doi:10.3772/j.issn.1002-0470.2019.09.003

改进的 HABE 算法在基于雾计算的 PHR 系统中的研究①

王 璇②* 邹 军** 杜 军***

(*南京信息职业技术学院电子信息工程学院 南京 210023)

(**清华大学电机工程与应用电子技术系 北京 100084)

(***中兴光电子技术有限公司 南京 210000)

摘 要 私人健康记录(PHR)系统基于云计算提供个性化的私人健康服务。利用雾计算技术可使 PHR 系统获得更好的移动化支持,但海量的雾设备对病人隐私也造成了巨大的威胁,因此急需一套严格的数据保护与访问权限控制方案。结合密文定长机制和外包解密机制,提出了一种支持外包解密的等级化属性加密(OHABE-CC)算法。首先采用等级化的属性权威提高方案可扩展性,使之适用于雾计算的动态环境,其次任何 PHR 明文经过加密都会转换成固定长度的密文。通过外包解密机制,用户仅需通过极少的计算就可以解密 PHR 密文。基于 q-DBDH 困难假设,证明了该算法满足 IND-RCCA2 安全性。功能与解密效率分析表明,该算法相比其他现有算法更加适用于基于雾计算的 PHR 系统。

关键词 私人健康记录(PHR), 雾计算, 属性加密(ABE), 密文定长, 外包解密

0 引 言

私人健康记录(personal health record,PHR)系统[1]是一种基于云计算的健康服务,它方便病人以及医生随时随地上传、访问、分析以及使用各类健康信息。随着物联网、云计算等技术的全面发展,关于个人健康信息的数据呈现井喷式的增长,私人健康记录分享系统逐渐发挥出了在健康管理方面的优势,例如身体状态预测、疾病预防、病史分析、用药分析等功能。随着物联网应用的崛起,过于集中化的 PHR 系统使得此类数据交换伴随着巨大的传输延迟从而严重降低了 PHR 云的服务质量。2011 年,Bonomi[2]提出了雾计算的概念,所谓雾计算就是赋予路由器、传感节点、智能汽车、智能手机等位于云与用户之间的设备一定的计算能力和计算任务,在网络边缘构成一组庞大的计算集群,为用户提供更好的移动化支持。由于严重依赖设备认证与审计来维护大量的不稳定连接[3],基于雾计算构建的 PHR 系统对病人的隐私造成了巨大的威胁。因此急需一套严格的数据保护与访问权限控制技术,使之既可以实现数据的安全加密,又能够方便加密者们自己自由地制定各种各样的访问策略。

基于属性的加密算法[4,5](attribute-based encryption,ABE)是近些年提出的一种功能加密体制,它的加解密过程与一套访问策略以及属性集合紧密相关,只有属性集合与访问策略之间足够地相似才能够正确地解密密文,如此就能够在多用户环境下实现安全的数据共享。ABE 不仅为加密信息提供了灵活的访问控制功能,还具备一定的容错性质,因此 ABE 在 PHR 系统当中有着广阔的应用前景[6,7],填补了传统密码学在这方面的应用空白。当前国内外学者在该领域进行了许多相关研究,文献[8]将 PHR 系统划分为公有域与私有域,从而较好地提高

① 国家自然科学基金(61872423)和江苏高校品牌专业建设工程(PPZY2015C242)资助项目。

② 女,1971 年生,硕士,副教授,高级工程师;研究方向:信息安全,密码学;联系人,E-mail: wangxuan_njcit@163.com

(收稿日期:2018-11-19)

PHR 数据分享的秘钥管理效率。文献[9]基于等级属性加密(hierarchical attribute-based encryption, HABE)设计了一种支持匿名的医疗数据访问控制算法,该方案能够充分保护用户的隐私。文献[10]提出了一种匿名密钥发布机制,使得属性权威在发布密钥过程中无法得知 PHR 系统用户的任何属性,从而保护了 PHR 系统用户的隐私。文献[11]提出了一种改进的代理重加密方法,使得代理人可以重定义 PHR 密文的访问策略。文献[12]提出了一种隐藏访问策略的 HABE 算法,使得密文当中的访问策略对 PHR 系统用户透明,避免访问策略暴露解密者身份的风险。

密文定长机制是一种对加密算法的优化,使得密文的尺寸常量化从而适应带宽受限的传输环境。在大部分 ABE 方案当中,密文尺寸的复杂度通常为 $O(|S|)$,其中 $|S|$ 为密文所包含的属性数量。这使得在涉及海量属性的 ABE 方案当中,密文随着属性数量的剧增而逐渐丧失可用性。密文定长机制通过代数原理可以将密文尺寸复杂度控制在 $O(1)$,从而极大地提高了 ABE 方案的可用性,尤其是在带宽受限的环境当中。文献[13]首次提出了一种密文定长机制,通过该方法将基于身份加密(identity-based encryption,IBE)的密文尺寸常量化。文献[14]提出了一种密文定长的 CP-ABE 算法,不过仅支持"与"门访问策略因而表达性有限。文献[15]提出了一种密文和密钥尺寸复杂度都为 $O(1)$ 的 CP-ABE 算法。尽管密文定长机制有利于提升算法的可用性,但是现有 ABE 方案的解密过程仍然需要执行大量的幂乘和双线性对运算,很难适用于性能受限的设备当中。

外包解密机制在保证数据安全性的同时,能够有效地将绝大部分的解密工作外包给外部服务器处理[16]。为了减轻移动设备的解密负担,文献[17]首次提出了可外包解密的 ABE 方案。文献[18]提出了一种可认证的外包解密方案。Lin 等人[19]进一步优化了外包认证的开销。文献[20]提出了一种支持离线加密的可外包 ABE 方案。因此构造一种适用于轻量级设备的 ABE 机制,使得加密和解密能够同时适用于轻量级设备,将有助于 ABE 在 PHR 系统中的广泛应用。

本文结合密文定长机制和外包解密机制,提出了一种支持外包解密的定长密文 HABE 算法(outsourced hierarchical attribute based encryption with constant ciphertext,OHABE-CC),并基于 q-DBDH 假设证明了该算法的安全性,该算法非常适用于构建基于雾计算的 PHR 系统访问控制系统。首先该算法继承 HABE 的优势,将属性权威的权力与运算负载分散并提高其可扩展性,使之适用于动态变化的雾计算环境。其次在加密过程中采用阈值门构建访问策略,并利用密文定长技术产生尺寸复杂度为 $O(1)$ 的密文。在此基础上利用外包解密技术修改密钥生成算法,首先输出一个原始私钥,用户在请求解密时利用密钥转换算法输出一个转换私钥与一个 Elgmal 型密钥。其中转换私钥交由解密服务器,而 Elgamal 型私钥由用户保存。解密服务器启动密文转换算法,利用转换私钥将密文转换为 Elgmal 型密文。最后用户仅需执行一次幂乘、一次乘法就可以恢复明文。

1　双线性映射与困难假设

1.1　双线性映射

定义 1 双线性映射(Bilinear Map):设 G_1 和 G_2 是两个阶为大素数 p 的循环群,g 是 G_1 的一个生成元,称映射 $e: G_1 \times G_1 \rightarrow G_2$ 是关于 G_1 和 G_2 的双线性映射,当且仅当 e 满足以下性质:

(1) 双线性。对于任意的 $u,v \in G_1$ 以及 $a,b \in Z_p$,都有 $e(u^a,v^b) = e(u,v)^{ab}$;

(2) 非退化性。存在生成元 g,使得 $e(g,g) \neq 1$,其中 1 是 G_2 的单位元;

(3) 可计算性。对于任意的 $u,v \in G_1$,存在多项式时间算法能够有效计算出 $e(u,v)$ 的值。

1.2　困难假设

定义 2 q-DBHE 假设(decisional q-parallel bilinear Diffie-Hellman exponent assumption):假设 G_1 是一个根据安全参数生成的阶为素数 p 的乘法循环群,$e: G_1 \times G_1 \rightarrow G_2$ 是从群 G_1 到群 G_2 的一个双线性映射,$\alpha \in Z_p$ 是一个秘密随机数。若已知包含了群

G_1 中 $2q+1$ 个元素的一组参数 $\{g, h, g^{\alpha}, g^{\alpha^2}, \cdots, g^{\alpha^q}, g^{\alpha^{q+2}}, \cdots, g^{\alpha^{2q}}\}$ 以及一个参数 Z，那么该敌手难以判断 Z 是否等于 $e(g,h)^{\alpha^{q+1}}$。

2 OHABE-CC 算法框架与安全模型

2.1 算法框架

ABE 算法内在的模糊匹配性质使其同时提供了强大的安全保护以及灵活的访问控制机制。本文所提出的 OHABE-CC 算法继承了 HABE 算法的优势，在 ABE 算法的基础上有效分散属性权威的权力与运算负载，可有效适应动态变化的计算环境。不仅如此，OHABE-CC 算法的密文长度复杂度为 $O(1)$，密文仅包含 5 个元素且不随属性数量的增加而增加。同时基于外包解密技术重新设计了密钥生成算法和解密算法，最后用户仅需通过一个次乘就可以恢复明文。因此该算法非常适用于构建基于雾计算的 PHR 系统访问控制系统。一个 OHABE-CC 算法包括以下子算法：系统设置算法、局部属性权威授权算法、用户授权算法、加密算法、密钥转换算法、密文转换算法、解密算法。

系统设置算法 输入安全参数 λ 以及全局属性集合 Ω，输出公钥 PK 以及根密钥 RSK。

局部属性权威授权算法 该算法是一个迭代算法，初次执行时输入局部属性集合 Ω_1 以及根密钥 RSK，输出一级局部属性权威私钥 SK_1；否则输入局部属性权威 Ω_{i+1} 以及属性权威的私钥 SK_i，其中 $i>1$，最终输出新的局部属性权威私钥 SK_i。该算法允许任意的属性权威不断向下授权，靠近用户侧的雾计算设备能够充分发挥边缘计算能力，为用户提供可靠、快捷的授权服务。

用户授权算法 输入属性集合 S 以及局部属性权威私钥 SK_i，最终输出用户私钥 SK_u。

加密算法 输入访问策略 $\gamma_{t,S}$、消息明文 M 以及公钥 PK，最终输出消息密文 CT。该算法中无论访问策略、消息明文以及公钥如何变化，消息密文的长度总是固定的，该特性保证了算法在涉及海量属性的系统中具备更高的加解密效率。

密钥转换算法 输入用户私钥 SK_u，输出转换私钥 TK 以及 Elgamal 型私钥 EK。

密文转换算法 输入消息密文 CT 以及转换私钥 TK，输出转换密文 CT'。

终端解密算法 输入转换密文 CT' 以及 Elgamal 型私钥 EK，输出消息明文 M。用户只需要进行两步简单的计算就可以获取明文。

2.2 安全模型

OHABE-CC 的安全模型由一个在敌手 A 和挑战者 C 之间进行的挑战游戏来定义。定义如下：

初始化阶段 A 产生挑战访问策略 γ_{t^*,S^*}，将之发送给 C。

系统设置阶段 C 运行系统设置算法，秘密保存根密钥 RSK，并将公钥 PK 发送给 A。

询问阶段 1 A 随机地、有限次地对 C 发送以下 2 种询问：

(1) 授权询问。A 提交属性集合 S 给 C，C 首先运行用户授权算法生成用户私钥 SK_u，然后运行密钥转换算法生成转换私钥 TK，如果 $\gamma_{t^*,S^*}(S)\neq 1$ 则将 SK_u 发送给 A，否则将 TK 发送给 A。

(2) 解密询问。A 提交密文 CT 给 C，C 运行解密算法将解密结果发送给 A。在该询问中，A 不可以询问关于挑战访问策略关联的密文的解密请求。

挑战阶段 A 提交等长的消息明文 M_0 和 M_1 给 C，C 随机地产生一个比特 $b\in\{0,1\}$ 并根据 γ_{t^*,S^*} 对 M_b 执行加密算法，生成挑战密文 CT^* 并发送给 A。

询问阶段 2 本阶段与询问阶段 1 相同。

猜测阶段 A 给出一个比特 b' 作为对 b 值的猜测，若 $b=b'$ 则 C 输出 1 表述 A 赢得游戏，否则输出 0 表示 A 挑战失败。

根据上述挑战游戏，敌手 A 赢得上述挑战游戏的优势定义为

$$Adv_{HABE\text{-}CC}^{CCA2}(\lambda)=\Pr[b=b']-\frac{1}{2}$$

定义 3 OHABE-CC 安全性：如果敌手 A 在多项式时间内无法以不可忽略的优势赢得上述挑战游戏，那么称 OHABE-CC 在选择访问策略模型下能够抵御自适应的选择密文攻击(selective security under adaptive chosen ciphertext attacks)。

3　OHABE-CC 算法

本节首先给出一种 OHABE-CC 算法中各个子算法的详细描述。

3.1　系统设置

系统设置算法输入安全参数 λ，输出公钥 PK 以及根密钥 RSK，其中 PK 以广播形式公开，而 RSK 作为秘密保存。算法的具体流程如下：

(1) 创建长度为 λ 比特的随机的大素数 p；

(2) 创建阶为 p 的整数循环群 Z_p^*；

(3) 选择双线性映射 $e: G_1 \times G_1 \rightarrow G_2$，其中 G_1 和 G_2 分别是大素数 p 为阶的加法循环群和乘法循环群；

(4) 创建 g 为 G 的一个生成元。

(5) 创建全局属性集合 $\Omega = \{att_1, att_2, \cdots, att_N\}$，其中 att_i 为系统中的任意真实属性字段；

(6) 创建傀儡属性集合 $\Omega = \{att_{N+1}, \cdots, att_{2N-1}\}$；

(7) 创建一个函数 $index()$，对于任意属性 $att_j \in \Omega \cup \Omega'$，$index(att_j)$ 返回其索引 j。

(8) 创建 Z_p^* 上的随机元素 x；

(9) 创建根密钥 RSK 并赋值 $RSK = x$；

(10) 创建 G_1 上的元素 g_1 并赋值 $g_1 = g^x$；

(11) 创建一组 G_1 上的随机元素：

$\{g_2, h_1, h_2, \cdots, h_{2N}, \delta_1, \delta_2, \delta_3\}$；

(12) 创建 G_2 上的元素 Z 并赋值 $Z = e(g_1, g_2)$；

(13) 选择一个哈希函数算法 $H: \{0,1\}^* \rightarrow Z_p^*$；

(14) 输出公钥 PK，具体如下为

$$PK = \{g, g_1, g_2, Z, h_0, h_1, \cdots, h_{2N-1}, \delta_1, \delta_2, \delta_3, H\}$$

3.2　局部属性权威授权

局部属性权威授权算法初次执行时输入局部属性集合 Ω_1 以及根密钥 RSK，输出一级局部属性权威的授权私钥 SK_1。非初次执行时，输入局部属性权威 Ω_{i+1} 以及授权私钥 SK_i，其中 $i > 1$，输出新的局部属性权威的授权私钥 SK_i。执行流程如下：

(1) 判断申请授权的局部属性权威 A_i 的等级 i，如果 $i = 1$ 跳转至第 2 步，否则跳转至第 8 步；

(2) 输入属性集合 Ω_1、根密钥 RSK 以及公钥 PK，创建一个次数为 $N - 1$ 的随机多项式 q 使得 $q(0) = x$；

(3) 对于任意属性 $att_j \in \Omega_1 \cup \Omega'$，创建 Z_p^* 上的随机元素 $r_{1,j}$；

(4) 创建 G_1 上的元素 $a_{1,j}$ 并赋值 $a_{1,j} = g_2^{q(j)}(h_0 h_j)^{r_{1,j}}$；

(5) 创建 G_1 上的元素 $b_{1,j}$ 并赋值 $b_{1,j} = g^{r_{1,j}}$；

(6) 创建 G_1 上的一组元素：

$c_{1,j,1}, \cdots, c_{1,j,j-1}, c_{1,j,j+1}, \cdots, c_{1,j,2N-1}$，

其中 $c_{1,j,1} = h_1^{r_{1,j}}$，$c_{1,j,2} = h_2^{r_{1,j}}$，并以此类推。

(7) 创建授权私钥 $SK_1 = \{sk_{1,j}\}_{att_j \in \Omega_1 \cup \Omega'}$，其中：

$sk_{1,j} = \{a_{1,j}, b_{1,j}, c_{1,j,1}, \cdots, c_{1,j,j-1}, c_{1,j,j+1}, \cdots, c_{1,j,2N-1}\}$

跳转至第(14)步。

(8) 输入属性集合 Ω_i、公钥 PK 以及上一级局部属性权威私钥 $SK_{i-1} = \{sk_{i-1,j}\}_{att_j \in \Omega_1 \cup \Omega'}$，其中：

$sk_{i-1,j} = \{a_{i-1,j}, b_{i-1j}, c_{i-1,j,1}, \cdots, c_{i-1,j,j-1}, c_{i-1,j,j+1}, \cdots, c_{i-1,j,2N-1}\}$

(9) 对于任意属性 $att_j \in \Omega_i \cup \Omega'$，创建 Z_p^* 上的随机元素 $r_{i,j}$；

(10) 创建 G_1 上的元素 $a_{i,j}$ 并赋值 $a_{i,j} = a_{i-1,j}(h_0 h_j)^{r_{i,j}}$；

(11) 创建 G_1 上的元素 $b_{i,j}$ 并赋值 $b_{i,j} = b_{i-1,j} g^{r_{i,j}}$；

(12) 创建 G_1 上的一组元素：

$c_{i,j,1}, \cdots, c_{i,j,j-1}, c_{i,j,j+1}, \cdots, c_{i,j,2N-1}$，

其中 $c_{i,j,1} = c_{i-1,j,1} h_1^{r_{i,j}}, c_{i,j,2} = c_{i-1,j,2} h_2^{r_{i,j}}$，并以此类推。

(13) 创建授权私钥 $SK_i = \{sk_{i,j}\}_{att_j \in \Omega_i \cup \Omega'}$，其中：

$sk_{i,j} = \{a_{i,j}, b_{i,j}, c_{i,j,1}, \cdots, c_{i,j,j-1}, c_{i,j,j+1}, \cdots, c_{i,j,2N-1}\}$

(14) 输出 A_i 的授权私钥。

3.3　用户授权

用户授权算法输入属性集合 S 以及局部属性权威私钥 SK_i，最终输出用户私钥 SK_u。流程如下：

(1) 对于任意属性 $att_j \in S \cup \Omega'$，创建 Z_p^* 上的一个随机元素 $r_{i+1,j}$；

(2) 创建 G_1 上的元素 $a_{i+1,j}$ 并赋值 $a_{i+1,j} = a_{i,j}(h_0 h_j)^{r_{i+1,j}}$；

(3) 创建 G_1 上的元素 $b_{i+1,j}$ 并赋值 $b_{i+1,j} = b_{i,j} g^{r_{i+1,j}}$；

(4) 创建 G_1 上的一组元素：

$c_{i+1,j,1}, \cdots, c_{i+1,j,j-1}, c_{i+1,j,j+1}, \cdots, c_{i+1,j,2N-1}$，

其中 $c_{i+1,j,1}=c_{i,j,1}h_1^{r_{i+1,j}}$，$c_{i+1,j,2}=c_{i,j,2}h_2^{r_{i+1,j}}$，并以此类推；

(5) 输出授权私钥 $SK_u=\{sk_{i+1,j}\}_{att_j\in S\cup\Omega'}$，其中：

$$sk_{i+1,j}=\{a_{i+1,j},b_{i+1,j},c_{i+1,j,1},\cdots,c_{i+1,j,j-1},c_{i+1,j,j+1},\cdots,c_{i+1,j,2N-1}\}$$

3.4 加 密

加密算法输入公钥 PK、消息明文 M 以及访问策略 $\gamma_{t,S}$，并输出密文 CT。下面对加密算法进行详细的阐述。

对于访问策略 $\gamma_{t,S}$ 而言，它满足 $S\subseteq\Omega$ 并且 $1\leqslant t\leqslant|\Omega|$，表示访问策略由一个全局属性集合 Ω 的真子集 S 构成，一个属性集合只有包含了 S 当中不少于 t 个的属性，才能满足这个访问策略并且最终能够正确地获取密文。例如一个访问策略为

$$\gamma_{t,S}=(t=n',S=\{att_1,att_2,\cdots,att_n\})$$

那么属性集合 $S=\{att_3,att_4,\cdots,att_{n'+2}\}$ 则满足这个访问策略，而另一个属性集合 $S=\{att_3,att_4,\cdots,att_{n'+1}\}$ 则不能满足这个属性。

为了基于访问策略 $\gamma_{t,S}$ 对明文 M 进行加密，加密算法首先产生一个随机的傀儡属性集合 $W=\{att_{N+1},att_{N+2},\cdots,att_{2N-t}\}$，使得 $W\subseteq\Omega'$，同时选择两个随机数 $s,r\in Z_p$。然后执行以下算法：

(1) 创建 G_2 上的元素 C_0 并赋值 $C_0=M\cdot Z^s$；

(2) 创建 G_1 上的元素 C_1 并赋值 $C_1=g^s$；

(3) 创建 G_1 上的元素 C_2 并赋值

$$C_2=(h_0\prod_{j\in S\cup W}h_j)^s;$$

(4) 创建 Z_p^* 上的元素 c 并赋值

$$c=H(\gamma_{t,S}\|C_0\|C_1\|C_2);$$

(5) 创建 G_1 上的元素 C_3 并赋值 $C_3=(\delta_1^c\delta_2^r\delta_3)^s$

(6) 返回密文 $CT=\{r,C_0,C_1,C_2,C_3\}$。

在密文当中 C_0 嵌入了明文 M，但同时嵌入了随机数 s，因此明文不可见，其安全性会在第 4 节给出相应的证明。而 r、C_1、C_2、C_3 则在解密过程中发挥作用，但前提是用户的属性集合满足访问策略。

由以上过程可以看出，无论访问策略如何变化，密文永远只由 5 个元素组成，其中包含 1 个 Z_p^* 上的元素，1 个 G_2 上的元素，3 个 G_1 上的元素。因此密文的长度是恒定的，而 QLZ 方法[8]、QDH 方法[9]以及 ZWM 方法[10]的密文长度均与访问策略所包含的属性数量相关，详细的对比将在第 4.2 节给出。因此与以上方案相比，随着 PHR 系统当中属性数量的增加，OHABE-CC 的加密算法的优势更加明显，更加节省带宽。

3.5 密钥转换

密钥转换算法输入用户私钥 SK_u，输出转换私钥 TK 以及 Elgamal 型私钥 EK。算法的执行流程如下：

(1) 创建 Z_p^* 上的随机元素 μ；

(2) 基于任意的 $sk_{i+1,j}\in SK_u$，创建 G_1 上的元素数组 $tk_{i,j}$，具体如下：

$$tk_{i+1,j}=\{a_{i+1}^{\mu},b_{i+1}^{\mu},c_{i+1,j,1}^{\mu},\cdots,c_{i+1,j,j-1}^{\mu},c_{i+1,j,j+1}^{\mu},\cdots,c_{i+1,j,2N-1}^{\mu}\}$$

(3) 创建转换私钥 $TK=\{tk_{i+1,j}\}_{\forall att_j\in S\cup\Omega'}$ 以及 Elgamal 型私钥 $EK=\mu$；

(4) 输出转换私钥 TK。

3.6 密文转换

密文转换算法输入消息密文 CT 以及转换私钥 TK，输出转换密文 CT'。具体流程如下：

(1) 输入密文 CT、转换私钥 TK，首先验证以下的等式是否成立：

$$e(g,C_2)=e(C_1,h_0\prod_{j\in S\cup W}h_j)$$

$$e(g,C_3)=e(C_1,\delta_1^c\delta_2^r\delta_3)$$

(2) 如果两等式均成立则说明密文为合法密文，跳转至下一步，否则结束转换并输出符号 $\perp$。

(3) 创建 G_1 上的元素 D_1，进行如下赋值计算：

$$D_1=\prod_{att_j\in S_{real}\cup W}(a_{i+1,j}\cdot\prod_{k\in S\cup W,k\neq j}c_{i+1,j,k})^{\Delta_{j,S'_{real}\cup W'(0)}}$$

(4) 创建 G_1 上的元素 D_1，进行如下赋值计算：

$$D_2=\prod_{att_j\in S_{real}\cup W}(b_{i+1,j})^{\Delta_{j,S'_{real}\cup W(0)}}$$

(5) 创建 G_2 上的元素 T'，进行如下赋值计算：

$$T'=\frac{e(C_1,D_1)}{e(C_2,D_2)}$$

(6) 输出转换密文 $CT'=\{C_0,T'\}$。

用户的属性集合被嵌入到用户私钥当中，因此如果用户的属性集合无法满足密文当中的访问策略，就无法在该算法中计算出正确的 D_1 和 D_2，也就不能获取正确的转换密文。该机制保证任何不满足

访问策略的用户都不能通过任何方式降低破解密文的困难程度。

3.7　终端解密

终端解密算法输入转换密文 CT' 以及 Elgamal 型私钥 EK, 输出消息明文 M。其执行流程如下:

(1) 创建 G_2 上的元素 T_0, 进行以下赋值计算:

$$T_0 = (T')^{\mu^{-1}}$$

(2) 创建 G_2 上的元素 M, 进行以下赋值计算:

$$M = C_0 \cdot \frac{1}{T_0}$$

如果用户的属性满足密文当中的访问策略,那么用户仅需通过以上两步基本运算就可以获取消息明文,即此时的元素 M 当中包含的信息。

4　安全性分析与性能比较

4.1　安全性分析

本方案的安全性基于 q-BDHE 假设,对此给出以下定理:

定理 1　OHABE-CC 安全性: 如果 q-BDHE 问题是难解的,那么一定不存在多项式时间内的敌手能以不可忽略的优势破解 OHABE-CC。

为了将 OHABE-CC 安全性规约到 q-BDHE 问题上,根据 OHABE-CC 安全模型设计了如下的模拟游戏:

初始化阶段　挑战者 C 获取 q-BDHE 问题的参数 $\{g, h, g^{\alpha}, g^{\alpha^2}, \cdots, g^{\alpha^q}, g^{\alpha^{q+2}}, \cdots, g^{\alpha^{2q}}, Z\}$, 同时敌手 A 产生将挑战访问策略 γ_{t^*,S^*} 发送给挑战者 C。

系统设置阶段　C 定义生成全局属性集合 $\Omega = \{att_1, att_2, \cdots, att_N\}$ 以及一个傀儡属性集合 $\Omega' = \{att_{N+1}, att_{N+2}, \cdots, att_{2N-1}\}$, 其中 $2N-1=q$。对于任意的属性 $att_j \in \Omega$, C 产生一个随机数 $r_j \in Z_p$ 并计算 $h_j = g^{r_j} g^{\alpha^{q-j+1}}$, 之后选择另一个随机数 $r_0 \in Z_p$ 并计算 $h_0 = g^{r_0} \prod_{j \in S^* \cup W^*} h_j^{-1}$。随后选择一个随机数 $\alpha' \in Z_p^*$, 使得 $g_1 = g^x = g^{\alpha'} g^{\alpha^q}$, 这里暗示了 $x = \alpha' + \alpha^q$, 但是在计算过程中并不知道 α 到底是多少。除此之外再选择一组随机数 $d_2, d_3, e_1, e_2, e_3 \in Z_p^*$, 最终 C 向 A 发送公钥 $PK = \{g, g_2, Z, h_0, h_1, \cdots, h_q, \delta_1, \delta_2, \delta_3, H\}$, 其中 $\delta_1 = g_2 g^{e_1}, \delta_2 = g_2^{d_2} g^{e_2}, \delta_3 = g_2^{d_3} g^{e_3}, Z = e(g_1, g_2) = e(g^{\alpha'}, g^{\alpha}) e(g^{\alpha^q}, g^{\alpha})$, 而函数 $H: \{0,1\}^* \to Z_p$ 是一个随机预言机。

询问阶段 3　A 随机地、有限次地对 C 发送以下 2 种询问:

(1) 授权询问。A 发出关于属性集合 S 的授权询问,C 定义属性集合 T、T' 以及 T'' 使得 $T = (S_{real} \cap S^*) \cup W^*, T \subseteq T' \subseteq (S^* \cup W^*), T'' = T' \cup \{0\}$。对于任意的属性 $att_j \in S \cup \Omega'$, C 获取一个 $N-1$ 次的随机多项式 $q(\cdot)$, 这个多项式保证 $q(0) = x = \alpha' + a^q$, 但 C 本身无法知道 x 的值是多少。如果 $att_j \in T'$, C 选择两个数 $t_j, r'_j \in Z_p$ 使得 $q(j) = t_j, r_j = \alpha^j + r'_j$, 然后计算 $sk_j = \{g_2^{q(j)} (h_0 h_j)^{r_j}, g^{r_j}, h_1^{r_j}, \cdots, h_{j-1}^{r_j}, h_{j+1}^{r_j}, \cdots, h_q^{r_j}\}$; 如果 $att_j \notin T'$, C 选择一个数 $r'_j \in Z_p$ 并计算 $r_j = r'_j - \Delta_{0,T''}(j)\alpha^j$, 然后通过拉格朗日插值法得到 $q(j) = \Delta_{0,T''}(j) q(0) + \sum_{i \in T''} \Delta_{0,T''}(i) q(j)$, 计算 $sk_j = \{g_2^{q(j)} (h_0 h_j)^{r_j}, g^{r_j}, h_1^{r_j}, \cdots, h_{j-1}^{r_j}, h_{j+1}^{r_j}, \cdots, h_q^{r_j}\}$。以上计算完成之后,C 调用密钥转换算法获取转换私钥 TK 以及 Elgamal 型密钥 EK, 如果 $|S \cap S^*| < t^*$ 那么 C 将私钥 SK_u 发送给 A,否则将转换私钥 TK 发送给 A。

(2) 解密询问。A 提交一个密文 $CT = (r, C_0, C_1, C_2, C_3)$ 并向 C 请求解密,但是敌手 A 不可以提交关于挑战密文的解密请求。假设密文当中嵌入的访问策略为 $\gamma_{t,S}$, 那么 C 首先计算 $c = H(\gamma_{t,S} \| C_0 \| C_1 \| C_2)$, 然后检验 $e(g, C_2) = e(C_1, h_0 \prod_{j \in S \cup W} h_j)$ 和 $e(g, C_3) = e(C_1, \delta_1^c \delta_2^r \delta_3)$ 是否成立。如果以上计算任意一个不成立则 C 返回 $\perp$, 否则进一步校验 $c + rd_2 + d_3 = 0$ 是否成立。如果成立则返回一个随机的消息给 A,否则计算并输出如下的消息:

$$M = \frac{C_0}{e\left(\frac{C_3}{C_1^{ce_1 + re_2 + e_3}}, g_1^{(c+rd_2+d_3)^{-1}}\right)}$$

挑战阶段　敌手 A 提交两个长度相同的挑战明文 M_0 和 M_1 给模拟器 B,然后模拟器 B 随机选择一个比特 $\beta \in \{0,1\}$ 并对明文 M_β 执行加密算法,计算

$C_0^* = M_\beta Z \cdot e(h, g_2^{\alpha'})$, $C_1^* = h$, $C_2^* = h^{r_0}$, $c^* = H(\gamma_{t^*, S^*} \| C_0^* \| C_1^* \| C_2^*)$, $r^* = -(c^* + d_3)/d_2$ 以及 $C_3^* = h^{c^* e_1 + r^* e_2 + e_3}$, 然后将挑战密文 $CT^* = (r^*, C_0^*, C_1^*, C_2^*, C_3^*)$ 发送给 A。

询问阶段 4 本阶段与询问阶段 3 相同。

猜测阶段 A 输出 β' 作为对 β 的猜测。如果 $\beta = \beta'$, C 输出 $\theta = 0$ 表示其认为 q-DBDH 问题的解为 $Z = e(g, h)^{\alpha^{q+1}}$。如果 $\beta \neq \beta'$, C 输出 $\theta = 1$ 表示模拟器 B 猜测 $Z \neq e(g, h)^{\alpha^{q+1}}$。当 $Z \neq e(g, h)^{\alpha^{q+1}}$ 的情况下,挑战密文实际上就是随机密文,A 无法从中获取任何有用的信息,因此有:

$$\Pr[\beta' = \beta \mid Z \neq e(g, h)^{\alpha^{q+1}}] = \Pr[\beta' \neq \beta \mid Z \neq e(g,h)^{\alpha^{q+1}}] = 1/2$$

当 $Z = e(g, h)^{\alpha^{q+1}}$ 的情况下,挑战密文与真实的 OHABE-CC 密文具有相同的分布。假设模拟器游戏中解密询问的次数为 q_d, 由于 A 破解 OHABE-CC 的优势为 ε, 在进行解密询问时游戏中断的概率最多为 q_d/p, 所以有:

$$\Pr[\beta = \beta' \mid Z = e(g, h)^{\alpha^{q+1}}] = \varepsilon - \frac{q_d}{p} + \frac{1}{2}$$

因此 C 成功解决 q-BDHE 难题的优势为

$$\varepsilon' = \frac{1}{2}\Pr[\beta = \beta' \mid Z = e(g, h)^{\alpha^{q+1}}] + \frac{1}{2}\Pr[\beta \neq \beta' \mid Z \neq e(g,h)^{\alpha^{q+1}}] - \frac{1}{2} = \frac{\varepsilon}{2} - \frac{q_d}{2p}$$

4.2 性能分析

本章首先给出 OHABE-CC 在基于雾计算的 PHR 系统中具体的工作流程,如图 1 所示。具体工作流程如下。

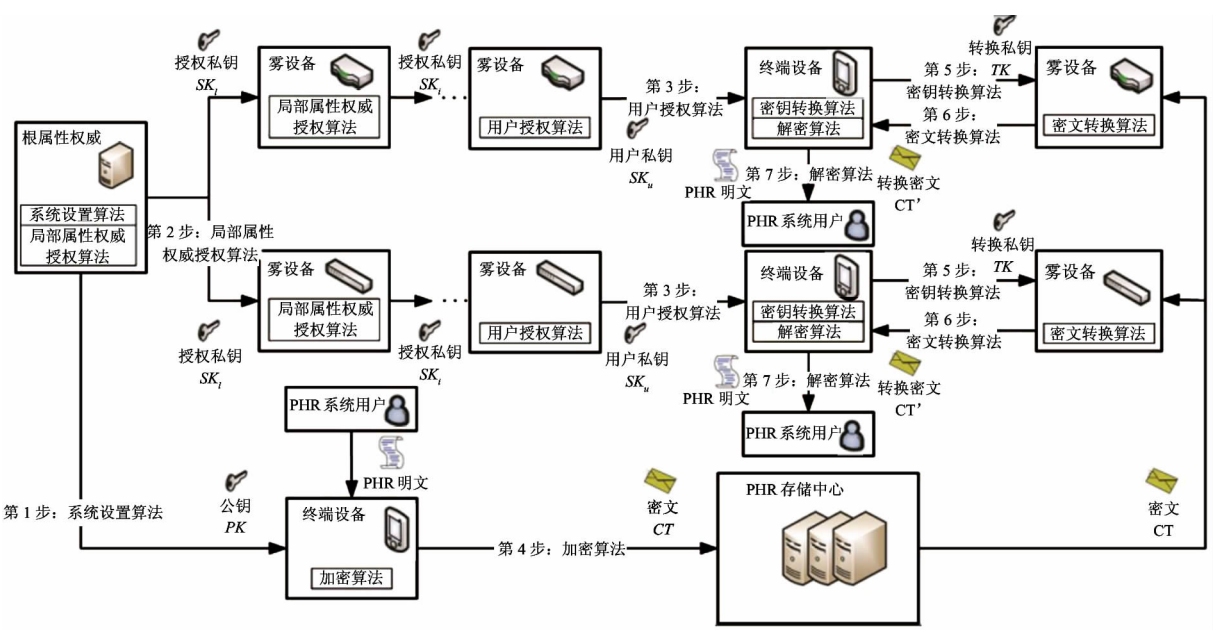

图 1 OHABE-CC 在基于雾计算的 PHR 系统中具体的工作流程

(1) 在基于雾计算的 PHR 系统中,可信的根属性权威启动系统设置算法向全网广播公钥 PK;

(2) 部署在网络边缘的可信雾设备向根属性权威申请授权,根属性权威启动局部属性权威授权算法进行授权,授权成功后雾设备获取授权私钥 SK_i 成为局部属性权威,可以向下继续授权其他雾设备或 PHR 系统用户;

(3) PHR 系统用户将属性集合 S 发送给临近的雾设备(局部属性权威)以申请授权,通过授权后将获得对应的用户私钥 SK_u;

(4) 持有 PHR 记录的 PHR 系统用户为自己的 PHR 记录 M 制定相应的访问策略 $\gamma_{t,S}$, 然后执行加密算法对 M 执行加密算法,加密后发送给 PHR 存储中心统一存储、管理;

(5) 当某个 PHR 系统用户发起解密请求时,PHR 存储中心将对应的密文 CT 发送给就近的雾设

备,与此同时该用户执行密钥转换算法将用户私钥 SK_u 转换为转换私钥 TK 和 Elgamal 型私钥,并将转换私钥 TK 发送给这个雾设备;

(6) 获取密文 CT 与转换私钥 TK 的雾设备执行密文转换算法,验证密文合法性的同时将转换密文 CT' 发送给请求解密的 PHR 系统用户;

(7) PHR 系统用户使用 Elgamal 型私钥对转换密文 CT' 进行进一步的解密,仅需要简单的计算就可以快速地获取 PHR 记录 M。

为进一步分析 OHABE-CC 的性能,本文将其与 QLZ 方法[8]、QDH 方法[9]以及 ZWM 方法[10]等 3 个类似的方案进行比较。为方便表述,定义了如下若干符号,如表 1 所示。

表 1　符号定义

符号	定义
A	访问策略
$A_{\min}$	满足访问策略 A 的最小访问策略
$\lvert * \rvert$	返回访问策略 * 当中包含属性的个数
$ST_{\cap}(*)$	返回访问策略 * 当中“与”门的阈值之和
$SG(*)$	返回访问策略 * 里阈值门的总数

算法功能分析　本文将 OHABE-CC 与 QLZ 方法[8]、QDH 方法[9]以及 ZWM 方法[10]在加密功能方面进行了比较,比较结果如表 2 所示。

QLZ 方法[8]采用了多属性权威机制,但是属性权威数量是固定的,并不支持属性权威的扩展。该方法的安全等级经证明达到了抵抗自适应选择明文攻击(IND-CPA2)。然而该方法密文冗长并且不支持密文的外包计算,在功能性上的考虑不够周全。

表 2　方案功能比较

方案	属性权威	可扩展	外包解密	安全等级
QLZ 方法	多个	否	否	IND-CPA2
QDH 方法	单一	否	否	IND-CPA2
ZWM 方法	单一	否	否	IND-CPA1
OHABE-CC	多个	支持	支持	IND-RCCA2

QDH 方法[9]和 ZWM 方法[10]都仅采用了单一属性权威的机制且不能扩展,同时都不支持密文的外包计算,因此很难适应云海量 PHR 系统用户的管理。在安全性方面 QDH 方法[9]能够抵抗自适应选择明文攻击(IND-CPA2),而 ZWM 只能抵抗选择明文攻击(IND-CPA1),即只能满足加密算法的基本安全需求,无法抵御更加强大的攻击。因此在安全性上都无法与 QLZ 方法[8]和 OHABE-CC 相比。

OHABE-CC 采用了等级化的多属性权威机制,可以随时利用海量的雾设备对属性权威进行扩展,而且其外包解密功能使得用户无需进行过于复杂的计算就可以获取密文,相比 QLZ 方法[8]、QDH 方法[9]以及 ZWM 方法[10]更加适用于雾计算场景,更加有利于海量 PHR 系统用户的管理。经证明其安全性能够抵抗自适应的重放选择密文攻击(IND-RCCA2),其安全性要高于 IND-CPA1 以及 IND-CPA2。综上所述,OHABE-CC 在功能上更加有利于 PHR 系统在雾计算环境当中的应用。

算法效率分析　本文对 QLZ 方法[8]、QDH 方法[9]、ZWM 方法[10]以及 OHABE-CC 的加解密效率进行了分析,主要考虑了密文长度复杂度以及解密开销两大方面,其中解密开销主要包含乘运算次数、幂运算次数以及双线性配对次数,比较结果如表 3 所示。

表 3　方案解密效率比较

方案	密文长度	解密开销		
		乘次数	幂次数	配对次数
QLZ 方法	$O(\lvert A \rvert)$	$ST_{\cap}(A_{\min}) - SG(A_{\min}) + 2$	$\lvert A_{\min} \rvert + 1$	$N \lvert A_{\min} \rvert + 2$
QDH 方法	$O(2 \lvert A \rvert)$	$2 \lvert A_{\min} \rvert - 1$	$\lvert A_{\min} \rvert$	$3 \lvert A_{\min} \rvert$
ZWM 方法	$O(\lvert A \rvert)$	$\lvert A_{\min} \rvert$	0	5
OHABE-CC	$O(1)$	1	1	0

QLZ 方法[8]基于树形结构构建了 PHR 系统的访问策略,树形结构本身具备了颇高的复杂度,不难看出其密文长度以及解密过程中乘运算、幂运算以及双线性配对运算的次数都与 $|A_{\min}|$ 呈现线性的关系,尽管访问策略的灵活性有所提高但牺牲了太多的计算效率。

QDH 方法[9]的密文长度复杂度为 $O(2|A|)$,在访问策略相同的情况下其密文长度是 QLZ 方法[8]的两倍。在解密效率方面,QDH 方法[9]相比 QLZ 方法[8]需要执行更多的乘运算,但执行双线性陪读运算次数要明显少于 QLZ[9],而且随着全局属性集合当中的属性数量增长这个优势愈发明显。

ZWM 方法[10]最显著的特点就是将双线性配对运算的次数降低为常数次数(5 次),同时不用进行任何的幂运算,但是解密需要的乘运算次数仍然与 $|A_{\min}|$ 线性相关。

OHABE-CC 不仅使得用户在执行解密时只需要进行 1 次乘运算和 1 次幂运算,而且将密文长度的复杂度控制在 $O(1)$,无论访问策略当中包含的属性有多少,密文永远只含有 5 个元素(3 个 G_1 上的元素,1 个 G_2 上的元素以及 1 个 Z_p^* 上的元素),因而能够极大地节省传输开销。综上所述 OHABE-CC 在计算开销和传输开销两方面均有所优化,因此能够更加适用于基于雾计算的 PHR 系统。

5 结 论

私人健康记录系统是一种基于云计算的健康服务。利用雾计算技术可以为 PHR 系统提供更好的移动化支持,但同时对病人的隐私也造成了巨大的威胁,因此急需一套严格的数据保护与访问权限控制技术。结合密文定长机制和外包解密机制,提出了一种支持外包解密的等级化属性加密(OHABE-CC)算法,首先将属性权威的权力与运算负载分散并提高其可扩展性,使之适用于动态变化的雾计算环境。其次在加密过程中将密文的长度复杂度控制为 $O(1)$。基于外包解密技术,用户仅需通过极少的计算就可以恢复 PHR 明文。基于 q-DBDH 困难假设,证明了 OHABE-CC 满足 IND-RCCA2 安全性,经分析该算法在功能以及解密计算效率上优于其他类似算法,更加适用于基于雾计算的 PHR 系统。

参考文献

[1] Lanranjo L, Neves A L, Vilanueva T, et al. Patient's access to their medical records[J]. *Acta Medica Portuguesa*, 2013, 26(3):265-270

[2] Bonomi F. Connected vehicles, the internet of things, and fog computing[C]. In: Proceedings of the 8th ACM International Workshop on Vehicular Inter-Networking, Las Vegas, USA. 2011. 13-15

[3] Stojmenovic I, Wen S. The fog computing paradigm: Scenarios and security issues[C]. In: Proceedings of 2014 Federated Conference on Computer Science and Information Systems, Warsaw, Poland, 2014. 1-8

[4] Sahai A, Waters B. Fuzzy identity-based encryption[C]. In: Proceedings of International Conference on Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, 2005. 457-473

[5] 苏金树,曹丹,王小峰,等. 属性加密机制[J]. 软件学报,2011,22(6):1299-1315

[6] Pearce C, Bainbidge M. A personally controlled electronic health record for Australia[J]. *Journal of the American Medical Informatics Association*, 2014, 21(4): 707-713

[7] Li M, Yu S, Zheng Y, et al. Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2013, 24(1): 131-143

[8] 刘琴,刘旭辉,胡柏霜,等. 个人健康记录云管理系统中支持用户撤销的细粒度访问控制[J]. 电子与信息学报,2017,39(5):1206-1212

[9] Zhang L, Wu Q, Mu Y, et al. Privacy-preserving and secure sharing of PHR in the cloud[J]. *Journal of Medical Systems*, 2016, 40(12): 267

[10] Qian H, Li J, Zhang Y, et al. Privacy-preserving personal health record using multi-authority attribute-based encryption with revocation[J]. *International Journal of Information Security*, 2015, 14(6):487-497

[11] Qin B, Deng H, Wu Q H, et al. Flexible attribute-based encryption applicable to secure e-healthcare records[J]. *International Journal of Information Security*, 2015, 14(6):499-511

[12] Zhang L, Wu Q, Mu Y, et al. Privacy-preserving and se-

cure sharing of PHR in the cloud[J]. *Journal of Medical Systems*, 2016, 40(12): 267

[13] Boneh D, Boyen X, Goh E J. Hierarchical identity based encryption with constant size ciphertext[C]. In: Proceedings of Advances in Cryptology-EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, 2005. 440-456

[14] Zhang Y, Zheng D, Chen X, et al. Computationally efficient ciphertext-policy attribute-based ecryption with constant-size ciphertexts[C]. In: Proceedings of the 8th International Conference on Provable Security, Hong Kong, China, 2014. 259-273

[15] Odelu V, Das A K, Rao Y S, et al. Pairing-based CP-ABE with constant-size ciphertexts and secret keys for cloud environment[J]. *Computer Standards & Interfaces*, 2017, 54(part 1): 3-9

[16] 崔勇,宋健,缪葱葱,等. 移动云计算研究进展与趋势[J]. 计算机学报,2017,40(2):273-295

[17] Green M, Hohenberger S, Waters B. Outsourcing the decryption of ABE ciphertexts[C]. In: Proceedings of USENIX Security Symposium, San Francisco, USA, 2011. 34-50

[18] Lai J, Deng R H, Guan C, et al. Attribute-based encryption with verifiable outsourced decryption[J]. *IEEE Transactions on Information Forensics and Security*, 2013, 8(8): 1343-1354

[19] Lin S, Zhang R, Ma H, et al. Revisiting attribute-based encryption with verifiable outsourced decryption[J]. *IEEE Transactions on Information Forensics and Security*, 2015, 10(10): 2119-2130

[20] Wang Z, Ma H, Wang J. Attribute-based online/offline encryption with outsourcing decryption[J]. *Journal of Information Science and Engineering*, 2016, 32(6): 1595-1611

A research of improved HABE algorithm in fog computing based PHR systems

Wang Xuan*, Zou Jun**, Du Jun***

(*School of Electronic Information Engineering, Nanjing Vocational College of Information Technology, Nanjing 210023)

(**Department of Electrical Engineering, Tsinghua University, Beijing 100084)

(***Zhongxing Optoelectronic Technology Co., Ltd., Nanjing 210000)

Abstract

Personal health record (PHR) system provides personalized health services for individuals based on cloud computing. By utilizing fog computing, PHR system will gain better support in terms of mobility. However, a large amount of involved fog devices compromise patients' privacy. Thus, it is urgent to deploy a strict scheme with data protection and access control. With the combination of constant ciphertext mechanism and outsourcing mechanism, an outsourced hierarchical attribute-based encryption with constant ciphertext is proposed. It first improves scalability by introducing hierarchical attribute authorities in order to adapt to dynamic environment of fog computing. Then any PHR plaintext can be encrypted to a ciphertext with constant length. Via outsourcing mechanism, it is a few computation that users need to do to decrypt PHR ciphertext. Based on q-DBDH complexity assumption, we prove it meets IND-RCCA2 security requirement. The analysis in terms of functionality and decryption efficiency demonstrates that this algorithm is more suitable for fog computing based PHR system than other existing algorithms.

Key words: personal health record (PHR), fog computing, attribute-based encryption(ABE), constant ciphertext, outsourced decryption