C. 安全模型

我们利用敌手 $\mathbb{A}$ 和挑战者 $\mathbb{C}$ 之间的游戏,定义了高效可验证的多授权机构属性基加密移动云存储数据访问控制的选择访问结构模型,其中,敌手和挑战者之间的游戏如下.

初始化阶段:敌手 $\mathbb{A}$ 提交恶意的属性授权机构集合 $\mathfrak{R}=\left(\widehat{A}_i\right)_{i\in I}$ 和访问结构 $\left(M_i^*,\rho_i^*\right)_{i\in I^*}$ 给挑战者 $\mathbb{C}$, 其中, $I\subseteq\{1,2,...,N\}$ 及 $I^*\subseteq\{1,2,...,N\}$. 挑战者 $\mathbb{C}$ 首先执行 *GlobalSetup* 算法产生公开参数 *Params*,然后把公开参数响应给敌手.挑战者 $\mathbb{C}$ 根据不同的授权机构产生不同的公私钥对,具体如下.

(1) 对于每个授权机构 $\widehat{A}_i\subseteq\mathfrak{R}$, 挑战者 $\mathbb{C}$ 运行 *AASetup* 算法产生公私钥对 $\left(SK_i,PK_i\right)$, 并把 $\left(SK_i,PK_i\right)$ 发送给敌手.

(2) 对于每个授权机构 $\widehat{A}_i\not\subset\mathfrak{R}$, 挑战者 $\mathbb{C}$ 运行 *AASetup* 算法产生公私钥对 $\left(SK_i,PK_i\right)$, 并把公钥参数 PK_i 发送给敌手.

查询阶段 1:敌手 $\mathbb{A}$ 可以选择不同数量的用户 $\{ID_1,ID_2,...,ID_q\}$ 及用户属性集合 $S_{ID_1},S_{ID_2},...,S_{ID_q}$, 然后向挑战者 $\mathbb{C}$ 进行多次(即 q 次)的密钥询问;其中,敌手 $\mathbb{A}$ 每次进行密钥询问时,提交的属性集合 S_{ID_i} 不仅不能满足访问结构 (M^*,ρ^*), 而且不能来自于恶意的属性授权机构 $\mathfrak{R}$. 挑战者 $\mathbb{C}$ 运行 *KeyGen* 产生密钥 SK_{ID},并把其响应给敌手 $\mathbb{A}$.

挑战阶段:敌手 $\mathbb{A}$ 提交两个长度相同内容不同的明文 M_0 和 M_1 给挑战者 $\mathbb{C}$; 挑战者 $\mathbb{C}$ 首先随机选择一个位数 $b\in\{0,1\}$, 然后运行离线加密算法 $Encrypt.OffL\left(Params,PK_i\right)$ 和线上加密算法 $Encrypt.OnL\left(Params,IT,ck,\left(M^*,\rho^*\right)\right)$, 产生挑战密文 CT^*, 最后把挑战密文 CT^* 发送给敌手 $\mathbb{A}$.

查询阶段 2:重复查询阶段 1.

敌手 $\mathbb{A}$ 的优势被定义为 $\Pr[\beta'=\beta]-1/2$, 其中, $\Pr[\beta'=\beta]$ 表示 $\beta'=\beta$ 的概率.

猜测阶段:敌手 $\mathbb{A}$ 输出一个作为 β' 对 β 的猜测.如果 $\beta=\beta'$, 则敌手 $\mathbb{A}$ 赢得游戏.

定义 3. 如果敌手 $\mathbb{A}$ 在多项式时间内赢得以上游戏的概率是可以忽略的,那么高效可验证的多授权机构属性基加密移动云存储数据访问控制是选择明文安全的(selective CPA-secure).

3 高效可验证的 MA-ABE 访问控制方案

在这一部分,本文具体构造了高效可验证的多授权机构属性基加密方案.假定该方案中 LSSS 访问矩阵最大的行数为 $P_{\max}$.本文方案包括如下几个阶段.

1) 系统初始化

通过执行 *GlobalSetup* 算法,产生系统公开参数 *Params*.首先利用 $\partial\partial(1^\kappa)\to(e,p,G,G_T)$ 构造一个双线性群满足 $e:G\times G\to G_T$; 其中,g 为 G 的生成元.然后随机选择 $h,u,v,w\in G$, 并构建抗碰撞的哈希函数 $H(\cdot):\{0,1\}^*\to Z_p$, $H_0(\cdot):G_T\to\{0,1\}^{\ell_{H_0}}$ 和 $H_1(\cdot):\{0,1\}^*\to\{0,1\}^{\ell_{H_1}}$ 以及安全的密钥提取函数 H'.其中,p 为一个素数, Z_p 为模 p 构成的有限域.另外,假设方案中有 N 个属性授权机构 $\left\{\hat{A}_1,\hat{A}_2,...,\hat{A}_N\right\}$, 每个属性授权机构管理一类属性集合 $\tilde{A}_i=\{A_{i,1},A_{i,2},...,A_{i,q_i}\}$. 其中, $A_{i,j}\in Z_p,i=1,2,...,N$ 和 $j=1,2,...,q_i$. 方案中的数据传递在安全的信道中进行.因此,系统的公开参数为

$$Params=\left(g,h,u,v,w,e,p,H,H_0,H_1,G,G_T,H'\right) \tag{1}$$

2) 授权机构建立

授权机构运行 *AASetup* 进行初始化操作,具体过程分为如下两步.

(1) 所有授权机构随机选择 $\alpha_i\in Z_p$, 并计算 $Y_i=e(g,g)^{\alpha_i}$, 然后将 Y_i 发送给其他授权机构,最后,每个授权机构独立计算 $Y=\prod_N^{i=1}Y_i=e\left(g,g\right)^{\sum_N^{i=1}\alpha_i}$.

(2) 对于每个授权机构 $\hat{A}_i$ 来说,具体操作如下.

(a) 随机选择 N–1 个整数 $s_{ik} \in Z_p\left(k \in \{1,...,N\}/\{i\}\right)$, 计算 $g^{s_{ik}}$,然后把其发送给其他授权机构 $\hat{A}_i(i \in \{1,\ldots,N\}/\{i\})$.

(b) 当收到来自其他授权机构 $\hat{A}_i\left(i \in \{1,...,N\}/\{i\}\right)$ 的 N–1 个组件 $g^{s_{ki}}$ 时,通过如下公式计算主要私钥 MK_i.

$$MK_i = \left(\prod_{k\in\{1,...,N\}/\{i\}} g^{s_{ik}}\right) \Big/ \left(\prod_{k\in\{1,...,N\}/\{i\}} g^{s_{ki}}\right) = g^{\left(\sum_{k\in\{1,...,N\}/\{i\}} s_{ik} - \sum_{k\in\{1,...,N\}/\{i\}} s_{ki}\right)} \tag{2}$$

其中, $\prod_{i\in\{1,...,N\}} MK_i = 1 \bmod p$.

(c) 为每个属性 $A_{i,j} \in \tilde{A}_i$, 计算组件 $u^{A_{i,j}}h$.

每个授权机构发布自己的公钥 $PK_i = (Y)$, 并保留自己的私钥 $SK_i = \left(\alpha_i, (u^{A_j}h)_{A_j\in\tilde{A}_i}, MK_i\right)$.

3) 密钥产生

当新的用户访问系统时,需要从属性授权机构请求私钥,授权机构通过执行密钥产生 *KeyGen* 算法为用户发布私钥.*KeyGen* 的具体过程分为如下两步.

(1) 每个授权机构 $\hat{A}_i$.

(a) 随机选择一个数 $\gamma_i \in Z_p$, 计算组件 $MK_i \cdot g^{\gamma_i}, MK_i \cdot v^{-\gamma_i}$ 和 $MK_i \cdot g^{\alpha_i}.w^{\gamma_i}$, 并将其共享给其他授权机构 $\hat{A}_i$ $(i\in\{1,\ldots,N\}/\{i\})$.

(b) 当收到来自其他授权机构的组件 $MK_i \cdot g^{\gamma_i}, MK_i \cdot v^{-\gamma_i}$ 和 $MK_i \cdot g^{\alpha_i}.w^{\gamma_i}$ 时,通过如下公式计算密钥组件 D_0、D_1 和 D_v.

$$\left.\begin{aligned} D_0 &= \prod_N^{i=1} MK_i.g^{\gamma_i} = g^{\sum_N^{i=1}\gamma_i} = g^r, \\ D_1 &= \prod_N^{i=1} MK_i.g^{\alpha_i}.w^{\gamma_i} = g^{\sum_N^{i=1}\alpha_i} w^r, \\ D_v &= \prod_N^{i=1} MK_i.v^{-\gamma_i} = v^{-r} \end{aligned}\right\} \tag{3}$$

其中, $r = \sum_N^{i=1}\gamma_i$.

(2) 授权机构首先为每个属性 $\tau \in \left[\left|S_{ID} \cap \tilde{A}_i\right|\right]$ 随机选择 $r_\tau \in Z_p$, 然后计算 $D_{j,2} = g^{r_j}$, $D_{j,3} = \left(u^{A_j}h\right)^{r_j H(ID)} \cdot D_v = \left(u^{A_j}h\right)^{r_j H(ID)} v^{-r}$.

用户从授权机构获得的私钥为 $SK_{ID}^i = \left(D_0, D_1, \left\{D_{j,2}, D_{j,3}\right\}_{j=1...\tau}, S_{ID} \cap \tilde{A}_i\right)$.

4) 离线加密

当数据拥有者的移动设备重新启动时,执行 *Encrypt.OffL* 算法产生临时密文,并将临时密文存储在移动设备中;*Encrypt.OffL* 算法的具体过程如下.

(a) 随机选择参数 $s \in Z_p$, 并计算 $key = Y^s, C_0 = g^s$.

(b) 为访问矩阵的每一行随机选择 $z_j, x_j, t_j \in Z_p$, 其中, $j = 1,2,...,P_{\max}$.

(c) 计算 $C_{j,1} = w^{z_j}v^{t_j}, C_{j,2} = (u^{x_j}h)^{-t_j}, C_{j,3} = g^{t_j}$.

方案临时密文为 $IT = \left(s, key, C_0, \left\{z_j, x_j, t_j, C_{j,1}, C_{j,2}, C_{j,3}\right\}_{j=1...P_{\max}}\right)$.

5) 在线加密

当数据拥有者外包数据到云存储服务器时,需要运行 *Encrypt.OnL* 加密数据 *MSG*,然后把密文外包到云存储服务器.*Encrypt.OnL* 算法的具体过程如下.

(1) 数据拥有者随机选择 $ck \in G_T$, 计算对称密钥 $sk = H'(ck)$, 使用对称密钥 sk 加密数据 MSG 生成数据密文 CT'.另外,计算验证令牌 $Token = H_1(H_0(ck) \| CT')$.

(2) 数据拥有者执行如下操作加密对称密钥 ck,具体过程如下.

(a) 定义访问控制结构 $(M_{\ell \times n}, \rho)$, 其中, $\ell \leqslant P_{\max}$.

(b) 选择随机向量 $\vec{y} = (s, y_2, y_3, ..., y_n)$, 其中, $y_2, y_3, ..., y_n \in Z_p$.

(c) 计算 $z'_j = M_j \vec{y}$ 和 $C = key \cdot ck$, 其中, M_j 是访问矩阵 M 的行向量.

(d) 计算 $C_{j,4} = z'_j - z_j \bmod p, C_{j,5} = t_j(x_j - \rho(j)) \bmod p$.

外包密文为 $CT = \left(CT', C, C_0, \left\{C_{j,1}, C_{j,2}, C_{j,3}, C_{j,4}, C_{j,5}\right\}_{j=1...\ell}, (M, \rho), Token\right)$.

6) 密文转换

数据使用者在移动设备重启时,执行 *GenToken* 算法产生转换密钥,并将其存储到移动设备上.*GenToken* 算法的具体过程如下,首先随机选择 $\mu \in Z_p$, 并计算 $TK_{ID}^i = \left(SK_{ID}^i\right)^{\mu}$ 获得转换密钥.当数据使用者访问云存储数据时,把转换密钥外包到云存储服务器,云服务器运行 *PDecrypt* 算法产生转换密文 *TD*.*PDecrypt* 算法的具体过程如下.

首先通过下式计算 key_{μ}.

$$key_{\mu} = \frac{e\left(C_0, (D_0)^{\mu}\right)}{e\left(w^{\sum_{j \in \ell} C_{j,4} \cdot \omega_j}, (D_1)^{\mu}\right) \cdot \prod_{j \in \ell}\left(e\left(C_{j,1}, (D_1)^{\mu}\right) \cdot e\left(C_{j,2}.u^{C_{j,5}}, (D_{j,2})^{\mu}\right)^{H(ID)}.e\left(C_{j,3}, (D_{j,3})^{\mu}\right)\right)^{\omega_j}} \tag{4}$$

其中, $\sum_{j=1}^{\ell} \omega_j z'_j = s$. 最后,云存储服务器把转换密文 $TD = \left(CT', C, key_{\mu}, Token\right)$ 发送给数据使用者.

7) 解密

数据使用者收到云存储服务器发送的转换密文后,执行 *Decrypt* 算法进行解密,具体过程如下:首先计算对称密钥 $ck = C / \left(key_{\mu}\right)^{1/\mu}$, 然后验证等式 $Token \neq H_1(H_0(ck) \| CT')$ 是否成立.若等式成立,则云存储服务器解密不正确,返回⊥;若等式不成立,则表示云存储服务器解密正确,使用对称密钥 ck 解密密文 CT',返回明文 MSG.

4 OO-MA-ABE 方案的安全性和其他性能分析

4.1 正确性分析

正确性:如果下面的等式成立,则本文的方案是正确的.从公式(4)中,可以得到如下计算组件:

$$\left.\begin{aligned}
&\prod_{j \in \ell}\left(e\left(C_{j,1}, (D_1)^{\mu}\right) \cdot e\left(C_{j,2} \cdot u^{C_{j,5}}, \left(D_{j,2}\right)^{\mu}\right)^{H(ID)} \cdot e\left(C_{j,3}, \left(D_{j,3}\right)^{\mu}\right)\right)^{\omega_j} \\
&= \prod_{j \in \ell}\left(e\left(w^{z_j} v^{t_j}, g^{\mu r}\right) \cdot e\left(\left(u^{x_j} h\right)^{-t_j}.u^{t_j\left(x_j - \rho(j)\right)}, g^{\mu r_i}\right)^{H(ID)}.e\left(g^{t_j}, \left(u^{A_j} h\right)^{\mu r_j H(ID)} v^{-r\mu}\right)\right)^{\omega_j} \\
&= \prod_{j \in \ell}\left(e\left(w^{z_j} v^{t_j}, g^{\mu r}\right) \cdot e\left(h^{-t_j}.u^{-\rho(j) t_j}, g^{\mu r_i}\right)^{H(ID)}.e\left(g^{t_j}, \left(u^{A_j} h\right)^{\mu r_j H(ID)} v^{-r\mu}\right)\right)^{\omega_j} \\
&= \prod_{j \in \ell}\left(e\left(w^{z_j} v^{t_j}, g^{\mu r}\right).e\left(g^{t_j}, v^{-r\mu}\right)\right)^{\omega_i} = \prod_{j \in \ell} e\left(w^{z_j}, g^{\mu r}\right)^{\omega_i} = e\left(w^{\sum_{j \in \ell} z_j \cdot \omega_j}, g^{\mu r}\right)
\end{aligned}\right\} \tag{5}$$

$$\prod_{i \in I_c} e(C_0, (D_0)^{\mu}) = \prod_{i \in I_c} e\left(g^s, g^{\mu \sum_N^{i=1} \alpha_i} w^{\mu r}\right) = \prod_{i \in I_c} e(g, g)^{\mu \sum_N^{i=1} \alpha_i s} e(w, g)^{\mu r s} \tag{6}$$

$$e\left(w^{\sum_{j \in \ell} C_{j,4} \cdot \omega_j}, (D_1)^{\mu}\right) = e\left(w^{\sum_{j \in \ell}\left(z'_j - z_j\right) \cdot \omega_j}, g^{\mu r}\right) = e\left(w^{\sum_{j \in \ell} z'_j \cdot \omega_j - \sum_{j \in \ell} z_j \cdot \omega_j}, g^{\mu r}\right) \tag{7}$$

最后,通过计算式(5)~式(7),可以得到

$$key_\mu=\frac{\prod_{i\in I_c}e\left(C_0,(D_0)^\mu\right)}{e\left(w^{\sum_{j\in\ell}C_{j,4}\cdot\omega_j},(D_1)^\mu\right)\cdot\prod_{j\in\ell}\left(e\left(C_{j,1},(D_1)^\mu\right)\cdot e\left(C_{j,2}.u^{C_{j,5}},\left(D_{j,2}\right)^\mu\right)^{H(ID)}.e\left(C_{j,3},\left(D_{j,3}\right)^\mu\right)\right)^{\omega_j}}=e(g,g)^{\mu\sum_N^{i=1}\alpha_i s} \quad (8)$$

4.2 安全性分析

理论 1. 本文方案在离散对数的假设下抵抗 N–1 个属性授权机构合谋攻击.

证明:每个属性授权机构随机产生 N–1 个随机整数 s_{ik}, 并把 $g^{s_{ik}}$ 共享给其他属性授权机构;每个授权机构根据收到的共享参数产生自己的主密钥 MK_i. 根据离散对数的假设可以知道,敌手很难从 $g^{s_{ik}}$ 中推断出 s_{ik}. 因此,即使有 N–2 个属性授权机构与敌手合谋,敌手仍然有一个参数不能确定,其不可能猜测到有效的 g^r, 所以说,敌手不可能构建出一个有效的私钥.因此,本文方案可以在离散对数的假设下抵抗 N–1 个属性授权机构的合谋攻击. □

理论 2. 在 q-type 假设(q-type assumption)成立的情况下,没有多项式时间敌手可以选择性攻破我们的方案.

证明:假定在选择安全性的情况下有多项式时间敌手 $\mathbb{A}$ 可以有不可忽略的优势打破我们的方案,那么敌手 $\mathbb{A}$ 可以构建出一个仿真者 $\mathbb{C}$ 以不可忽略的优势解决 q-type 问题.具体过程如下.

初始化:敌手 $\mathbb{A}$ 提交挑战的访问结构 (M^*,ρ^*) 和妥协的属性授权机构 $\Re=\left(\tilde{A}_i\right)_{i\in I}$ 的索引集合I,其中,M^*是一个 $\ell\times n$ 的二维数组,且 $\ell,n\leqslant p$. 假定 (M^*,ρ^*) 不可能满足每次敌手 $\mathbb{A}$ 用于密钥询问的属性集合.另外,访问结构中的属性不能来自于妥协的属性授权机构.

建立阶段:仿真者 $\mathbb{C}$ 运行 *AASetup* 算法和 *GlobalSetup* 算法,对于每个授权 $\tilde{A}_i\subseteq\Re$ 直接运行算法产生公开参数;对于每一个授权机构 $\tilde{A}_i\not\subset\Re$, 仿真者 $\mathbb{C}$ 首先随机选择 $\tilde{\alpha}\in Z_p$ 及 $a\in Z_p$, 然后计算设置 $\alpha=a^{q+1}+\tilde{\alpha}$ 保证 $e(g,g)^\alpha=e\left(g^a,g^{a^q}\right)\cdot e(g,g)^{\tilde{\alpha}}$; 最后仿真者 $\mathbb{C}$ 随机选择 $\tilde{u},\tilde{h},\tilde{v}\in Z_p$, 并通过下式计算 u,h,w,v.

$$\left.\begin{aligned}u&=g^{\tilde{u}}\prod_{(j,k)\in[\ell,n]}\left(g^{a^k/b_j^2}\right)^{M^*_{j,k}},\\h&=g^{\tilde{h}}\prod_{(j,k)\in[\ell,n]}\left(g^{a^k/b_j^2}\right)^{-\rho^*(j)M^*_{j,k}},\\w&=g^a,\\v&=g^{\tilde{v}}\prod_{(j,k)\in[\ell,n]}\left(g^{a^k/b_j}\right)^{M^*_{j,k}}\end{aligned}\right\} \quad (9)$$

仿真者生成的公钥为 $PK_i=\left\{e(g,g)^\alpha=e\left(g^a,g^{a^q}\right)\cdot e(g,g)^{\tilde{\alpha}}\right\}$.

查询阶段 1:在这一阶段,敌手 $\mathbb{A}$ 可以选择不同数量的用户 $\{ID_1,ID_2,...,ID_q\}$ 及用户属性集合 $S_{ID_1},S_{ID_2},...,S_{ID_q}$, 然后向仿真者 $\mathbb{C}$ 进行多次(即 q 次)的密钥询问.其中,敌手 $\mathbb{A}$ 提交的属性集合来自于未妥协的属性授权机构且不满足挑战访问结构 (M^*,ρ^*). 仿真者 $\mathbb{C}$ 先随机选择一个 $\tilde{r}\leftarrow Z_p$, 并挑选向量 $\vec{\omega}=(\omega_1=-1,\omega_2,...,\omega_n)^T\in Z_p^n$ 使其满足 $\vec{M}_i^*.\vec{\omega}=0$. 其中, $i\in\left\{i\,|\,i\in\left[\ell\wedge\rho^*(i)\in S_{ID_q}\right]\right\}$. 从LSSS的定义可以知道,由于 S_{ID_q} 不满足矩阵 M^*,因此,向量 $\vec{\omega}$ 一定存在.然后定义 $r=\tilde{r}+H(ID)(\omega_1a^q+\omega_2a^{q-1}+...+\omega_na^{q+1-n})=\tilde{r}+H(ID)\sum_{i\in[n]}\omega_ia^{q+1-i}$, 随后构建密钥组件 D_0、D_1 如下:

$$\left.\begin{aligned}D_0&=g^{\tilde{\alpha}/\beta}\left(g^{a/\beta}\right)^{\tilde{r}}\prod_{i=2}^n\left(g^{a^{q+2-i}/\beta}\right)^{H(ID)\omega_i},\\D_1&=g^{\tilde{r}}\prod_{i\in[n]}^n\left(g^{a^{q+1-i}}\right)^{ID\omega_i}\end{aligned}\right\} \quad (10)$$

另外,对于每个属性 $A_\tau \in S_{ID_q}$,其中,$\tau \in [|S|]$;仿真者 $\mathbb{C}$ 设置 r_τ,具体如下:

$$r_\tau = \tilde{r}_\tau + \tilde{r}.\sum_{\substack{i' \in [\ell] \\ \rho^*(i') \notin S}} \frac{b_{i'}}{A_\tau - \rho^*(i')} + \sum_{\substack{i' \in [n,\ell] \\ \rho^*(i') \notin S}} \frac{\omega_i b_{i'} a^{q+1-i}}{A_\tau - \rho^*(i')} \tag{11}$$

最后,为每个属性计算如下密钥组件,其中,$A_\tau \in S_{ID_q}$.

$$D_{\tau,2} = g^{r_\tau} \tag{12}$$

$$\left.\begin{aligned} D_{\tau,3} = v^{-\tilde{r}} \prod_{i\in[n]} \left(g^{a^{q+1-i}}\right)^{-H(ID)\tilde{v}\omega_i} \cdot \prod_{(i,j,k)\in[n,\ell,n], i\neq k} \left(g^{\frac{a^{q+k+1-i}}{b_j}}\right)^{-H(ID)\omega_i M^*_{j,k}} \cdot \\ \left(u^{A_\tau} h\right)^{H(ID)\tilde{r}_\tau} \cdot \left(K_{\tau,2} \Big/ g^{\tilde{r}_\tau}\right)^{H(ID)(\tilde{u}A_\tau + \tilde{h})} \cdot \prod_{(i,j,k)\in[\ell,\ell,n], \rho^*(i')\notin S} g^{H(ID)\tilde{r}(A_\tau - \rho^*(j)) M^*_{j,k} b_{i'}^k} \Big/ (A_\tau - \rho^*(i')) b_j^2 \cdot \\ \prod_{(i,i',j,k)\in[n,\ell,\ell,n], \rho^*(i')\notin S} g^{H(ID)(A_\tau - \rho^*(j))\omega_i M^*_{j,k} b_{i'}{}^{a^{q+1+k-i}}} \Big/ (A_\tau - \rho^*(i')) b_j^2 \\ = \left(u^{A_\tau} h\right)^{r_\tau H(ID)} v^{-r} \end{aligned}\right\} \tag{13}$$

仿真者响应私钥 $SK^* = \left(D_0, D_1, \left\{D_{\tau,2}, D_{\tau,3}\right\}_{\tau\in[|S_{ID_q}|]}, S_{ID_q}\right)$ 给敌手 $\mathbb{A}$.

挑战阶段:敌手 $\mathbb{A}$ 提交两个长度相同内容不同的信息 M_0 和 M_1 给仿真者.首先仿真者 $\mathbb{C}$ 随机选择一个数 $\theta \in \{0,1\}$;其次产生密文组件 $C = m_\theta \cdot T \cdot e(g,g^s)^{\tilde{\alpha}}$ 和 $C_0 = g^s$.然后仿真者 $\mathbb{C}$ 随机选择 $(z_1,...,z_\ell, z'_1,...,z'_\ell, \tilde{y}_2,..., \tilde{y}_\ell) \in Z_p$,并使用向量 $\vec{y} = \left(s, sa + \tilde{y}_2, sa^2 + \tilde{y}_3,..., sa^{n-1} + \tilde{y}_n\right)^T$ 共享秘密值 s.另外,为 $\tau \in [\ell]$ 设置 $\lambda_\tau = \sum_{i\in[n]} M^*_{\tau,i} sa^{i-1} + \sum_{i=2} M^*_{\tau,i} \tilde{y}_i = \sum_{i\in[n]} M^*_{\tau,i} sa^{i-1} + \tilde{\lambda}_\pi$ 和 $t_\tau = -sb_\tau$;然后通过下式设置密文组件 $C_{\tau,1}, C_{\tau,2}, C_{\tau,3}, C_{\tau,4}, C_{\tau,5}$.

$$\left.\begin{aligned} C_{\tau,1} &= w^{\tilde{\lambda}_\tau} \cdot \left(g^{sb_\tau}\right)^{-\tilde{v}} \cdot \prod_{\substack{(j,k)\in[\ell,n] \\ j\neq\tau}} \left(g^{sa^k b_\tau / b_j}\right)^{-M^*_{j,k}} . w^{-z_j}, \\ C_{\tau,2} &= \left(g^{sb_\tau}\right)^{-\left(\tilde{u}\rho^*(\tau)+\tilde{h}\right)} \cdot \prod_{\substack{(j,k)\in[\ell,n] \\ j\neq\tau}} \left(g^{sa^k b_\tau / b_j^2}\right)^{-\left(\rho^*(\tau)-\rho^*(j)\right)M^*_{j,k}} . u^{z'_j}, \\ C_{\tau,3} &= \left(g^{sb_\tau}\right)^{-1}, \\ C_{\tau,4} &= z_j, \\ C_{\tau,5} &= z'_j \end{aligned}\right\} \tag{14}$$

最后,仿真者 $\mathbb{C}$ 把挑战密文 $CT^* = \left(M^*, C, C_0, \left\{C_{\tau,1}, C_{\tau,2}, C_{\tau,3}, C_{\tau,4}, C_{\tau,5}\right\}_{\tau=1...\ell}\right)$ 发送给敌手 $\mathbb{A}$.

查询阶段 2:重复查询阶段 1.

猜测阶段:敌手 $\mathbb{A}$ 输出一个作为 β' 对 β 的猜测.如果 $\beta = \beta'$,则仿真者 $\mathbb{C}$ 输出 0,即 $T = e(g,g)^{a^{q+1}s}$;否则,仿真者 $\mathbb{C}$ 输出 1,即 T 是一个随机数.如果 $T = e(g,g)^{a^{q+1}s}$,则仿真者 $\mathbb{C}$ 进行真实的仿真,因为 $C = m_b \cdot T \cdot e(g,g^s)^{\tilde{\alpha}} = m_b \cdot e(g,g)^{\alpha s}$;如果为随机数,则敌手 $\mathbb{A}$ 的优势为 0.因此,敌手 $\mathbb{A}$ 以不可忽略的优势打破以上游戏,仿真者 $\mathbb{C}$ 能以不可忽略的优势打破 q-type 假设. □

4.3 性能分析

在表 1 中,本文方案从访问结构的类型、是否离线加密、外包解密、外包可验证及授权机构数量方面,与以前的 ABE 方案进行了对比.从对比中可以看出,本文方案同时实现了多授权机构 ABE 模型下的离线加密、外

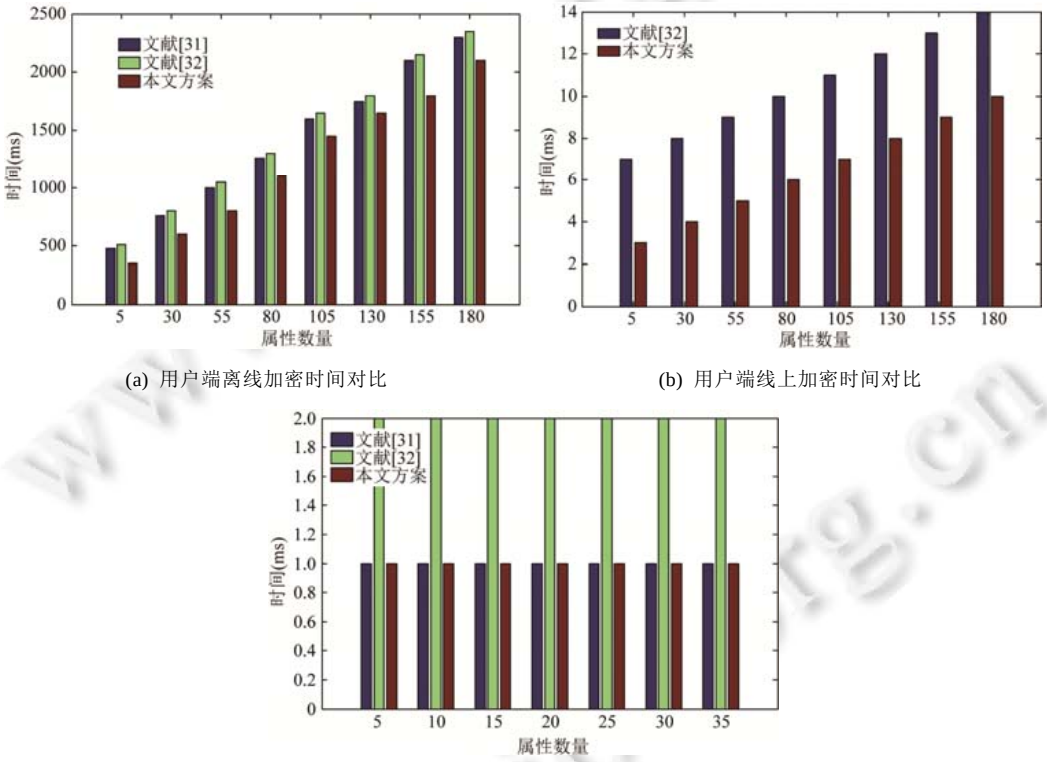
包解密以及外包结果可验证的功能,文献[15]中的方案、文献[31,32]中的方案仅实现了其中部分功能.相比而言,本文方案支持的功能更加丰富,实用性更强.

我们对本文提出的方案、文献[31]中的方案及文献[32]中的方案进行了仿真实验,并对 3 个不同方案的离线加密、线上加密和用户端解密的时间进行了对比分析.所有实验程序均采用 Java 语言编写,并在 Eclipse 下运行,微机环境为 Windows 操作系统 Intel(R) Core(TM) i3 CPU 2.0GHz 和 2GB RAM 内存.同时,本文方案采用了 JPBC 中提出的基于椭圆曲线 $y^2=x^3+x$ 构造的 160 位椭圆曲线群.另外,为降低实验中随机因素的影响,我们针对不同程序的每一种情况都独立运行 20 次实验,实验结果如图 2 所示.

Table 1 Comparison of flexibility of OO-MA-ABE scheme

表 1 OO-MA-ABE 方案功能对比

方案	访问结构	离线	外包	可验证	授权机构数量
文献[31]	任意‘LSSS’	否	是	否	多个
文献[32]	任意‘LSSS’	是	是	否	多个
文献[15]	任意‘LSSS’	是	是	否	单个
本文方案	任意‘LSSS’	是	是	是	多个



(a) 用户端离线加密时间对比 (b) 用户端线上加密时间对比

(c) 用户端解密时间对比

Fig.2 Computation time analysis of OO-MA-ABE scheme

图 2 OO-MA-ABE 方案计算时间分析

图 2(a)中,我们首先给出了数据拥有者的离线加密时间对比.从图中可以看出,随着系统中属性数量的不断增加,本文方案中数据拥有者在离线阶段所花费的加密时间要明显少于文献[31]和文献[32]的方案.这对系统的整体效率来说是一个非常大的提升.然后,我们对数据拥有者的线上加密时间进行了对比分析,如图 2(b)所示.在移动云计算环境下,提高移动设备的线上运行效率,降低能耗,延长设备使用时间是非常重要的.从图中可以看出,我们的方案在数据拥有者端加密耗费的时间要远远少于文献[32]的方案,这说明,相较其他方案,本文方案在移动云环境下具有更强的实用性.最后,我们对用户端的解密代价进行了分析,如图 2(c)所示.在用户解密阶段,

本文方案和文献[31]的方案用户端均只需进行一次指数运算(除法运算花费时间很少,可忽略不计),解密花费时间相同.而在文献[32]所提方案中,用户在解密阶段需要进行两次指数运算,所消耗时间远远超过本文方案需要的解密时间.所以,本文方案中用户所花费的代价相较而言是最低的.综上所述,本文的方案在性能和功能上均优于其他方案.

5 结 论

本文为了处理多授权机构属性基加密访问控制方案中加密和解密计算代价问题,提出了高效可验证的多授权机构属性基加密云存储数据访问控制方案.该方案通过把加密过程分为两部分,即离线加密和线上加密;把加密阶段所有的配对操作在离线阶段预处理,来减少线上加密阶段的计算开销.另外,本文方案通过外包解密的方式减少用户端解密计算的代价,同时对外包的计算进行了验证,保证了云存储服务器解密的正确性.本文方案可以抵抗单个授权机构获取用户的所有属性,一定程度上保护了用户的身份隐私.最后,对本文提出的方案进行了安全性分析和仿真实验,结果表明了方案的高效性及安全性,可用于部署到移动云存储平台.

References:

[1] Yao X, Han X, Du X. A lightweight access control mechanism for mobile cloud computing. In: Proc. of the 2014 IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS). IEEE, 2014. 380–385.

[2] Ren W, Zeng L, Liu R, Cheng C. F2AC: A lightweight, fine-grained, and flexible access control scheme for file storage in mobile cloud computing. Mobile Information Systems, 2016.

[3] Xie Y, Wen H, Wu B, Jiang Y, Meng J. A modified hierarchical attribute-based encryption access control method for mobile cloud computing. In: Proc. of the Cloud Computing. 2016.

[4] Nag A, Choudhary S, Dawn S, Basu S. Secure data outsourcing in the cloud using multi-secret sharing scheme (MSSS). In: Proc. of the 1st Int'l Conf. on Intelligent Computing and Communication. Singapore: Springer-Verlag, 2017. 337–343.

[5] Chattopadhyay AK, Nag A, Majumder K. Secure data outsourcing on cloud using secret sharing scheme. IJ Network Security, 2017,19(6):912–921.

[6] Wang S, Zhou J, Liu JK, Yu J, Cheng J, Xie W. An efficient file hierarchy attribute-based encryption scheme in cloud computing. IEEE Trans. on Information Forensics and Security, 2016,11(6):1265–1277.

[7] Xu J, Wen Q, Li W, Jin Z. Circuit ciphertext-policy attribute-based hybrid encryption with verifiable delegation in cloud computing. IEEE Trans. on Parallel and Distributed Systems, 2016,27(1):119–129.

[8] Lei L, Cai QW, Jing JW, Wang Z, Chen B. Enforcing access controls on encrypted cloud storage with policy hiding. Ruan Jian Xue Bao/Journal of Software, 2016,27(6):1432–1450 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5003.htm [doi: 10.13328/j.cnki.jos.005003]

[9] Wang Z, Huang D, Zhu Y, Li B, Chung CJ. Efficient attribute-based comparable data access control. IEEE Trans. on Computers, 2015,64(12):3430–3443.

[10] Wang H, Zheng Z, Wu L, He D. New large-universe multi-authority ciphertext-policy ABE scheme and its application in cloud storage systems. Journal of High Speed Networks, 2016,22(2):153–167.

[11] Jung T, Li X, Wan Z, Wang M. Control cloud data access privilege and anonymity with fully anonymous attribute-based encryption. IEEE Trans. on Information Forensics and Security, 2015,10(1):190–199.

[12] Hohenberger S, Waters B. Online/Offline attribute-based encryption. In: Public-Key Cryptography-PKC 2014. Berlin, Heidelberg: Springer-Verlag, 2014. 293–310.

[13] Shao J, Zhu Y, Ji Q. Privacy-Preserving online/offline and outsourced multi-authority attribute-based encryption. In: Proc. of the 16th IEEE/ACIS Int'l Conf. on Computer and Information Science (ICIS). IEEE, 2017. 285–291.

[14] Qin B, Deng RH, Liu S, Ma S. Attribute-Based encryption with efficient verifiable outsourced decryption. IEEE Trans on Information Forensics and Security, 2015,10(7):1384–1393.

[15] Shao J, Lu R, Lin X. Fine-Grained data sharing in cloud computing for mobile devices. In: Proc. of the 2015 IEEE Conf. on Computer Communications (INFOCOM). IEEE, 2015. 2677–2685.

[16] Sahai A, Waters B. Fuzzy identity-based encryption. Eurocrypt, 2005,3494:457–473.

[17] Han J, Susilo W, Mu Y, Zhou J, Au MHA. Improving privacy and security in decentralized ciphertext-policy attribute-based encryption. IEEE Trans. on Information Forensics and Security, 2015,10(3):665–678.

[18] Tang H, Cui Y, Guan C, Wu J, Weng J, Ren K. Enabling ciphertext deduplication for secure cloud storage and access control. In: Proc. of the 11th ACM on Asia Conf. on Computer and Communications Security. ACM, 2016. 59–70.

[19] Li J, Yao W, Zhang Y, Qian H, Han J. Flexible and fine-grained attribute-based data storage in cloud computing. IEEE Trans. on Services Computing, 2016.

[20] Zhou Z, Huang D, Wang Z. Efficient privacy-preserving ciphertext-policy attribute based-encryption and broadcast encryption. IEEE Trans. on Computers, 2015,64(1):126–138.

[21] Yanli C, Lingling S, Geng Y. Attribute-Based access control for multi-authority systems with constant size ciphertext in cloud computing. China Communications, 2016,13(2):146–162.

[22] Phuong TVX, Yang G, Susilo W. Hidden ciphertext policy attribute-based encryption under standard assumptions. IEEE Trans. on Information Forensics and Security, 2016,11(1):35–45.

[23] Ruj S, Stojmenovic M, Nayak A. Decentralized access control with anonymous authentication of data stored in clouds. IEEE Trans. on Parallel and Distributed Systems, 2014,25(2):384–394.

[24] Chase M. Multi-Authority attribute based encryption. In: Proc. of the Conf. on Theory of Cryptography. LNCS 4392, Berlin, Heidelberg: Springer-Verlag, 2007. 515–534.

[25] Lewko A, Waters B. Decentralizing attribute-based encryption. In: Advances in Cryptology–EUROCRYPT 2011. Berlin, Heidelberg: Springer-Verlag, 2011. 568–588.

[26] Guo F, Mu Y, Chen Z. Identity-Based online/offline encryption. In: Proc. of the Int'l Conf. on Financial Cryptography and Data Security. Berlin, Heidelberg: Springer-Verlag, 2008. 247–261.

[27] Even S, Goldreich O, Micali S. On-Line/Off-Line digital signatures. In: Proc. of the Conf. on the Theory and Application of Cryptology. New York: Springer-Verlag, 1989. 263–275.

[28] Hohenberger S, Waters B. Online/Offline attribute-based encryption. In: Proc. of the Int'l Workshop on Public Key Cryptography. Berlin, Heidelberg: Springer-Verlag, 2014. 293–310.

[29] Rouselakis Y, Waters B. Practical constructions and new proof methods for large universe attribute-based encryption. In: Proc. of the 2013 ACM SIGSAC Conf. on Computer & Communications Security. ACM, 2013. 463–474.

[30] Green M, Hohenberger S, Waters B. Outsourcing the decryption of ABE ciphertexts. In: Proc. of the USENIX Security Symp. 2011. 34.

[31] Yang K, Jia X. DAC-MACS: Effective data access control for multi-authority cloud storage systems. In: Security for Cloud Storage Systems. New York: Springer-Verlag, 2014. 59–83.

[32] De SJ, Ruj S. Decentralized access control on data in the cloud with fast encryption and outsourced decryption. In: Proc. of the 2015 IEEE Global Communications Conf. (GLOBECOM). IEEE, 2015. 1–6.

附中文参考文献:

[8] 雷蕾,蔡权伟,荆继武,林璟锵,王展,陈波.支持策略隐藏的加密云存储访问控制机制.软件学报,2016,27(6):1432–1450. http://www.jos.org.cn/1000-9825/5003.html [doi: 10.13328/j.cnki.jos.005003]

仲红(1965－),女,安徽固镇人,博士,教授,博士生导师,CCF 专业会员,主要研究领域为网络(无线传感网,车联网,SDN 软件定义网),信息安全(大数据隐私保护,云安全,边缘计算).

崔杰(1980－),男,博士,副教授,CCF 专业会员,主要研究领域为网络(无线传感网,车联网,SDN 软件定义网),信息安全(大数据隐私保护,云安全,边缘计算).

朱文龙(1990－),男,硕士,主要研究领域为云计算,大数据隐私保护.

许艳(1982－),女,博士,讲师,主要研究领域为云计算,数据隐私保护,物联网安全.