第 38 卷 第 3 期
Vol.38 No.3

计 算 机 工 程
Computer Engineering

2012 年 2 月
February 2012

• 安全技术 •

文章编号：1000—3428(2012)03—0126—04 文献标识码：A 中图分类号：TP309.2

去中心化的安全分布式存储系统

贾亚茹[1]，刘向阳[2]，刘胜利[3]

(1. 河北无极中学，石家庄 052460；2. 河北省无极县教育局，石家庄 052460；
3. 上海交通大学计算机科学与工程系，上海 200240)

摘 要：提出一种去中心化的安全分布式存储系统。通过公钥加密和单钥加密相结合的方法，提高存储数据的保密性。对每个数据源使用不同的对称密钥进行分布式加密，采用分布式纠删码对加密后的数据进行编码。使用 RS 多项式编码和 List Decoding 译码方法存储加密的对称密钥，以保证系统的鲁棒性。分析结果表明，该方案的计算复杂度较低。

关键词：分布式存储；分布式纠删码；去中心化存储；加密

Decentralized Secure Distributed Storage System

JIA Ya-ru[1], LIU Xiang-yang[2], LIU Sheng-li[3]

(1. Hebei Wuji High School, Shijiazhuang 052460, China; 2. Wuji Education Bureau of Hebei Province, Shijiazhuang 052460, China;
3. Dept. of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200240, China)

【Abstract】This paper proposes the decentralized secure distributed storage system. The privacy of data is fulfilled by the combination of public key system and symmetric key system. Symmetric key for data from different sources are different, hence encryptions can be implemented in a distributed way. The employment of decentralized erasure code, together with the distributed encryptions, makes possible the decentralization of the system. RS codes are used to store the encryption of symmetric keys, and the list decoding of RS codes ensures the robust. Analysis result shows that the computing efficiency of this system is higher.

【Key words】distributed storage; distributed erasure code; decentralized storage; encryption

DOI: 10.3969/j.issn.1000-3428.2012.03.043

1 概述

随着计算机和网络技术的发展，存储模式也由个人集中式的存储发展为分布式存储。分布式存储就是将数据存储在多台独立的存储服务器上。分布式存储通过备份、冗余编码等手段，可以提高系统的可靠性、可用性和存取效率，还易于扩展[1]。因此，越来越多的用户将自己的大量数据外包给存储服务方进行分布式存储。这种模式在给用户带来方便和快捷的同时，也带来了新的问题。例如：用户如何保证数据的隐私性。为解决这个问题，研究者提出了新的方案[2]。但该方案完全使用公钥加密，导致方案的加解密效率不高，而且加密时仍然需要集中式的处理，没有实现真正意义下的“去中心化”。本文针对分布式存储，研究如何将“去中心化”和用户数据进行高效加解密结合，提出一种去中心化的安全分布式存储系统。

2 相关工作

分布式存储通常以提高可靠性为目的。为提高可靠性，存储数据的一般方法是通过编码增加冗余信息，利用纠删编码的原理，对数据进行纠删编码，将编码后的各个数据片段分布式地存储于异地的存储服务器中。

分布式存储原理如图 1 所示。具体如下：对于一个文件 M，首先将其分成 k 个等长的分组(最后一组不足可填充 0)，形成向量 $\boldsymbol{M}'=(M_1,M_2,\cdots,M_k)$，然后利用$(n,k)$纠删码对其进行编码来增加冗余，得到与 $\boldsymbol{M}'$相关的码字向量 $\boldsymbol{C}=(C_1,C_2,\cdots,C_n)$。将码字的 n 个部分分别存储在 n 个存储服务器中。数据恢复时，只需存取 n 个存储服务器中的任意 k 个即可解码恢复出原始数据 M。

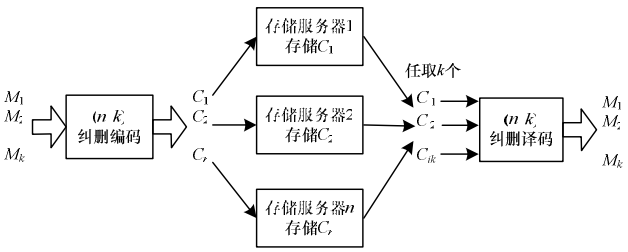

图 1 分布式存储原理

目前分布式存储的研究重在研究系统的计算复杂度、通信效率、鲁棒性等。一个好的分布式存储系统应满足计算复杂度低、通信代价小、具有较好的抗灾难性等。

在一般情况下，分布式存储系统是将分组消息形成的向量 $\boldsymbol{M}'=(M_1,M_2,\cdots,M_k)$ 集中进行编码处理的。但如果数据来源是异地的，则需先将所有的数据汇聚在某个服务器上集中进行编码处理，之后再将处理好的分组数据发送给各个异地的存储服务器。

文献[1]讨论了异地数据源(如传感器网络中的传感器采

基金项目：国家自然科学基金资助项目(60873229)；上海市青年科技启明星基金资助项目“模糊保密数据中的密钥提取和保护”(09QA1403000)

作者简介：贾亚茹(1975－)，女，中学一级教师、学士，主研方向：安全存储；刘向阳，中学一级教师、学士；刘胜利，教授、博士

收稿日期：2011-05-25 **E-mail：**slliu@sjtu.edu.cn

集的数据)分布式存储中如何去中心化，即：使数据的编码不再集中进行，而转由各个存储服务器分布式地进行。其思想是使用分布式的纠删码原理实现分布式编码以去中心化的。由于省去了集中化处理所需的汇聚步骤，编码过程转由各个存储服务器分布式地完成，因此整个系统的架构有了较大的简化。同时，由于所采用的纠删码所使用的生成矩阵实际上是一个稀疏矩阵，因此编码和译码的效率较高。去中心化的分布式存储原理如图 2 所示。

分布式存储对于如何保证数据的保密性方面研究相对较少。文献[2]在去中心化分布式存储的基础上增加数据的保密性。该文利用了特殊的具有同态性的公钥密码算法，使对密文的纠删编码可以直接进行解密后再进行纠删解码，其系统结构如图 3 所示。

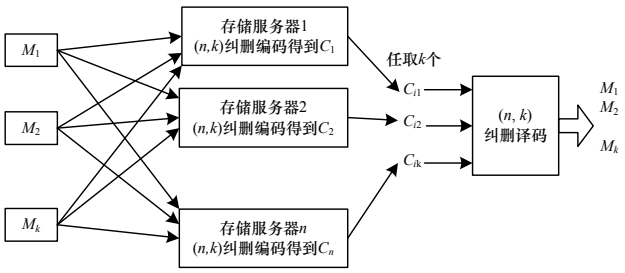

图 2　去中心化的分布式存储原理

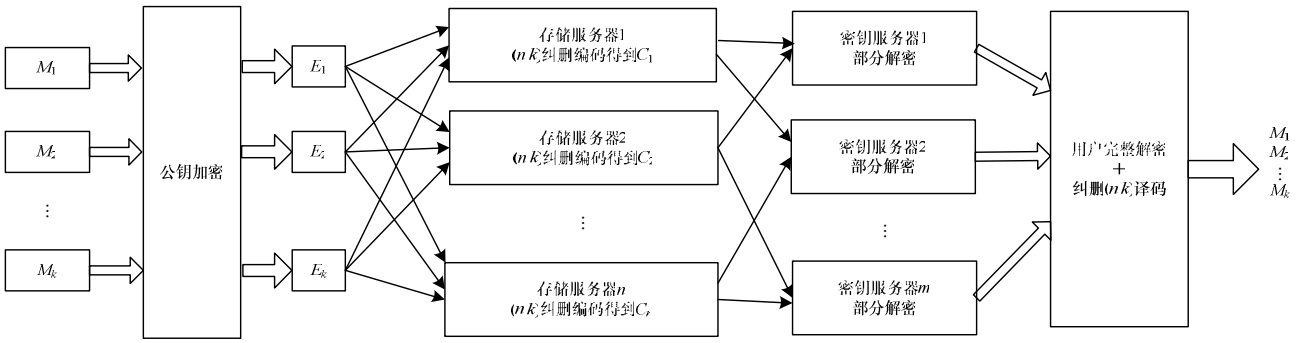

图 3　文献[2]系统结构

但公钥密码算法的算法复杂度实质是非常高的，几乎从来不会场直接应用于对大量数据加解密，而分布式存储的应用场景很多都是大量数据的存储，因此，文献[2]方案的实用性较低。该方案具有以下缺点：

(1)方案使用了公钥算法，至使其数据的加解密速度慢，难以达到实时性要求。由于公钥算法的安全性一般都基于一些数学上的困难问题，如离散对数、大整数分解等，因此公钥算法的算法复杂度高，加密和解密速度很慢，一般只是用于对少量数据(如密钥)的加密。以 RSA 公钥算法为例，它加解密所针对的数据一般是 1 024 bit。因此，对于大量的数据，只能将其以 1 024 bit 为一组进行分割，然后逐组进行加密，解密也是分组进行。而在分布式存储中，用户一般都是借助第三方服务器来存储大量数据的，这样公钥算法对大量数据的加解密都难以达到实时性的要求，更不适合于计算能力弱的网络环境(如无线传感器网等)。

(2)数据的分布存储并没有完全实现去中心化。虽然方案借助了文献[1]中的去中心化的纠删码的思想，但之前的加密操作却是集中处理的。由于所有的密文需要共享一个相同的标识 $h_{ID}=H(M_1,M_2,\cdots,M_n)$，而且该标识参与了加密运算，这就导致了方案中的加密过程不可能去中心化，因而就失去了文献[1]方案“去中心化”的特性。

(3)系统架构复杂。方案中解密时需要借助分布式的解密服务器，而且要求合谋的服务器个数不超过 t 个。在文献[2]中，解密服务器是独立于存储服务器的，专门用于为用户进行解密操作。用户的私钥通过(t, m)门限体制存储于 m 个密钥服务器中。文中假设解密服务器的安全级别比存储服务器要高：存储服务器可以允许任意多个服务器合谋，而对于解密服务器却中允许不超过 t 个服务器间的合谋。

(4)在方案的解密操作中，分布式的解密服务器需要进行的是计算复杂度很高的模指数算法，而用户最后的合成阶段，即使利用了所使用的公钥算法的同态性质也需要至少 $O(k)$次模指数操作和复杂度更高的双线性配对操作。

3　去中心化的安全分布式存储系统

本文针对文献[2]方案，提出一个新的通用的去中心化的安全存储系统，如图 4 所示。

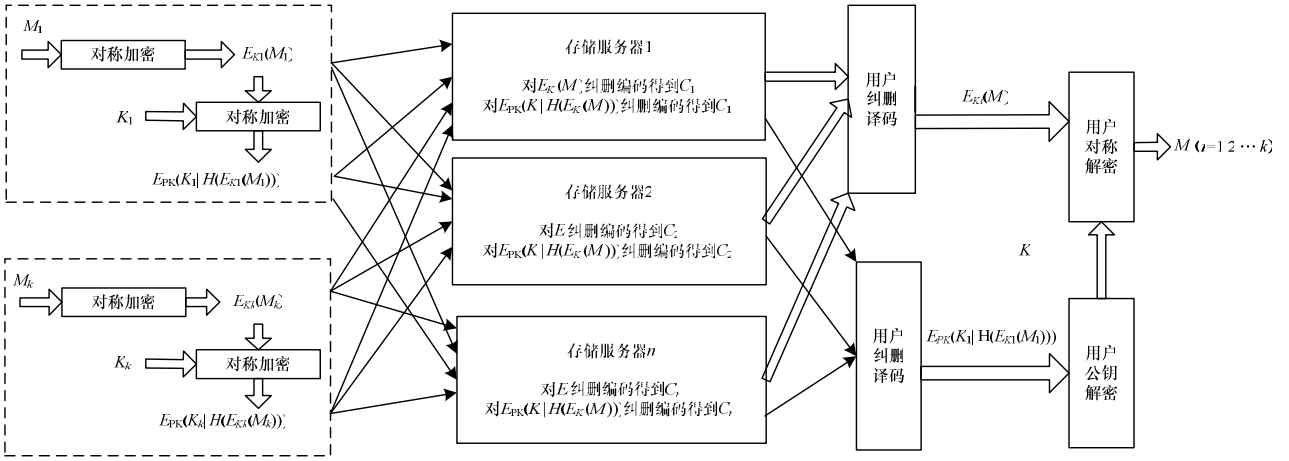

图 4　去中心化的安全分布式存储系统

具体的符号说明如下：

$E_K(M)$：利用对称密码算法，以 K 为密钥，对明文 M 进行加密；

$E_{PK}(M)$：利用公钥密码算法，以 PK 为公钥，对明文 M 进行加密；

$H(M)$：求消息 M 的 Hash 值；

$M_1, M_2, \cdots, M_k$：k 个明文分组，每个明文分组的来源可能不同(异地)，也可能相同(如对一个文件的分割)；

$M_1 \| M_2$：M_1 和 M_2 的级联；

ID_{S_j}：存储服务器的标识信息。

设 $M_i \in F_q$，本文所有的运算都在有限域 F_q 上进行。

为保证数据的安全性，本文系统将采用对称加密算法(单钥加密算法)中的分组密码对数据分组 M_i 进行加密得到密文 $E_{K_i}(M_i)$。加密时，不同的分组 M_i 采用的是不同的密钥 K_i，这也就保证了加密这一步骤可以在异地分布式地完成。而对于每个分组所使用的密钥 K_i，再利用公钥密码体制，利用用户的公钥 PK 进行加密，得到密钥 K_i 和 $E_{K_i}(M_i)$ 的 Hash 值所对应的密文 $E_{PK}(K_i \| H(E_{K_i}(M_i)))$。

对于密文 $E_{K_i}(M_i)$，使用类似文献[1]中的分布式存储思想。将 $E_{K_i}(M_i)$ 发送给存储服务器后，服务器将采用去中心化的纠删码来对其进行纠删编码得到码字分量 C_i 存储于服务器上。同时，服务器对密文 $E_{PK}(K_i \| H(E_{K_i}(M_i)))$ 则采用 RS 纠错码进行编码，存储所得的码分量 C_i'。RS 编码的好处在于即使码字中的若干分量有错误，也可以将错误纠正。

从各个存储服务器采集各个码字分量 C_i，进行纠删译码，恢复出 $E_{K_i}(M_i)$。为了从 $E_{K_i}(M_i)$ 解密得到 M_i，必须将相应的密钥 K_i 恢复出来。为了保证在分量 C_i'有多个错误的情况下也能将 K_i 恢复出来，译码时笔者将使用 list decoding 的方法[3-4]。译码得到 K_i 后，即可从 $E_{K_i}(M_i)$ 中解密恢复出原始数据分组 M_i。

具体描述如下：

(1)加密预处理：完成对数据的加密。

1)从密钥空间中随机选择密钥 K_i，计算 M_i 的密文：$E_{K_i}(M_i)$。

2)计算该密文的 Hash 值：$H(E_{K_i}(M_i))$。

3)利用用户的公钥 PK 计算 $K_i \| H(E_{K_i}(M_i))$ 相对应的密文：$E_{PK}(K_i \| H(E_{K_i}(M_i)))$。

4)所需存储的数据为：$E_{K_i}(M_i) \| E_{PK}(K_i \| H(E_{K_i}(M_i)))$。将其发送给各个存储服务器。

(2)数据的存储：任何一个存储服务器 j(j=1,2,⋯,m)都使用去中心化的纠删码对所接收到的数据编码后存储。

1)令 $D_i = E_{K_i}(M_i)$。设 $N(j)$ 是服务器 j 所得到的所有数据 D_i 的下标集合。$\forall i \in N(j)$，随机选择 $\alpha_{ij} \in F_q$，并计算 $C_j = \sum_{i \in N(j)} \alpha_{ij} D_i$。

2)令 $x_j = H(ID_{S_j})$，$D_i' = E_{PK}(K_i \| H(E_{K_i}(M_i)))$。将 D_i' 划分成 t 个域 F_q 上的元素，设 $\boldsymbol{D_i'} = (u_{i0}, u_{i1}, \cdots, u_{i(t-1)})$，其中，$u_{il} \in F_q$。令多项式 $f(x) = u_{i0} + u_{i1}x + \cdots + u_{i(t-1)}x^{t-1}$。服务器 j 计算 $C_{ji}' = f(x_j) = u_{i0} + u_{i1}x_j + \cdots + u_{i(t-1)}x_j^{t-1}$，其中 i=1,2,⋯,k。

3)将编码系数 $\{\alpha_{ij}, i \in N(j)\}$、码字分量 C_j 及码字向量 $(C_{j1}', C_{j2}', \cdots, C_{jk}')$ 存储于服务器 j 上。

(3)数据的恢复：用户利用 n 个服务器中任意 k 个服务器提供其所存储的编码系数和码字分量来恢复所存储的密文。

1)用户利用服务器 j 所存储的编码系数 $\{\alpha_{ij}, i \in N(j)\}$ 确定一个 k 维列向量 $(\alpha_{1j}, \alpha_{2j}, \cdots, \alpha_{kj})^{\mathrm{T}}$，当 $i \notin N(j)$ 时，$\alpha_{ij} = 0$。

2)K 个服务器 $j_1, j_2, \cdots, j_k$ 所确定的 k 个 k 维行向量构成一个方阵：

$$\boldsymbol{A} = \begin{pmatrix} \alpha_{1j_1} & \alpha_{1j_2} & \cdots & \alpha_{1j_k} \\ \alpha_{2j_1} & \alpha_{2j_2} & \cdots & \alpha_{2j_k} \\ \alpha_{3j_1} & \alpha_{3j_2} & \cdots & \alpha_{3j_k} \\ \vdots & \vdots & & \vdots \\ \alpha_{kj_1} & \alpha_{kj_2} & \cdots & \alpha_{kj_k} \end{pmatrix}$$

相应的 k 个码字分量组成一个码字 $(C_{j_1}, C_{j_2}, \cdots, C_{j_k})$。

3)设接收到密文 D_i 的服务器的个数为 $N(i)$。根据文献[1]中所证明的，如果 $N(i) > 5\frac{n}{k}\ln k$ 时，矩阵 $\boldsymbol{A}$ 为奇异矩阵的概率小于 $\frac{k}{q} + O(1)$。因此，只要有限域 F_q 足够大，那么矩阵 $\boldsymbol{A}$ 不可逆的概率就可以忽略。

4)用户从存储服务器中收集的码字 $(C_{j_1}, C_{j_2}, \cdots, C_{j_k})$ 中计算所存储的密文 D_i：

$(D_1, D_2, \cdots, D_k) = (C_{j_1}, C_{j_2}, \cdots, C_{j_k}) \cdot \boldsymbol{A}^{-1}$。

5)用户利用 list decoding 译码方法[3-4]从存储服务器中恢复所存储的密文 D_j'：

①从各个存储服务器中收集得到所有的码字向量 $(\boldsymbol{C}_{j1}', \boldsymbol{C}_{j2}', \cdots \boldsymbol{C}_{jk}')$，$j$=1,2,⋯,$n$。存储器中无相应分量的则设为 0。

②令 $\tau = \left\lfloor \frac{n-t+1}{2} \right\rfloor$，$v = 1$。从码字向量中得到 $(\boldsymbol{C}_{i1}', \boldsymbol{C}_{i2}', \cdots, \boldsymbol{C}_{in}')$。计算下列线性方程组：

$$\sum_{l=0}^{v} \begin{pmatrix} (\boldsymbol{C}_{i1}')^l & 0 & 0 & 0 \\ 0 & (\boldsymbol{C}_{i2}')^1 & \cdots & 0 \\ \vdots & \vdots & & 0 \\ 0 & 0 & \cdots & (\boldsymbol{C}_{in}')^l \end{pmatrix} \cdot \begin{pmatrix} 1 & x_1 & \cdots & x_1^{k_l} \\ 1 & x_2 & \cdots & x_2^{k_l} \\ \vdots & \vdots & & \vdots \\ 1 & x_n & \cdots & x_n^{k_l} \end{pmatrix} \cdot \begin{pmatrix} Q_{l,0} \\ Q_{l,1} \\ \vdots \\ Q_{l,k_l} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

其中，$k_l = n - \tau - 1 - l(t-1)$。

③设多项式 $Q_l(x) = Q_{l,0} + Q_{l,1}x + \cdots + Q_{l,k_l}x^{k_l}$，设 $Q(x,y) = Q_0(x) + Q_1(x)y + \cdots + Q_v(x)y^v$。

④分解多项式 $Q(x,y)$ 成 $y - f(x)$ 的形式，且 $f(x)$ 的次数必须小于 t。

⑤输出 v 个形如 $(C_{i1}', C_{i2}', \cdots, C_{in}')$ 的码字。

6)用户检测所译得的 $\boldsymbol{D_i'}$ 是否正确。对每个码字 $(C_{i1}', C_{i2}', \cdots, C_{in}')$ 逐个进行如下测试：

①进行 Lagrange 插值，恢复出编码前的消息 $\boldsymbol{D_i'} = (u_{i0}, u_{i1}, \cdots, u_{it})$。

②将 $\boldsymbol{D_i'}$ 看作 $\boldsymbol{D_i'} = E_{PK}(K_i \| H(E_{K_i}(M_i)))$。利用自己的私钥进行解密得到 $K_i \| H(E_{K_i}(M_i))$，验证 $H(D_i) = H(E_{K_i}(M_i))$ 是否成立。若成立则将 K_i 解密密文 $D_i = E_{K_i}(M_i)$，得到 M_i。

③如果 v 个码字都不能通过②的测试，则 l 的值加 1，转上一步的 list decoding 重新进行译码，并检测。

以上方案描述假设了每个 M_i 都是有限域 F_q 上的元素，如果 M_i 很多，可以将其分成若干个 F_q 上的元素，逐个参与以上的各个步骤。同一个 M_i 中的各块数据是共享一个对称密钥 K_i 的，而且对其密文的纠错编码时，编码时共享同一组编码系数。

4 性能分析

对本文提出的分布式存储系统的性能分析如下：

(1)本方案中通过公钥体制和对称密码体制之间的结合，且消息的各个部分消息 M_i 使用了不同的对称密钥 K_i，使得整个加密操作都可以分布式完成，实现了加密阶段的去中心化，且分布式存储服务器的存储代价与文献[2]方案的相同。

(2)计算复杂度高的公钥密码只用于加密对称密钥，而其它的大量数据都使用对称密码，使得加密步骤的计算复杂度大大降低。

(3)采用了文献[1]中的分布式纠删编码方法，实现编码阶段的去中心化。由于所采用的纠删码的生成矩阵实质上是一个随机的稀疏矩阵，因此其编码和译码的计算复杂度都小于 $O(n^3)$。

(4)从密文中恢复出明文，对称密钥的恢复非常重要。为了保证密钥的恢复，这里采用 RS 纠错码中的多项式编码[5-6]和 List Decoding 的译码方法[3-4]。纠错码的应用保证了即使某些存储服务器提供的数据有误，在不知道错误存储器的位置的情况下，只要发生服务器出错个数不超过其纠错能力 $v=\left\lfloor \frac{n-t+1}{2} \right\rfloor$，一般的 RS 译码都可以将其纠正。使用 List Decoding，其纠错能力更加强大，它可以纠 RS 码纠错能力之外的错误。通过放松与接收码字的距离，可以输出多个码字，而其中的一个必定是其中的正确解。

5 结束语

本文提出一种去中心化的安全分布式存储系统。该系统具有如下特点：(1)只需要存储服务器，而无需解密服务器，从而简化了系统的架构；(2)保证系统中的加密操作分布式进行，实现真正意义上的去中心化；(3)利用公钥密码体制和单钥密码体制相结合的传统思想，保证了数据的快速加解密；(4)无论是存储服务器还是解密服务器，都可以达到防止全合谋的特性，即无论多少个服务器合谋，其安全性都不会受到破坏。下一步工作将研究如何提高系统效率。

参考文献

[1] Dimakis A G, Prabhakaran V, Ramchandran K. Decentralized Erasure Codes for Distributed Networked Storage[J]. IEEE Transactions on Information Theory, 2006, 52(6): 2809-2816.

[2] Lin H Y, Tzeng W G. A Secure Decentralized Erasure Code for Distributed Networked Storage[J]. IEEE Transactions on Parallel and Distributed Systems, 2010, 21(11): 1586-1594.

[3] Sudan M. Decoding of Reed-solomon Codes Beyond the Error Correction Bound[J]. Journal of Complexity, 1997, 13(1): 180-193.

[4] McGowan J. Implementing Generalized Reed-solomon Codes and a Cyclic Code Decoder in GUAVA[EB/OL]. (2005-04-04). http://usna.edu/Users/math/wdj/mcgowan/mcgowan_mathhonors2004-2005.pdf.

[5] Berlekamp E R. Algebraic Coding Theory[M]. New York, USA: McGraw-Hill, 1968.

[6] Gemmell P, Sudan M. Highly Resilient Correctors for Polynomials[J]. Information Processing Letters, 1992, 43(4): 169-174.

编辑 金胡考

(上接第 125 页)

表 3 计算量对比

协议	标签	服务器
文献[5]协议	3E+4H+2PRNG	3D+13H+PRNG
本文协议	2E+4H+2PRNG	2D+11H+PRNG

表 4 存储数据对比

协议	标签	服务器
文献[5]协议	$h(ID)$, ID, k, n	$h(ID)$, ID, k, k_{old}, n, p, q
本文协议	ID, k, n	ID, k, k_{old}, n, p, q

从表 2 可以看出，本文所提出的改进协议满足系统的所有安全和隐私要求。文献[5]协议虽然也能满足要求，但其标签的计算量和存储数据量都比本文协议要大，标签需要进行 3 次二次剩余加密运算，服务器在最坏情况下需要进行 3 次二次剩余解密运算、13 次 Hash 运算，而本文协议标签只需进行 2 次加密运算，服务器则需要 2 次解密运算和 11 次 Hash 运算。文献[4-5]协议存储标签 ID 的 Hash 值，并以此作为搜索标签信息的索引。本文协议证明，直接利用标签 ID 在保证信息安全的同时同样可以避免穷搜索，存储 $h(ID)$加大了标签对内存的要求，导致标签成本提高。

6 结束语

本文在文献[4]协议的基础上，提出了一种增强型 RFID 认证协议，解决了已有协议面临的攻击和隐私问题，可有效抵抗假冒攻击，拒绝服务攻击和跟踪攻击。相对于其他改进协议，该协议中标签和服务器需要较少的计算量和存储量，符合 RFID 系统低成本、低消耗的要求。

参考文献

[1] Miles S B, Sarma S E, Williams J R. RFID Technology and Applications[M]. New York, USA: Cambridge University Press, 2008.

[2] Weis S, Sarma S, Rivest R, et al. Security and Privacy Aspects of Low-cost Radio Frequency Identification Systems[C]//Proc. of the 1st International Conference on Security in Pervasive Computing. Boppard, Germany: [s. n.], 2003.

[3] 张恒山, 常 军, 管会生. 基于混合加密方法的 RFID 安全认证协议[J]. 计算机工程, 2011, 37(1): 134-136.

[4] Chen Yalin, Chou Jue-Sam, Sun Hung-Min. A Novel Mutual-authentication Scheme Based on Quadratic Residues for RFID Systems[J]. Computer Networks, 2008, 52(12): 2373-2380.

[5] Yeh Tzu-Chang, Wu Chien-Hung, Tseng Yuh-Min. Improvement of the RFID Authentication Scheme Based on Quadratic Residues[J]. Computer Communications, 2011, 34(3): 337-341.

[6] Forouzan B A. 密码学与网络安全[M]. 马振晗, 贾军保, 译. 北京: 清华大学出版社, 2009.

编辑 金胡考