一种抵抗假冒攻击的移动 RFID 双向认证协议

何 小 平
(广东培正学院数据科学与计算机学院 广东 广州 510800)

摘 要 分析现有移动双向认证协议存在的安全缺陷,提出一种能够抵抗攻击者假冒攻击的移动双向认证协议。采用计算量更小的位运算对信息进行加密,能保障信息安全的同时,亦能够降低系统整体的计算量;对于所有信息并不是单纯转发,而是通过加密后再发送,以此抵抗假冒攻击。通过形式化分析,证明了该协议的正确性;安全性分析表明,该协议能够抵抗攻击者发起的常见攻击;性能分析表明,该协议具备低计算量的特征。

关键词 物联网 射频识别 移动系统 位运算 假冒攻击 双向认证

中图分类号 TP393.08 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2020.03.055

A MOBILE RFID MUTUAL AUTHENTICATION PROTOCOL AGAINST COUNTERFEITING ATTACK

He Xiaoping
(*School of Data and Computer Science, Guangdong Peizheng College, Guangzhou 510800, Guangdong, China*)

Abstract Based on the analysis of the security flaws of existing mobile mutual authentication protocols, this paper proposes an improved mobile mutual authentication protocol, which can resist attackers' counterfeiting attacks. It used bit operation with less computation to encrypt the information, which ensured information security and reduced the overall calculation of the system. The improved protocol did not simply forward all information, but sent it after encryption, so as to resist counterfeiting attacks. The correctness of the protocol is proved by formal analysis. The security analysis of the protocol shows that the protocol can resist common attacks initiated by attackers. The performance analysis of the protocol shows that the protocol has the characteristics of low computational load.

Keywords Internet of things RFID Mobile system Bit operation Counterfeiting attack Mutual authentication

0 引 言

无线射频识别技术在很多方面都有广泛的应用,该技术在应用过程中无需与实体接触,即可识别出其中信息,适用于很多近距离通信,最典型的是 RFID 系统的应用。一般 RFID 系统包含有三个部分:标签、读写器、后台服务器[1-2]。在传统的 RFID 系统中,读写器是固定式的,即读写器与后台服务器之间的通信基于有线传输方式进行,一般认为安全可靠。随着科技的发展和社会的进步,人类的需求不断增加,传统的 RFID 系统已经无法满足人们的复杂需求,逐渐产生了移动式的 RFID 系统。在移动式的 RFID 系统中,读写器不再是固定式的,而是移动式的,即读写器与后台服务器之间的通信改为基于无线通信方式。无线通信方式因其自身固有的属性,使得该通信信道并不安全,容易被攻击者监听或进行其他类型的攻击,一般认为安全性较低[3-5]。

为能够确保移动式 RFID 系统的推广应用,必须保障通信过程中信息的安全,适用于传统 RFID 系统的双向认证协议并不能很好地运用在移动式 RFID 系统中,因此需要设计出新的能够适用于移动式 RFID 系统中的双向认证协议[6-8]。

收稿日期:2019-06-13。教育部高等教育司 2018 年第二批产学合作协同育人项目(201802068001)。**何小平**,讲师,主研领域:网络工程,信息安全。

1 相关工作

文献[9]给出一个适用于移动系统的认证协议，协议基于共享密钥机制。对协议进行分析，其因缺少标签一端对读写器一端的认证，使得攻击者可以假冒读写器给标签发送信息，从而实施假冒攻击。文献[10]基于交叉位运算提出一个移动双向认证协议，协议采用按位运算实现信息加密，能够极大程度上降低系统整体的计算量。对协议进行分析，攻击者可以通过窃听手段获取一个完整会话全部消息，可对获取的消息进行分析，协议无法提供后向安全保障，攻击者可通过获取消息分析出上一轮通话过程中部分隐私信息。

文献[11]基于物理不可克隆设计出一种移动双向认证协议，因物理不可克隆函数产生的共享密钥是无法进行复制的，使得协议具备较高的安全性。虽然无法对协议中的重要信息进行复制，但对于截获的消息进行多次重放，几轮重放之后，将会导致标签与后台数据库之间用到的共享密钥信息失去一致性，从而协议无法抵抗攻击者发起的去同步化攻击。

文献[12]在综合考虑多种因素之后提出一种双向认证协议，适用于移动式 RFID 系统。协议虽然增加了移动读写器与后台服务器之间的双向认证，但忽略了标签与后台服务器之间的双向认证，使得攻击者可以假冒其中一方进行通信，因此协议无法抵抗攻击者发起的假冒攻击。

文献[13]采用位运算进行加密，提出了一种移动双向认证协议。协议为降低系统的计算量，摒弃基于哈希函数加密的想法，采用位运算进行加密。但协议在设计过程中，未能考虑到物理入侵方式，使得攻击者可以通过物理入侵的手段获取共享密钥，从而采用反向克隆的技术发起假冒攻击，因此协议无法抵抗假冒攻击。

文献[14]基于哈希函数设计出一种移动双向认证协议，但协议存在多种安全缺陷。

基于上述问题，本文在详细分析文献[14]中协议存在的安全不足基础之上，提出一种改进的协议。改进的协议先进行通信实体之间的识别验证，若验证通过，则进行后续操作，否则，协议终止。该协议能够有效抵抗重放攻击及假冒攻击等。

2 文献[14]协议的分析

2.1 认证过程描述

文献[14]中提出的双向认证协议的流程图如图 1 所示。

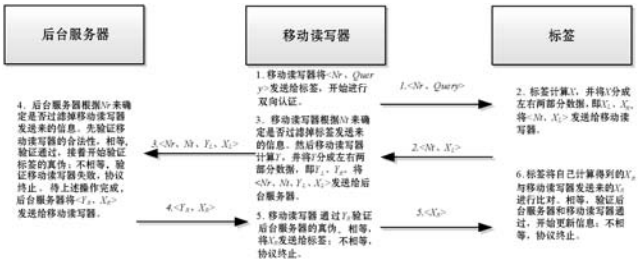

图 1 文献[14]中双向认证协议流程图

结合图 1，描述出具体的认证过程如下（有关下面描述中涉及到的符号所表示的含义，可以参考文献[14]）：

（1）移动读写器将 $<Nr、Query>$ 发送给标签，开始进行双向认证。

（2）标签计算 X，并将 X 分成左右两部分数据，即 X_L、X_R。将 $<Nt、X_L>$ 发送给移动读写器。

（3）移动读写器根据 Nt 来确定是否过滤掉标签发送来的信息。然后移动读写器计算 Y，并将 Y 分成左右两部分数据，即 Y_L、Y_R。将 $<Nr、Nt、Y_L、X_L>$ 发送给后台服务器。

（4）后台服务器根据 Nr 来确定是否过滤掉移动读写器发送来的信息。

先验证移动读写器的合法性。后台服务器计算得到一个 Y_L 与移动读写器发送来的 Y_L 进行比对。若相等，验证通过，接着开始验证标签的真伪；若不相等，验证移动读写器失败，协议终止。

再验证标签的合法性。后台服务器先利用 $<IDt_{new}、Kt>$ 计算得到一个 X_L 与移动读写器发送来的 X_L 进行比对。若相等，标签通过验证，开始更新信息；若不相等，将用 $<IDt_{old}、Kt_{old}>$ 替换 $<IDt_{new}、Kt>$ 计算得到一个 X'_L 与移动读写器发送来的 X_L 再次进行比对。若相等，表明标签通过验证，开始更新信息；若仍然不相等，标签验证失败，协议终止。

待上述操作完成，后台服务器将 $<Y_R、X_R>$ 发送给移动读写器。

（5）移动读写器将自己计算得到的 Y_R 与后台服务器发送来的 Y_R 进行比对。若相等，验证后台服务器通过，并将 X_R 发送给标签；若不相等，验证失败，协议终止。

（6）标签将自己计算得到的 X_R 与移动读写器发送来的 X_R 进行比对。若相等，验证后台服务器和移动读写器通过，开始更新信息；若不相等，协议终止。

2.2 协议具体分析

对上述协议进行分析可以发现，协议无法提供标

签对移动读写器的认证,同时协议无法抵抗攻击者发起的假冒攻击。

协议无法提供标签对移动读写器的认证具体分析如下:

结合上述协议过程中的第(5)步和第(6)步,后台服务器发送给移动读写器的 $<Y_R、X_R>$ 信息,移动读写器在对 Y_R 验证结束后,直接将 X_R 转发给标签,并没有进行任何的操作。标签接收到 X_R 后,通过 X_R 的验证,只能验证出后台服务器的真伪,无法验证出读写器的真伪,因为 X_R 的计算过程与移动读写器没有任何关系。因此,协议在最后一步无法提供标签对移动读写器的认证。

协议无法抵抗攻击者发起的假冒攻击具体分析如下:

通过上面的分析已经得出:协议无法提供标签对移动读写器的认证。基于此,攻击者可以通过监听的手段,获取移动读写器与后台服务器之间的第(4)步通信的消息,即攻击者可以监听得到 $<Y_R、X_R>$ 信息。攻击者在截获上述信息的基础之上,阻塞合法移动读写器与合法标签之间的第(5)步的通信;此时攻击者假冒成移动读写器,将截获的 X_R 信息发送给标签。标签在收到信息后,按照第(6)步进行操作,攻击者假冒的移动读写器也能够通过验证。因此,协议无法抵抗攻击者发起的假冒攻击。

针对文献[14]协议存在的安全缺陷,本文给出一种改进的能够抵抗攻击者假冒攻击的移动 RFID 双向认证协议。改进的协议主要从以下方面进行修改:(1) 协议中用到的所有数据采用密文方式进行传输,即所有信息加密后再传输;(2) 移动读写器发送给标签的信息,不再是简单的转发,而是再次加密后再传送给标签,以此应对攻击者发起的假冒攻击和提供标签对移动读写器的认证;(3) 为降低系统整体的计算量,不再采用哈希函数进行加密,而是采用计算量更少的位运算进行加密。

3　协议设计

3.1　协议的初始化及符号说明

在现有的移动 RFID 系统中,后台服务器和移动读写器两个通信实体,都具备强大的计算能力、充足的存储空间、强有力的查询能力。标签则不具备上述优势,计算能力较为薄弱,无法进行复杂的计算,存储空间受限,无法存放大量的数据[15-16]。

认证协议在开始之前会存在一个初始化过程,初始化的主要目的是完成协议开始之前密钥的分发以及安全存储等操作。具体地,在文中设计的协议中,后台服务器一端将会产生如下数据:ids_R、ids_T、key、key_T、key_R。协议在未开始之前,赋值 $ids_{T\_new}=ids_{T\_old}=ids_T$。后台服务器将会通过安全路径将信息 $<ids_R, key, key_R>$ 发送给移动读写器,并安全存放在移动读写器中;同时后台服务器也将会通过安全路径将信息 $<ids_T, key, key_T>$ 发送给标签,并安全存放在标签中;而后台服务器自身则会存放信息 $<ids_R, ids_T, key, key_T, key_R, key_{T\_old}, ids_{T\_old}>$,同时令 $key_{T\_old}=0$。

协议中出现的符号所表示的含义如表 1 所示。

表 1　协议中符号的含义

Server	后台服务器
Reader	移动读写器
Tag	标签
ids_R	*Reader* 的假名
ids_T	*Tag* 的假名
key	*Server*、*Reader*、*Tag* 三者之间的共享密钥
key_T	*Server*、*Tag* 之间的共享密钥
key_{T_old}	*Server*、*Tag* 之间上次认证的共享密钥
key_R	*Server*、*Reader* 之间的共享密钥
ids_{T_new}	*Server* 存放的当前 *Tag* 的假名
ids_{T_old}	*Server* 存放的上次认证 *Tag* 的假名
r_R	*Reader* 产生的随机数
r_T	*Tag* 产生的随机数
$\oplus$	异或运算
$Sac(X\oplus Y)$	自组合交叉位运算

为便于文中后续描述,约定用 $Sac(X\oplus Y)$ 符号表示自组合交叉位运算。下面给出自组合交叉位运算的定义:X 和 Y 表示二进制序列,长度都为偶数位,X 和 Y 进行异或运算得到一个新的二进制序列 Z,即 $Z=X\oplus Y$;Z 的最高位和最低位保持不变,从第二位开始至倒数第二位结束进行遍历,遍历过程中,将偶数位上面的那个数放在新二进制序列的奇数位上,将奇数位上的那个数放在新二进制序列的偶数位上[17]。

下面以一个具体的例子对自组合交叉位运算进行详细描述。$X=$1001 1100,$Y=$0011 0010,在这里 X 和 Y 两个二进制序列的长度均为 8,满足偶数位要求。根据上述定义,$Z=X\oplus Y=$1010 1110,$Sac(X\oplus Y)=Sac(Z)=$1101 0110。该例子的具体流程图如图 2 所示。

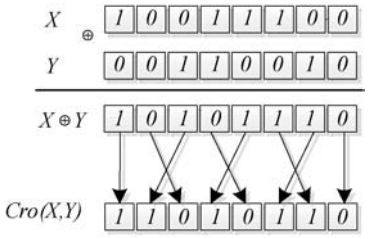

图 2 自组合交叉位运算流程图

3.2 协议描述

改进的协议流程图如图 3 所示。

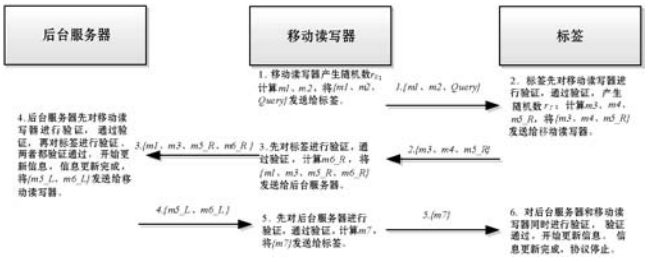

图 3 改进的协议流程图

结合图 3，可将改进的协议具体步骤描述如下：

Step 1 移动读写器产生一个随机数 r_R。用自身产生的随机数 r_R、存放的共享密钥 key 的右半部分 key_R 计算得到 $m1$；用自身产生的随机数 r_R、存放的共享密钥 key 的左半部分 key_L 计算得到 $m2$。然后将 $\{m1、m2、Query\}$ 一起发送给标签，开始进行认证协议，$Query$ 表示双向认证请求命令。

$m1$、$m2$ 按照如下规则进行计算所得到：

$m1 = r_R \oplus key\_R$、$m2 = r_R \oplus key\_L$。

Step 2 标签接收到 $\{m1、m2、Query\}$ 信息后，先对移动读写器进行验证，只有在验证通过之后，再进行后续操作，否则，协议终止。

标签利用接收到的 $m1$、存放的共享密钥 key 的右半部分 key_R 计算 $m1 \oplus key\_R$；标签利用接收到的 $m2$、存放的共享密钥 key 的左半部分 key_L 计算 $m2 \oplus key\_L$。比较 $m1 \oplus key\_R$ 的值与 $m2 \oplus key\_L$ 的值是否相等，不相等，则标签验证移动读写器失败，协议终止；反之，表示移动读写器通过标签的验证，同时可以通过计算得到移动读写器产生的随机数 r_R。

标签产生一个随机数 r_T。用自身产生的随机数 r_T、计算得到的随机数 r_R 计算得到 $m3$；用自身产生的随机数 r_T、存放的共享密钥 key 计算得到 $m4$；用自身产生的随机数 r_T、存放的共享密钥 key_T、存放的假名 ids_T 计算得到 $m5$，并将 $m5$ 分成左右位数相同的两部分 $m5\_R$、$m5\_L$。然后将 $\{m3、m4、m5\_R\}$ 一起发送给移动读写器，以此作为标签对移动读写器的响应。

r_R、$m3$、$m4$、$m5$ 按照如下规则进行计算所得到：

$r_R = m1 \oplus key\_R$（或 $r_R = m2 \oplus key\_L$）

$m3 = r_R \oplus r_T$ $m4 = Sac(r_T \oplus key)$

$m5 = Sac(r_T \oplus key_T \oplus ids_T)$

Step 3 移动读写器接收到 $\{m3、m4、m5\_R\}$ 信息后，先对标签进行验证，只有在验证通过之后，再进行后续操作，否则，协议终止。

移动读写器利用接收到的 $m3$、自身产生的随机数 r_R 计算得到一个随机数 r'_T；利用计算得到的随机数 r'_T、存放的共享密钥 key 计算得到 m'_4，然后比较计算得到的 $m4'$ 与接收到的 $m4$ 是否相等，不等，标签未通过验证，协议终止；相等，表明标签验证通过，且 $r'_T = r_T$、$m4' = m4$。

接着移动读写器再利用自身产生的随机数 r_R、存放的共享密钥 key_R、存放的假名 ids_R 计算得到 $m6$，并将 $m6$ 分成左右位数相同的两部分 $m6\_R$、$m6\_L$。最后将 $\{m1、m3、m5\_R、m6\_R\}$ 一起发送给后台服务器。

r'_T、$m4'$、$m6$ 按照如下规则进行计算所得到：

$r'_T = m3 \oplus r_R$

$m4' = Sac(r'_T \oplus key)$

$m6 = Sac(r_R \oplus key_R \oplus ids_R)$

Step 4 后台服务器接收到 $\{m1、m3、m5\_R、m6\_R\}$ 信息后，先对移动读写器进行验证，只有在验证通过之后，再对标签进行验证。当且仅当移动读写器和标签都通过验证之后，才会进行后续操作；否则，协议终止。

（1）*后台服务器对移动读写器的验证*。后台服务器利用接收到的 $m1$、存放的共享密钥 key 的右半部分 key_R 计算得到一个随机数 r'_R。利用计算得到的随机数 r'_R、存放的共享密钥 key_R、存放的假名 ids_R 计算得到 $m6'$，取 $m6'$的右半部分 $m6\_R'$，然后比较计算得到的 $m6_R'$与接收到的 $m6\_R$ 是否相等，不等，移动读写器未通过验证，协议终止；相等，表明移动读写器验证通过，且 $r'_R = r_R$、$m6' = m6$，然后后台服务器开始验证标签的真伪。

r'_R、$m6'$按照如下规则计算得到：$r'_R = m1 \oplus key\_R$、$m6' = Sac(r'_R \oplus key_R \oplus ids_R)$。

（2）*后台服务器对标签的验证*。后台服务器利用计算得到的随机数 r_R、接收到的 $m3$ 计算得到一个随机数 r'_T。用计算得到的随机数 r'_T、存放的共享密钥 key_T、存放的假名 ids_T 计算得到 $m5'$，取 $m5'$的右半部分 $m5\_R'$，然后比较计算得到的 $m5_R'$与接收到的 $m5\_R$ 是否相等，相等，表明标签通过验证，且 $r'_T = r_T$、$m5' = m5$，然后开始更新信息：$ids_{T\_old} = ids_{T\_new}$、$key_{T\_old} = key_T$、$key_T = Sac(r_T \oplus key_T)$、$ids_{T\_new} = Sac(r_T \oplus ids_{T\_old})$；不相等，后台服务器取出上一次认证用到的

key_{T_old}、ids_{T_old}替换 key_T、ids_T 计算得到 $m5''$,取 $m5''$的右半部分 $m5\_R''$,然后比较计算得到的 $m5_R''$与接收到的 $m5\_R$ 是否相等,若仍不相等,则标签未通过验证,协议终止,反之,标签通过验证,且表明标签与后台服务器之间再次恢复同步性,并开始更新信息:$key_T = Sac(r_T \oplus key_T)$、$ids_{T\_new} = Sac(r_T \oplus ids_{T\_old})$。

待信息更新完成,后台服务器取计算得到的 $m6$ 的左半部分 $m6\_L$,取计算得到的 $m5$ 的左半部分 $m5\_L$,最后将$\{m5_L、m6_L\}$一起发送给移动读写器。

r'_T、$m5'$、$m5''$按照如下规则计算得到:$r'_T = m3 \oplus r_R$、$m5' = Sac(r'_T \oplus key_T \oplus ids_T)$、$m5'' = Sac(r'_T \oplus key_{T\_old} \oplus ids_{T\_old})$。

Step 5　移动读写器接收到$\{m5_L、m6_L\}$信息后,先对后台服务器进行验证,只有验证通过之后,才会进行后续操作,否则,协议终止。

移动读写器将接收到的 $m6\_L$ 与自身之前计算得到的 $m6\_L$ 进行比较,不等,后台服务器未通过验证,协议终止;相等,后台服务器验证通过,移动读写器利用接收到的 $m5\_L$、自身产生的随机数 r_R、计算得到的随机数 r_T 计算得到 $m7$。最后将$\{m7\}$发送给标签。

$m7$ 按照如下规则进行计算所得到:$m7 = m5\_L \oplus r_R \oplus r_T$。

Step 6　标签接收到$\{m7\}$信息后,先对移动读写器和后台服务器进行验证,验证通过之后,再进行后续操作;否则,协议终止。

标签利用自身之前计算得到的 $m5\_L$、自身产生的随机数 r_T、计算得到的随机数 r_R 计算得到 $m7'$。然后标签将接收到的 $m7$ 与计算得到的 $m7'$进行比较,不等,后台服务器和移动读写器未通过验证,表明后台服务器和移动读写器两者之中,至少有一方是伪造的,协议终止;相等,后台服务器和移动读写器验证通过,接着标签一端开始更新信息:$key_T = Sac(r_T \oplus key_T)$、$ids_{T\_new} = Sac(r_T \oplus ids_{T\_old})$,待信息更新完成,标签、移动读写器、后台服务器三个通信实体之间的双向认证完成。

$m7'$按照如下规则进行计算所得到:$m7' = m5\_L \oplus r_R \oplus r_T$。

4　改进的协议安全性分析

4.1　双向认证

协议最基本的功能是能够提供通信实体之间的认证。协议中通信实体有标签、后台服务器、移动读写器三方,需提供三者之间的相互认证,本协议能够提供该功能。

(1) 标签与移动读写器之间的双向认证。协议中,标签对移动读写器的验证包含两部分:标签通过接收到的 $m1$、$m2$ 对移动读写器进行第一次验证,具体的验证方法在 Step 2 中已详细阐述;标签通过接收到的 $m7$ 对移动读写器进行第二次验证,具体验证可参见 Step 6。移动读写器对标签的验证是在 Step 3 中完成,通过接收到的 $m3$、$m4$ 对标签的验证,具体验证可参见 Step 3。

(2) 标签与后台服务器之间的双向认证。协议中,标签对后台服务器的验证在 Step 6 中完成,通过接收到的 $m7$ 对后台服务器的验证。后台服务器对标签的验证是在 Step 4 中完成,通过接收到的 $m3$、$m5\_R$ 对标签的验证,具体过程可见 Step 4。

(3) 移动读写器与后台服务器之间的双向认证。移动读写器对后台服务器的验证是在 Step 5 中完成,通过接收到的 $m6\_L$ 对后台服务器的验证,具体过程可见 Step 5。后台服务器对移动读写器的验证是在 Step 4 中完成,通过接收到的 $m1$、$m6\_R$ 对移动读写器的验证,具体过程可见 Step 4。

重点阐述最后一步骤中,标签是如何通过 $m7$ 同时完成对移动读写器和后台服务器的验证。$m7$ 中包含 $m5\_L$ 信息,该信息是后台服务器计算得到。如果后台服务器是伪造的,则不可能计算得到正确的 $m5\_L$ 信息,因此标签通过计算得到的 $m7$ 与接收到的 $m7$ 进行比较,即可识别真伪。同时 $m7$ 中还包含有 r_R、r_T 信息,该信息是移动读写器自身产生和计算所得。如果移动读写器是伪造的,则不可能得到 r_R、r_T 信息,从而计算得到的 $m7$ 也是错的。因此,基于上述,只要标签一端计算得到的 $m7$ 与接收到的 $m7$ 不相等,即可表明移动读写器和后台数据库之间,至少有一方是伪造的。

4.2　重放攻击

协议中所有信息加密过程中均混入随机数,从而可以确保前后两次通信信息的互异性。随机数的混入也能够保证通信信息的新鲜性。攻击者会通过监听的手段获取一轮的通信信息,准备通过重放该信息以获取通信实体的隐私信息,但无法成功。虽然攻击者通过重放监听得到的信息能够通过一方通信实体的认证,当通信实体回复消息给攻击者的时候,攻击者因为缺乏相对应的参数信息,使得攻击者无法破解得到隐私信息。因此,攻击者重放攻击失败,协议可抵抗攻击者的重放攻击。

4.3　去同步化攻击

去同步化攻击是指标签与后台服务器之间认证用

到的共享密钥失去一致性。具体地,标签与后台服务器两者之间有一方进行了密钥更新操作,而另一方没有进行密钥更新操作,从而使得两者之间再次认证用到的共享密钥不再一致。文中设计的协议为了能够抵抗攻击者的去同步化攻击,在后台服务器一端不仅存放当前一轮通信过程中用到的认证共享密钥 key_T 信息、标签假名 ids_{T_new} 信息,还存放有上一轮认证过程中用到的认证共享密钥 key_{T_old} 信息、标签假名 ids_{T_old} 信息。当后台服务器用 $< key_T 、ids_{T\_new} >$ 信息无法验证标签真伪时,将会用 $< key_{T\_old} 、ids_{T\_old} >$ 替换 $< key_T 、ids_{T\_new} >$ 信息再次验证标签的真伪。当且仅当上述两次验证都无法通过之时,才断定标签是伪造的。因此,协议能够抵抗攻击者发起的去同步化攻击。

4.4 暴力破解攻击

当前计算机的计算能力非常强大,攻击者在无法进行其他攻击手段获取隐私信息时,可能会用更为直接和暴力的穷举方式穷尽出隐私信息,因此协议必须能够抵抗攻击者的暴力破解攻击。协议设计过程中,所有信息都是加密之后传送,且加密过程中均混入随机数,使得攻击者无法暴力破解任何有用的隐私信息。比如:$m1 = r_R \oplus key\_R$ 中,攻击者通过监听等手段可以获取 $m1$ 信息,但 $m1$ 是密文,攻击者只有破解出该密文才可以获取隐私信息。对于攻击者来说无法穷举出,因 $m1$ 中有 r_R 和 key_R 信息是攻击者无法知晓的,且随机数 r_R 信息每次都是随机产生,无法提前预测,前后两次的 $m1$ 信息又是不相同的,因此攻击者根本无法穷举出任何隐私信息。基于上述,协议能够抵抗攻击者的暴力破解攻击。

4.5 假冒攻击

假冒攻击是指攻击者通过其他手段获取一些通信消息,然后通过重放消息的手段,以达到蒙混过关认证。协议能否抵抗假冒攻击的关键在于,协议是否能够提供通信实体之间的双向认证。在 4.1 节中已经详细分析改进的协议能够提供通信实体之间的双向认证。基于此,即便是攻击者通过监听的方式获取一些通信消息,然后通过重放手段,通过了三个通信实体其中一方的验证,并且得到该通信实体的响应信息,但攻击者因缺乏必备的参数信息,无法从接收到的响应信息中分析出有用的隐私信息,因此,协议能够抵抗攻击者发起的假冒攻击。

4.6 后向安全

后向安全是指攻击者通过一定的手段获取当前的通信消息,企图从获取的通信消息中,推导出下次的通信消息,从而分析出其中隐私信息。本文协议在设计过程中,所有通信消息都是密文传送,为保证通信消息的安全性,所有通信消息加密过程中均混入随机数。比如:$m1 = r_R \oplus key\_R$ 中,攻击者不知道 r_R 和 key_R 即便是攻击者通过监听等手段获取了本轮的 $m1$ 信息,想去推导出下轮的 $m1$ 信息,也无法成功。因为随机数 r_R 的混入使得攻击者根本无法推导出。其一,随机数每轮都是随机产生,具备无法预测性;其二,随机数每轮随机产生的时候,均是不同的。基于以上两点,攻击者无法预测下轮加密过程中用到的随机数具体数值,所以不可能推导出下轮 $m1$ 信息,因此,协议具备前向安全性。

表 2 是协议的安全性比较。

表 2 协议的安全性比较

攻击类型	文献[9]	文献[10]	文献[11]	文献[12]	文献[13]	文献[14]	本文协议
双向认证	√	√	√	√	√	×	√
假冒攻击	×	√	√	×	×	×	√
重放攻击	√	√	√	√	√	√	√
异步攻击	√	√	×	√	√	√	√
暴力破解	√	√	√	√	√	√	√
后向安全	√	×	√	√	√	√	√
拒绝服务	√	√	√	√	√	√	√
注:√表示能够抵抗,×表示无法抵抗							

5 改进的协议性能分析

RFID 系统中通信实体虽然有三个,但移动读写器及后台服务器具备强大的计算能力、充足的存放空间,而标签却不具备上述优势,因此性能分析仅选择标签作为对象,从标签一端的计算量、标签一端的存储量角度进行深入分析各协议的性能。具体分析见表 3。

表 3 协议的性能比较

指标	计算量	存储量
文献[9]	3PR + H	2l
文献[10]	8C + 14XOR	3l
文献[11]	PR + 7H + 2PUF	2l
文献[12]	5P + 3M	5l
文献[13]	P + C	3l
文献[14]	3P + 2M + 2PUF	4l
本文协议	5XOR + 4Sac	3l

表 3 中:H 表示哈希函数的计算量;PR 表示产生随机数的计算量;PUF 表示物理不可克隆函数的计算

量;P 表示伪随机数函数的计算量;M 表示模运算的计算量;C 表示交叉运算的计算量;XOR 表示异或运算的计算量;Sac 表示自组合交叉位运算的计算量。所有的通信消息长度均用 l 表示。

上述运算可以分为两种,一种是轻量级的计算量,另一种是超轻量级的计算量。H、PR、PUF、P、M 这几种运算均属于轻量级的计算;C、XOR、Sac 这几种运算均属于超轻量级的计算。根据不同量级的运算定义,轻量级的计算量一般是超轻量级的计算量的若干倍。由表 3 可知,除本文协议之外,其他协议均存在轻量级的运算,因此在标签一端的计算量方面,本文协议具有一定的优势。从标签的存储空间角度出发,可以看出本文协议与其他文献中所提出的协议在数据的存储量方面是相当的。基于性能分析和安全性分析,本文协议在标签一端的计算量方面有一定的改进和提升,同时也能够弥补其他协议存在的安全缺陷问题。

6 结　语

传统的 RFID 系统在广泛运用的同时,逐渐面临新的困境。移动式的 RFID 系统的出现能够很好地弥补传统 RFID 系统的不足,但其无法提供通信信息的安全。为了确保通信消息的安全,本文设计了适用于移动式的 RFID 系统双向认证协议。在 Wang 等所提的协议基础上,给出改进的协议。改进的协议必须在通信实体之间完成双向认证之后,才会进行后续的操作,从而可以避免攻击者的重放攻击及假冒攻击;同时为能够抵抗攻击者其他类型的攻击,通信信息均加密后再传输,且加密过程中均混入随机数,使得攻击者无法通过当前窃听的信息推导出上一轮或下次的通信消息。对协议进行安全性分析,表明协议能够满足移动式的 RFID 系统的安全需求;对协议进行性能分析,表明协议在计算量方面能够适用于当前的移动式 RFID 系统中。下一步研究方向:将加载有该协议的移动式 RFID 系统原型实现出来,研究一个完整通信所需时间及计算量等具体参数。

参 考 文 献

[1] Xie R, Jian B Y, Liu D W. An improved ownership transfer for RFID protocol[J]. International Journal of Network Security, 2018, 20(1): 149 - 156.

[2] 刘道微,凌捷. 一种改进的满足后向隐私的 RFID 认证协议[J]. 计算机科学,2016,43(8):128 - 130.

[3] Huang Z, Xu R, Chu C, et al. A novel cross layer anti-collision algorithm for slotted ALOHA-based UHF RFID systems [J]. IEEE Access, 2019, 7: 36207 - 36217.

[4] Sidorov M, Ong M T, Sridharan R V, et al. Ultralightweight mutual authentication RFID protocol for blockchain enabled supply chains[J]. IEEE Access,2019, 7: 7273 - 7285.

[5] 刘鹏. 一种 RFID 系统多标签共存证明协议设计[J]. 兵器装备工程学报,2018,39(2):124 - 126.

[6] Xu H, Shen W W, Li P, et al. Novel implementation of defence strategy of relay attack based on cloud in RFID systems [J]. IJICS,2019, 11(2): 120 - 144.

[7] Xie R, Ling J, Liu D W. Wireless key generation algorithm for RFID system based on bit operation[J]. International Journal of Network Security, 2018, 20(5): 938 - 949.

[8] Zhang Y L, Chen S G, Zhou Y, et al. Monitoring bodily oscillation with RFID tags[J]. IEEE Internet of Things Journal,2019, 6(2): 3840 - 3854.

[9] 王国伟,贾宗璞,彭维平. 基于动态共享密钥的移动 RFID 双向认证协议[J]. 电子学报,2017,45(3):612 - 618.

[10] 占善华. 基于交叉位运算的移动 RFID 双向认证协议[J]. 计算机工程与应用,2019,55(7):120 - 126.

[11] Kaul S D, Awasthi A K. Privacy model for threshold RFID system based on PUF[J]. Wireless Personal Communications, 2017, 95(3): 2803 - 2828.

[12] Sundaresan S, Doss R, Piramuthu S, et al. A secure search protocol for low cost passive RFID tags[J]. Computer Networks, 2017, 122: 70 - 82.

[13] Fan K, Jiang W, Li H, et al. Lightweight RFID protocol for medical privacy protection in IoT[J]. IEEE Transactions on Industrial Informatics, 2018, 14(4): 1656 - 1665.

[14] 汪杰,汪学明. 改进的轻量级移动 RFID 双向认证协议[J]. 计算机工程与设计,2018,39(4):912 - 917.

[15] Fan K, Zhu S S, Zhang K, et al. A lightweight authentication scheme for cloud based RFID healthcare systems[J]. IEEE Network,2019, 33(2): 44 - 49.

[16] Ibrahim A, Dalkiliç G. Review of different classes of RFID authentication protocols [J]. Wireless Networks, 2019, 25(3):961 - 974.

[17] Bai Z, He Y G. Recognition of the anti-collision algorithm for RFID systems based on tag grouping[J]. IJICT,2019, 14(1): 81 - 88.

(上接第 313 页)

[26] Fan L. A blind signature protocol with exchangeable signature sequence[J]. International Journal of Theoretical Physics, 2018, 57(12):3850 - 3858.

[27] Liang X Q, Wu Y L, Zhang Y H, et al. Quantum multi-proxy blind signature scheme based on four-qubit cluster states[J]. International Journal of Theoretical Physics,2019, 58(1):31 - 39.