國立交通大學

資訊科學與工程研究所

碩士論文

PLMN/IP 雙網架構的行動付款系統

A Mobile Payment System for PLMN/IP Dual Networks

研 究 生：江俊賢

指導教授：張明峰　教授

中華民國九十五年六月

PLMN/IP 雙網架構的行動付款系統

A Mobile Payment System for PLMN/IP Dual Networks

研究生：江俊賢　　Student：Chun Hsien Chiang

指導教授：張明峰　　Advisor：Ming-Feng Chang

國立交通大學

資訊科學與工程研究所

碩士論文

A Thesis
Submitted to Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Master
in

Computer Science

June 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年六月

PLMN/IP 雙網架構的行動付款系統

學生：江俊賢　　　　　　　　　　　　指導教授：張明峰 博士

國立交通大學資訊工程學系（研究所）碩士班

摘要

隨著電腦技術與通訊網路的進展，網路服務愈來愈多樣化。一般結合銀行帳戶或信用卡的網路即時付款系統缺乏行動性。雖然由行動通訊業者提供的行動付款系統，由於其無線網路涵蓋廣大的範圍，行動性較佳。但是結合行動通訊網路的行動付費大多透過簡訊或互動式語音服務(IVR)操作來達成付款確認。這種方式對使用者來說操作的複雜度較高，對業者來說建置系統的門檻也相對高出許多。

完善的行動付費服務要能同時滿足行動性、安全性、操作便利性、低建置成本等要求。在本論文中，我們嘗試解決上述的問題，設計一個以電話號碼為基礎但不需透過行動通訊業者支援的開放式行動付費平台。我們利用雙網認證機制對雙網設備認證，並使用 Kerberos 架構結合 Diffie-Hellman 金鑰交換演算法達到足夠的安全性。

系統主要功能有即時轉帳、匿名消費、按時計費。使用者須下載安裝軟體，在完成第一次的雙網認證後方可使用本行動付款服務。使用者只需輸入對方的電話號碼及金額即可完成轉帳的功能。在匿名消費時，只要兩個步驟即可完成交易。首先，使用者要向伺服器動態申請一個消費號碼，並告知店家。之後會收到要求確認的訊息，經由使用者再次確認後，交易即可順利完成。為了做到按時計費的功能，在本系統中我們使用一種有效率的計時資料結構。最後，我們測量了我們系統的效能以確保能符合使用上的需求。

A Mobile Payment System for PLMN/IP Dual Networks

Student: Chun Hsien Chiang Advisor: Dr. Ming-Feng Chang

Department of Computer Science and Information Engineering

National Chiao Tung University

ABSTRACT

With the progress of the computer and communication network development, novel network services have been provided regularly. A general network payment services using bank accounts or credit cards for real-time credit transfer provide little mobility support. Mobile payment systems provided by PLMN operators support user mobility better because of the wide coverage of their radio access networks. A mobile payment transaction usually involves a telephone call to an IVR or a short message exchange. From the user's point of view, it is inconvenient. From the payment service provider's point of view, the deployment cost is high since operations on the PLMN cost more than that on the Internet.

An ideal mobile payment service should provide mobility support, adequate security model, low operation complexity, and low deployment cost. In the thesis, we present a solution where user accounts are based on the telephone number and can be deployed on an open platform without any PLMN operator support. We adopt an authentication method for dual-network (PLMN/Internet) devices. In addition, we combine Kerberos architecture and Diffie-Hellman secret key agreement protocol to ensure the security of the system.

Our mobile payment system provides functions including real time credit transfer, anonymous payment, and charging by time. The users can download the software, and use the service after passing the first time E.164 authentication. After that, the user can do credit transfer just specify the payee's telephone number and the amount to be transferred. When shopping anonymously, there are two steps to do of the user. First the user need to request an one time shopping number from our server and then deliver it to the merchant in a secure manner. Second, the user has to confirm the payment as receiving the confirmation request from the payment server. In addition, service can be charged by time; this can be used for communicating sessions, such as viewing a video, which last a time period. To support this service, we use an efficient time-wheel data structure and improve it for load balancing for per-tick operations. Finally, we measure the maximum system capacity of our payment server.

誌謝

首先要感謝張明峰老師的費心指導下，得以完成此論文的著作及實作。在這兩年期間老師的督促及訓練獨立思考研究，讓我無論在作研究或者做人處事方面成長了很多，謝謝老師辛勞的教誨。

還有要感謝網路通訊實驗室孟達及弘鑫學長的指導及同學的關心，讓我可以在研究所這段期間的生活增添不少色彩。在這感謝你們，也很高興能夠認識你們！

最後謹在此將論文獻給我最親愛的家人及佩怡，由於你們的支持，讓我可以退伍之後重回學校求學的過程中一路順坦，沒有後顧之憂的完成學業。

Tables of Contents

List of Figures

1896

List of Tables

Chapter 1 Introduction

1.1 Overview

With the rapid progress of the Internet technologies, many new applications have been deployed. One of them is electronic payment (E-payment) [1] service which allows monetary value to be transferred between different accounts through the computer network. The existing electronic payment systems include online credit card, electronic cash and smart card [10]. The online credit card scheme is based on the traditional credit card payment but all information is exchanged across the Internet. The electronic cash scheme usually links with the banks. The users must first have a bank account and install the dedicated electronic cash software on a PC or PDA. The software can manage the deposit and withdrawing of the user's electronic cash from the user's bank account for payment. A smart card has a storage memory that can be used to store the monetary value for payment. Some emerging smart card is also equipped with a microprocessor to support cryptographic computation in a transaction. The purpose of these payment tools is to provide a convenient, secure, low cost and robust transaction platform to human beings. Apart from these payment tools, there is still another channel for payment － mobile payment which is usually deployed by the PLMN (public land mobile network) operators.

Mobile payment can be convenient to pay for financial transactions in our day life. However, current mobile payment systems almost rely on the services provided by GSM (global system for mobile communication) [6] networks, such as user authentication mechanism, IVR (interactive voice response) service and short message service. This raises the deployment cost of payment service provider because of the expensive core equipment of PLMN. To the consumer, each transaction usually involves a phone call or a short message transmission that would also increases per-transaction cost. To lower the cost described above, the whole payment platform should be extracted from the PLMN.

1.2 Related work

One of the most important issues of the mobile payment is security. As we mentioned before, the current mobile payment systems rely on GSM network. The security services in GSM include the user authentication and data confidentiality. User authentication prevents illegitimate MSs (mobile stations) from accessing the network resource. Data confidentiality reduces the risk of message intercepting at the access network. Each user has a secret key Ki to support these security services. Ki is stored on the network side and in the SIM (subscriber identity module) which is usually distributed at POS (point of sale) to the user.

To initiate the authentication procedure [6], the MS sends a request to the network. Then the network generates a random number and calculates a response value using the secret key Ki of the MS. After that, the network sends the random number to the MS as an authentication challenge. Finally, since no one except the legitimate MS and the network has Ki, the network could verify the response value calculated by the MS and determines if the MS identity is authentic. In addition, a secret key Kc would be derived from Ki and the random number to encrypt the transmission data to ensure the confidentiality. Since the Ki is only shared by SIM and PLMN operator, a third mobile payment service provider need to budget more cost for authentication and each payment transaction. However, if the mobile payment system is extracted from the PLMN and deployed in the Internet, the same security issues must be considered accordingly.

E.164 number defined by ITU can be used to identify user devices on the Internet. However, to avoid masquerade, a device's identifier (E.164) number must be authenticated on the Internet. An E.164-number-based user authentication method for VoIP [4] (voice over internet protocol) communications has been presented by Lin [6]. During the user authentication procedure, the user device is requested to make a GSM call to a caller-ID receiver, and the caller-ID receiver decodes the caller-ID of the incoming call. After verifying the caller-ID received, the server and the user device perform a secret key agreement algorithm to establish a shared secret key for user authentication hereafter, and the client can use the E.164 number as its identifier in the Internet. However, such procedure involved a GSM call which raises the overhead of each transaction. To reduce the operation overhead, the Diffie-Hellman is used to establish a secret key between the client and the server to enable future authentication based on cryptographic scheme.

Various types of mobile payment system have been proposed in recent years. Unfortunately, most of them were deployed by the PLMN operators which raise the deployment and transaction cost due to the reliance upon the GSM services. In addition, it is hard to provide multimedia session charging because of the limitation of the bandwidth in GSM/GPRS network.

1.3 Objectives

The thesis focuses on design and implementation of a mobile payment system that deployed over the IP network. It uses E.164 numbers as user's identifier and supports multimedia session charging. Besides, it allows users to transfer credit from one account to another and supports an anonymous payment model where the user's identifier is not released to the merchant. To take advantage of the system, the users just need to pass the E.164 authentication at the first time he uses the system. The users can pay to everyone that uses the system securely anywhere anytime. To deploy the system, the operators don't have to own very expensive hardware equipment for authentication purpose.

We also try to reduce operation and maintenance overhead by using a Kerberos-like architecture. Since Kerberos [14] is designed to authenticate the clients in a distributed network, it can be used to provide a single authentication server for multiple services. The mobile payment can be one of the services provided.

1.4 Summary

The remaining of thesis is organized as follows. Chapter 2 describes the essential knowledge background of the security schemes of our system. Chapter 3 shows the details of our system design. Chapter 4 presents the implementation issues. Conclusion is given in Chapter 5.

Chapter 2 Background

In this thesis, we design a mobile payment platform on the PLMN/IP network. All transaction messages will be transferred in the insecure IP network. Therefore, we first introduce the security services and mechanisms in the chapter. Second we explain those schemes we used in our system design.

2.1 Security Services

Security service [2] is a communication service used to protect valuable system resources. Security services can be divided into four categories, data confidentiality, data integrity, authentication, non-repudiation. Each of them has different mechanism to support.

Confidentiality provides protection of private message from interpreting by an eavesdropping party. The confidential service could be different of the protection level. The wider form of service protects all transmitted data between two nodes over a period of time. For example, in a given TCP connection, this protection prevents the contents transmitted in the connection being read by a third party until the connection is released. The narrower form of service is usually considered as the protection of a single message or certain fields in a message. System designer could use different level of protection for efficiency purpose.

Data integrity is a protection of data against modification by unauthorized entities. As well as confidentiality, integrity could be applied on a connection, a message or data fields of a message. In general, integrity assures the messages are received as sent. It detects not only the modification on message, but also duplication, insertion, reordering or replaying.

Authentication mechanism provides assurance of relevant identities to communicating parties. Usually there are two phase about authentication. For example, giving a communicating session, first the client request should be authenticated and the server should prove its identity to the client. In other word, authentication must assure that the two entities in a session are authentic. Second, the authentication service must guarantee that there is no third party can masquerade as a legitimate entity of the session.

Non repudiation is a prevention mechanism against the communicating parties to deny that it had played a part in a certain transaction. That is, when a message is received, the receiver can prove that the message is sent by the sender. Similarly, the sender can prove that the receiver received the message indeed.

2.2 Cryptographic Schemes

Cryptography is the most important security technique for the open network and is used to support the confidentiality and authenticity of the security services. The operation of cryptographic algorithm is based on two general principles: substitution and transposition. The substitution transforms the plaintext by replace a pattern that may consist of one or several bits or letters by another element. In transposition, patterns in a plaintext are rearranged. As figure 2-1 shows, cryptography is usually done by transforming the contents of a plaintext in an encrypt algorithm to ciphertext that can’t be decode by any other unauthorized party even if it is intercepted. Note that encrypt and decrypt algorithm is well known and the key used to encrypt need not be the same as the one used to decrypt. Following explains the terms in the figure.

1. Plaintext：This is the original message as the input of encrypt algorithm.
2. Key：A parameter of encrypt or decrypt algorithm.
3. Encrypt algorithm：A function that performs transformation on the plaintext base on the parameter key.
4. Ciphertext：The output of encrypt algorithm that is usually a block of binary data which is linkable to the plaintext.

5. Decrypt algorithm：A function that transform the ciphertext into the plaintext base on the parameter key.

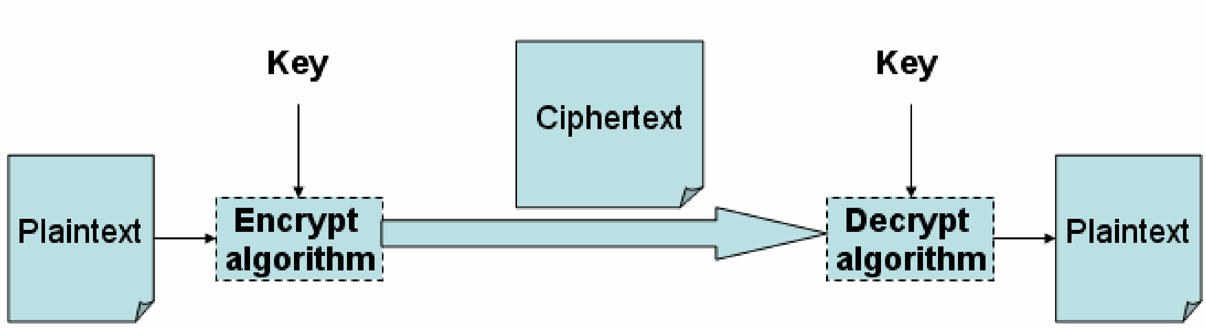


Figure 2-1 Cryptographic model

Cryptography can be classified into two different types by the number of keys used. If the key used to feed into encrypt and decrypt algorithm is the same, it is referred as symmetric encryption. In other word, if the key used in encrypt algorithm is different from the one used in decrypt algorithm, it is referred as asymmetric encryption.

Symmetric encryption encodes and decodes messages with a single secret key shared by all participants in a given communication session. It is also referred as the secret key, conventional or single key encryption. This encryption scheme is developed in the late 1970s and widely used of the two type of the encryption in many application packages. Current popular symmetric encryption algorithm is listed in the Table 2-1 [2].

Depend on the type of algorithm, the symmetric encryption algorithm could be divided into two categories block and stream algorithm. The block algorithm processes the plaintext and produced the ciphertext in a fixed-size block one by one. Thus the size of output may be larger than the size of input if the latter one isn’t divisible by the predefined block size. The stream algorithm processes the plaintext and produced the ciphertext word by word continuously. Thus the size of input and the size of output are the same. To use symmetric encryption, the parties in a communicating session must have the same key. Generally speaking, the common ways to distribute the secret key is as following. (Assume there are parties A and B)

1. A selects a key and physically delivered to B.
2. A third party C selects a key and physically delivered to A and B.
3. If A and B already share a key, each of them could delivered a new key encrypted in the old one to another.
4. If C shares a key with A and B separately, C could distribute a new key encrypted by the old one.

Table 2-1 Conventional encryption algorithms

Algorithm	Key Size(bits)	Block Size(bits)	Number of Rounds
DES	56	64	16
Triple DES	168	64	48
AES	256	128	14
IDEA	128	64	8
RC5	Max to 2048	64	Max to 255

1896

In contrast to using single secret key of symmetric encryption, public-key encryption involves a key pair composed of a private key and a public key. Public-key algorithms are based on mathematical axiom to ensure that a message encrypt by one of the key pair is only could be decrypted correctly by another key. The public-key encryption is a new scheme of cryptography in recent two decades. As the name shows, the public key of the key pairs must be well known to others and the private key must keep private. To employ the public key encryption, the essential scenarios are usually as following：

1. Assume two communicating parties A and B, each of them must have a key pair first.
2. Before transmitting message, A and B must exchange the public key of their key pair.
3. If A want to send a message to B and want B certain that the message is indeed from him, A encrypts the message with his private key first and then sends it to B.

4. While B receives the ciphered message sent from A, B could decrypt it with the public key of A correctly if the ciphered message is really sent from A.
5. If B wish to send message without being interpreted by others. B just sends the ciphered message that encrypted with A’s public key to A.
6. Since only A has its own private key, no other people could decrypt the ciphered message sent by B.

However the asymmetric encryption is not certainly more secure than symmetric encryption. The security of any encryption scheme depends on two factors:

1. The key length：As a rule, if the algorithm is the same, the longer length of the key the more security that encryption scheme could provide.
2. The Design of Algorithm：The protection level of an algorithm depends on the computational complexity to break a cipher.

Even though the public key encryption doesn’t necessarily provide a strong protection, it has some properties that could not be substituted by secret key encryption. The primary application is digital signature which addresses the shortcoming of conventional encryption to provide the non repudiation service. We will make a description of the digital signature in later section.

In general, the length of key of asymmetric encryption is longer than that one of symmetric encryption and the computational complexity of asymmetric encryption algorithm such as RSA is usually higher than symmetric encryption algorithm (e.g. DES, 3DES). A suitable mean to reduce the encryption time and keep the non repudiation service is to put them all in use. For example, the sender first generates a session key for encrypting. Then he sends a message consisted of the ciphered message and the ciphered session key that encrypted by the sender’s public key to the recipient. In this case, the session key provides confidentiality and integrity for the sender, the public key provides authentication and non repudiation for the recipient.

2.3 Approaches to Message Authentication

In addition to data confidentiality, message authentication assures the recipient that the message is authentic. The common method to prove the truth of identity is to show the knowledge of certain secret that bind with that identity. Both symmetric key and public key schemes could be applied on the method.

A simple way to provide message authentication is just encrypt the entire message or an important part of the message in a shared secret key. Since no one except sender and receiver possess the secret key, then only the genuine sender would be able to produce the ciphered message such that the receiver could decrypt it to original message. By this way, both confidentiality and authentication are assured. The Kerberos adopt this scheme to verify the request in the ticket granting server which will be introduced later. Besides, the public-key cryptographic could also be used to perform authentication capability.

The popular way to take advantage of public-key cryptography is to make a digital signature. A digital signature is a block of data that only the sender of a message can generate it. One can apply a hash function on the entire message and encrypt the hash value to produce a digital signature. Figure 2-3 illustrates this work flow. Notice that in this situation, the key and key' composed a key pair of asymmetric encryption.

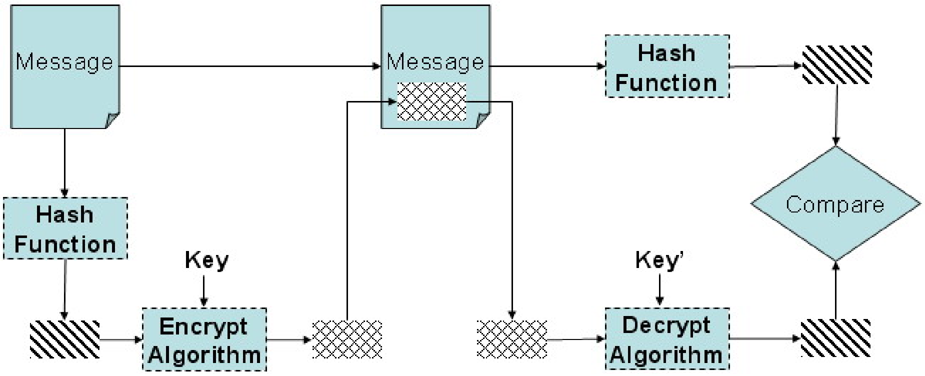


Figure 2-2 Digital signature work flow

2.4 VoIP Authentication

VoIP authentication is a dual network authentication mechanism. It assumes that no one can impersonate a fake E.164 number through the PLMN/PSTN network. Since impersonation of E.164 number in PLMN/PSTN network is impossible, the caller identity extract from a call or short message can be trusted as the E.164 number of user. Base on the concept, VoIP authentication utilize RS232 interface to retrieve the caller id of client and compare with the one that the client claimed through the IP network.

There were centralized and decentralized model of the VoIP authentication. We adopt the centralized mode to integrate the VoIP authentication with our system. To initialize the authentication procedure, the client sends a SIP request REGISTER encapsulate its phone number to the server. A SIP 401 response and a REFER will be sent by the server. When the client receive the REFER, it extract the phone number from the refer-to header field and place the call immediately. The GSM call will be routed to a caller id receiver that controlled by the server through RS232 interface. After the call disconnected by the server, the client sends a NOTIFY message to the server to indicate that the GSM call is complete. If the caller id received from the caller id receiver is the same with that one encapsulated in the REGISTER, then the authentication is successful. Figure2-4 illustrates the signal flow.

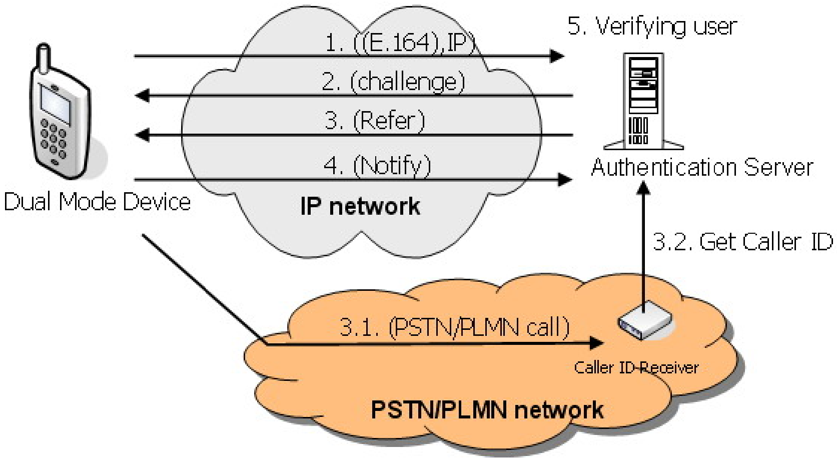


Figure 2-3 Procedure of the VoIP authentication

2.5 Diffie-Hellman Key Agreement Protocol

In the Internet communication, a shared secret key is required to support end to end secure message transmission. Since the VoIP authentication provides a mechanism to authenticate the E.164 number in the Internet, it involved a GSM call in the procedure, which is not efficient and practical for our mobile payment system. To improve the efficiency, we authenticate the E.164 number via using the symmetric encryption algorithm after the first-time VoIP authentication. The Diffie-Hellman [2] algorithm is proposed to build a shared secret key among parties of a communication session without prior share secret. It has two system parameters P and G. The P is a prime number and G is a primitive root of P. The system must allow all the users to access these parameters.

The security of the Diffie-Hellman algorithm is based on the mathematical complexity of logarithm rather than computation of exponentials modulo a big prime number. Figure 2-5 shows an example of Diffie-Hellman protocol. Assume that there are two users A and B. First they generate a private value XA and XB. Then they exchange a public value YA and YB. When both of user A and user B receive the remote public value, the secret key K could be calculated by the local private value and remote public value.

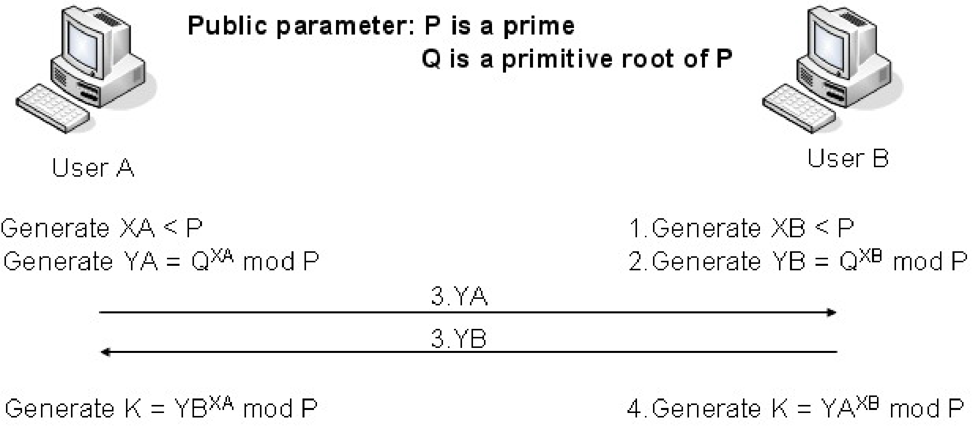

Public parameter: P is a prime
Q is a primitive root of P
User A
User B
1.Generate XA < P
2.Generate $YA = Q^{XA}$ mod P
1.Generate XB < P
2.Generate $YB = Q^{XB}$ mod P
3.YA
3.YB
4.Generate $K = YB^{XA}$ mod P
4.Generate $K = YA^{XB}$ mod P

Figure 2-4 Diffie-Hellman key agreement protocol

2.6 Kerberos Authentication Service

Kerberos is an authentication service that addresses the problem of authorization to users at workstations wishing to access services on servers throughout a distributed network. To reduce operation and maintenance overhead, it is effective to apply Kerberos to the authentication server and the payment server of our mobile payment system. The Kerberos is based on the symmetric encryption to provide the basic security services of confidentiality, integrity and authentication.

The components of Kerberos included authentication server, ticket granting server (TGS), application server and the client. The authentication server is responsible for handling the ticket request sent from client and generating the ticket-granting ticket (TGT) that encrypted in the client's secret key that derived from its password. Each client's password is store in the database of authentication server in advance such that there is no problem of distribution. The TGS uses the TGT to verify the authenticator claimed by the client and delivers the ticket for application server too. Finally, the application server provides services to clients throughout a distributed network.

The Kerberos protocol proceeds with several message exchanges. In the first exchange, the client sends a request for a ticket for the desired application server. The reply with TGT encrypted in the client's secret key is sent. The TGT is encrypted in secret key shared by authentication server and TGS. Thus no one except TGS could decrypt the TGT. In the second exchange, the client sends a request to the TGS with TGT generated by the authentication server. The reply with the ticket for desired server is encrypted in the session key from the TGT. Thus no one can get the ticket without having the secret key of the original user.

2.7 Mobile Payment

Mobile payment is a payment made through a mobile device, such as a cell phone, smart phone or PDA. Using mobile payment, a person can pay for an item in a store more efficient than traditional payment tool likes cash or checks. In this section, we introduce three different mobile payment systems Paybox, GiSMo and Sonera then we give a brief comment about them.

Paybox is an operator-independent payment platform and its main stockholder is Deutsche Bank. The consumers that using Paybox pay directly to the bank account of the merchant. The payment service provider plays a neutral role in the transaction. The merchant need not be special to the Paybox but should bear the 3% fee for every transaction cost. Figure 2-6[3] illustrates the whole payment flow of Paybox. (1)To initialize a transaction, the user should send the phone number to the merchant. (2)Then the user merchant sends the phone number and amount of the transaction to the Paybox server. (3)The user would be notified by a GSM call placed by the Paybox server for confirmation. (4)Finally, the user enters the PIN code to confirm the transaction. Depend on the payment flow there are two disadvantages we could conclude. One is that the Paybox does not provide anonymity of phone number for the users. Another is the additional cost of the GSM call placed by the Paybox server in each transaction.

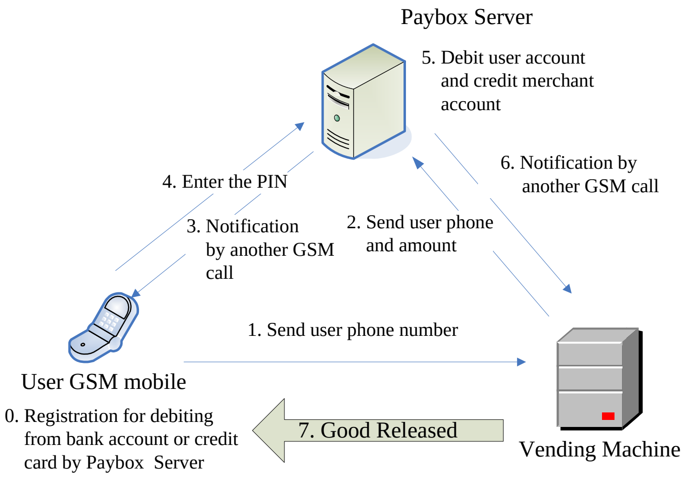

Paybox Server
5. Debit user account and credit merchant account
6. Notification by another GSM call
4. Enter the PIN
3. Notification by another GSM call
2. Send user phone and amount
1. Send user phone number
User GSM mobile
0. Registration for debiting from bank account or credit card by Paybox Server
7. Good Released
Vending Machine

Figure 2-5 Payment flow of Paybox

GiSMo is a supplementary business of Millicom International Cellular. The operator links the account identity created by the users to his phone number and his bank account or credit card. As figure 2-7 [3] shows, (1) the consumer should send the order details received from Internet and his account identity. (2) The GiSMo server then send a short message with a dynamic code to the phone number that bound to the account identity. Only that one who owns the legitimate SIM card with the phone number would get the dynamic code. After that, (3) the consumers should send the dynamic code by a PC to the GiSMo server to confirm the transaction. Finally, (4) the GiSMo server will debit the user account and credit the merchant account after verifying the dynamic code. In the payment flow, the order details and dynamic code should be sent from a PC such that reduce the mobility.

In addition, the GiSMo server is vulnerable to active attack because it relies on the dynamic code to authenticate each transaction. Some one who does not own the legitimate GSM phone could initialize the transaction by sending a valid account identity and order detail repeatedly.

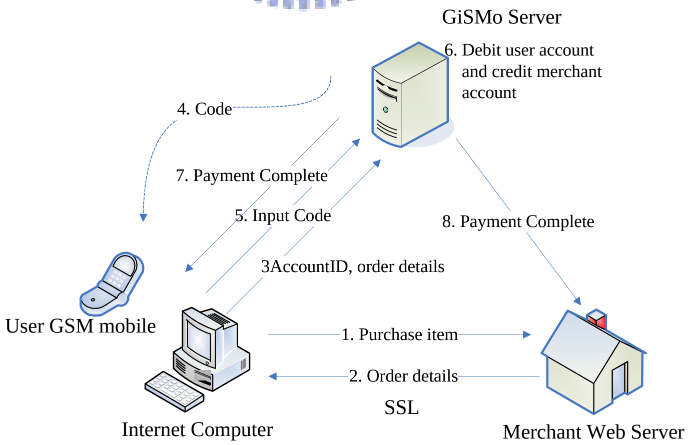


Figure 2-6 Payment flow of GiSMo

Sonera mobile pay is deployed mainly in Sweden and Finland. It is a platform that deployed by the PLMN operator. Sonera provides prepaid and postpaid for the users. In addition, it allows the consumer to pay by the credit card. The figure 2-8 illustrates the payment flow. (1)First, the consumer sends a short message with the amount of the transaction to Sonera. (2)Then the Sonera generates a one time dynamic payment identity and sends it to the consumer by a short message. (3)Then the user transfers that payment identity to the merchant. (4)After that, the merchant sends the payment information including the dynamic payment identity to Sonera. (5)Finally, the Sonera server processes the credit transferring after verifying the dynamic identity and the payment information.

There are two disadvantages we can observe. First, the whole payment flow consists of two transmissions of short message such that increase the cost of each transaction. Second, it is hard to provide the session charging because the amount is decided at first step.

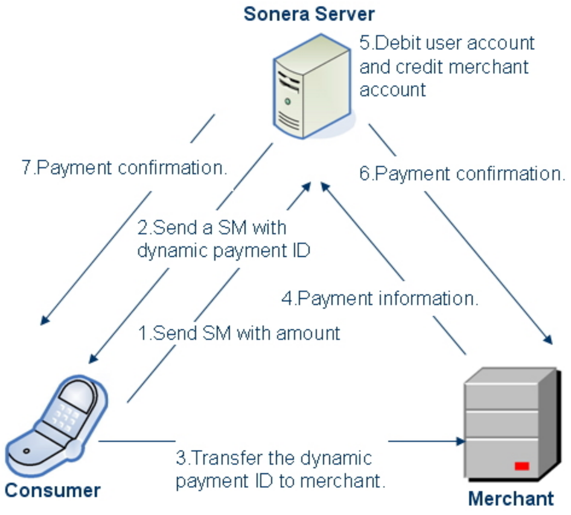


Figure 2-7 Payment flow of Sonera.

2.8 Time wheel data structure

Since our mobile payment system provides a service of charging by time, we need a timer module to support this service. Conventional algorithms to implement an operating system timer module take O(n) time to start or maintain a timer, where n is the number of outstanding timers. The time wheel data structure [7] uses a circular buffer to store the timers and takes O(1) time to start, stop and maintain timers within the range of the wheel.

The model of time wheel has four component routine.

- StartTimer：The client calls this routine to set a timer into the module. The client must at least supply an interval indicating how long does the timer count and a expiry action that the timer module should invoke when the timer is expired.
- StopTimer：The client calls this routine to stop a timer in the module.
- PerTickBookkeeping：The routine is invoked every T units of time, which T is the resolution of the timer module. It is responsible for checking the outstanding timer, if there are timers expired, the corresponding expiry action will be performed.
- ExpiryProcessing：This routine does what the expiry action specified.

The data structure of time wheel consisted of several array which hierarchically present the level of time. For example, if the timer module could present a maximum timer as one day. Thus there should be three arrays A_H, A_M, A_S to compose the hierarchy structure. The three arrays stand for the wheel of hour, minute, second respectively and could be considered as a virtual clock in the timer module. Therefore, the A_H must have twenty-four elements, A_M must have sixty elements and A_S must have sixty elements. Each array has an index to indicate the current time of the virtual clock. As Figure 2-9 illustrates, if the client starts a timer which will expire in 50 min and 45 s into the timer module and the current time of the virtual clock is 10 h, 24 min, 30 s. First the absolute time of that timer will be calculated as 11h, 15m, 15s and a timer tag will be inserted into at the 11th slot of A_H. The PerTickBookkeeping will increase the index of A_S at every tick.

Even there is no timer tag in the timer module there should be a default timer tag at the 0th of the A_S and A_M to update the index of A_M and A_H respectively. As the index of A_H become 11, the timer tag will be moved to the 15th elements of A_M. In the same way, the timer tag will be moved to the 15th elements of A_S when the index of A_M becomes 15 later. Finally, the ExpiryProcessing will be invoked when the index of A_S become 15.

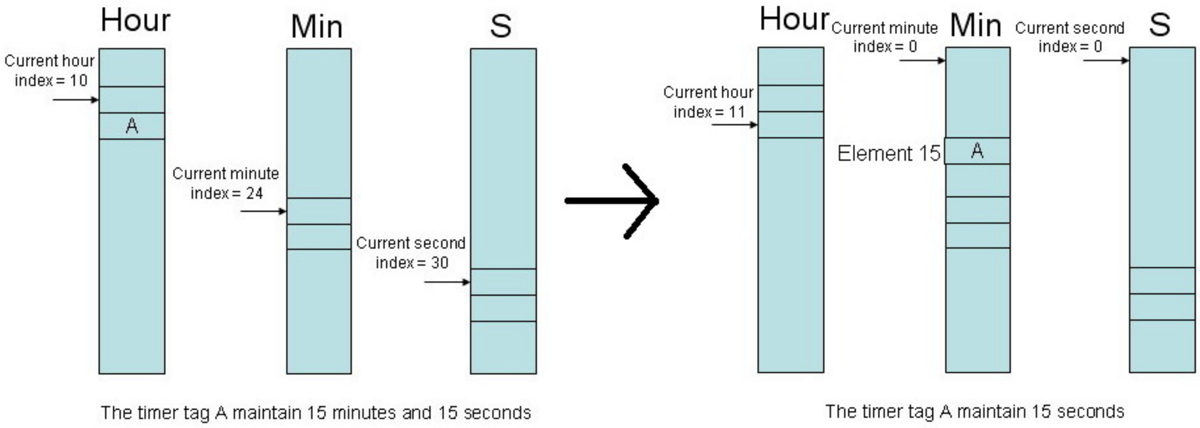

Hour
Min
S
Current hour index = 10
A
Current minute index = 24
Current second index = 30
The timer tag A maintain 15 minutes and 15 seconds
Hour
Current minute index = 0
Min
Current second index = 0
S
Current hour index = 11
Element 15
A
The timer tag A maintain 15 seconds

Figure 2-8 Concept of Time Wheel

2.9 Summary

Most of the current mobile payment systems rely on the GSM services such as authentication, SMS and IVR, which will increase the transaction cost and deployment cost to the user and the mobile payment service provider respectively. We will design a payment platform over the PLMN/IP dual network to reduce the cost and integrate the VoIP authentication, Diffie-Hellman and Kerberos to build up the security capabilities. Furthermore, we improve the Time Wheel data structure in our system to balance the per-tick operation.

Chapter 3 Design of Our System

3.1 Overview

Our mobile payment system accounts user’s telephone number as his identity. We assume each user uses a dual mode device (DMD) with a legitimate SIM card for the system. Also, each user has credits in the payment system. There are two kinds of servers in our system. One of them is the authentication server that performs the VoIP authentication to the requesting users and builds a secret key for them. Another server is the payment server which maintains the accounting information for subscribed users and verifies their authority. They can pay easily through our system if the merchant has a registered dual mode device.

We emphasize the convenience and security of the system, each financial transaction among users and servers can be done in real time and in everywhere. Our system supports anonymous payment capability that allows users to request payment from anyone who has a one time credit number (OTCN).

Furthermore, our payment system enables the merchant to charge a user for a service in a period of time. To support this function, we use an effective timer wheel data structure and improve it to balance the per-tick operation.

3.1.1 Architecture

As we mentioned before, the whole system consists of two types of server and the DMD. The system architecture is illustrated in Figure 3-1. Each user can get the service through the DMD. The main protocol of our system is based on SIP [11] such that it can support mobility easily. The payment server acts as a SIP registrar which maintains the location record of each DMD. For flexibility purpose, our payment system could interwork with additional e-commerce platform on world wide webs to provide further service such as GiSMo. Our system is deployed as a client-server model. Every request message must be sent to payment server for authentication purpose. Only those authenticated request could be processed and forwarded by the payment server. Therefore the payment server could be considered as a SIP proxy in VoIP network. On the consumer side, one can pay by DMD directly at counter or companion using a WWW browser on Internet. On the merchant side, one can use a POS or a web server to charge the user through our system if they can negotiate with the DMD by SIP message.

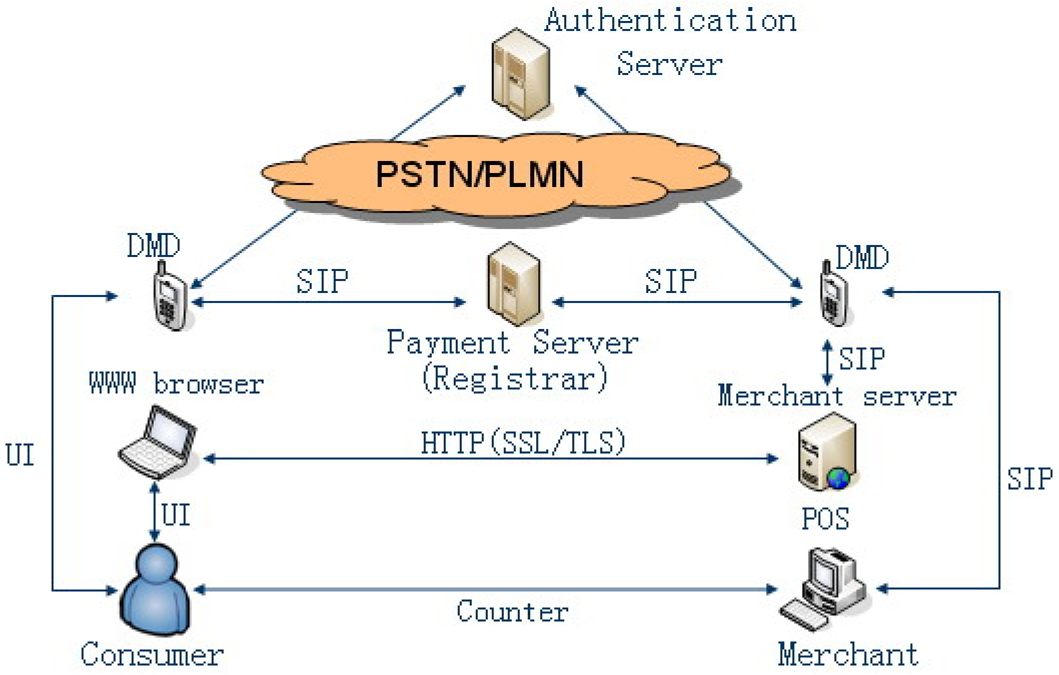


Figure 3-1 Architecture of our mobile payment system

3.1.2 Components

The components of our mobile payment system are described as follow.

- Dual mode device：In the thesis, a DMD is defined as a device that could access the GSM and WLAN network. It is most likely a PDA phone or smart phone.
- POS system：A point of sale system used by the merchant at counter to support the payment of transactions.
- Authentication server：Each user must be authenticated by the authentication server to before he can obtain the subscribed services. It is the first entry point in our system. Upon receiving a request from the client, it initiates the VoIP authentication procedure immediately. In the process, the Diffie Hellman algorithm is used to establish a session key between the server and the client. In addition, the authentication server generates a ticket for the client. The ticket includes several necessary data fields for the later verification of the payment server. Finally, a reply encapsulates a random secret key and the ticket is delivered to the client.
- Payment server：The payment server manages the account information of users and processes all payment requests. Three databases account, location and one time credit number information are maintained by the payment server. Account database store the subscriber data for each subscribed user. Location database contains the binding of identifier and IP address for registered user and is used to find out the destination of each request. The payment server behaves like a gateway between the users and the financial network. Each transaction request will be verified by the payment server at first. Besides, it creates the dynamic bindings for users to support anonymous payment.

3.1.3 Functionality

Our mobile payment system has three basic functionalities of credit transferring, event charging and charged by time respectively. The credit transferring capability enables the users to send his money to another user. The payer just needs to know the phone number of payee and decides the amount to transfer. As the figure 3-2 shows, transaction is similar to the SIP non invite transaction. As receiving a request, the payment server transfers the credit from the payer to the payee and forwards the request to the payee to inform that a payment is incoming. If the payee's DMD is active, the payment server will receive a confirmation from the payee and forward it back to the payer. If the payment server does not receive a confirmation in a certain period of time (TimerA), it will notify the payer automatically. In this case, some exception handling manners could be chosen such as sending email or and a short message to the payee.

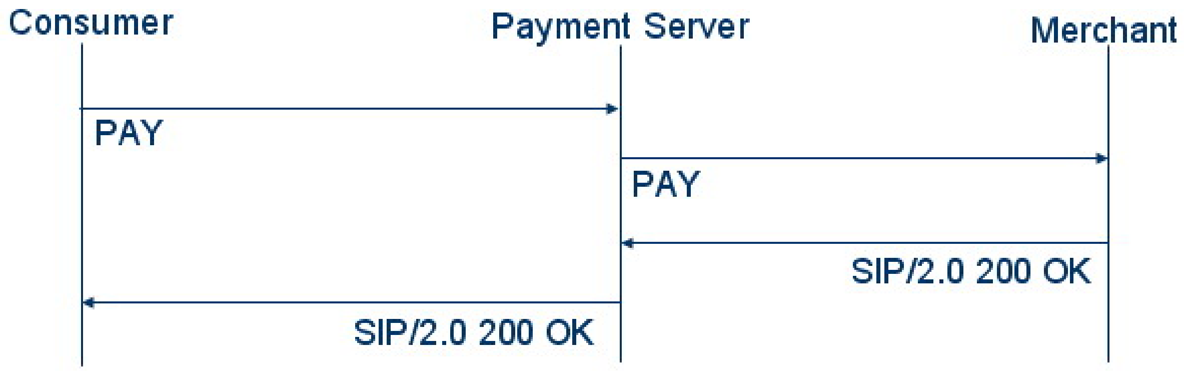


Figure 3-2 Message flow of credit transfer

The event charging means that the amount of the payment is given before the transaction terminated. Most of transactions in our day life are based on this model such as shopping in a convenience store, pay the fees for a taxi or pay the bill in a restaurant. In figure 3-3, to initial the transaction, the merchant's DMD sends a RECEIVE request to the payment server to indicate that it want to request money from the consumer. The consumer's DMD respond a 180 message to indicate that the request has been received but the consumer has not confirmed. After the consumer makes a confirmation, the DMD will send a 200 response to the payment server to agree this request. Finally, the amount of this transaction will be transferred from the consumer to the merchant account immediately when the payment server received the 200 response. The ACK is used to inform the consumer that the merchant has received the confirmation.

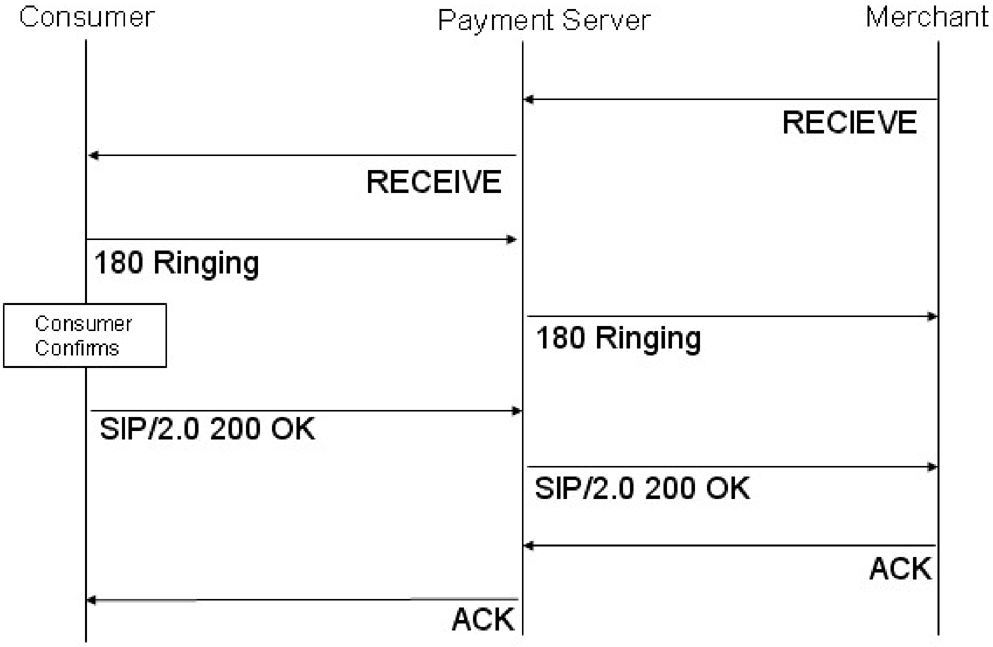


Figure 3-3 Message flow of event charging

ES 1896

The charged by time function enables the merchant to charge the consumer in a communicating session. The amount of transaction depends on how long does the session exist. All message flow of this service is similar to the event charging apart from the BYE message at the end of session. After the payment server receives the ACK message sent from the merchant, it starts two timers. One timer counts for the period of time of the session, the other is set as the maximum time that the session could exist. The value of maximum timer depends on the balance of the consumer and should be dynamic adjusted if the consumer served by many sessions at the same time. In addition, the session not only could be terminated by the consumer or the merchant at anytime, it also could be terminated by the payment server when the maximum timer expired. Figure 3-4 illustrates the whole message flow.

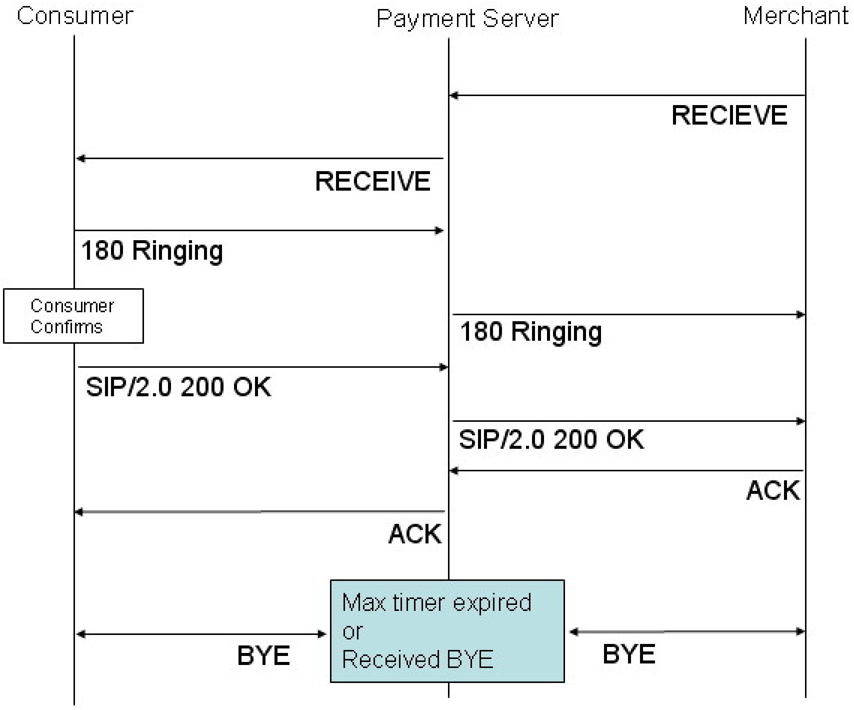


Figure 3-4 Message flow of charged by time

1896

3.2 Authentication

Our solution adopts a Kerberos-like architecture, for which we could obtain several benefits. First, the authentication and access control capabilities could be provided in independent entities to reduce the complexity of each entity. Thus, the maintainability would be better accordingly. Second, when we integrate many kinds of application servers to our system, there might be different format of authenticator corresponding to each application server. Therefore we could diminish the overhead of message exchange between the clients and the authentication server by perform authentication and access control separately.

The authentication mechanism of our mobile payment system can be divided into two phases. First, the VoIP authentication procedure combining with Diffie-Hellman algorithm is performed and a secret key will be established for the DMD at the end. Second, we use the authentication mechanism that is similar to Kerberos authentication service for subsequent message transmission. We assume a secret key K_{PS} is shared between the payment server and the payment server beforehand.

3.2.1 The Authentication Server

The authentication server maintains a database of the user's identifier and its secret key. As Figure 3-5 illustrates, it responds in the first phase authentication of the E.164 number claimed by an unauthorized user via VoIP authentication procedure. At the end of VoIP authentication, the authentication server calculates the secret key K by Diffie-Hellman algorithm and store K for the DMD.

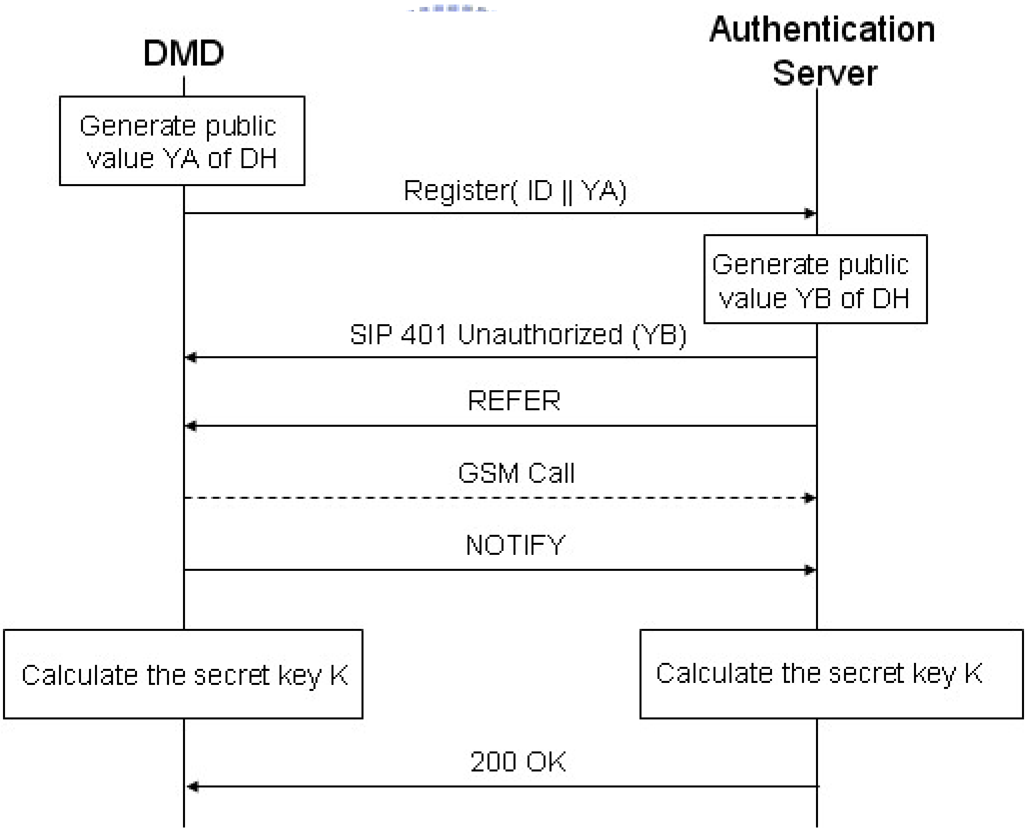

DMD
Authentication Server
Generate public value YA of DH
Register(ID || YA)
Generate public value YB of DH
SIP 401 Unauthorized (YB)
REFER
GSM Call
NOTIFY
Calculate the secret key K
Calculate the secret key K
200 OK

Figure 3-5 Message flow of first phase authentication

At the second phase, the authentication server generates a ticket for those users that has been authenticated in the first phase. Figure 3-6 depicts the message flow and data format of ticket request. To request a new ticket from the authentication server, the DMD send a plain request with its identifier and IP address to the authentication server. The data filed “Times” stores the expiration time of the ticket. The nonce is used to perform the mutual authentication to the authentication server. $K_{C,PS}$ is the secret key used for the DMD and the payment server. K_{PS} is the secret key shared by the authentication server and the payment server. The ticket is encrypted in K_{PS} and $K_{C,PS}$ is encrypted in K to ensure that the content of ticket is unalterable and no one could retrieve $K_{C,PS}$ except the legitimate DMD.

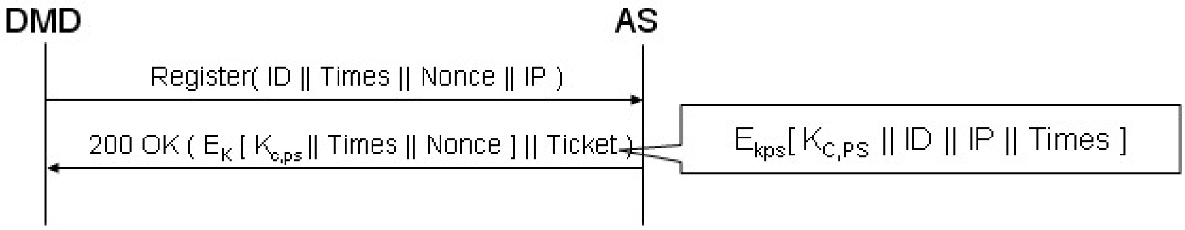

DMD
AS
Register(ID || Times || Nonce || IP)
200 OK (E_K [$K_{c,ps}$ || Times || Nonce] || Ticket)
E_{kps}[$K_{C,PS}$ || ID || IP || Times]

Figure 3-6 Message flow of ticket request

3.2.2 The Payment Server

The payment server behaves like the ticket granting server of Kerberos authentication service. Figure 3-7 illustrates the procedure of authenticator verification. After the DMD obtains a ticket from the authentication server, the ticket must be encapsulated in the subsequent request sent by the DMD. To verify the incoming request, the payment server first decrypts the ticket encapsulated in the request and using the $K_{C,PS}$ stored in the ticket to decrypt the authenticator. Then the payment server compares the identifier and IP address in the authenticator with those in the ticket. Since no one can provides the correct authenticator other than the authenticated user and the content of ticket is unalterable, the request is legitimate only if the identifier in the authenticator and IP address is matched with those in the ticket. In this manner, we can provide protection against masquerade attack of user identifier and IP address.

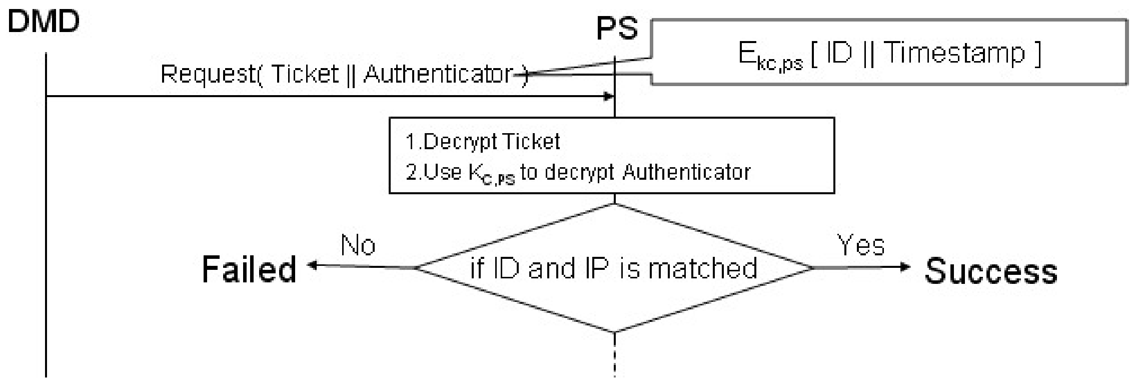

DMD
PS
Request(Ticket || Authenticator)
$E_{kc,ps}$ [ID || Timestamp]
1.Decrypt Ticket
2.Use $K_{c,ps}$ to decrypt Authenticator
No
Failed
if ID and IP is matched
Yes
Success

Figure 3-7 Verify a request

3.3 Using the Time Wheel in Our Solution

Although using original time wheel algorithm we can maintain timer in O(1), there has a computational burst problem of moving too many timer when the highest level wheel updates. As Figure 3-8 illustrates. The average processing load of each time slot of A_H, A_M and A_S are 3600, 60 and 1 respectively. This means there are too many timers to be moved when A_H updated and nothing to do when A_S updates.

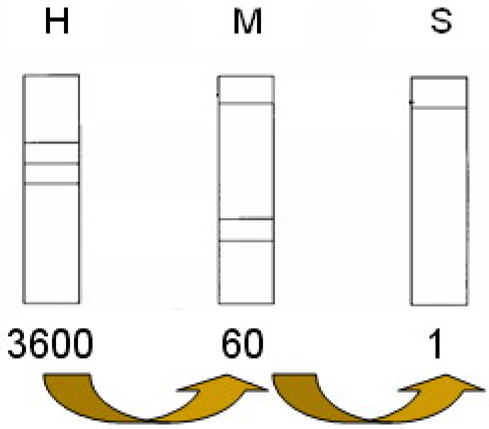


Figure 3-8 Unbalanced processing load of Time Wheel

Consider the same example in previous chapter (chapter 2). We made an improvement on the original time wheel. An overlap array is appended at the end of A_M and A_S to buffer the timer tags that should be triggered in the next time slot in the next level of wheel. To balance computational load, we move partial timers from next array to the overlap array smoothly as figure 3-9 shows. Assume that the current index of A_S, A_M and A_H are two, two and one respectively and there are sixty timers in the 3th slot of A_M, six hundred timers in the 2th slot of A_H. We first calculate a rate to move these timers partially. Since the A_M has sixty timers in the next time slot (3th), the rate should be (1,1) that means move one timer per minute. Identically, the rate of A_H should be (1,6) that means move one timer per six minutes. When the A_S updates, we first move one timer from the 3th slot of A_M, then check if we need to move timer from the 2th slot of A_H.

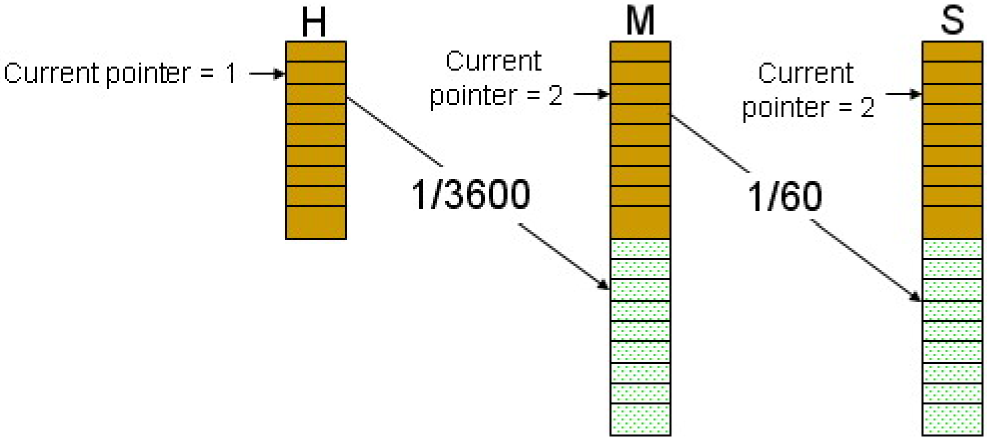


Figure 3-9 Concept of our improvement

3.4 Mobility Support

To support device mobility, the payment server act as a SIP registrar that store routing record of each mobile device. Each DMD should register its location to the payment server after it is authenticated. The binding of the location and phone number helps the payment server to route a request to the destination. However, the binding for each DMD is not permanent. It will be out of date after expire time specified by the DMD. Thus the DMD should register to the payment server periodically or when its location information is changed.

3.5 Anonymity

3.5.1 One time credit number (OTCN)

For keeping personal privacy, our mobile payment system allows the consumers to pay without showing his identifier. The OTCN is a unique random string regulated by the payment server to support anonymous charging. The consumer should request an OTCN from the payment server for each anonymous transaction. For security consideration, an OTCN will become invalid after a period of time and only can be used once. To perform credit transfer anonymously, the work flow is illustrated in Figure 3-10(a). A parameter Conceal indicating the anonymity is add to the Call-Info header. The payment server will conceal the payer’s true identifier from the payee if parameter Conceal is presented.

To use OTCN in event charging and charged by time service, the work flow is illustrated in Figure 3-10(b). After the DMD obtains an OTCN, the consumer must deliver it to the merchant in a secure manner. Thus, only the right merchant could have a valid OTCN. Then the merchant issues a charging request with the OTCN to the payment server. When the payment server extracts the OTCN, it uses the OTCN to find the correct consumer and forwards the charging request to the consumer’s DMD. As the consumer receives a charging request, they don’t need to check the source of the charging request. He only has to confirm the amount of the request. In this manner, the same malicious attack may be happened as previous description.

The malicious merchant could try different OTCNs to cheat the payment server to forward its charging request to the victim. This may confuse the user that received the malicious charging request. It can be addressed by applying a policy to the payment server such as blocking the DMD if the payment server detects that its charging request has retried with different failed OTCN several times in a period of time.

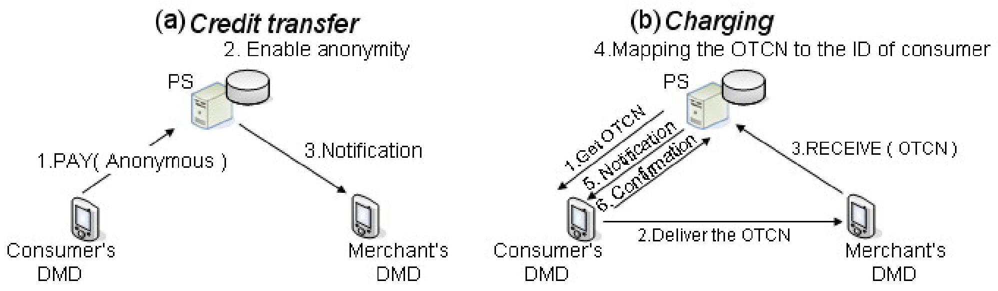
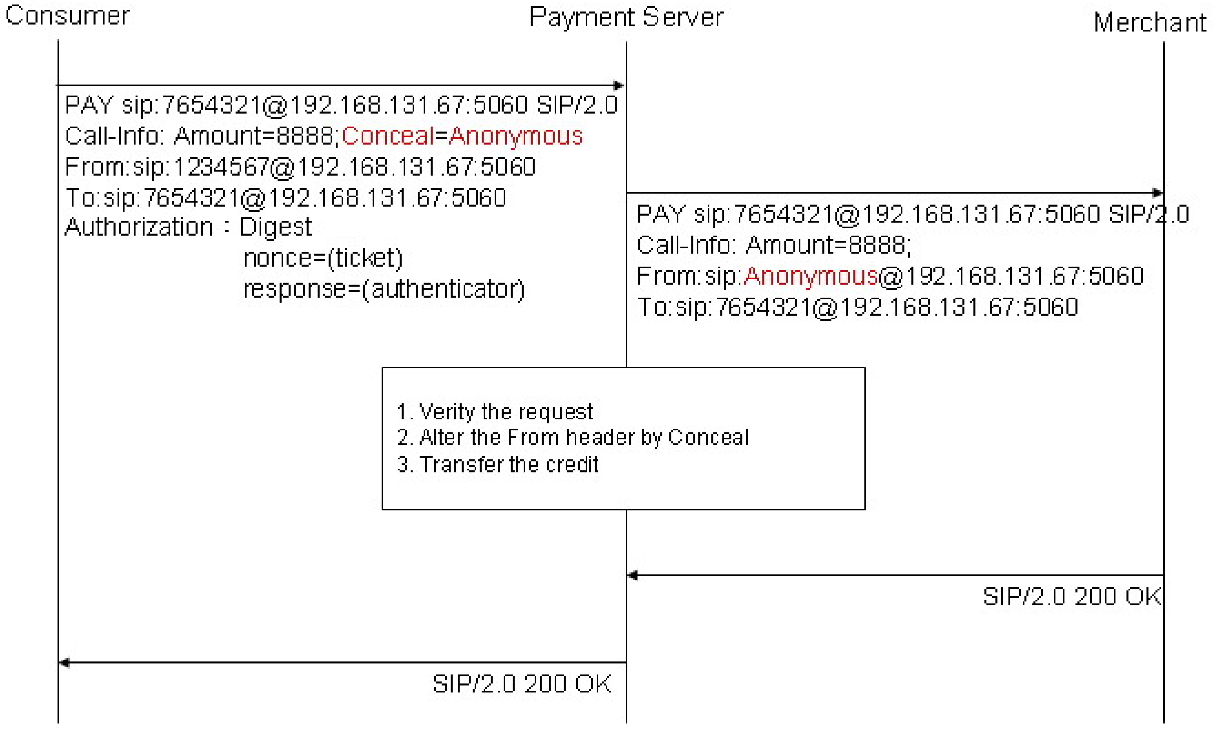


Figure 3-10 Work flow of anonymity

3.5.2 Message flows

Figure 3-11 illustrates the message flow of anonymous credit transfer.



```
PAY sip:7654321@192.168.131.67:5060 SIP/2.0
Call-Info: Amount=8888;Conceal=Anonymous
From:sip:1234567@192.168.131.67:5060
To:sip:7654321@192.168.131.67:5060
Authorization : Digest
                nonce=(ticket)
                response=(authenticator)
```

```
PAY sip:7654321@192.168.131.67:5060 SIP/2.0
Call-Info: Amount=8888;
From:sip:Anonymous@192.168.131.67:5060
To:sip:7654321@192.168.131.67:5060
```

1. Verity the request
2. Alter the From header by Conceal
3. Transfer the credit

```
SIP/2.0 200 OK
```

```
SIP/2.0 200 OK
```

Figure 3-11 Message of anonymous credit transfer

Figure 3-12 illustrates the message flow of anonymous charging.

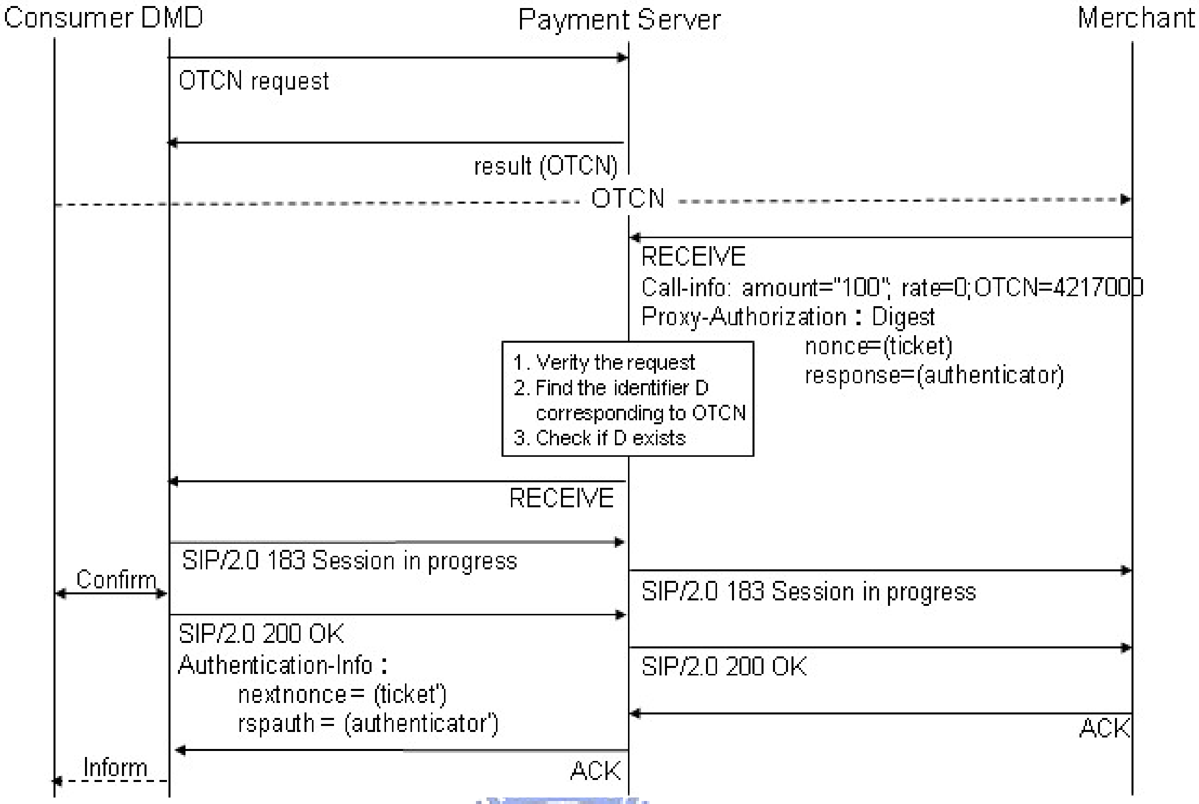


Figure 3-12 Message flow of anonymous charging

Chapter 4 Implementation

4.1 Implementation of Time Wheel

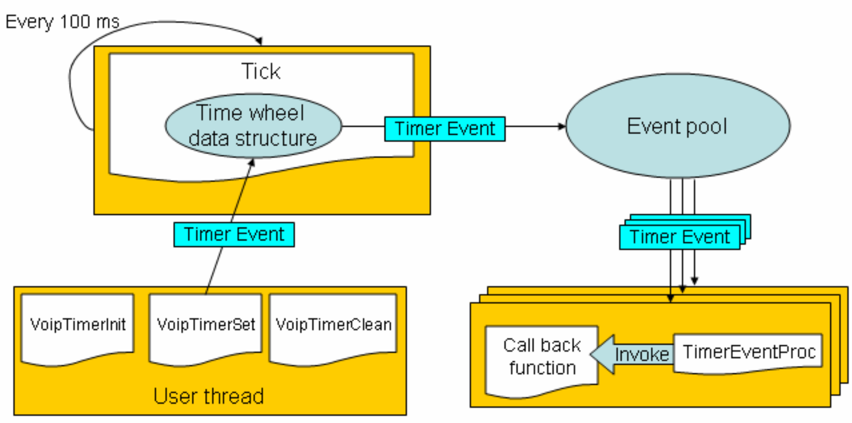


Figure 4-1 Model of our timer module

Figure 4-1 shows the design model of the time wheel. We provide three APIs for the user thread. At least two threads are activated after the user calls the VoipTimerInit routine. One of them executes the Tick function every 100ms, the other execute the TimerEventProc. Following explains the function of each routine.

- VoipTimerInit：Initialize the timer module.
- VoipTimerSet：Add a timer event into the time wheel data structure.
- VoipTimerClean：Delete a timer even from the time wheel data structure.
- Tick：Update the time wheel data structure, and check if there are timer events expired. Move the expired timer events to the event pool. A single thread running across this function.
- TimerEventProc：Retrieve the expired timer event from the event pool and invoke the call back function specified by the user. Multithreads running across this function.
- Call back function：The expiry action that should be performed when the timer is

expired.

4.2 Implementation of Authentication Server

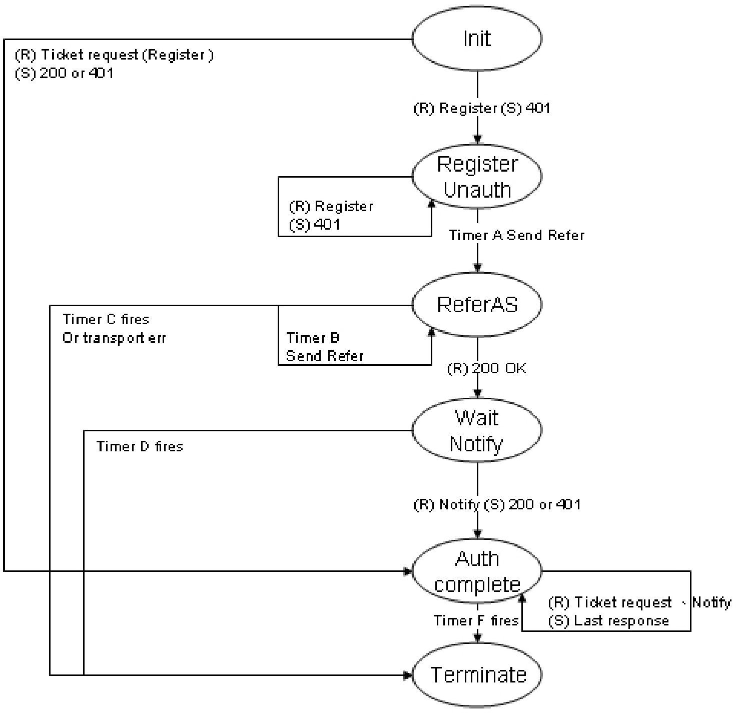


Figure 4-2 State diagram of each transaction in the authentication server

Figure 4-2 shows the state diagram of each transaction in the authentication server which regards the SIP REGISTER message as a request for authentication procedure.

1. When receiving a REGISTER message, the authentication server sends a 401 response to the client and the state changes to RegisterUnauth. The REGISTER message may be retransmitted by the client if the 401 response is lost. The state RegisterUnauth is designed to process these retransmissions of REGISTER.
2. After timer A expired, the state will change to ReferAS which retransmit a SIP REFER to the client until receiving a response. If the authentication server has not received the response after time C expired, the transaction will be terminated. The state WaitNotify is used to wait a NOTIFY sent from the client. The NOTIFY indicates that the client

have already placed the GSM call after receiving the REFER. The NOTIFY should be sent before the expiration of timer D, if not, the transaction will be terminated too.

3. Upon the authentication server receives the NOTIFY in time, it will generate a ticket for the client and enter the next state AuthComplete which will automatically jump to Terminate after timer F expired.
4. In addition, if the client just wishes to renew the old ticket, it will show its old one in the REGISTER message. After verification the ticket, the state will become AuthComplete directly. In this case, the authentication server will handle the retransmission of REGISTER until the timer F expired.

4.3 Implementation of Payment Server

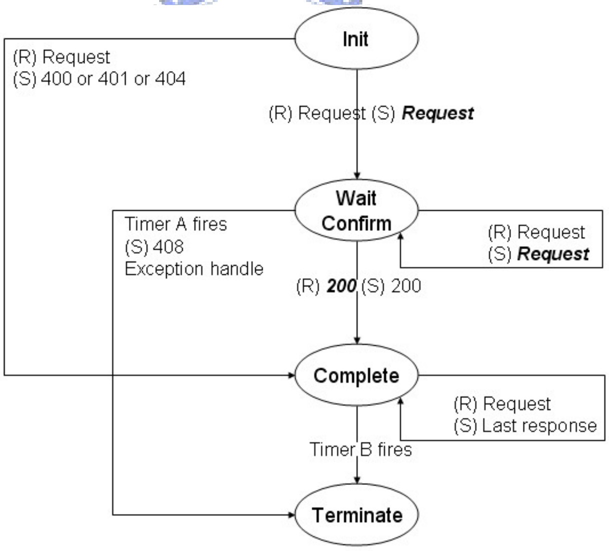


Figure 4-3 State diagram of credit transfer

The Figure 4-3 illustrates the state diagram of credit transfer service. The bold and italic word means that the message is sent to or received from the payee and the normal form is for the payer.

1. After authenticating the request of credit transfer, the payment server simply check the balance of the payer and check if the payee is registered to the system. If balance is

not enough, a 400 response will be sent to the payer, else if the payee is not found, the payment server sends a 404 response to the payer.

2. After transferring the credit from payer to the payee, the request will be forwarded to payee. A 408 response will be sent back to the payer if timer A is expired. In this case, it indicates that the payee is not online and this exception must be handled.

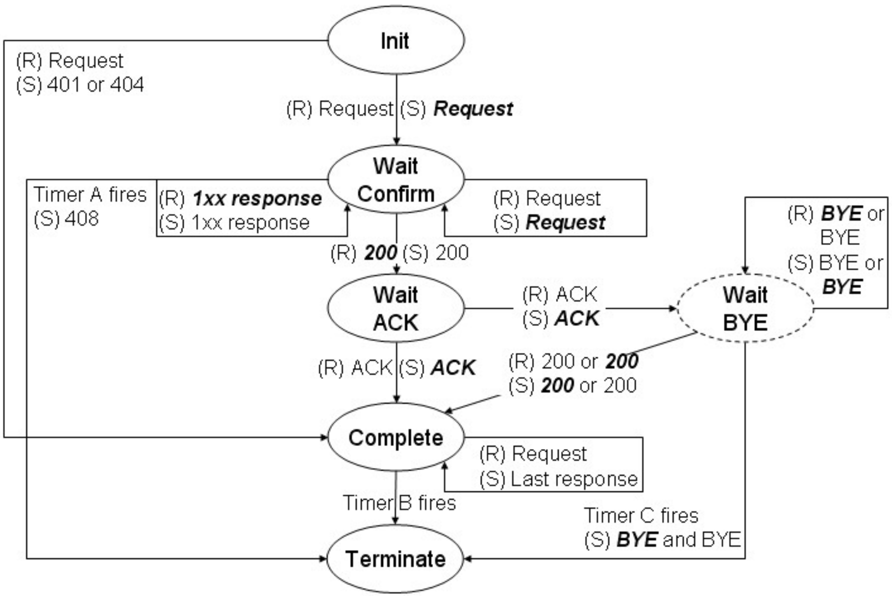


Figure 4-4 State diagram of anonymous event charging and charged by time

The figure 4-4 illustrates the state diagram of anonymous event charging and charged by time service. The bold and italic word means that the message is sent to or received from the consumer and the normal form is for the merchant.

1. After the payment server receives a request for event charging or charged by time, it simply checks if the consumer is registered or not, if not, the payment server will inform the merchant by a 404 response.

2. Otherwise the payment server forwards the request to the consumer and enters the WaitConfirm state. In the WaitConfirm state, the payment server listens to the 1xx response sent from the consumer and processes the retransmission of request. Besides, a timer A will be used to wait the confirmation from the consumer.

3. The transaction will enter the WaitACK state as receiving the 200 response from the

consumer. There are two kinds of transition path of at this point. If the whole transaction is event charging, the state will be changed to the Complete state when an ACK arrived. Otherwise, the state will be changed to the WaitBYE state and a timer C will be started. The calculation of timer C is base on the credit of the consumer and the charging ratio of the transaction.

4. As the payment server receives a BYE message, it will forward the BYE to the opposite DMD but the state remained unchanged. Until the payment server receives a 200 response from the opposite DMD, the state will be changed to the Complete which will become Terminate automatically.

4.4 Measurement of Payment Server

4.4.1 Environment

Figure 4-5 depicts our environment to measure the system capacity of the payment server. In which all message are transferred on the Ethernet to reduce the effect of network. The protocol is UDP. We put two SIM-DMDs that simulate the real DMD in use. One behaves as the payer and sends credit transfer request periodically to the payment server. The other behaves as the payee and reply a confirmation as receiving a request forwarded by the payment server.

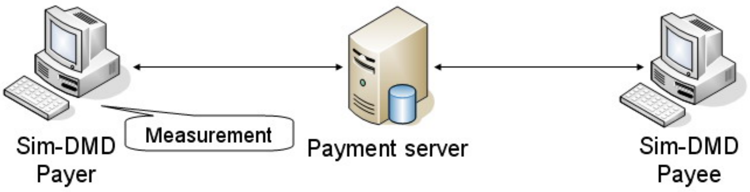


Figure 4-5 Layout of our environment

4.4.2 Measurements

We use Sim-DMD to simulate the users that can send request to the payment server simultaneously. There are 1000 uniform distributed requests for each test. A request is considered as successful one if the payer receives a corresponding response forwarded from the payment server.

We measure the average processing time of successful request and the success probability of each test in different arrival rate which means the quantity of request received by the payment server per minute. We change the arrival rate for each test. The lower arrival rate means the delay between each request is longer. Table 4-1 shows the result of our test. In this table, we could obtain the system capacity at a given success rate and average processing time from the result. Figure 4-6 plots the curve of the result. From the result, we can obtain a threshold of 1600 (reqs/min.) for which performance becomes worse (lower than 90%) if the arrival rate is higher than the threshold.

Table 4-1 Result of our measurement

Arrival rate (reqs/min)	Success prob. (done/1000)	Avg time (sec)	Arrival rate (reqs/min)	Success prob. (done/1000)	Avg time (sec)
1000~1400	100%	0.01	1900	82.5%	1.665
1500	99.8%	0.12	2000	81.2%	1.812
1600	95.2%	0.756	2100	77.1%	1.929
1700	91.5%	1.265	2200	73.5%	2.107
1800	87.1%	1.345	2300	71.7%	2.255

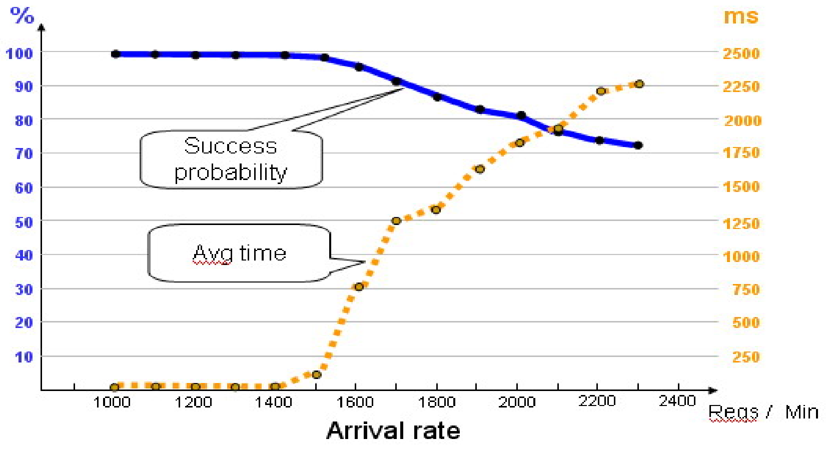


Figure 4-6 Curves of the result

Chapter 5 Conclusions

It becomes more and more popular to pay on the Internet in recent years. A lot of e-commerce system provides the emerging payment channel for the users. In the thesis, we first introduce the necessary security application used in our mobile system. Then we use them to develop a mobile payment system based on the telephone number. The whole system is deployed on the Internet and uses the telephone number as user's identifier.

Before using our system, the users need to own a DMD and a valid SIM card to request a ticket from the authentication server first. The authentication server first performs user authentication procedure to verify the validation of the client's identity. If the authentication is successful, the authentication server would generate a ticket to the client for subsequently authentication purpose. Once the client obtained the ticket, the user can access the payment server before the ticket expires. The users can send money to any registered user conveniently and the merchant could request money from the consumer by two means include event charging and charged by time. For the event charging or charged by time service, the consumer could remain anonymously by using the OTCN and the merchant could trade in tangible good and invisible service, such as online music or movie. In addition, to support services charged by time, we use an efficient time wheel data structure which can maintain timers in O(1) and balance per-tick maintenance operations.

Although our mobile payment system provides data confidentiality, integrity and authentication for each transaction, it is not sufficient for macro-payment service due to the lack of non-repudiation service which is the primitive weakness of symmetric encryption. In addition, the Diffie-Hellman is also vulnerable to the man-in-the-middle attack which will disable the authentication capability of the subsequent Kerberos procedure. To address these shortcomings of our payment system, it is effective to use the asymmetric encryption instead of symmetric encryption. Therefore, the future considerations are the capabilities that support public key distribution or integration with PKI for the present system.

References

[1] M. Peirce, “Multi-Party Electronic Payments for Mobile Communications” Ph.D. Thesis, University of Dublin, Trinity College, Oct. 2000.

[2] William Stallings, “Network Security Essentials: Applications and Standard Second Edition”, Prentice Hall, 2002.

[3] Yi-Sheng Huang, “Anonymous Mobile Payment for 3G Networks”, National Chiao Tung University June 2004.

[4] Daniel Collins, “Carrier Grade Voice over IP”, McGraw Hill, New York, 2001.

[5] Wen-Chen Hu, Chung-Wei Lee and Weidong Kou, “Advances in Security and Payment Methods for Mobile Commerce”, IGP, 2005.

[6] Yi-Bing Lin and Imrich Chlamtac, “Wireless and Mobile Network Architectures”, WILEY, 2001.

[7] Chi-Fan Lin, “VoIP Authentication System”, National Chiao Tung University June 2005.

[8] George Varghese and Anthony Lauck, “Hashed and Hierarchical Timing Wheels: Efficient Data Structure for Implementing a Timer Ficility”, IEEE/ACM Trans. Networking, Vol. 5, No. 6, December 1997.

[9] Muxiang Zhang and Yuguang Fang, “Security Analysis and Enhancements of 3GPP Authentication and Key Agreement Protocol”, IEEE Trans. Wireless Commun., Vol. 4, No. 2, March 2005.

[10] Jing-Shyang Hwu, Rong-Jaye Chen and Yi-Bing Lin, “Authenticated Public-Key Distribution over WLAN/Cellular Dual Networks”, Department of CSIE, National Chiao Tung University, June 2005.

[11] Bo Meng and Qianxing Xiong, “Reaserch on Electronic Payment Model”, 8th International IEEE, Conference on Computer Supported Cooperative Work in Design Proceedings, 2003.

[12] J. Rosenberg, Henning Schulzrinne, G. Gamarillo, E. Schooler and Mark Handley etc…, “SIP: Session Initiation Protocol”, RFC 3261, IETF, June 2002.

[13] J. Franks, P. Hallam Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen and L. Stewart, “HTTP Authentication: Basic and Digest Access Authentication”, RFC 2617, IETF, June 1999.

[14] T. Yu, S. Hartman, K. Raeburn, “The Kerberos Network Authentication Service (V5)”, RFC 4120, IETF, July 2005.