Design and Implementation of an Integrated RFID and VoIP System for Supporting Personal Mobility

Pen-ho Chang and Tsan-Pin Wang

Department of Computer and Information Science, National Taichung University, Taichung, Taiwan
axer@ms1.boe.tcc.edu.tw; tpwang@mail.ntcu.edu.tw

Abstract

As a user moves to different locations, how to make the communication service in use follows the user to the current location without being broken off depends on the support of Personal Mobility. Therefore, supporting personal mobility for the "follow-me service" is the goal of this research. We integrate Radio Frequency Identification (RFID), and SIP Express Router (SER) to create a VoIP system which can achieve personal mobility.

In our proposed system, as soon as the user moves to different locations, the sensors of the doors/locations can read the RFID Tag of the user and the server can activate an according phone and register to the SER immediately. To avoid multiple registrations, our system will close the phone which the user used before at the same time. The advantage of this research is that it's unnecessary to change the settings of the SER, and neither to use the specific phone. In the implementation, we introduce two modules, Remote Call Server (RCS) and Remote Call Client (RCC), to support personal mobility. Both modules are written in Microsoft© Visual C#.NET and use the same MySQL database with the SIP Proxy Server for reducing the deployment cost. Moreover, the RFID reader directly connects to RCS via a regular USB port. As a result, the reader can immediately transfer the raw data to RCS when it reads some tags. These features make deploying a personal mobility architecture easier and promising.

Keywords

Personal mobility, VoIP, RFID

1. Introduction

Traditional Public Switched Telephone Network (PSTN) telephone transmits voice or sounds by circuit switched networks, while *Internet Telephony* converts voice or sounds into data packets and transmits them by internet. Internet telephony uses VoIP (Voice over Internet Protocol) [1] [2] to transmit voice packets in the internet or other packet-switched networks.

Currently, the most important VoIP protocol is Session Initiation Protocol (SIP) [3] that is designed by IETF. SIP is a signaling protocol and widely used for setting up and tearing down multimedia communication sessions such as voice and video calls. SIP is merely an initiation protocol for establishing multimedia sessions, and SIP uses Session Description Protocol (SDP) [4] that describes the set of media formats, addresses, and ports to negotiate an agreement between the two communication terminals as to the types of media they're willing to share.

A proxy server bases on SIP protocols is called SIP Proxy Server. In these days, one of the most widely used software for SIP Proxy Server is SIP Express Router (SER) [5] and its branch distribution OpenSER (or OpenSIPs) [6], both are open source free wares. The later one is the service (or daemon) software for VoIP that we used in this research.

The internet phone, called UA (User Agent), basically separates into two parts: hard phone and soft phone. A hard phone is similar to a traditional telephone. When someone dials the numbers on a hard phone, the hard phone will then transform the numbers into a SIP INVITE message and sent it to the SIP Proxy Server. Generally, the hard phone is produced to fit with users' usual practice. The major difference is that the hard phone is connected with an internet line instead of a telephone line. Being able to use a soft phone is one of the benefits of internet telephony because soft phone is not only a phone but more importantly a communication program. The communication program uses a microphone as a telephone transmitter, a speaker as a receiver. Particularly, we can install it in any personal computer, PDA or cell phone easily.

When a soft phone starts on, it will register to the SIP proxy server by default and complete the REGISTER procedure. If the user starts another soft phone at the same time in the different locations, the SIP Proxy Server will also allow multiple registrations of different IP address with the same ID. All of phones which share with the same ID will ring at the same time when someone calls in to this number. Moreover, when the user picks up any phone call in one location, the SIP Proxy Server transmits the SIP message of CANCEL to the other phones. Consequently, the ring stops the other phones on receiving the CANCEL message [7].

Radio Frequency Identification (RFID) [8] [9] is an automatic identification method, relying on storing and remotely retrieving data using devices called RFID tags and a RFID reader. This RFID tag is an object with specific circuit in it and the RFID reader can create variable electromagnetic waves. When an RFID tag is within the effect of electromagnetic waves, or induced range, it will produce electric current by electromagnetic induction. This induced current is enough to response the tag ID, called "TagID" hereafter, that is stored in the RFID tag. Each user has his RFID tag to represent the user by TagID. As a result, RFID is a device with character of personalization.

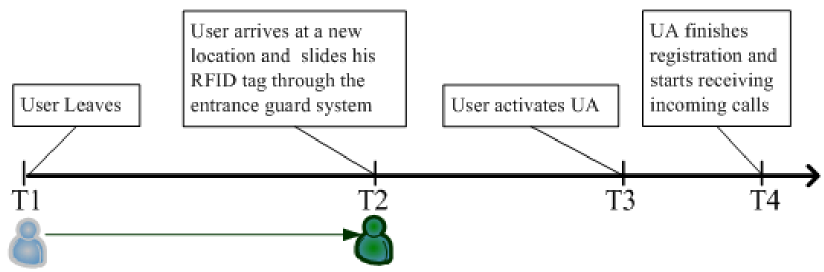


Figure 1: The timing diagram of a user's movement.

There is a phenomenon of *call-miss* when someone calls in during the time the user is moving to another place. The call-miss also happens when the user moves to a new position and forgot starting on the UA in its location. However, the probability of call-miss can be reduced if the UA can be start on as earlier as possible at the time of the` user slides his RFID tag through the entrance guard system in the new location. According to Figure 1, user leaves at the timing of T1 then arrives at a new location at T2 and he slides his RFID tag through the entrance guard system. He passes through the door, walks into his place in the room and takes a seat, etc. Finally, he activates UA at the timing of T3 and at the last time point of T4, UA can start to receive the call. The time difference (T2-T1) is according to the user's moving speed. The time difference (T3-T2) is decided by the moving speed of user and the distance between the entrance guard and the location of UA. (T4-T3) is the time difference of UA from starting on to completely registration which can be calculated according to the costs of UA starting, server handling messages and the transmission in network.

To carry out the goal of personal mobility, the idea of this work is to let the UA be activated as earlier as possible after the entrance guard system is being used. Moreover, we implement a personal mobility remote call system that consists of a server part program RCS (Remote Call Server) and a client part program RCC (Remote Call Client). When a TagID is authorized successfully, RCS will notify the RCC to start the soft phone on and at the same time the RCS will send a command to close the previous RCS in the original location. On receipt of the close command, RCS will notify its RCC to end up its soft phone. This proposal not only achieves the concept of follow-me service but also avoids the drawback of multiple registrations in the case of someone who might answer the phone in the original location.

2. Proposed personal mobility architecture

Now we consider only soft phones are adopted in VoIP, while the user is moving to a new location, for example, a user moves from laboratory to office. In the existing system, he has to operate his computer in the office and activate the UA manually. Because the SIP Proxy Server allows multiple-registration by default, if someone called in, the both UAs at laboratory and the office will ring simultaneously so that this call might be caught by someone else in the laboratory. Moreover, there will be definitely a call-miss if the user thoroughly forgot to start on the UA after arriving at office. Section 2.1 shows our improvement. Section 2.2 explains the important stages of the improvement and Section 2.3 roughly depicts the formats of control message in our proposed system.

2.1 Personal mobility architecture

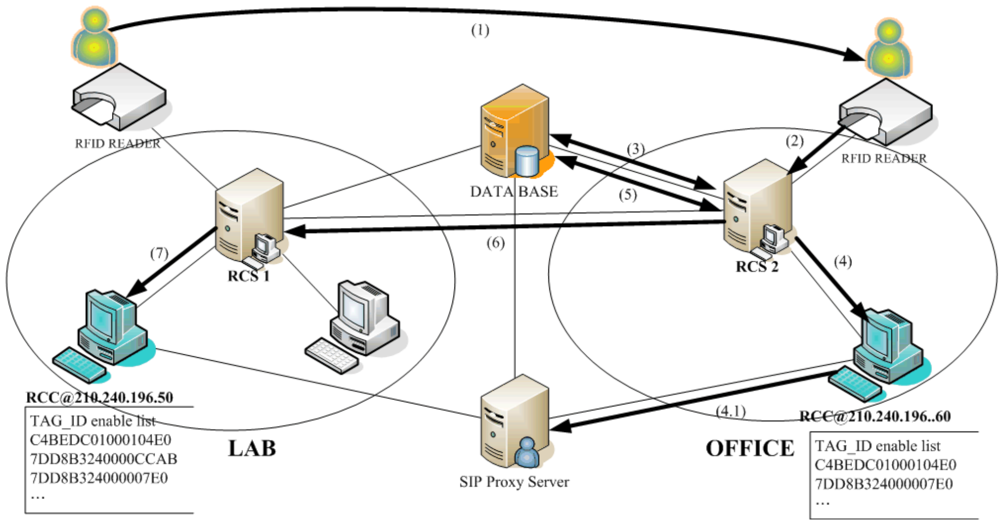


Figure 2: Personal mobility architecture.

The main components include two modules RCS and RCC. Figure 2 shows that when a user moves to a different location, he uses RFID tag to log on the entrance guard system. In a word, we integrate RFID into our system to achieve the follow-me service of personal mobility in VoIP systems. The proposed architecture and its operation steps are shown in the following.

(1) A user moves from LAB to OFFICE and uses his RFID tag to process authorization of the entrance guard.

(2) The RFID reader reads the TagID and transmits it to RCS2.

(3) RCS2 queries the database to acquire the IP address where the TagID belongs to.

(4) RCS2 sends the "activation command" to the RCC to activate its UA. The IP address of RCC is acquired in Step (3).

(4.1) UA starts on and registers to SIP Proxy Server. The account of SIP must be set in advance on the UA.

(5) RCS2 queries the database again and obtains the IP address of other online RCSs.

(6) RCS2 sends "close command" including the TagID from Step (2) to all online RCSs.

(7) RCS1 receives the "close command" and closes the UA program which was originally activated by RCS1.

2.2 The execution stages of the personal mobility remote call system

The main program of our system has separated in several stages: RCS starts on, client logs in RCC, RFID reader reads a tag, and RCS receives close command as follows:

(1) RCS starts on: After RCS starts on, RCS updates the online time in database in a specific interval periodically, so that other RCSs know its state "online" or "alive". Meanwhile it listens to a specific TCP port waiting for establishing connections from other RCC or other RCS.

(2) Client logs in RCC: When an RCC has established connection, RCS will keep remeber the IP address and the peer port of RCC. Meanwhile, RCS periodically checks the connection state and removes any non-existed connection.

(3) RFID reader reads a Tag: RFID reader sends TagID to its RCS and RCS verifies if the TagID has been registered before. This stage has several steps depicts as follows.

Step 1: When the RFID reader reads an RFID tag, it will transmit what it reads to RCS. RCS parses the TagID from raw data and then consults the "RCS TagID Enable List" table. The table includes available TagIDs and their IP address of RCC, the full path of UA execution file. The table and its description are shown in Table 1.

Step 2: RCS looks up Table 1 and gets column "ClientIP" value where column "TagID" is the same with TagID from Step 1. If column "ClientIP" value exists and the owner (i.e. RCC) of this "ClientIP" is online, continue to Step 3 or suspend if not.

Step 3: Similar to Step 2, RCS queries Table 1 again and gets column "ExeProgramDir" value, which is the execution path of UA. Carry out the Step 4 and 5.

Step 4: RCS gets the online RCSs from database. RCS will be treated offline without updating database in 30 minutes.

Step 5: To dispatch the close message to all other online RCSs, the message includes the TagID of user: *Send_close_command_to_all_of_SrvIPs('TagID from RFID reader');*

Table 1: RCS TagID Enable List and its description

Column Name	Description
TagID	RFID Tag ID
ClientIP	The IP address of RCC
LastDateTime	The last time that RFID reader reads the Tag ID.
Description	Remark
ExeProgramDir	The full path of UA execution file.
SrvIP	The IP address of RCS

(4) RCS receives close command

When RCS receives close command, containing argument of TagID, from other RCS. Similar to the Step 2 of Stage 3, RCS can obtain the IP address of RCC with this TagID and transmits "close" command to this RCC. RCC will close or kill the UA process as it receives the close command from RCS.

2.3 Format of control messages

In order to simplify the system, we adopt the socket program to exchange messages instead of Remote Procedure Call (RPC). Because eXtensible Markup Language (XML) is a mature developed markup language on the internet, it is designed to transmit and carry data so that it is very suitable for the exchange of data. Furthermore, even if RCS and RCC are not written in the same language, they can still use the common language, XML, to communicate with each other. This consideration escalates the scalability of the system. On the contrary, if messages exchange in the method of RPC, it will face problems in cross-language and cross-platform.

We, therefore, adopt an XML-like message format into the control messages between RCS and RCC as well as between RCS and RCS. We announce the types of message as an "enum" type.For example, we define variable Operation_state as enum type:

public enum Operation_state { E_Message,E_Exe_Cmd, … ,E_End_Exe };

The format of control message: <enum tag>message argument </enum tag>.

For example, the message of conversation:

<E_Message>Conversation Content Body</E_Message>

The message for RCS to order RCC to execute its UA execution file:

<E_Exe_Cmd>full path of UA execution file </E_Exe_Cmd> or

<E_Exe_Cmd>C:\Program Files\CounterPath\X-Lite\x-lite.exe</E_Exe_Cmd>

3. Implementation

RCS and RCC are written in Microsoft© Visual C#.NET and the system use the same MySQL database with the SIP Proxy Server for decreasing the deployment cost. MySQL Connector/.Net[10] needs to be installed for C#.NET[11][12][13] programs to connect with MySQL database. Section 3.1 lists the experimental environment; In Section 3.2, we show some snapshots of the program execution at runtime.

3.1 Experimental environment

(1) RCS and RCC

OS: Microsoft Windows XP Professional ver. 2002, Service Pack 3.
Framework: Microsoft.NET Framework 2.0 [14]
Using Language C# .NET

(2) Database and Operation system

MySQL Ver 14.12 Distrib 5.0.45, for redhat-linux-gnu (i386) using readline 5.0
OS: Linux 2.6.23.15-80.fc7 #1 EST 2008 i686 i686 i386 GNU/Linux

(3) RFID reader and tag

RF frequency: 13.56MHz, identified distance up to 10cm,USB interface [15]
RFID tag: Standard of ISO 15693 [16]

(4) Soft phones implemented: x-lite[17].

3.2 Program snapshots

(1) RCS is listening and waiting for connection from RCC.

As shown in Figure 3, RCS starts to listen to all interfaces when the “Start” button is pressed. The figure shows two connections established with RCCs. Each established connection will be added to the “Channels” list.

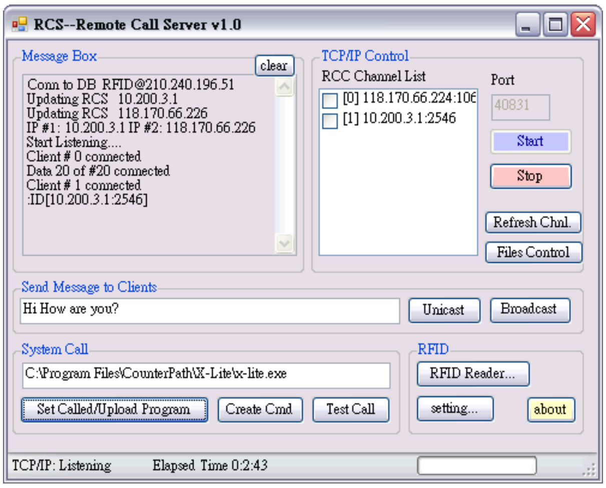


Figure 3: RCS is listening and waiting for connection from RCC.

(2) RFID Reading Program in RCS is running.

RCS deploys an RFID Reading Program that consists of an “RCS TagID Enable List” table, where lists all of its RCCs. As shown in Figure 4, when Reader reads a TagID, it will check if the TagID is in the list. If it is in the table, RCS will automatically start on the soft phone which path is in column “File Path of Soft Phone”.

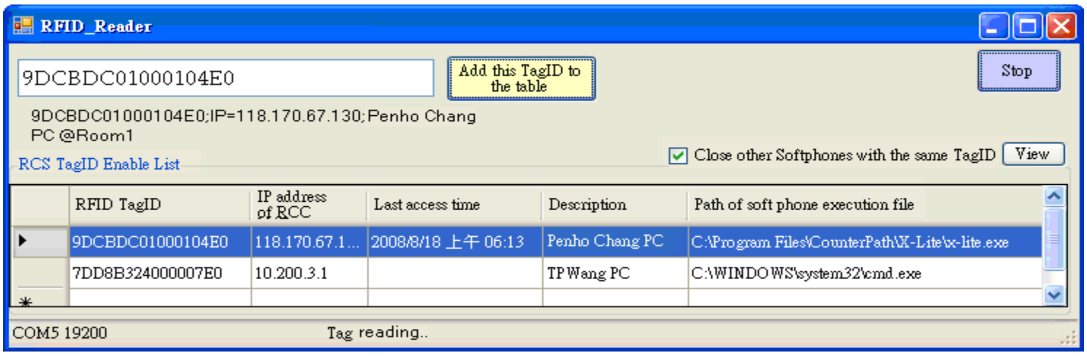


Figure 4: Program snapshot that RFID Reading Program in RCS is running

(3) RCC receives a “start” command and then activates its accompanied UA.

As shown in Figure 5, as long as RCC receives a command, it will execute this command faithfully. We can either predefine or leave empty the path of UA program in the “CONTROL PANEL” of RCC. RCC will adopt the path in the “start” command signaling from RCS as a priority, unless the path in the “start” command signaling is empty.

4. Performance Analysis

Our goal is to reduce the probability of call-miss as possible as we can. In this section, we will compare the original system with our proposed one in terms of the probability of call-miss. Section 4.1 shows the qualitative analysis of two systems. The analytical model is derived in Section 4.2 and the results of analysis and test are written in Section 4.3.

(a) (b)

Figure 5: Program snapshots (a) an RFID reader is reading a tag (b) RCC receives a "start" command and then activates an accompanied UA program.

4.1 Qualitative Analysis

In the subsection, we describe the benefits and weakness for the existing system and the proposed system. The advantages of the existing system are simple and easy to maintain, while its disadvantages include higher call-miss rate and may arise the call misrouting if SIP proxy server allows multi-registration. On the other hand, although the proposed system needs to install and setup extra modules, it has many benefits including reduced call-miss rate, achieving follow-me service, integrated VoIP and RFID in entrance guard system, without modifying any setting in SIP server, and unnecessary for specific soft phone, etc.I

4.2 Analytical Model

For the convenience to measure and to depict, we divide into several intervals from the user arrives at the new location to the end of UA finishes registering to the SIP proxy server. In other words, we divide the interval of time point from T2 to T4 in Figure 1 into another three intervals s_1, s_2 and s_3.as shown in Figure 6(a). The definition is listed in the following.

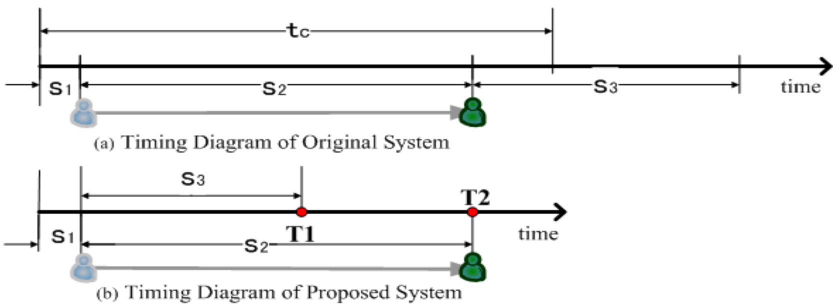


Figure 6: The timing diagram of (a) the original system and (b) the proposed system

s_1: The time that the RFID reader needs for reading TagID and RCS completes the authorization.

s_2: The time for the user's moving from the entrance guard spot to his new location (or his seat).

s_3: The time that UA is activated and UA finishes registering to the SIP proxy server.

We denote personal *handoff interval* t_h as the needed time from the time that the user arrives at the new location to that UA finishes registration. Hence, t_h is given:

$t_h = s_1 + s_2 + s_3$ (1a)

In our work, we substitute the method of activating UA automatically for the manual operation of human. The system receives the message from RCC and starts on the UA immediately. Due to omitting from the slowly operation from human, the interval s_3 could be much shorter while s_1, s_2 remain unchanged.

Refer to Figure 6(b), at the time point T1, the UA has finished registration and can receive the incoming calls; at the time point T2, the user arrives at his new location and is able to answer the phone call. Thus, we define $t_h^{'}$ as the *personal handoff interval* in the proposed system. Eq. (1a) can be derived as

$$\begin{cases} t_h^{'} = s_1 + s_2 \ \ (s_2 \ge s_3) \\ t_h^{'} = s_1 + s_3 \ \ (s_2 < s_3) \end{cases} \qquad (1b)$$

In general condition, s_2 is bigger than s_3. To reduce complexity, Eq. (1b) is only considered the situation of $s_2 \ge s_3$: $t_h^{'} = s_1 + s_2$.

By passing the time t_h, the user can answer the phone without call-miss; t_c is the inter-arrival time, which is the interval from the last call to the previous one. We assume that t_c is an exponential distributed random variable with mean λ_c. Thus, the call miss rate $P_{callmiss}$ given as

$$P_{callmiss} = P(t_c < t_h) = \int_0^{\infty} \int_0^{t_h} \lambda_c e^{-\lambda_c t_c} dt_c \cdot f_h(t_h) dt_h \qquad (2)$$

The function $f_h(t)$ in Eqs(2) is a probability density function of the personal handoff interval. And s_1, s_2 and s_3 are also random variables, assume that they are all exponential distributed with mean values are $1/\lambda_1, 1/\lambda_2, 1/\lambda_3$ respectively. The probability density function $f_h(t_h)$ in the existing system from Eq (1a) we have

$$f_h(t_h) = f_h(s_1, s_2, s_3) = \int_{-\infty}^{\infty} f_{s_1}(u) \int_{-\infty}^{\infty} f_{s_2}(v) f_{s_3}(s-u-v) dv du$$
$$= \lambda_1 \lambda_2 \lambda_3 \int_0^s e^{-\lambda_1 u} \int_0^{s-u} e^{-\lambda_2 v} e^{-\lambda_3 (s-u-v)} dv du$$

I $f_{s_1}(u), f_{s_2}(v) = 0$ if $u, v < 0$ and $f_{s_3}(s-u-v) = 0$ if $u+v>s$

$$= \frac{-\lambda_1 \lambda_2 \lambda_3}{\lambda_2 - \lambda_3} \int_0^s e^{-\lambda_1 u} e^{-\lambda_3 (s-u)} (e^{-(\lambda_2 - \lambda_3)(s-u)} - 1) du \quad \text{where } \lambda_2 \ne \lambda_3$$
$$= \frac{\lambda_1 \lambda_2 \lambda_3}{\lambda_2 - \lambda_3} \left(\frac{1}{\lambda_1 - \lambda_2} (e^{-\lambda_1 s} - e^{-\lambda_2 s}) - \frac{1}{\lambda_1 - \lambda_3} (e^{-\lambda_1 s} - e^{-\lambda_3 s}) \right) \qquad (3a)$$

Similarly, the probability density function $f_h^{'}(t_h)$ of our proposed system from Eq (2b) is

$$f_h^{'}(t_h^{'}) = f_h^{'}(s_1, s_2) = \int_{-\infty}^{\infty} f_{s_1}(t) f_{s_2}(s-t) dt = \int_0^s f_{s_1}(t) f_{s_2}(s-t) dt \quad \text{where s = s1 + s2}$$

$$a \ f_{s_1}(t) = 0 \text{ if } t < 0 \text{ and } f_{s_2}(s-t) = 0 \text{ if } t > s$$

$$= \int_0^s \lambda_1 e^{-\lambda_1 t} \cdot \lambda_2 e^{-\lambda_2 (s-t)} dt = \lambda_1 \lambda_2 \cdot e^{-\lambda_2 s} \int_0^s e^{-(\lambda_1 - \lambda_2)t} dt = \frac{-\lambda_1 \lambda_2}{\lambda_1 - \lambda_2} e^{-\lambda_2 s} (e^{-(\lambda_1 - \lambda_2)s} - 1) \text{ if } \lambda_1 \neq \lambda_2$$

$$= \frac{-\lambda_1 \lambda_2}{\lambda_1 - \lambda_2} (e^{-\lambda_1 s} - e^{-\lambda_2 s}) \text{ where } \lambda_1 \neq \lambda_2$$

$$\text{and } f_h^{'}(t_h^{'}) = \lambda_1^2 s e^{-\lambda_1 s} \quad \text{where } \lambda_1 = \lambda_2$$

(3b)

Together with Eqs. (3a) and (2), we can obtain the probability of call-miss in the existing system

$$P_{callmiss} = P(t_c < t_h) = \frac{\lambda_1 \lambda_2 \lambda_3}{\lambda_2 - \lambda_3} \int_0^\infty (1 - e^{-\lambda_c t_h}) \left(\frac{1}{\lambda_1 - \lambda_2} (e^{-\lambda_1 t_h} - e^{-\lambda_2 t_h}) - \frac{1}{\lambda_1 - \lambda_3} (e^{-\lambda_1 t_h} - e^{-\lambda_3 t_h}) \right) dt_h$$

$$= \frac{\lambda_1 \lambda_2 \lambda_3}{\lambda_2 - \lambda_3} \left(\frac{1}{\lambda_1 - \lambda_2} \left(\frac{1}{\lambda_1} - \frac{1}{\lambda_2} - \frac{1}{\lambda_c + \lambda_1} + \frac{1}{\lambda_c + \lambda_2} \right) - \frac{1}{\lambda_1 - \lambda_3} \left(\frac{1}{\lambda_1} - \frac{1}{\lambda_3} - \frac{1}{\lambda_c + \lambda_1} + \frac{1}{\lambda_c + \lambda_3} \right) \right) \quad \text{(4a)}$$

Substituting Eq (3b) to Eq (2), we obtain the probability of call-miss in the proposed system

$$P'_{callmiss} = P'(t_c < t_h) = \frac{-\lambda_1 \lambda_2}{\lambda_1 - \lambda_2} \int_0^\infty (1 - e^{-\lambda_c t})(e^{-\lambda_1 t} - e^{-\lambda_2 t})\, dt$$

$$= \frac{-\lambda_1 \lambda_2}{\lambda_1 - \lambda_2} \left(\frac{1}{\lambda_1} - \frac{1}{\lambda_2} - \frac{1}{\lambda_1 + \lambda_c} + \frac{1}{\lambda_2 + \lambda_c} \right) \quad \text{(4b)}$$

Finally, we determine that the improvement ratio R is given by

$$R = \left(1 - \frac{P'_{callmiss}}{P_{callmiss}}\right) \times 100\% \quad \text{(5)}$$

4.3 Numerical Results

Table 2: The needed time for each interval

Inter val	Needed time in average	Description
s_1	500 ms	Reading frequency of RFID Reader is set once per second and the time for processing of program is neglected (less than 5ms).
s_2	10 s	This value depends on the user's moving speed, the average setting is 10s.
s_3	5 s	The time needed for a user to run the UA program manually, the average setting is 5s.
	110 ms	The average spending time that the system run the UA program automatically.
	1500 ms	The time to start on the program is about 1150ms, the needed time for SIP to complete registration is about 350ms. This value is the sum of both.

The needed time to measure each interval and the results are showed in Table 2. The value of s_3 is considered with two situations. The average time will be 5 seconds if the user operates UA by himself. On the contrary, the average time takes only 110ms if the system starts on the UA automatically. No matter which situation is, it always takes averagely about 1500ms to run the program of UA and to finish the registration.

To measure performance, we adopt Wireshark[18] to capture the packets. By analyzing the difference of timestamps of each SIP message packet, we obtain an averagely needed time for UA to complete the registration. It takes around 350ms. On the other hand, the needed time for the program of UA to run is roughly measured about 1150ms. Thus we conclude it takes averagely 1500ms to run the program of UA and to finish the registration. As a result, $s_3 = 5 + 1.5 = 6.5$s in the existing system and $s_3 = 0.11 + 1.5 = 1.61$s in our proposed system.

In the following section, we will discuss the relationship of improvement ratio R and probability of call-miss $P_{callmiss}$ according to various conditions.

4.3.1 The call-miss rate according to different inter-arrival times. Figure 7 reveals the relationship between $P_{callmiss}$ and inter-arrival time t_c. While λ_2 is equal to $\lambda_1/10$ and$\lambda_1/20$ respectively, if the calling frequency increases, or t_c is smaller, the probability of call-miss $P_{callmiss}$ also increases. In contrast, if the calling frequency gets lower, or t_c steps up, $P_{callmiss}$ decreases. Note that no matter what t_c is, the R at $\lambda_2=\lambda_1/20$ is always smaller than that at $\lambda_2=\lambda_1/10$, which means when λ_2 is increasing, our proposed system will achieve higher improvement ratio. When λ_2 is reduced, the improvement ratio is also decreased. In any case, our system performs better than the existing one, and at least exceeds 31.9% in improvement ratio.

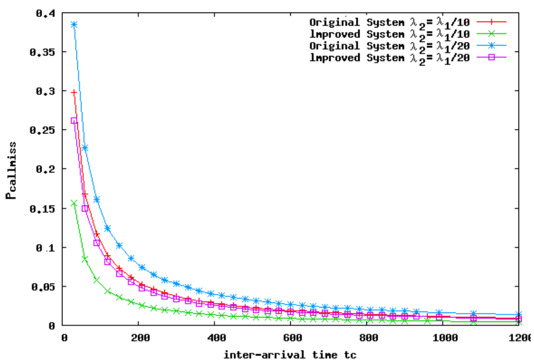


Figure 7: Impact of inter-arrival time on call-miss rates

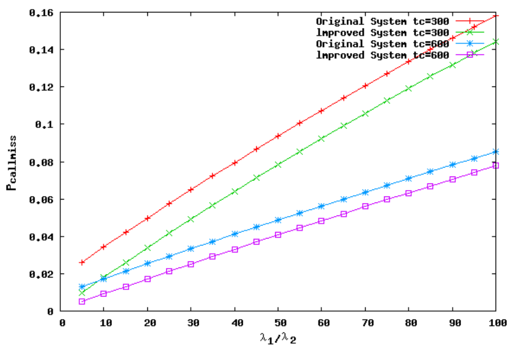
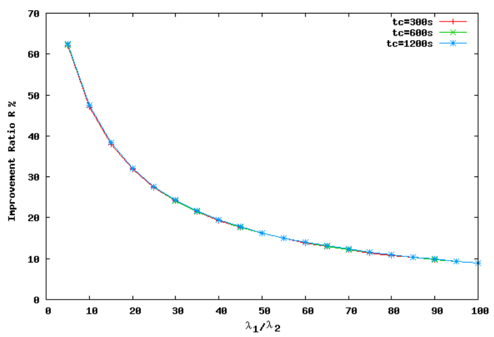


Figure 8: Impact of mobility on call-miss rates



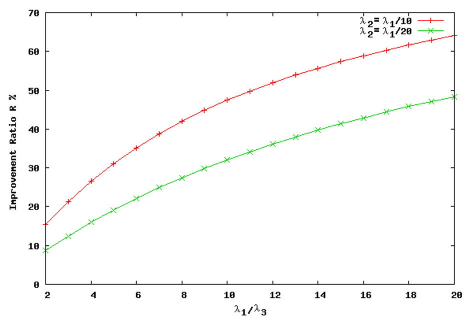


Figure 9: Impact of mobility on improvement ratio

Figure 10: Impact of different operating time on improvement ratio

4.3.2 The call-miss rate with different mobility. Figure 8 shows the relationship between $P_{callmiss}$ and λ_1/λ_2. We focus on the impact of mobility on call-miss rates. In general, the higher the mobility is, the lower λ_1/λ_2 is. For example, a user walks fast or the computer in the new location is close to the entrance guard. As the mobility decreases, $P_{callmiss}$ will increase. Here we use the improvement ratio as y-axis and the λ_1/λ_2 as the x-axis as shown in Figure 9. It shows that the improvement ratio with different t_c is hardly to distinct from each other, owing to t_c is much great than s_1, s_2, s_3, the **improvement ratio** is **irrelevant** to **inter-arrival time**..

4.3.3 The probability of call-miss in different operating time. We would like to reduce the operating time s_3 as soon as possible. Basically, s_3 depends on various manipulations of operation, bandwidth of networks, and process time of UA or SIP Proxy Server, etc. And λ_3 is the reciprocal of s_3, the bigger λ_1/λ_3 means the worse operation. On the other hand, as the operation is highly practiced, the λ_1/λ_3 value will be smaller. Figure 10 shows the impact of operating time on improvement ratio. If λ_1/λ_3 increases, the improvement ratio increases as well. It means when the operation is highly practiced, the improvement ratio is low

5. Conclusion and Future Work

This paper integrates RFID to support personal mobility in VoIP systems. In our implementation, as a user slides his card through the secure guard system, his UA of this location is started on immediately. Thus our proposed system not only avoids the forgetfulness of a user to start on the soft phone but also greatly reduces the time of starting on for a UA. More importantly, our proposed system implements two extra modules without modifying or adjusting any setting of SIP Proxy Server, and thus avoids the problems as authorization of management, scalability or compatibility.

Our numerical results show that as the inter-arrival time increases, the probability of call miss increases and the improvement ratio slightly increases as well. Moreover, the lower the mobility is, the higher the probability of call miss will be. However, the improvement ratio decreases with the personal mobility. On the other hand, we use the mode of single registration substitutes for the mode of multiple registrations on a SIP proxy server for a single user. This way not only reduces the unnecessary signaling but also supports personal mobility toward the ideal of follow-me service.

Our implementation may reveal a security weakness because of the plain-text format of signaling. If a malicious attacker listens to the network, he may probably hijack the information of signaling including the RFID tag ID, and then he can easily make a copy of it. Therefore, we plan to encrypt the content body signaling in the future work. Besides, a user has to set his account onto the UA in advance. Consequently, the system will be inconvenient and less scalable. To improve this drawback, we have to implement a UA which can dynamically download a user's profile and take this profile to process a SIP registration automatically. This extension will be our future work.

Acknowledgement

Previous version of this paper has been presented in NISS2009, Beijing, China. This research was supported by National Science Council, Taiwan under grants NSC 97-2221-E-142-002-MY2.

References

[1] U. Black, Voice Over IP. Prentice Hall, 2000.

[2] B. Goode, "Voice over Internet Protocol (VoIP)," Proc. IEEE, vol. 90, no. 9, 2002.

[3] RFC 3261 SIP: Session Initiation Protocol, J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler June 2002.

[4] RFC 2327 SDP: Session Description Protocol, M. Handley, V. Jacobson, April 1998.

[5] SER-SIP Express Router, iptel.org, http://www.iptel.org/ser

[6] OpenSER or OpenSIPS (OpenSIPS is the new name for the OpenSER project.), http://www.opensips.org/

[7] Tsan-Pin Wang and KauLin Chiu, "An Efficient Scheme for Supporting Personal Mobility in SIP-based VoIP Services," IEICE Trans. on Communications (Special Section on Mobile Multimedia Communications), Vol. 89-B, No.10, pp.2706-2714, Oct. 2006.

[8] Dr. Klaus Finkenzeller, RFID HANDBOOK, 2nd Edition, Wiley, 2003.

[9] Bill Glover, Himanshu Bhatt, RFID Essentials, O'Reilly, 2006.

[10] Connector/Net 5.2, release of MySQL's fully managed ADO.Net provider to date. http://dev.mysql.com/downloads/connector/net/5.2.html

[11] Robert Powell , Richard L. Weeks, C# and the .NET Framework 2001.8, Sams 2001

[12] Paul Yao (Author), David Durant , .NET Compact Framework Programming with C# Microsoft .NET Development Series 2004.

[13] David Makofske, Michael J. Donahoo, Kenneth L. Calvert TCP/IP Sockets in C#: Practical Guide for Programmers, The Practical Guides, Morgan Kaufmann, 2004 .

[14] Microsoft .NET Framework 2.0 Service Pack 1 x86 http://www.microsoft.com/download/

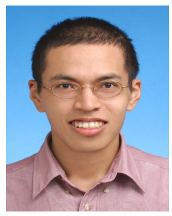
[15] Summit Automation Co., Ltd. http://www.summitco.com.tw/index.php,

[16] RFID SUMMIT U-Reader Specifications http://www.summitco.com.tw/reader-1-1.php

[17] X-Lite Softphone http://www.counterpath.com/index.php?menu=products&smenu=download

[18] Wireshark, Wireshark Foundation, http://www.wireshark.org/

Authors



Pen-ho Chang received his B.S. in Physics from National Kaohsiung Normal University and M.S. from National Taichung University in Taiwan in 1997 and 2009. He is currently a teacher of junior high school.

Tsan-Pin Wang is currently a professor in the Department of Computer and Information Science at National Taichung University, Taichung, Taiwan. He received the B.S. degree in applied mathematics; M.S. and Ph.D. in computer science and information engineering, from National Chiao Tung University, Taiwan, ROC, in 1990, 1992, and 1997, respectively. From 1992 to 1993, he was a system engineer in the R&D Division of Taiwan NEC Ltd. From 1997 to 2001, he was an assistant professor at Providence University. From 2001 to 2005, he was an associate professor at Providence University. From 2006 to 2009, he was the chairman of the department of CIS, National Taichung University. His research interests include mobile computing, mobile communications, Wireless Internet, and Voice over IP.