Using RFID tags to Improve Pilgrimage Management

Hamad Binsalleeh, Noman Mohammed, Parminder S. Sandhu,
Feras Aljumah, and Benjamin C. M. Fung
CIISE, Concordia University
Montreal, Quebec, Canada H3G 1M8
{h_binsal, no_moham, p_sand, f_aljum, fung}@ciise.concordia.ca

Abstract

Every year millions of Muslims from all around the world gather to perform their pilgrimage in Makkah, Saudi Arabia. Due to the massive number of pilgrims of different languages, cultures and countries, it is highly challenging for the authorities to manage and provide proper services to these pilgrims. In this paper, we propose a pilgrim tracking system with the help of Radio Frequency identification (RFID) that can be deployed to improve the present situation. We also investigate possible security and privacy issues and propose a new secure protocol which meets the requirements for this application. Our proposed authentication protocol provides privacy, security and it is efficient particularly for this application.

1 Introduction

Hajj is the largest annual pilgrimage in the world. Millions of Muslims from all around the world gather to perform their pilgrimage in Makkah, Saudi Arabia. According to statistics, the number of pilgrims in 2007 was over 2.4 million. With this number of people in one place along with communication barriers many people get lost, while others are taken into emergency rooms without any information. Sometimes, it becomes impossible to retrieve the essential medical history. Thus each year a large number of people die due to natural causes and also from mismanagement. Figure 1 gives the number of deaths in Hajj from the past five years.

In the current *Hajj* management system, the authorities/governments set a policy that each pilgrim should be associated with a company, which provides various types of services (e.g., transportation and accommodation). Pilgrims are not allowed to take the services provided by other companies. Therefore, it is important for a company to correctly identify an associated pilgrim and the specific services to which this pilgrim is entitled. A common method is to use bracelets on which personal and medical information, as well as information about the associated company, are written. Although these solutions help identify the pilgrims and thus provide them certain services, it is not sufficient for monitoring the individuals especially when they are moving around. As there are millions of pilgrims who come to Hajj, it becomes difficult if not impossible to manage them properly by these methods. Thus, to improve the present management system, we propose a Radio Frequency identification (RFID) tag based pilgrim tracking/monitoring system that can substantially alleviate several present problems.

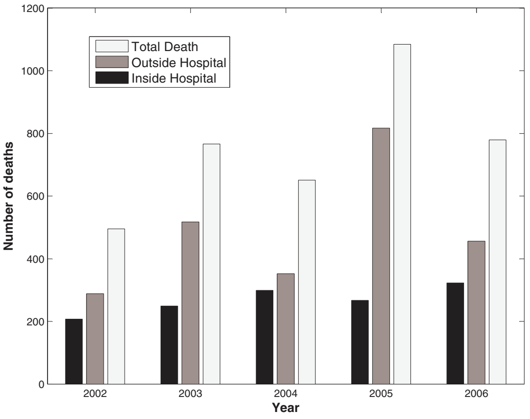

Figure 1. Statistics of pilgrims [4]

RFID is an Automatic Data Collection (ADC) technology that uses radio-frequency waves to transfer data between a reader and a RFID tag, for the purpose of identifying, categorizing, and tracking a movable item that the tag is attached [1]. RFID is fast, reliable, and does not require physical sight or contact between reader/scanner and the tagged item. This non-line of sight property makes RFID suitable for our application, compared to barcodes or other optically read technologies.

In Section 2, we briefly present our RFID based system architecture. Section 3 describes the security and privacy requirements. Then, we present our secure protocol in Section 4 which fulfills the system requirements. We analysis

978-1-4244-5700-7/10/$26.00 ©2009 IEEE

our protocol in Section 5 followed by a literature survey in Section 6. Finally, we conclude in section 7.

2 The System Architecture

In this section, we briefly present the system design and major components of the proposed scheme.

2.1 Design Principle

Each pilgrim is provided a RFID tag by a wrist-band containing only a unique number. All the associated data, such as name, age, medical history, physical movement, etc., will be stored in a central database maintained by an authority. When the pilgrim moves from one place to another, the location will be updated in the database so that the location information can be obtained promptly during emergencies. The proposed scheme can easily track pilgrims' movement since RFID readers can read multiple items simultaneously. Besides the fixed-position readers/sensors, wireless portable scanners may be used by authorities to identify the pilgrims whenever needed.

Compared to current solutions, the task of managing pilgrims is much easier for the officials in our scheme. They can monitor how many people are passing through a certain road, and thus response spontaneously to reduce traffic. They can also identify a pilgrim quickly along with complete health history during an emergency. In case of lost pilgrims, the company can track down the pilgrims easily. In summary, the proposed scheme provides a method that can readily collect information that are valuable, sometimes critical, in effectively and efficiently organizing the event.

2.2 System Components

Following are the major components of the system:

- Tags: These are "read only" tags that will be used by the pilgrim to identify themselves to the readers. These RFID tags contain secret keys, which are linked with the corresponding data in the central database such as the pilgrim's name, age, medical history, company, camp location, physical movement, etc.
- Readers: These are radio frequency devices designed to detect and read tags to obtain the stored information. These RFID readers will be used to track pilgrims' physical movement. They will be connected directly to the central database.
- Exit sensors: These sensors will be used to read the RFID tags of the pilgrims that pass through the sensor's gates. These sensors will also be directly connected to the central database. The difference of exit sensors with the readers is that readers are like antennas that have a range where they can read all the tags around it, but exit sensors are like gates that only read the tags that pass through them.
- Wireless portable scanner: These are portable RFID readers that will be used by officials to identify the pilgrims. These scanners will not be connected to the central database, but it contains the pilgrim information such as pilgrim's name, company, etc. While the scanners only contain the person specific records, the central database contains the movement pattern of the pilgrims along with personal information.
- Terminals: These terminals will be used in police stations and hospitals to retrieve the pilgrims' data and they will be connected to the central database directly. With the help of the terminal, the authority will be able to see the past movements of the pilgrim.
- Server: The server stores and provides all the necessary information regarding all the pilgrims securely.

In the remaining part of this paper, unless specifically stated, a "reader" means either a reader or an exit sensor.

3 Potential RFID Challenges

The benefits of the system can only be obtained when the system is secure, private and scalable. Since the tags are very light, heavy cryptographic operations are not possible to implement. This leads to different attacks on RFID tags and as well as readers [2]. In this section, we first present the assumptions of our scheme, and then describe the security and privacy requirements.

3.1 Assumptions

The proposed scheme has the following assumptions:

- Every tag has the ability to generate random numbers and calculating hash and keyed hash functions.
- The communication between the server and each reader is secure.
- All the tags must be initialized securely with the necessary parameters by trusted authority.

3.2 Security/Privacy Requirements

The proposed scheme is expected to satisfy the following security and privacy requirements.

Mutual Authentication The entities in communication should be able to verify each other so as to ensure that they are communicating with the expected valid entity.

Information Privacy The sensitive information about pilgrims should be protected from malicious users.

Location Privacy An adversary cannot link two communication sessions between the same tag and different readers/exit sensors. Otherwise, the adversary can trace the movement of a pilgrim.

Backward Untraceability Given that an adversary compromises a tag and obtains its current secret key, such information cannot be used to deduce previous secrets used by this tag and the previous movement information of the tag holder, i.e., the pilgrim.

Besides the security and privacy requirements, ***scalability*** is another issue that must be addressed. In this application, the system should be able to read a huge number of tags simultaneously. As a result, the scheme should require minimum computation for authenticating each tag.

A property similar to backward traceability is **forward untraceability** [3]. It means that, given that an adversary obtains a tag's current secret key through ownership transfer, such information cannot be used to deduce future secret keys used by this tag and thus monitor future movements of the tag holder, i.e., the pilgrim. For our application, the same tag will not be transferred from one pilgrim to another. Therefore, the proposed scheme do not need to provide forward traceability.

4 Secure Protocol Design

Our proposed protocol consists of two phases: *Initialization* and *Authentication*. Before describing the protocol, we present the used notations in Table 1.

4.1 Initialization

Following are the necessary operations that must be done at the beginning of the system by the trusted authority.

- Every tag T_i has to be assigned to a unique identification string u_i of l bits. Then, we have to store just the $t_i = h(u_i)$ inside every T_i.
- Server divides all the system tags T_i into different groups g_i. Then, every tag T_i has to be assigned to a specific group identification string g_i of l bits. This value has to be stored in every T_i.
- The server must hold four entries $[(u_i, t_i)_{new}, (u_i, t_i)_{old}, g_i, D_i]$ for each T_i. The first two entries are reflecting the new and old values of u_i and t_i respectively. After that, g_i represent the group index which this tag belongs to. The last entry is holding any information regarding this tag according to the used application.

Symbol	Description
l	The string length of tag identifier.
h	A hash function
f_k	A keyed hash function.
g_i	Group key of tags (l bits long).
G	Number of groups.
N	Number of tags.
i	Integer number $(1 \leq i \leq N)$.
T_i	The i^{th} tag.
D_i	Information associated with tag T_i.
u_i	A string assigned to each T_i.
t_i	T_i's identifier which is equal to $h(u_i)$.
r	random number.
x_{new}	The value of x after last updating process.
x_{old}	The value of x before last updating process.

Table 1. Notations

4.2 Authentication Process

Step 1. `Reader:` A reader generate a random number $r_1 \in \{0,1\}^l$ and then send it to T_i.

Step 2. `Tag:` The tag T_i receives r_1 from reader. Then, it generates a random number $r_2 \in \{0,1\}^l$ and then compute the following messages:

a. $M_1 = g_i \oplus r_2$

b. $M_2 = f_{g_i}(r_1 \oplus r_2)$

c. $M_3 = t_i \oplus r_2$

After that, T_i sends M_1,M_2 and M_3 to the reader.

Step 3. `Reader:` The reader receives all the three messages from T_i and then it send them with r_1 to the server.

Step 4. `Server:` It receives M_1,M_2,M_3 and r_1 from the reader. Then, it execute the following algorithm:

```
$t_i = \epsilon$
for $i = 0$ to $G$ do
  $r_2 \leftarrow M_1 \oplus g_i$
  $M_2' \leftarrow f_{g_i}(r_1 \oplus r_2)$
  if $M_2 = M_2'$ then
    $t_i \Leftarrow M_3 \oplus r_2$
  end if
  if $t_i \neq \epsilon$ then
    $M_4 \leftarrow u_i \oplus r_2$
    {send $M_4$ and $D_i$ to the reader}
    $(u_i)_{old} \leftarrow (u_i)_{new}$
    $(t_i)_{old} \leftarrow (t_i)_{new}$
```

Table 2. The Authentication Protocol

Server	Reader		Tag
$[(u_i, t_i)_{new}, (u_i, t_i)_{old}, g_i, D_i]$			$[t_i, g_i]$
	$r_1 \in \{0, 1\}^l$	r_1 --→	$r_2 \in 0, 1^l$. $M_1 = g_i \oplus r_2$ $M_2 = f_{g_i}(r_1 \oplus r_2)$ $M_3 = t_i \oplus r_2$
←-- M_1, M_2, M_3, r_1		←-- M_1, M_2,M_3	
For each g_i value: $r_2 \leftarrow M_1 \oplus g_i$ $M_2' \leftarrow f_{g_i}(r_1 \oplus r_2)$ *if $M_2 = M_2'$ then* $t_i \Leftarrow M_3 \oplus r_2$ $M_4 \leftarrow u_i \oplus r_2$			
M_4, D_i --→		M_4 --→	
$(u_i)_{old} \leftarrow (u_i)_{new}$ $(t_i)_{old} \leftarrow (t_i)_{new}$ $(u_i)_{new} \leftarrow (u_i)_{new} \oplus (t_i)_{new} \oplus r_1 \oplus r_2$ $(t_i)_{new} \leftarrow h((u_i)_{new})$			$u_i \leftarrow M_4 \oplus r_2$ *if $h(u_i) = t_i$ then* $t_i \Leftarrow h(u_i \oplus t_i \oplus r_1 \oplus r_2)$

```
        $(u_i)_{new} \leftarrow (u_i)_{new} \oplus (t_i)_{new} \oplus r_1 \oplus r_2$
        $(t_i)_{new} \leftarrow h((u_i)_{new})$
    else {$M_2 \neq M_2'$ for all $g_i$}
        {send error message to the reader and
        terminate the connection.}
    end if
end for
```

Step 5. `Reader`: The reader receives M_4 and D_i messages from the server and then it send only M_4 to the intended T_i.

Step 6. `Tag`: The Tag receives M_4 message from the reader and then it executes the following algorithm:

```
$u_i \leftarrow M_4 \oplus r_2$
if $h(u_i) = t_i$ then
    $t_i \Leftarrow h(u_i \oplus t_i \oplus r_1 \oplus r_2)$
else {$h(u_i) \neq t_i$}
    {send error message to the reader and terminate the connection.}
end if
```

5 Analysis

In this section, we analyze how the security and privacy requirements defined in Section 3.2 are achieved.

Mutual Authentication Due to the existence of a secure channel between the server and a reader and relative strong computation capability, it is trivial to achieve mutual authentication between the server and the reader. In addition, in our scheme, a reader acts as a transparent forwarder. Therefore, in the following, we view the combination of the server and a reader as a single entity, and thus simplify the discussions on mutual authentication between a tag and the server.

Upon receiving r_1 from a reader, a valid triplet (M_1, M_2, M_3) is generated from the tag *ID* (i.e., t_i) and the group *ID* (i.e., g_i). Since both t_i and g_i are unknown to the attacker, she/he cannot compute the triplet. Therefore, the server can authenticate a tag by checking the validity of the triplet. To generate a valid message from the server to a tag, an adversary has to know both r_2 and u_i. The former can be recovered from a valid triplet with the knowledge of all the group *ID*s in the system, which is, the same as the latter, stored only at the server. In other words, a tag can easily authenticate the server by checking the validity of M_4.

Information and Location Privacy In our scheme, messages transferred over an insecure channel, i.e., the channel between a tag and a reader, are indistinguishable from random bit strings, unless the receiver holds the relevant secrets, i.e., g_i and t_i. In other words, an adversary cannot deduce any confidential information through sniffing the traffic over the insecure channel.

We notice that, although an adversary fails to impersonate a reader to a tag because of the inability of generating a valid M_4 corresponding to a triplet, she/he may trigger the generation of a triplet. However, any pair of triplets that are triggered by the same r_1 and are generated by the same tag not only are indistinguishable from random bit strings, but also are unlinkable due to the fact that they are calculated from different r_2's.

Backward Untraceability In our scheme, the new secret value (i.e., t_i) is generated by hashing the previous value together with random numbers (r_1 and r_2). Even if an attacker can compromise the current secret, it will not be able to recalculate the previous secrets due to the one-way property of hash functions. Thus, the attacker is unable to backtrack either u_i or previous t_i's.

Scalability Our protocol has a reasonable overheads. We grouped the tags with different group keys so as to lower down the computational overheads of authentication at the server side. In reality, since the pilgrims will be with different companies, we can assign a group key for each company in the initialization phase. The Hajj event lasts only seven days. Hence it is unnecessary to update the group key in between. However, in case that the group key for a given company is disclosed, e.g., a tag has been compromised, this company can manually update the group key of the tags of their pilgrims.

In our protocol, the storage requirements for the server and the tag are $(4N + G)l$ and $2l$, respectively. The computational complexity at the server side is $(C_H + 2 * C_X)G + C_H + 5 * C_X$, while that at the tag side is $3 * C_H + 7 * C_X$, where C_H and C_X represent the computation cost of performing a hash and a XOR operation, respectively.

6 Related Work

The idea of using RFID tags to improve pilgrim management was first proposed by Mohandes et al. in [5]. They conducted an experiment over 1000 people. Each pilgrim was given a tag where the tag contains all the information of the pilgrim. However, they did not propose a complete model how the deployment can be done for all the pilgrims. Moreover, they did not address the security and privacy concerns of this application.

Though there have been proposed a lot of protocols by different researchers, we could not find any protocol which exactly fits all the requirements for this application. For example, Molnar and Wagner proposed an authentication protocol for Library RFID systems [6]. However, once a tag compromised the adversary can determine the past communications from the tag. Hence, the protocol is backward traceable. In [7], a one-way hash chain based authentication protocol has been proposed. This protocol is backward untraceable due to the one-way hash chain, however it is vulnerable to reply attack. Lim and Kwon proposed a RFID authentication protocol which provides both forward and backward untraceability. The protocol maintains two hash chains: one for tag anonymity and the other for server validation [3]. However, the protocol is not scalable, since the server needs to perform costly pre-computation and maintain two hash chains for each tag.

The protocol which is most suitable for this application is proposed by Song and Mitchell in [8]. Though the protocol is not resilient against forward traceability but it is secure against all the other attacks. This protocol requires to compute a hash and to verify a condition in the whole database to identify one tag. Therefore, the verification process becomes too costly when the number of tag is more than two millions. To improve the scalability, we introduced the concept of grouping of the pilgrims in our scheme, which substantially reduces the authentication delay and at the same time fulfills the security requirements.

7 Conclusion

The main contribution of this paper is two-fold. First, we designed a system making use of RFID tags to manage the Hajj event. The system provides each pilgrim a unique RFID tag which can be used to track the pilgrim in case of emergency, and the collected location information can be used to enhance the effectiveness of the present management system. We also defined the security and privacy requirements required in this application and proposed an authentication scheme accordingly. Our analysis showed that the proposed scheme can fulfill the defined requirements.

References

[1] A. Juels. RFID security and privacy: a research survey. In *IEEE Journal on Selected Areas in Communications*, 2006.

[2] A. Juels, S. Garfinkel, and R. Pappu. RFID privacy: An overview of problems and proposed solutions. In *IEEE Security and Privacy, 2005.*

[3] C. Lim and T. Kwon. Strong and robust RFID authentication enabling perfect ownership transfer. In *International Conference on Information and Communications Security, Raleigh, USA, 2006.*

[4] Kingdom of Saudi Arabia-Ministry of health. Health statistical year book, 2006. Last accessed: Jun 2008.

[5] M. Mohandes, M. A. Kousa, and A. A. Hussain. Software and hardware development of an RFID system for pilgrim identification. In *The First Saudi Intelligence Conference on Data Security, Riyadh, 2007.*

[6] D. Molnar and D. Wagner. Privacy and security in library RFID issues, practices, and architectures. In *Proceedings of the ACM CCS'04*, Washington, DC, 2004.

[7] M. Ohkuno, K. Suzki, and S. Kinoshita. Cryptographic approach to "privacy-friendly" tags. In *RFID Privacy Workshop, MIT, USA, 2003.*

[8] B. Song and C. J. Mitchell. RFID authentication protocol for low-cost tags. In *Proceedings of the WiSec'08*, Alexandria, Virginia, 2008.