This is achieved by using timestamp to check freshness of the key.

The protocol provides secrecy of communication session as no one other than the three parties involved in communication knows about the communication taking place.

The proposed protocol provided trust based relationship when the two computers are unknown to each other. However, the protocol needs to be augmented to be able to provide 1) uniqueness 2) reliability 3) verifiability. This does not fall within scope of current research.

4.3.2 Electronic Healthcare Management System

For electronic healthcare management system, following properties in protocol were identified: 1) privacy; 2) trustworthiness; 3) authentication and 4) responsibility [101].

The protocol does not have any features that can hide identity of a patient whose records are being accessed.

The protocol works on symmetric key cryptography system that has its basis in trust based relationships between communicating devices. There is also a trusted third party that all other devices in network trust. This is the only computer or group of computers that knows secret key of devices other than devices itself.

The protocol helps in establishing a secure connection on which data can be encrypted and then exchanged, such that that even if a malicious user eavesdrops, or captures packets, he or she is not able to decrypt information.
4.3.3 Mobile Based Payment System

For mobile based payment systems security requirements for protocol are: 1) authentication; 2) confidentiality; 3) integrity and 4) non-repudiation.

The protocol works on confidentiality based on trust. It is assumed that if the message is coming from a device that can identify itself and is in the list of trusted devices in network, it can be trusted to be confidential in communication.

It is possible for a trusted third party to keep track of all the requests. It can track source and destination of requests. Thus, if any one party ceases communication, the trusted party can provide proof of involvement.

4.4 Architecture of Application for Testing Protocol

An e-commerce system was selected by the authors based on their study of different systems that could affect the future of Indian citizens. As most of the security requirements for an e-commerce system were supported by the protocol proposed by the authors, an e-commerce system was selected for testing the proposed protocol. The most integral part of an e-commerce system is the payment sub system where the user shares personal information such as credit/debit card number or bank details. A key exchange protocol can be helpful in making the exchange of this data secure.

The architecture adopted for development of an e-commerce system is a standard 3-domain architecture used for development of systems where user is required to pay by credit/debit cards or net banking for availing goods or services provided by website, web application or a mobile application. The authors selected this architecture as this is a standard architecture supported by major electronic payment service providers such as Visa, MasterCard and American Express [102]. The figure 4.2 is followed by description on how architecture operates.

Fig. 4.2 E-commerce System Architecture
Following is the description of all the steps in figure and parameters of all requests and responses [103] [104] [105]. These requests and responses are based on a generic system. The authors focus on strengthening communication between the client and the server.

1. In this step the client sends billing address and payment details to a merchant server. Following information is sent by client to merchant server.
   a. OrderNo.: Order number from merchant’s side.
   b. ProductDetail: Details of a product to be purchased.
   c. Total_Amt: Total amount of transaction.
   d. ACCT: Bank account no./Credit/Debit Card no.
   e. ExpDate: Expiry date of Credit/Debit card.
   f. CVV: CVV number on Credit/Debit card.
   g. FirstName: Cardholder’s First Name.
   h. LastName: Cardholder’s Last Name.
   i. Street: Street name and number of cardholders.
   j. Zip: Zip code of cardholder.

2. In this step merchant sends payment information sent by client to its payment gateway.
Universally Composable Session Key Exchange Protocol for Mobile Computers

a. In this step the merchant’s payment gateway sends Bank account no./Credit/Debit Card no. and ExpDate information to a third party authenticator in interoperability domain.

3. In this step third party authenticator will sign a message with its digital signature and send information it has received from merchant’s payment gateway to issuer for verification.

4. Issuer will verify the digital signature of the sender and if it is verified, it proceeds further to verify details of the customer sent in message. Issuer responds with an ACK code. ACK code can have either of 3 values, 0, 1 or 2. 0 means details are incorrect. 1 means details are correct. 2 means it cannot be verified and that there is an error. Issuer signs this response with its own digital signature.

a. 3rd party authenticator receives acknowledgement message in the form of ACK from issuer and then checks digital signature to verify issuer’s identity, and logs it in for record purpose. It then forwards acknowledgement status message with its own digital signature to merchant’s payment gateway.

The payment gateway will then verify the signature for identification purpose and then decide next action based on the value of ACK. If it is a failure (value is 0 or 2), it will go to step 10 and send back transaction failure message to the web server. In COMMENT field the reason of failure can be specified based on value of ACK. If ACK value is a success then it will move to step 5.

5. Payment gateway will send the details of customer to acquirer.

6. Acquirer will sign the message and send details to interbank payment gateway in interoperability domain, this message will have all the details including ACK message.

a. Internal payment gateway will verify message by signature and forward the message to issuer after signing it digitally. It will also have a field PAYID which will help internal payment gateway to keep track of payment transaction.

Issuer will verify signature and after successful verification, it will check for funds available with cardholder or account owner, if funds are available it will go to step 7, otherwise it will return transaction failure error to step 9, with acceptance AuthCode 2. AuthCode 2 means funds are not available with card holder or account owner. AuthCode
1 means the user authorized transaction. AuthCode 0 means the transaction is not authorized by cardholder or account owner.

7. Issuer generates an OTP that is delivered to card holder/account owner through SMS on his registered mobile number. Issuer will also generate a TranID and TIMESTAMP to keep track of particular transaction.

8. Customer authorizes transaction by entering OTP received in SMS on his registered number. If OTP is entered correctly, then transaction is authorized, otherwise it is not authorized.

9. In this step issuer will return AuthCode value and transaction ID, TIMESTAMP with a message signed by it to internal payment gateway in interoperability domain.
   a. Interoperability domain will verify the message received by issuer with signature and log it for record purpose. It will also forward message with its own signature on it to acquirer on merchant’s side. This message will also contain field PAYID to keep track of payment transaction on internal payment gateway.

10. Acquirer will verify message with signature and associate transaction with ORDNO. It will then checkAuthCode field. It will then forward AuthCode with its signature to merchant’s payment gateway based on MerchantID.
    a. Payment gateway will verify the message sent by Acquirer with signature and forward AuthCode and OrderNo to merchant’s server. Payment gateway will have the following extra fields of information apart from those mentioned in step 1.
       i. ACK: Acknowledgement of identity of cardholder or account owner by a third party.
       ii. PAYID: Payment ID of particular transaction which will allow user to track the transaction on internal payment gateway.
       iii. OTP: It is an authorization code generated by issuer and used by customer to authorize the transaction to issuer.
       iv. TIMESTAMP: It is used to keep track of date and time of transaction.
       v. AuthCode: It is an authorization code that is used by issuer. Value of this code indicates if the transaction is authorized by customer or not.

11. Merchant will send payment confirmation details to client. It will include following details:
a. OrderNo
b. ProductDetails
c. Total_Amount
d. ACCT: Bank account no. / Credit/Debit Card no. (Partial No.)
e. FirstName: Cardholder’s First Name
f. LastName: Cardholder’s Last Name
g. Street: Street name and number of cardholders
h. Zip: Zip code of cardholder
i. AuthCode: Authorization code (Success/Failure)

The challenge of E-commerce system with architecture discussed above is to securely exchange sensitive user information such as Credit/Debit card numbers or bank account numbers. This problem can be solved by encrypting information that is exchanged between the client and server computers. The encryption can be done with a shared secret key or by mathematical function that implements certain mathematical properties such as multiplicative group of integers modulo p.

The shared secret key method uses public key infrastructure and relies on a trusted third party for successful key exchange. This method can be computationally intensive. The mathematical functions on the other hand are easy to implement and execute. These functions are not as computationally intensive unless they are using large or very large numbers. This mathematical technique is at risk when one of the hosts where this technique is implemented is compromised. Hence, the authors selected shared secret key method.

The figure 4.3 depicts an e-commerce system with a layered architecture. In diagram below the e-commerce system is divided into 5 layers. User interface layer is responsible for UI of app on mobile device and web application server. Modified DS protocol layer is a new layer added to establish a secure connection with the server. Secure connection is established by sending a secret session key to server and later encrypting payment request and payment confirmation messages using this key while sending them. The part where secure connection is provided by modified DS protocol is highlighted.
The figure below shows the process of message passing between three entities: e-commerce app client, e-commerce application server and trusted third party. Here we have used concepts of public and private key protocols. The messages are encrypted by public keys and can only be decrypted by private keys. To begin the process of exchanging secret session key, the client sends a message to a trusted third party with its identification and identification of server with which it wants to establish a secure connection. The trusted third party then generates the key.
and sends it to client. The message also includes an encrypted message to server. The trusted third party also sends a synchronization sequence to both client and server that indicates the client wants to communicate.

**Fig. 4.4 Message Passing Sequence**

4.5 Conclusion

There are many different security requirements for different systems. For electronic voting system features like verifiability and auditability are important, however, for electronic healthcare management system trustworthiness, privacy and authentication are essential. For a mobile based payment system feature such as non-repudiation is crucial.
There are other factors also affecting security requirements of these applications. Also due to variation in working domains, applications work with a variety of data. For example, health record management system stores mostly biological details related to patient, whereas a payment system deals with data of financial nature such as shares and currency. Due to these variations in nature of information that these applications deal with, they have such different security requirements. The analysis of security requirements for electronic voting system and electronic healthcare management system and comparison of features with properties of protocol can tell that protocol is not suitable for these applications. However, analysis of security requirements of mobile based payment system shows that application is perfectly suitable for testing the proposed protocol.