基于流水线实现的加密电路指纹认证方案①

赵 建, 冯 全, 杨 梅, 贺 康, 刘 娟

(甘肃农业大学 工学院, 兰州 730000)

摘 要: 针对计算资源受限设备在开放网络中指纹认证的隐私保护问题, 采用了基于双调排序网络的指纹细节点匹配方案, 该方案使用加密电路实现双方交集计算, 具有高的计算和通信效率; 在此基础上, 设计了流水线的实现方式避免设备存储整个加密电路, 使计算资源受限设备能够安全、有效地进行现场指纹细节点集合与注册模板集合的匹配程度进行比较. 分析表明, 该方案适合在低内存的移动、嵌入式设备上运行.

关键词: 计算资源受限; 双调排序; 加密电路; 细节点; 流水线

Pipeline Based Implementation for Fingerprint Authentication Scheme Using Garbled Circuit

ZHAO Jian, FENG Quan, YANG Mei, HE Kang, LIU Juan

(College of Engineering, Gansu Agricultural University, Lanzhou 730000, China)

Abstract: In order to protect privacy of fingerprint when remote authentication is carried out by using equipment with limited computing resource, a scheme which performs the minutiae matching based on bitonic sorting network is adopted. The protocol achieves high efficiency on computation and communication by computing private two-party set intersection problem with garbled circuit. The processing of the garbled gates is pipelined to avoid the need to store the entire circuit and to improve the running time. The proposed scheme is capable of implementing remote authentication on mobile phone or embedded equipment with less memory.

Key words: limited computing resource; bitonic sorting; garbled circuit; minutiae; pipeline

在开放网络中进行生物认证的隐私保护问题逐渐引起了研究者的关注. 隐私保护通常包含两个要点: 一是生物特征模板保护问题; 二是安全计算协议设计问题. 文献[1]总结了一些典型的生物模板保护方法. 最近一系列结合安全计算协议的生物认证方案已经被提出[2-6], 这些认证协议主要采用密码术中的同态加密和二进制加密电路(Garbled Circuit, GC). 近些年来, 一些移动、嵌入式设备采用生物识别技术与远端服务器交互以实现身份认证, 但其安全性一直为外界所诟病; 上述隐私保护协议虽能解决这一问题, 但其占用的计算和通信资源非常大, 只适合于台式机, 对计算资源相对较弱的移动、嵌入式设备并不适用. 当前同态加密算法均为非对称加密算法, 虽然其占用内存相对 GC 较少, 但是计算和通信负担很重; 而 GC 的特点是通信量和内存占用量大, 但算法相对简单. 指纹认证是最常用的一种生物认证方式, 而细节点是指纹认证中最常用的特征, 因此基于细节点的指纹认证协议非常具有实用性. 文献[7]提出了一种基于 GC 技术的指纹细节点匹配算法, 将细节点匹配视为两集合元素之间的匹配, 该算法能有效减少计算量和通信量. 虽然该算法还使用了“free XOR[8]”、“GRR[9]”等一系列优化措施, 但对于移动、嵌入式应用来说还是远远不够的. 例如文献[7]中, 设现场细节点数和模板细节点数均为 128(一般指纹细节点数不会超过一百多个), 一个细节点用 16 位二进制数表示, 每个门的每根导线采用 80 位 garbled 值, 则所有门输入导线所对应的 garbled 表和 garbled 值约 1.61MB, 这还不包括传输整条 GC 时需要执行的 OT 协议所产生的通信量; 再且非智能手机一般可使用内存不超过 4MB, 一些嵌入式设备甚至只有几百 KB. 因此本文在文献[7]的基础上进行了

① 基金项目:国家自然科学基金(61062012)

收稿时间:2013-04-17;收到修改稿时间:2013-05-06

优化并设计了流水线实现方式，使之能适应移动、嵌入式设备中内存不足的问题.

1　基于加密电路的指纹细节点匹配方案

文献[7]中将细节点量化并映射成整数，服务器端 S 的细节点模板为 $M^T=\{X_i \mid 1\leqslant i\leqslant N_T,\ 1\leqslant X_i<N_T\}$，客户端 C 采集的现场指纹细节点集合为 $M^Q=\{Y_i \mid 1\leqslant i\leqslant N_Q,\ 1\leqslant Y_i<N_Q\}$. C 通过 OT 协议从 S 获得 M^Q 对应的 garbled 值，S 将 M^T 对应的 garbled 值直接发送给 C. 随后 S 生成整个电路的 GC，C 利用该 GC 对由 M^T 和 M^Q 构成的双调序列排序，并对排序后序列的两个相邻位置用整数相等性测试电路(IET)逐个判断两个整数是否相等，若相等则进行累计(其原理是集合元素不能重复，两个集合构成的排序序列中若相邻元素相等，则说明两集合中有匹配元素)，最后得到 $|M^T\cap M^Q|$ 的 garbled 值；C 将该 garbled 值发送至 S，S 查找出其对应的真值，并判断该真值是否大于等于阈值 T，从而决定是否接受用户身份. 本方案中 C 不能获得 $|M^T\cap M^Q|$ 以及计算的中间值，因为所有计算数据均是 garbled 值.

图 1(a)所示为双调排序器结构，它有 2^m 个输入(采用 Batcher 双调排序网络进行排序[10]，M^T 升序排列，M^Q 降序排列. 匹配时 M^T 在前，M^Q 在后，构成双调序列)和 m 级. 第 k 级($1\leqslant k\leqslant m$)由 2^{k-1} 个 half-clearner 组成，每个 half-clearner 有 2^{m-k} 个比较-交换器，处理 2^{m-k+1} 个输入数据；图 1(b)给出了第 k 级的一个 half-clearner 的内部结构，图中两个较粗黑点和其间连线表示一个比较-交换器，每个比较-交换器分别对第 j 和 $2^{m-k}+j$ 个输入数据进行比较交换($1\leqslant j\leqslant 2^{m-k}$). 比较-交换器的电路结构可见文献[7].

双调序列经双调排序器排序后形成从小到大的顺序，匹配上的细节点出现在序列中相邻位置，因此可采用图 2 所示电路对匹配细节点数进行累加：先在相邻位置对两个数据采用 IET 判断是否相等，当两个数据相等时输出为 1，反之为 0. 由于 M^T 和 M^Q 都是集合，其元素不能重复，故排序后每 3 个连续位置至多有 2 个匹配元素，利用这一特点可使每 2 个 IET 接一个或门，以此减少加法器的数目.

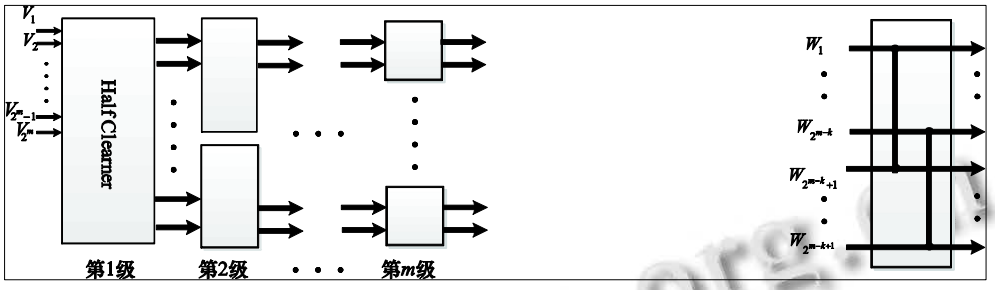

图 1(a)　双调排序器结构

图 1(b)　第 k 级 half-clearner 结构

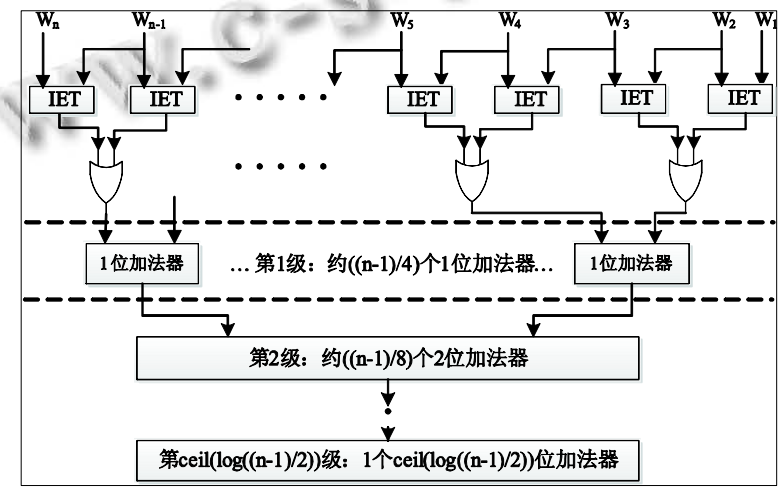

图 2　计算的电路结构

2 基于流水线的实现方式

由于采用“free XOR[8]”技术时, GC 中异或门不需要传输 garbled 表, 故在实现上述电路可尽量采用异或门. 由文献[7]可知, 电路的双调排序器含有比较-交换器个数约为(log$n$)×n/2, 即所需要的非异或门数量约为 $lnlogn$; 含有 IET 的数目为 n-1, 即需要(n-1)×(l-1)个非异或门; 所有加法器折合成 1 位加法器个数约为 $\frac{n-1}{4}+\frac{n-1}{8}\times2+\frac{n-1}{16}\times3+\cdots+\frac{n-1}{2^{ceil(\log((n-1)/2))}}\times ceil(\log((n-1)/2))$, 每个 1 位加法器需要 1 个非异或门; 其中 l 为表示 1 个细节点需要的二进制位数. 假定 n 不超过 256, l 为 16 位, 则双调排序器和 IET 部分的非异或门总量都分别占了整条电路所有非异或门的绝大部分. 通常服务器一次生成整个 GC 并传给客户端, 但为了能在移动、嵌入式设备中使用上述方案, 本文设计了上述电路的流水线实现方式(主要针对双调排序器和 IET 部分), 该流水线能根据当前客户端可供使用的内存量而作出相应的调整, 以保证每次传输给客户端的数据量满足当前客户端的内存条件.

当C要求与S进行指纹认证时, C将可使用的内存量 M 发给 S. S 按执行的层次和顺序对电路中所有门以及每个门的输入和输出编号, 建立电路描述并生成每个非异或门的 garbled 表. 由于本文中所有非异或门均采用双输入-单输出, 因此所有非异或门的 garbled 表的数据量是一样的, 设为 N_a 字节, 则整个电路大约需要 N=[$lnlogn$+(n-1)×(l-1)]×N_a 字节. S 按照电路中门的编号将电路所有门分成 ceil(N/M)批次, S 不需要一次生成这些电路的对应的 GC, 而是先生成一批 GC 并传输给 C, 然后删除自身内存中的 GC, 再生成下一批次的 GC. C 执行完每批 GC 后反馈给 S 一个完成消息, S 收到此消息后才会将下一批 GC 并发送给 C. 当一批 GC 执行完后, C 都会立即将这批 GC 以及无利用价值的 garbled 值删除, 只保留本批 GC 的输出, 作为下一批次GC的输入, 这种按照流水线的生成GC的方式可以减轻移动、嵌入式设备的内存压力.

为了实现上述功能, S 必须将每批次的电路描述传给 C, 我们采用类似 Fairplay[11]的伪代码来描述 GC, 该伪代码所占带宽相比于 garbled 表的数据量, 几乎可以忽略不计. 下面我们以一位比较电路为例, 说明如何进行电路描述, 图 3 中 x_1 为 C 的输入, y_1 为 S 的输入, c_1 为进位, c_2 为最后的输出. 图 3 电路描述如下:

1 1,

6 (0 1) (1 2) [3 4] (0 5),

6,

第 1 行为输入描述, 第一个 1 表示 C 有一根输入导线(即 w_0, 对应着 x_1 的 garbled 值), 第二个 1 表示 S 有一根输入导线(即 w_2, 对应着 y_1 的 garbled 值); 第 2 行为门描述, 6 表示最后一个异或门的输出导线 w_6(即该异或门的输出 c_2 所对应的 garbled 值, 下文所述的 w_3、w_4 和 w_5 也分别对应着各自门的输出所对应的 garbled 值), (0 1)和(1 2)分别代表导线 $w_0 \oplus w_1$ 和 $w_1 \oplus w_2$, 其中 w_1 为进位 c_1 对应的 garbled 值, [3 4]代表的是以导线 w_3 和 w_4 为输入对当前需执行的非异或门进行查表和解密计算(此处的解密并不是得到真实值, 而是得到真实值对应的 garbled 值); (0 5)代表的是导线 $w_0 \oplus w_5$; 第 3 行为需要输出的导线描述, 6 代表的是导线 w_6.

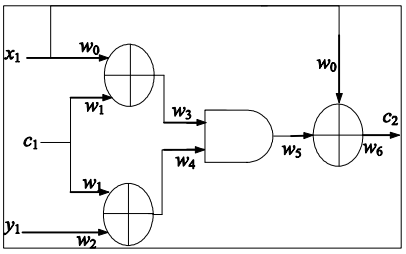

图 3 一位比较电路

3 采用的OT协议

鉴于有限的带宽资源, 本方案采用 extending OT[12]技术(计算复杂度为 $O(t^2)$)取代传统的 OT 技术(计算复杂度为 $O(xt)$), 即采用 t 次 OT 协议来取代大量的 x 次 OT 协议(x 为客户端输入二进制比特数, t 为安全参数); 考虑到移动、嵌入式设备相对较弱的计算能力, 本文进一步利用 NPOT[13]和预处理[14]将 OT 中大量的复杂计算在准备阶段完成(大量随机数的产生以及 t 次 NPOT[13]非对称运算, 事实上这些计算只需被执行一次即可, 无需每次指纹认证时都执行), 从而使执行 OT 协议时真正需要在线交互计算的只有一些简单的对称加密(约 2(x+t)次)和对称解密(约 x+t 次)运算, 具体运算流程图可见文献[6].

4 实验结果

我们使用 Huang[15]的 GC Java 库在 MyEclipse 10 环境中开发出相应的 Java 程序, 在 PC 上对本方案进

行了仿真. 我们设 N_T=N_Q=128, n=N_T+N_Q=256, l=16, 每个门的每根导线采用 80 位 garbled 值, 安全参数为 80, 通过在 PC 上的运行结果可知: N_a=0.03125KB, N=1.143MB, 本方案通过流水线只需传输总量约为 1.143MB 的 GC(即 garbled 表); 且本方案采用的 OT 协议在执行时只需占用约 86.8KB 的带宽资源(OT 协议准备阶段需占用带宽资源约 22.1KB). 假设当前要求进行指纹认证的嵌入式设备可供使用内存量为 M=300KB, 则 ceil(N/M)≈4, 即 S 会将整条 GC 按编号分成 4 个批次传输至 C.

5 结语

如今手机已经成为人们生活中必不可少的工具, 它存储着大量的个人隐私, 而这些隐私正是用户不想为外人所知的, 而本方案为手机在与外界进行交互处理时提供了可靠的安全保障, 从而极大的避免了个人隐私的泄漏. 随着手机配置和网速的提升, 本方案的可实施性将会提高.

参考文献

1 冯全,苏菲,蔡安妮.生物加密综述.计算机工程,2008,34(10): 141–143.

2 Feng Q, Su F, Cai AN. Secure remote authentication using fingerprint and fuzzy private matching. 2009 International Symposium on Intelligent Information Systems and Applications (IISA 2009). Qing Dao. Academy Publisher. 2009. 290–292.

3 Erkin Z, Franz M, Guajardo J, Katzenbeisser S, Lagendijk I, Toft T. Privacy-preserving face recognition. Privacy Enhancing Technologies Symposium (PETS'09). Berlin. Heidelberg, Springer. 2009. 235–253.

4 Sadeghi AR, Schneider T, Wehrenberg I. Efficient privacy-preserving face recognition. International Conference on Information Security and Cryptology(ICISC). Berlin, Heidelberg. Springer. LNCS. 2009, 5984. 235–253.

5 Barni M, Bianchi T, Catalano D, Raimondo MD, Labati RD, Failla P, Fiore D, Lazzeretti R, Piuri V, Scotti F, Piva A. Privacy-preserving fingercode authentication. ACM Workshop on Multimedia and Security(MM&Sec). 2010. 231–240.

6 Huang Y, Malka L, Evans D, Katz J. Efficient privacy-preserving biometric identification. 18th Network and Distributed System Security Conference(NDSS 2011). San Diego, California. Internet Society. 2011.

7 冯全,杨梅,康立军,吴丽丽,赵建,贺康.基于二进制加密电路的指纹细节点匹配.四川大学学报(工程科学版),2013,45(2): 75–80.

8 Kolesnikov V, Schneider T. Improved garbled circuit: Free XOR gates and applications. ICALP'08. Berlin, Heidelberg, Springer. 2008. 486–498.

9 Huang Y, Shen C, Evans D, Katz J, Shelat A. Efficient secure computation with garbled circuits. 7th International Conference on Information Systems Security. 2011.15–19.

10 Wang G, Luo TB, Goodrich MT, Du WL, Zhu ZT. Bureaucratic protocols for secure two-party sorting, selection, and permuting. ASIAC CS 2010. ACM. 2010. 226–237.

11 Malkhi D, Nisan N, Pinkas B, Sella Y. Fairplay—a secure two-party computation system. USENIX Security Symposium(Security'04). USENIX Association. 2004.

12 Ishai Y, Kilian J. Extending oblivious transfers efficiently. 23rd International Conference on Cryptology(CRYPTO). 2003.

13 Naor M, Pinkas B. Efficient oblivious transfer protocols. ACM-SIAM Symposium on Discrete Algorithms, 2001.

14 Beaver D. Precomputing oblivious transfer. 15th International Conference on Cryptology(CRYPTO). 1995.

15 Huang Y, Evans D, Katz J, Malka L. Faster secure two-party computation using garbled circuits. USENIX Security Symposium. 2011.