执行理想协议并得到$f_2(A,B)$后,关于A的信息熵为

$$H\left(A\middle|f_2(A,B)\right)=-\sum P\left(A=A_i\middle|f_2(A,B)\right)\times\log P\left(A=A_i\middle|f_2(A,B)\right).$$

类似地,可以定义得到$f_1(A,B)$后,关于B的信息熵,以及执行实际协议π之后关于A,B的条件信息熵.

同样得到的关于A,B的信息越多,信息熵的减少就越多.显然,理想协议执行后信息熵的减少是最少的,其他协议信息熵的减少都不会少于理想协议.因此,我们可以用比值$H_\pi\left(A\middle|f_2(A,B)\right)\Big/H\left(A\middle|f_2(A,B)\right)$衡量协议的信息泄露量.

定义 3. 对于计算多项式时间函数f的协议π,如果满足以下条件:

$$H_\pi\left(A\middle|f_2(A,B)\right)=H\left(A\middle|f_2(A,B)\right),$$

$$H_\pi\left(B\middle|f_1(A,B)\right)=H\left(B\middle|f_1(A,B)\right),$$

则认为保密地计算f的协议π与理想协议是等价的.

本文中,我们用条件概率来度量协议中 Bob 的信息泄漏量.首先以拥有单个字符的字符串的保密排序为例分析协议的信息泄露量(本文协议记为π).

假设字母表为$U=\{a,b,...,z\}$,$A=d,B=m$. A,B 在 U 上均匀分布(实际上不是均匀分布的,但分析方法相同,只是分析的计算过程稍微复杂一些).在这样的假设条件下,在保密计算前,关于 A,B 的先验概率都是 1/26,即$P(A)=P(B)=1/26$ 执行协议后 Bob 知道字符串A排在字符串B的前面,除此之外没有更多的信息,只能假设A在$\{a,b,c,d,e,f,g,h,i,j,k,l\}$上均匀分布,因此

$$P_\pi(A\mid f_2(A,B))=1/12=1/((B)_{\text{ord}}-1).$$

类似地,可以计算$P_\pi(B\mid f_1(A,B))=1/22=1/(26-(A)_{\text{ord}})$. 简单的分析计算可知, $P(A\mid f_2(A,B))=1/12$, $P(B\mid f_1(A,B))=1/22$. 所以有:

$$P_\pi(A\mid f_2(A,B))/P(A\mid f_2(A,B))=1,\ P_\pi(B\mid f_1(A,B))/P(B\mid f_1(A,B))=1.$$

其他情况下,无论是A排在字符串B的后面,或是A排在字符串B的同样位置,简单的分析都可以得出同样的结论.因此我们的协议所导致的信息泄露和理想协议的信息泄露一样,这部分信息泄露完全是函数 $f(A,B)$的信息泄露,这是无法避免的.

当字符串是多个字符而不是单个字符时,用排列组合的乘法原理也可以计算相应的条件概率,只是计算更复杂一些.经过分析可以得出如下结论,我们的协议和理想协议是等价的,泄露的信息量也和理想协议相同,都达到了最少的信息泄露.

2.5 协议1正确性分析

(1) 协议 1 在保密字符的加密过程中将计算$R_i=(R_{i1},R_{i2})$外包给云,不会影响最终的解密结果,随机变量R_α可以在解密运算中被成功消除.因为

$$R_i=(R_{i1},R_{i2})=(g^{r_i}\bmod p,h^{r_i}\bmod p),$$

$$R_j=(R_{j1},R_{j2})=(g^{r_j}\bmod p,h^{r_j}\bmod p).$$

所以可以得到

$$\begin{aligned}R_\alpha&=(R_{\alpha1},R_{\alpha2})\\&=R_i\cdot R_j\bmod p\\&=(R_{i1}\cdot R_{j1},R_{i2}\cdot R_{j2})\\&=(g^{r_i}\cdot g^{r_j}\bmod p,h^{r_i}\cdot h^{r_j}\bmod p)\\&=(g^{r_i+r_j}\bmod p,h^{r_i+r_j}\bmod p)\\&=(g^{r}\bmod p,h^{r}\bmod p).\end{aligned}$$

因此,将计算$R_i=(R_{i1},R_{i2})$外包给云服务器执行并不会影响解密结果.

(2) Alice 按照新的编码方式将字符串A表示在表格T'中,按照协议 1 将 Bob 构造的集合$\Pi(S')$解密,如果

解密的集合 $D(\Pi(S'))$ 中出现 2,则说明 Bob 按照字符串 B,从表格 T'中所取的字符至少有一个是排在字符串 A 中字符的前面,只有这样,在解密后的集合 $D(\Pi(S'))$ 中才会出现 2.同理,如果解密的集合 $D(\Pi(S'))$ 中不出现 2 且不均为 1,则说明 Bob 按照字符串 B,从表格 T'中所取的字符至少有一个是排在字符串 A 中字符的后面.如果解密的集合 $D(\Pi(S'))$ 中均为 1,那么字符串 B 为字符串 A 的子串.

因此协议 1 能够正确地判断字符串 A 和 B 按照字典序排序的位置关系.

2.6 性能分析

目前没有任何关于保密地判断两个字符串按照字典序排序位置关系的协议,因此在本节只对协议 1 进行效率分析和实验验证.本文的协议是用同态加密算法解决字符串按照字典序顺序排序的问题,基本运算都是模乘运算.

计算复杂性分析.本文在协议 1 中利用同态加密方案 E 计算 $R_i=(R_{i1},R_{i2})$是在预处理阶段由云服务器完成的,在加密过程中,只需通过对 R_i 执行一定次数的模乘运算($R_\alpha=R_i\cdot R_j \bmod p$),就可以秘密地得到 $g^r \bmod p$ 和 $h^r \bmod p$,而不需要再做复杂的模指数运算.如果忽略预处理时间,用方案 E 加密 1 次只需要进行 1 次乘法运算和 1 次模乘运算.Alice 用加密方案 E 最多加密 nt 次,解密 m 次.加密 1 次需要 1 次模乘运算,解密 1 次需要 $\lg p$ 次模乘运算,故协议 1 的计算复杂性为 $m\lg p+nt$ 次模乘运算.

通信复杂性分析.衡量通信复杂度的指标一般用协议交换信息的比特数,或者用通信轮数,在安全多方计算研究中通常用轮数.本文中协议 1 需要进行 3 轮通信.

2.7 实验数据分析

实验测试环境.Windows 10 64 位操作系统,处理器是 Intel(R)Core(TM)i5-6600 CPU @3.30GHz,内存是 8.00GB,用 Java 语言在 MyEclipse 上运行实现.本文所做模拟实验均在此环境下进行.

实验方法.我们通过模拟实验来测试本文执行协议 1 所用时间,可通过协议执行的时间来验证方案的效率.本实验以字符串 A 和字符串 B 为例,设定字符串 A 的字符个数 n=20,字符串 B 的字符个数分别为 m=1,2,…,20,对 m 的每个设定值进行 1 000 次模拟实验测试,忽略协议中的预处理时间,统计协议执行时间的平均值.图 1 描述了判断字符串排序的执行时间随字符串字符个数增长的变化规律.

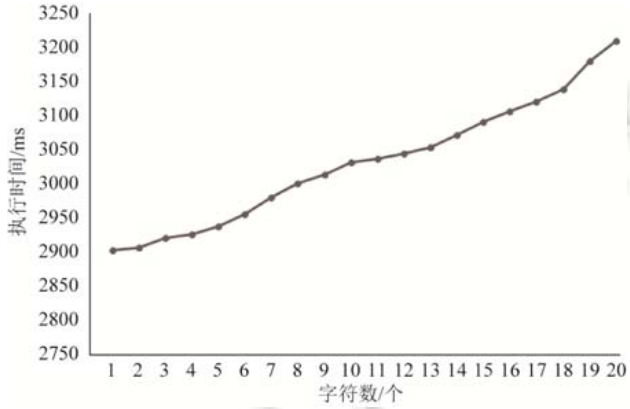

Fig.1 The execution time of the string sort increases with the number of characters in the string

图 1 字符串排序的执行时间随字符串字符个数增长的变化规律

3 应 用

本节我们将利用保密地判断两个字符串位置关系的协议来解决大数据情况下的百万富翁问题和保密地判断字符串模式匹配问题.

3.1 大数据情况下的百万富翁问题

问题描述:Alice 拥有大数据 x,Bob 拥有大数据 y.双方想在不泄露任何 x 和 y 信息的情况下知道 x 和 y 的大小关系.

我们可以将大整数 x 和 y 看成是用十进制表示的特殊类型的字符串,那么就可以将保密地判断大整数 x 和 y 的大小问题转化成保密的字符串排序问题,即通过判断字符串的位置关系来确定 x 和 y 的大小关系.假设 Alice 和 Bob 协商将大整数 x 和 y 表示成 n 位的十进制数,不足的位数分别用 0 补齐(此处为高位补 0).如:大整数 $x=x'=x_1x_2\ldots x_n$, 大整数 $y=y'=y_1y_2\ldots y_n$.

由事实 1 可知:根据集合 S 中的元素值可判断字符串 A 和字符串 B 的位置关系,于是有:如果 $x'<y'$, 那么通过调用协议 1 得到的集合 S 中的元素值没有出现 2 且不都为 1.如果 $x'=y'$, 那么集合 S 中所有元素值均为 1.如果 $x'>y'$, 那么集合 S 中出现 2.

为方便表达,定义如下谓词:

$$P(A,B)=\begin{cases}1, & x=y\\ 0, & x>y\\ 2, & x<y\end{cases}.$$

协议 2.大数据情况下的百万富翁问题.

输入:Alice 输入私有数据 x,Bob 输入私有数据 y.

输出:$P(A,B)$.

(1) 假设 Alice 和 Bob 协商将大整数 x 和 y 表示成 n 位的十进制数,不足的位数分别用 0 补齐.如:大整数 $x=x'=x_1x_2...x_n$, 大整数 $y=y'=y_1y_2...y_n$.

(2) Alice 和 Bob 调用协议 1,根据解密结果 $D(\Pi(S'))$ 中的值来判断两个大整数 x 和 y 的大小关系.如果 $D(\Pi(S'))$ 中所有的元素值均为 1,输出 $P(A,B)$=1,此时大整数 x=y;如果 $D(\Pi(S'))$ 中元素值出现 2,输出 $P(A,B)$=0,此时大整数 x>y;如果 $D(\Pi(S'))$ 中的元素值没有出现 2 且不都为 1,输出 $P(A,B)$=2,此时大整数 x>y.

协议效率分析.

在协议 2 中最多需要 $n\lg p$ 次模乘运算(n 为机密数据的长度),Alice 和 Bob 之间需要进行 3 轮通信.

文献[23]和本文的方案都可以一次性解决大数据情况下的百万富翁问题,而不需要重复调用多次基本协议,同时都对保密数据进行了编码.忽略方案中随机数选择的计算开销和双方准备阶段的计算开销,且将两个方案中的模都统一为 p 进行比较分析.

Table 2 The efficiency of the protocol 2

表 2 协议 2 性能分析与比较

	文献[23]	本文协议 2
计算复杂性	$5n\log p+4n-6$	$n\log p+2n$
通信复杂性	3	3

由表 2 可知,本文协议 2 的通信复杂性和文献[23]的通信复杂性一样,计算复杂性低于文献[23]的计算复杂性.当两个很大的数据比较大小时,p 为固定数值,$\log p$ 不会随着 n 的变化而线性增大.当 n 的取值很大时,本文中协议 2 的计算速率比文献[23]快 5 倍多.在适用范围方面,文献[23]不能完全判断两个保密数的小于和等于关系,而本文的协议 2 不仅解决了两个数比较大小的问题,也能区分两个数是否相等的问题.

实验数据分析.

实验方法.我们通过模拟实验来测试本文协议 2 和文献[23]中协议所用时间,可通过协议执行的时间来验证方案的效率.本实验假定数 A 和数 B 长度为 n,n 的变化范围为 20,21,…,40,对 n 的每个设定值进行 1 000 次模拟实验测试,忽略协议中的预处理时间,统计协议执行时间的平均值.图 2 描述了大数据情况下百万富翁协议的执行时间随机密数据长度增长的变化规律.

协议 2 的安全性依赖于保密地判断两个字符串位置关系协议的安全性,应用证明定理 1 所用的方法很容易

证明协议 2 的安全性,本文在这里省略证明过程.

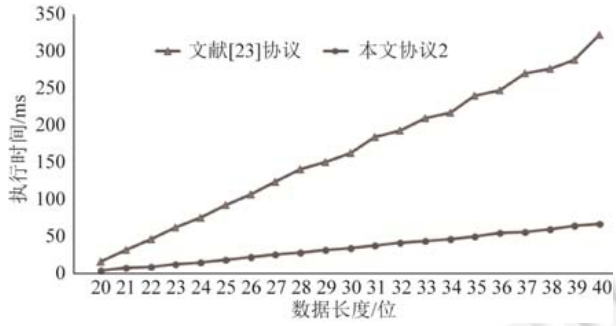

Fig.2 The execution time of the millionaires' problem increases with the number of characters in the string

图 2 大数据情况下百万富翁协议的执行时间随字符串字符个数增长的变化规律

3.2 保密地判断字符串模式匹配问题

问题描述:Alice 有一个字符串(文本串) $A=a_1\ldots a_n$, Bob 有一个字符串(模式串) $B=b_1\ldots b_m(m\leqslant n)$. 在不泄露任何 A 和 B 私有信息的情况下判断 A 中是否存在一个子串与 B 相等,也就是说,在字符串 A 中找到一个子串 $S_i=a_ia_{i+1}\ldots a_{i+m-1}$, 使 $S_i=B$.

利用协议 1 这种将保密的字符串构造为编码表的方法,适当地作处理就可以解决字符串模式匹配的问题.Alice 根据字符串 A 利用文章前面提到的编码规则构造表格 T'.根据字符串 B 中每个字符在字典集中的位置,Bob 对应地从表格 T'中第 1 行开始取值(直到第 $n-m+1$ 行为止),并将取值相乘得到结果 $E(S_i)$,记作集合 $W=\{w_1,w_2,\ldots,w_{n+m-1}\}=\{E(s_1),E(s_2),\ldots,E(s_{n+m-1})\}$. Alice 解密,如果集合 W 中出现 1,则说明字符串 A 中存在子串与字符串 B 相等.

为方便表达,定义如下谓词:

$$P(A,B)=\begin{cases}1, & \text{字符串}A\text{ 和字符串}B\text{ 匹配}\\ 0, & \text{字符串}A\text{ 和字符串}B\text{ 不匹配}\end{cases}.$$

协议 3.保密地判断两个字符串是否模式匹配.

输入:Alice 拥有保密的字符串 $A=a_1a_2\ldots a_n$,Bob 拥有保密的字符串 $B=b_1b_2\ldots b_m(m\leqslant n)$.

输出:$P(A,B)$.

(1) 令 $i=1$.

(2) Alice 调用协议 1 编码保密数据的方法构造表格 T',Bob 根据字符串 B 中每个字符在字典集中的位置对应地从表格 T'中取出相应的数据,并计算

$$E(s_i)=T'[i][(b_1)_{\mathrm{ord}}]\times\ldots\times T'[j][(b_j)_{\mathrm{ord}}]\times T'[m+i-1][(b_m)_{\mathrm{ord}}].$$

(3) 令 $i=i+1$.Bob 循环执行以上第(2)步,直到 $i=n-m+1$ 为止.

(4) Bob 将循环得到的结果 $E(s_i)$记为 $W=\{w_1,w_2,\ldots,w_{n+m-1}\}=\{E(s_1),E(s_2),\ldots,E(s_{n+m-1})\}$.为了不泄露字符串 A 中有几个子串和字符串 B 相等,再在 W 中随机选取 z 个元素值,将它们随机插入到 W 中,得到集合 $W'=\{w_1,w_2,\ldots,w_{n+m-1},w_{n+m},w_{n+m+1},\ldots,w_{n+m-1+z}\}$, 最后将集合 W'中元素做置换,将得到的结果 $\psi(W')$ 发送给 Alice.

(5) Alice 根据解密结果 $D(\psi(W'))$ 中的值来判断两个字符串是否模式匹配.如果 $D(\psi(W'))$ 中元素值出现 1,则输出 $P(A,B)=1$,此时字符串 A 中存在一个子串 $S_i=a_ia_{i+1}\ldots a_{i+m-1}$, 使得 $S_i=B$;否则,输出 $P(A,B)=0$.

协议效率分析.

本节将与 2010 年的文献[32]中所提出的同样使用 ElGamal 同态加密算法和新的保密数据编码方法解决字符串匹配问题的协议作比较.忽略方案中随机数选择的计算开销和双方准备阶段的计算开销,对两个方案进行对比.

Table 3 The efficiency of the protocol 3

表 3 协议 3 性能分析与比较

	文献[32]	本文协议 3
计算复杂性	$mn(4nk+1)\lg p$	$(n+m-1)\lg p$
通信复杂性	mn^2+mn	3

在表 3 中,n 为字符串 A 的字符个数,m 为字符串 B 的字符个数,k 为字符串 A 和 B 每个字符对应的 ASCII 值的二进制位数,p 为大素数.$mn(4nk+1)\lg p$ 表示执行文献[32]中协议需要进行的模乘次数.由表 3 可知,本文协议 3 的计算复杂性远小于文献[32],而且文献[32]的通信复杂性和字符串 A,B 的长度有关,会随着字符串长度的增加而越来越大,本文协议 3 的通信复杂性为定值,远低于文献[32].

实验数据分析.

实验方法.我们通过模拟实验来测试本文协议 3 和文献[32]协议所用的时间,可通过协议执行的时间来验证方案的效率.本实验以字符串 A 和字符串 B 为例,设定字符串 A 的字符个数为 26,字符串 B 的字符个数分别为 m=1,2,…,20,对 m 的每个设定值进行 1 000 次模拟实验测试,忽略协议中的预处理时间,统计协议执行时间的平均值.图 3 描述了字符串模式匹配的执行时间随模式串字符个数增长的变化规律.

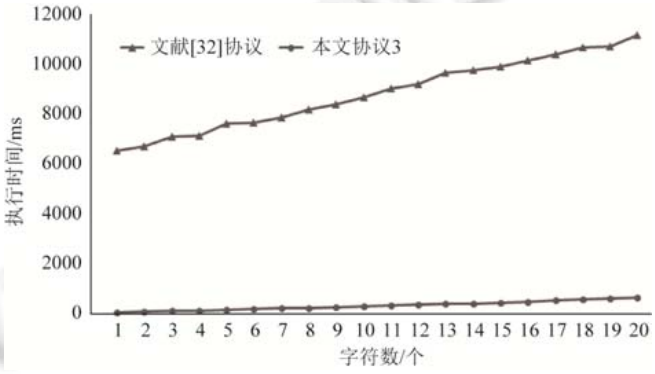

Fig.3 The execution time of the string matching problem increases with the number of characters in the string

图 3 字符串模式匹配的执行时间随模式串字符个数增长的变化规律

协议 3 的安全性可以应用证明定理 1 所用的方法,本文在这里省略证明过程.

4 结 论

字符串排序问题是安全多方计算中新的研究问题,具有重要的研究意义和应用前景.本文基于一种新的编码方式和一种云外包计算下的同态加密方案设计了两个字符串保密排序的安全多方计算协议,并利用模拟器证明了方案的安全性,同时将保密的字符串排序协议应用于解决大数据情况下的百万富翁问题和保密地判断字符串模式匹配问题.本文研究的问题都是基于半诚实模型的,对于安全多方计算的研究与应用有重要的理论意义,但恶意模型的安全性更高、更具有实际意义,所以如何实现恶意模型下的字符串保密排序问题是我们今后研究的问题.

References:

[1] Goldwasser S. Multi party computations: Past and present. In: Proc. of the 16th Annual ACM Symp. on Principles of Distributed Computing. ACM, 1997. 1–6.

[2] Goldreich O. Secure multi-party computation. Manuscript, Preliminary Version, 1998. 86–97.

[3] Freedman MJ, Nissim K, Pinkas B. Efficient private matching and set intersection. In: Proc. of the Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer-Verlag, 2004. 1–19.

[4] Lynn B, Prabhakaran M, Sahai A. Positive results and techniques for obfuscation. In: Advances in Cryptology-EUROCRYPT 2004. Berlin, Heidelberg: Springer-Verlag, 2004. 20–39.

[5] Aggarwal G, Mishra N, Pinkas B. Secure computation of the *k* th-ranked element. In: Advances in Cryptology-EUROCRYPT 2004. Berlin, Heidelberg: Springer-Verlag, 2004,3027:40–55.

[6] Fitzi M, Holenstein T, Wullschleger J. Multi-Party computation with hybrid security. In: Advances in Cryptology-EUROCRYPT 2004. Berlin, Heidelberg: Springer-Verlag, 2004,4:419–438.

[7] Ishai Y, Kushilevitz E. On the hardness of information-theoretic multiparty computation. In: Advances in Cryptology-EUROCRYPT 2004. Berlin, Heidelberg: Springer-Verlag, 2004,3027:439–455.

[8] Golle P, Juels A. Dining cryptographers revisited. In: Advances in Cryptology-Eurocrypt 2004. Berlin, Heidelberg: Springer-Verlag, 2004,3027:456–473.

[9] Yao AC. Protocols for secure computations. In: Proc. of the 23rd Annual Symp. on Foundations of Computer Science. IEEE, 1982. 160–164.

[10] Goldreich O, Micali S, Wigderson A. How to play any mental game. In: Proc. of the 19th Annual ACM Symp. on Theory of Computing. ACM, 1987. 218–229.

[11] Yasin S, Haseeb K, Qureshi RJ. Cryptography based e-commerce security: A review. Int'l Journal of Computer Science Issues, 2012,9(2):132–137.

[12] Sharma R. Review paper on cryptography. Int'l Journal of Research, 2015,2(5):141–142.

[13] Kumar SN. Review on network security and cryptography. Int'l Trans. of Electrical and Computer Engineers System, 2015,3(1): 1–11.

[14] Ioannidis I, Grama A. An efficient protocol for Yao's millionaires' problem. In: Proc. of the 36th Annual Hawaii Int'l Conf. on System Sciences. IEEE, 2003. 6–9.

[15] Li SD, Dai YQ, You QY. Efficient solution to Yao's millionaires' problem. Dianzi Xuebao (Acta Electronica Sinica), 2005,33(5): 769–773 (in Chinese with English abstract).

[16] Li SD, Wang DS. Efficient secure multiparty computation based on homomorphic encryption. Dianzi Xuebao (Acta Electronica Sinica), 2013,41(4):798–803 (in Chinese with English abstract).

[17] Sheikh R, Mishra DK, Kumar B. Secure multiparty computation: From millionaires problem to anonymizer. Information Security Journal: A Global Perspective, 2011,20(1):25–33.

[18] Grigoriev D, Shpilrain V. Yao's millionaires' problem and decoy-based public key encryption by classical physics. Int'l Journal of Foundations of Computer Science, 2014,25(4):409–417.

[19] Karimian AN. Efficient non-interactive secure two-party computation for equality and comparison [Ph.D. Thesis]. University of Calgary, 2015.

[20] Lipmaa H, Toft T. Secure equality and greater-than tests with sublinear online complexity. In: Proc. of the Int'l Colloquium on Automata, Languages, and Programming. Berlin, Heidelberg: Springer-Verlag, 2013. 645–656.

[21] Li SD, Wang DS, Dai YQ. *et al*. Symmetric cryptographic solution to Yao's millionaires' problem and an evaluation of secure multiparty computations. Information Sciences, 2008,178(1):244–255.

[22] Li SD, Wang DS, Dai YQ. Symmetric cryptographic protocols for extended millionaires' problem. Science in China Series F: Information Sciences, 2009,52(6):974–982.

[23] Lin HY, Tzeng WG. An efficient solution to the millionaires' problem based on homomorphic encryption. In: Proc. of the Int'l Conf.on Applied Cryptography and Network Security. Berlin, Heidelberg: Springer-Verlag, 2005. 456–466.

[24] Atallah MJ, Du W. Secure multi-party computational geometry. In: Proc. of the Workshop on Algorithms and Data Structures. Berlin, Heidelberg: Springer-Verlag, 2001. 165–179.

[25] Du W, Atallah MJ. Secure multi-party computation problems and their applications: A review and open problems. In: Proc. of the 2001 Workshop on New Security Paradigms. ACM, 2001. 13–22.

[26] Li SD, Wu CY, Wang DS, *et al*. Secure multiparty computation of solid geometric problems and their applications. Information Sciences, 2014,282:401–413.

[27] Li SD, Wang DS, Dai YQ. A secure multi-party computation solution to intersection problems of sets and rectangles. Progress in Natural Science, 2006,16(5):538–545.

[28] Lindell Y, Pinkas B. Privacy preserving data mining. Journal of Cryptology, 2002,15(3):177–206.

[29] Rivest RL, Adleman L, Dertouzos ML. On data banks and privacy homomorphisms. Foundations of Secure Computation, 1978,4(11):169–180.

[30] Xu MZ, You L. Information Security and Cryptology. Beijing: Tsinghua University Press, 2007. 96–98 (in Chinese).

[31] Goldreich O. Foundations of Cryptography: Volume 2, Basic Applications. London: Cambridge University Press, 2004. 599–764.

[32] Luo Y, Shi L, Zhang C, *et al*. Privacy-Preserving protocols for string matching. In: Proc. of the 4th Int'l Conf. on Network and System Security (NSS). IEEE, 2010. 481–485.

附中文参考文献:

[15] 李顺东,戴一奇,游启友.姚氏百万富翁问题的高效解决方案.电子学报,2005,33(5):769–773.

[16] 李顺东,王道顺.基于同态加密的高效多方保密计算.电子学报,2013,41(4):798–803.

[30] 徐茂智,游林.信息安全与密码学.北京:清华大学出版社,2007.96–98.

李顺东(1963－),男,河南平顶山人,博士,教授,主要研究领域为密码学,信息安全.

杨晓艺(1993－),女,硕士,主要研究领域为密码学,信息安全.

亢佳(1992－),女,硕士生,主要研究领域为密码学,信息安全.

窦家维(1963－),女,博士,副教授,主要研究领域为密码学,应用数学.