专题

云计算中基于可转换代理签密的可证安全的认证协议

谢琪^{①②*}, 吴吉义^{②④}, 王贵林^③, 刘文浩^①, 陈德人^④, 于秀源^①

① 杭州师范大学信息科学与工程学院, 密码及应用研究所, 杭州 310036

② 杭州师范大学电子商务与信息安全重点实验室, 杭州 310036

③ School of Computer Science & Software Engineering, University of Wollongong, Wollongong NSW 2522, Australia

④ 浙江大学计算机学院, 杭州 310027

* 通信作者. E-mail: qixie68@yahoo.com.cn

收稿日期: 2011-08-29; 接受日期: 2012-01-18

国家自然科学基金 (批准号: 61070153) 资助项目

摘要 云用户与公有云之间的双向认证是云计算中用户访问公有云的重要前提. 2011年, Juang等首次提出了云计算环境下采用代理签名的认证协议, 其优点是用户只需到私有云中注册, 然后在私有云的帮助下通过公有云的认证. 但是, 该方案存在3个缺陷: 1) 为保护用户的隐私, 每次会话都需更新用户公钥; 2) 当私有云中的许多用户同时登录不同的公有云时, 私有云会遭遇网络拥堵; 3) 用户的私有云与访问的公有云之间需要预先共享秘密. 为弥补上述不足, 提出了一种保护用户隐私的可证安全的可转换代理签密方案, 基于该方案设计了一种一轮云计算认证协议. 新方案的优点在于用户向私有云注册后, 就能通过公有云的认证, 而不需要私有云的帮助, 并且它还能保护用户的隐私性、抗抵赖性. 协议不需要在每次会话开始前更新用户公钥, 同时私有云与访问的公有云之间不再需要预先共享秘密. 在随机谰言机模型下证明了新协议的安全性, 并且比较说明新协议在效率方面优于 Juang 等的协议.

关键词 云计算 认证协议 可转换 代理签名 签密 可证安全

1 引言

云计算是通过网络为用户提供计算、存储数据、软件以及平台等服务的一种计算模式. 在云计算环境下, 用户不再需要购买相应的基础设施与计算设备等硬件和软件, 从而减少管理成本. 用户通过网络从云计算中心获得所需服务、并支付相应的费用给云计算服务提供商^[1,2]. 因此, 云计算具有广泛的应用前景. 但是, 随着云计算的发展, 由欺诈行为驱动的“黑云”则发展更快, 导致云计算资源的拥有者、提供商与被提供服务的用户间出现信任危机, 阻碍了云计算成为主流服务平台. 因此, 企业需要增强身份认证与欺诈识别能力以应对未经授权的接入、网络钓鱼、恶意软件, 甚至是知识产权偷窃行为带来的安全威胁^[3].

云计算中的云分为私有云、公有云和混合云3种. 其中, 私有云是为某个单位内部用户单独使用而构建的, 它能实现对数据、安全性和服务质量的最有效控制. 并且, 该单位拥有基础设施, 可以控制

引用格式: 谢琪, 吴吉义, 王贵林, 等. 云计算中基于可转换代理签密的可证安全的认证协议. 中国科学: 信息科学, 2012, 42: 303-313, doi: 10.1360/112011-915

图 1 Juang 等协议模型

Figure 1 Juang et al.'s protocol model

在此基础设施上部署应用程序的方式; 私有云一般部署在单位数据中心的防火墙内, 也可被部署于主机托管场所. 而公有云通常指第三方提供商通过自己的基础设施直接向外部用户提供服务. 这样, 外部用户就可通过互联网以低廉的价格获得公有云提供的服务. 分布在不同地方的不同公有云能够为用户提供不同的服务. 无论是从安全性还是从费用支付的角度看, 用户与公有云之间实现双向认证是不可少的. 为解决这个问题, 最简单和直接的方法是要求用户向不同的公有云进行注册, 然后在需要提供服务时进行双向认证. 很显然, 这种做法存在许多弊端: 用户必须记住在不同公有云中所注册的口令, 而公有云则需要维护注册用户的信息. 这既不方便, 又增加了额外管理成本, 同时用户注册表容易受到偷窃攻击的威胁, 降低了系统安全性^[4]. 2005 年, 针对移动用户的漫游问题, Lee 等^[5]提出了基于代理签名的认证协议, 其优点是用户只需向自己的移动网络注册, 然后在自己的移动网络的帮助下就能通过访问网络的认证. 但是, Lee 等^[6]和 Wang 等^[7]指出文献^[5]的方案是不安全的, 因为该方案不能保护用户的隐私, 同时它也不具有抗抵赖性的安全属性. 2011 年, Juang 等^[8]基于文献^[5~7]的设计思想, 提出了第一个可用于云计算环境中的认证协议. 在他们的方案中, 用户只需向自己所属的私有云中注册, 然后在私有云的帮助下, 即可实现与公有云的双向认证.

图 1 为 Juang 等^[8]提出的协议模型, 其中 HSC 代表私有云, User 表示私有云 HSC 中的用户, VC 表示用户 User 想要访问的公有云. Juang 等的模型分为用户注册、用户与访问公有云之间的双向认证两个阶段. (1) 在注册阶段, 用户提交身份等基本信息到所属的私有云, 私有云产生代理签名密钥并发送给用户; (2) 在双向认证阶段, 用户首先向访问公有云发送认证请求; 公有云把收到的用户请求消息与自己产生的消息经共享秘密加密后一起发送给用户所属的私有云, 请求其帮助认证用户的合法性; 用户所属的私有云认证用户的合法性, 若合法则更新用户的公钥, 并把结果经共享秘密加密后反馈给访问公有云; 访问公有云收到私有云的认证结果后, 回复认证信息给用户, 让用户验证公有云的合法性; 用户验证公有云的合法性. 仔细分析文献^[8]的方案, 发现他们的方案中存在 3 个缺点: 1) 为了保护用户的隐私, 私有云需要更新用户的公钥, 而私有云更新用户的公钥缺乏权威性, 应该由密钥管理中心更新比较合理, 但也不能在每次认证过程中更新; 2) 当同一个私有云中的许多用户需要同时登入不同的公有云时, 私有云会遇到网络堵塞; 3) 因为用户要实现与公有云的双向认证, 需要私有云的帮助, 这样, 他们的方案中私有云与公有云需要预先共享秘密. 除此之外, 他们的方案需要多轮认证, 也没有对方案进行安全性证明.

为解决这些问题, 我们提出了新的协议模型 (图 2 所示), 同样分为用户注册、用户与访问公有云之间的双向认证两个阶段. 1) 在注册阶段, 用户与私有云交互来产生代理签名密钥; 2) 在双向认证

图 2 提出的协议模型

Figure 2 The proposed protocol model

阶段, 用户首先向访问公有云发送认证请求; 访问公有云认证用户的合法性, 若合法则回复认证信息给用户, 让用户验证公有云的合法性. 为了能够实现该协议, 提出了一种保护用户隐私的可证安全的可转换代理签密方案, 基于该方案设计了云计算认证协议, 优点是用户只要到自己的私有云中注册, 在不需要私有云的帮助下实现用户与公有云的一轮双向认证. 同时, 提出的方案具有保护用户隐私性、抗抵赖性、不需要更新用户的公钥、私有云与公有云不需要预先共享秘密等优点, 而且在随机谰言机模型下是可证安全的, 效率优于已有方案.

2 提出的方案

本节首先提出了一种保护用户隐私的可转换代理签密方案; 然后, 基于该方案提出了一种应用于云计算的一轮认证协议.

2.1 可转换代理签密方案

密钥产生中心 KGC 选取大素数 q , $E(GF_q)$ 是定义在有限域 $GF(q)$ 上的椭圆曲线, P 是 $E(GF_q)$ 的生成元, l 是安全参数, KGC 为原始签名人、代理签名人、指定接收人产生公私钥对 $(x_0 \in Z_q^*, y_0 = x_0P)$, $(x_d \in Z_q^*, y_d = x_dP)$ 和 $(x_v \in Z_q^*, y_v = x_vP)$, $H(): \{0, 1\}^* \rightarrow \{0, 1\}^{256}$ 是安全的 Hash 函数 (例如 SHA-2), $E_\varepsilon()$ 和 $D_\varepsilon()$ 是以 ε 为密钥的安全对称加密和解密算法 (例如 AES). 提出的签名方案主要由代理签名密钥产生、签名、消息恢复与签名验证、签名转换组成.

1) 代理签名密钥产生.

代理签名密钥的生成采用 Nicolosi 等 [9] 提出的可证明安全的两方签名方案, 过程如下:

Step1: 原始签名人随机选择 $k_0 \in Z_q^*$, 计算 $r_0 = k_0P$, $c_0 = H(r_0)$, 构造代理证书 m_w , 然后, 他把 (c_0, m_w) 发送给代理签名人.

Step2: 代理签名人收到 (c_0, m_w) 后, 随机选择 $k_1 \in Z_q^*$, 计算并发送 $r_1 = k_1P$ 给原始签名人.

Step3: 原始签名人验证 $qr_1 = O$ 是否成立, 这里 O 为椭圆曲线上的无穷远点. 如果成立, 则计算 $r = r_0 + r_1$, $\sigma = k_0 + x_0H(r||m_w||y_d||y_0)$, 然后把 (r_0, σ) 发送给代理签名人.

Step4: 代理签名人计算 $r = r_0 + r_1$, 通过下面两式验证 (c_0, m_w, σ) 的合法性:

$$c_0 = H(r_0), \quad c_0 = H(\sigma P - y_0 H(r || m_w || y_d || y_0)).$$

如果都成立, 则代理签名人计算代理签名密钥 $x_s = \sigma + k_1 + x_d H(r \| m_w \| y_d \| y_0)$, 其对应的公钥为 $Y = r + H(r \| m_w \| y_d \| y_0)(y_0 + y_d)$; 否则要求原始签名人重新计算 (r_0, σ) .

2) 签名.

代理签名人想要对消息 m 签名, 他随机选择 $a \in Z_q^*$, $R_1 = aP$, $R_2 = ay_v$, $c = a + x_s H(R_1 \| y_v \| R_2 \| m)$, $R_3 = E_{R_2}(c \| m_w \| y_0 \| r \| y_d \| m)$. 最后, 代理签名人把 (R_1, R_3) 发送给指定接收人.

3) 消息恢复与签名验证.

当指定接收人收到来自代理签名人的 (R_1, R_3) 以后, 计算 $R_2 = x_v R_1$, $c \| m_w \| y_0 \| r \| y_d \| m = D_{R_2}(R_3)$, 恢复出原始签名人和代理签名人的公钥 y_0 和 y_d , 代理证书 m_w 和参数 r , 部分签名 c 以及签名的消息 m , 然后从 KGC 的公告牌中检查 y_0 和 y_d 的合法性, 最后通过下面的式子验证签名:

$$cP = R_1 + H(R_1 \| y_v \| R_2 \| m)(r + H(r \| m_w \| y_d \| y_0)(y_0 + y_d)).$$

如果成立, 则签名有效.

提出的代理签密方案与一般的代理签密方案的区别是: 原始签名人和代理签名人的公钥、代理证书和参数不能作为公开参数传输, 也需要同消息一样进行恢复, 这样可以保证他们身份的保密性, 以应用于云计算的认证协议.

4) 签名转换.

如果代理签名人事后抵赖, 则指定接收人可以公开参数 $(c, r, m_w, R_1, R_2, m, y_0, y_d)$, 则仲裁者就可以通过验证式验证签名的有效性.

2.2 提出的认证协议

参数的选取同 2.1 小节, 同时 KGC 为用户 User、用户所属的私有云 HSC、想要访问的公有云 VC 产生公私钥对, 即 $x_i (i \in VC, u, HSC)$ 是私钥, 对应的公钥为 $y_i = x_i P$. 提出的方案主要由两部分组成: 用户 User 到所属的私有云 HSC 中注册, 用户与需要访问的公有云 VC 进行双向认证.

1) 用户注册.

用户 User 希望成为私有云 HSC 的合法用户, 他需要到 HSC 中注册, 注册完成后, HSC 与 User 共同产生代理签名密钥. HSC 需要构造代理证 m_w , 包含用户 User 的身份、指明是 HSC 中的合法用户、有效期限、具体的权限等信息. 而代理签名密钥的生成依然采用 Nicolosi 等^[9]提出的可证明安全的两方签名方案, 具体过程同 2.1 小节中的代理签名密钥生成过程. 最后, User 获得代理签名密钥 $x_s = \sigma + k_1 + x_u H(r \| m_w \| y_u \| y_{HSC})$, 其中 $\sigma = k_0 + x_{HSC} H(r \| m_w \| y_u \| y_{HSC})$, 其对应的公钥为 $Y = r + H(r \| m_w \| y_u \| y_{HSC})(y_{HSC} + y_u)$.

2) 双向认证协议.

User 希望访问公共云 VC, 他和 VC 一起执行以下协议实现双向认证, 并获得会话密钥.

Step1: User $\rightarrow$ VC: (R_1, R_3) .

User 随机选择 $a, e \in Z_q^*$, $R_1 = aP$, $R = eP$, $R_2 = ay_{vc}$, $c = a + x_s H(R_1 \| y_{vc} \| R_2 \| t \| r \| R)$, $R_3 = E_{R_2}(c \| m_w \| y_u \| t \| r \| R)$, 这里 t 是当前的时间. 最后, User 把 (R_1, R_3) 发送给 VC.

Step2: VC $\rightarrow$ User: (R_4, V) .

VC 在时间 t' 收到来自 User 的 (R_1, R_3) 以后, 计算 $R_2 = x_{vc} R_1$, $c \| m_w \| y_u \| t \| r \| R = D_{R_2}(R_3)$, 恢复出 User 的公钥 y_u , 代理证书 m_w, c, R , 以及 r 和 t ; 然后从 m_w 确认 User 是 HSC 中的一个合法用户, 检查 $t' - t \leq \Delta t$ (Δt 是合理的时间间隔) 和 $cP = R_1 + H(R_1 \| y_{vc} \| R_2 \| t \| r \| R)(r + H(r \| m_w \| y_u \| y_{HSC})(y_{HSC} + y_u))$.

如果全部成立, 则 VC 确信 (R_1, R_3) 来自于 HSC 中的合法用户 User. 然后 VC 随机选择 $b \in Z_q^*$, 计算 $R_4 = bP, SK = bR = beP, V = h(SK)$, 最后把 (R_4, V) 发送给 User. 否则, 放弃.

如果 User 事后存在抵赖行为, 则 VC 可以公开 $(c, R_1, R, R_2, t, m_w, r)$, 则仲裁者可以判定 User 是否存在抵赖行为.

Step3: User 收到 (R_4, V) 后, 计算 $SK' = eR_4 = ebP$, 验证 $h(SK') = V$ 是否成立. 如果不成立, 则终止协议. 如果成立, 则 User 确信 VC 是合法的, 因为只有 VC 拥有 x_{vc} 才能计算 R_2 , 并获得正确的 R , 这样他们共享会话密钥 $SK = beP$. 这样 User 和 VC 之间可以通过 SK 作为对称加密密钥传输信息, 获得服务. 而且共享的会话密钥 SK 具有完全的前向安全性.

3 安全性证明

首先, 证明提出的可转换代理签密方案在随机谰言机模型下是安全的. 继而, 证明新提出的协议也是安全的.

3.1 可转换代理签密方案的安全性证明

定义 1 可转换代理签密方案的安全性主要包含不可伪造性、消息保密性、不可验证性和不可转换性. 针对不可伪造性, 有如下 3 类攻击者:

A_1 : 拥有原始签名人、代理签名人、指定接收人的公钥, 但可以与指定接收人合谋; 目标是对选定的消息生成有效的签名.

A_2 : 拥有原始签名人、代理签名人、指定接收人的公钥, 但可以与指定接收人合谋, 并控制原始签名人; 目标是对原来的代理证书或者伪造的代理证书、对选定的消息生成有效的签名.

A_3 : 拥有原始签名人、代理签名人、指定接收人的公钥, 但可以与指定接收人合谋, 并控制代理签名人; 目标是伪造代理证书、对选定的消息生成有效的签名.

显然, 一般的代理签名方案若能够抵抗 A_2 或者 A_3 的攻击, 那么就能抵抗 A_1 的攻击. 在提出的方案中, 代理签名密钥的生成采用 Nicolosi 等提出的可证明安全两方签名方案. 因此, 无论是原始签名人还是代理签名人都无法单独生成代理签名密钥. 也就是说, 无论原始签名人还是代理签名人都无法成功修改代理证书. 因此, 提出方案的不可伪造性只需要证明攻击者 A 可以与指定接收人和原始签名人合谋, 不能修改代理证书, 并且对选定的任意消息不能生成有效的签名即可.

定理 1 在随机谰言机模型下, 如果攻击者 A 在概率多项式时间 t 内至多 q_s 次签名问询, q_h 次 Hash 问询, 以不可忽略的概率 $\varepsilon \geq 10(q_h + q_s)(q_s + 1)/2^t$ 能伪造可转换代理签密, 那么离散对数问题能在概率多项式时间 t' 内以不可忽略的概率 ε' 得到解决, 其中

$$\varepsilon' \geq 1/9, t' \leq 120686q_h t / \varepsilon + 3(q_s + q_h + 2)T_M.$$

T_M 表示椭圆曲线上的点乘计算时间.

证明 挑战者把公共参数以及原始签名人、代理签名人、指定接收人的公钥、指定接收人和原始签名人的私钥给 A , 要求 A 不能修改代理证书、对选定的消息生成有效的签名. A 能够向挑战者询问签名 oracle 问询、Hash oracle 问询、验证 oracle 问询、转换 oracle 问询. 挑战者事先建立 Hash 问询表 h -list, Sign 问询表 S -list, 初始时空. 令代理签名的公钥为 $Y = x_s P$.

签名问询: A 发送 (m, r, m_w) 给挑战者, 并问询签名 oracle. 挑战者查找 S -list, 如果在 S -list 中存在 (m, r, m_w) , 则返回 (R_1, R_3) 给 A , 并令 $\text{flag} = 1$; 否则, 挑战者随机选择 $c, H \in Z_q^*$, 计算

$R_1 = cP - HY, R_2 = x_v R_1, R_3 = E_{R_2}(c \| m_w \| y_0 \| r \| y_d \| m)$; 然后把 $(m, r, m_w, y_v, c, H, R_1, R_3, R_2, \text{flag} = 0)$ 记入 $S\text{-list}$, 把 (R_1, y_v, R_2, m, H) 记入 $h\text{-list}$, 这里 $H = H(R_1 \| y_v \| R_2 \| m)$, flag 是布尔 flag , 用来区分攻击者是否已经询问过签名. 最后, 返回 (R_1, R_3) 给 A , 并令 $\text{flag} = 1$.

Hash 问询: A 发送 (y_v, m) 给挑战者, 并问询 Hash oracle. 挑战者查找 $h\text{-list}$, 如果在 $h\text{-list}$ 中存在 (y_v, m) , 则返回 H 给 A ; 否则, 挑战者随机选择 $c, H \in Z_q^*$, 计算 $R_1 = cP - HY, R_2 = x_v R_1, R_3 = E_{R_2}(c \| m_w \| y_0 \| r \| y_d \| m)$; 然后把 $(m, r, m_w, y_v, c, H, R_1, R_3, R_2, \text{flag} = 0)$ 记入 $S\text{-list}$, 把 (R_1, y_v, R_2, m, H) 记入 $h\text{-list}$, 这里 $H = H(R_1 \| y_v \| R_2 \| m)$. 最后, 返回 H 给 A .

验证问询: A 发送 (R_1, R_3) 给挑战者, 并问询验证 oracle. 挑战者查找 $S\text{-list}$, 如果存在 (R_1, R_3) , 则返回 1; 否则, 返回 $\perp$.

转换问询: A 发送 (R_1, R_3) 给挑战者, 并问询转换 oracle. 挑战者查找 $S\text{-list}$, 如果存在 (R_1, R_3) , 则返回 (c, r, m_w, R_1, R_2, m) , 并令 $\text{flag} = 1$; 否则, 返回 $\perp$.

输出签名: 经过多项式次问询以后, A 针对消息 m^* 构造一个有效签名 (R_1^*, m^*, c^*, H^*) , 这里 $H^* = H(R_1^* \| y_v \| R_2^* \| m^*), R_3^* = E_{R_2^*}(c^* \| m_w \| y_0 \| r \| y_d \| m^*)$. 最后, 把 (R_1^*, R_3^*) 发送给挑战者. 挑战者计算 $R_2^* = x_v R_1^*$, 解密 R_3^* , 恢复 $y_0, y_d, r, m_w, c^*, m^*$, 然后在 $h\text{-list}$ 中查询 m^* 和 R_2^* , 如果存在则获得 H^* ; 否则, 失败. 在 $S\text{-list}$ 中查询 (m^*, r, m_w) , 如果存在且 $\text{flag} = 1$, 则攻击失败; 否则验证 $c^*P = R_1^* + H^*Y$ 是否成立.

对消息 m^* 而言, 提出的签名 (R_1^*, m^*, c^*, H^*) 是标准的 Schnorr 签名. 根据分叉引理^[10], A 利用重放技术, 对相同的消息 m^* 可以构造另外一个有效的签名 $(R_1^*, m^*, c^{**}, H^{**})$, 这里 $H^{**} = H'(R_1^* \| y_v, R_2^* \| m^*), R_3^{**} = E_{R_2^*}(c^{**} \| m_w \| y_0 \| r \| y_d \| m^*)$. 这样, 挑战者获得关于消息 m^* 的另外一个有效签名 (R_1^*, R_3^{**}) , 同样挑战者通过解密和查表获得所有参数、并可以验证签名的正确性. 所以, 挑战者得到: $c^*P = R_1^* + H^*Y, c^{**}P = R_1^* + H^{**}Y$.

因为 $Y = x_s P$, 所以 $x_s = (c^* - c^{**}) / (H^* - H^{**})$, 即椭圆曲线中的离散对数可解.

下面, 我们分析解离散对数的时间和概率. 如果能对同一个消息构造两个有效的签名, 那么挑战者能以 $\varepsilon' \geq \varepsilon$ 的概率解决离散对数问题. 因为我们的方案采用了标准的 Schnorr 签名, 因此根据引理 15^[10], 在不知道签名密钥的前提下, 我们方案中签名者或者模拟者产生签名的概率与 Schnorr 签名是一样的, 而解离散对数的时间是 A 的运算时间加上挑战者回答签名问询、Hash 问询、验证问询、转换问询、验证签名和计算签名密钥 x_s 的时间之和. 设 T_M 表示椭圆曲线上的点乘计算时间、 q_s 表示签名问询的次数, q_h 表示 Hash 问询的次数, 则挑战者回答签名问询、Hash 问询、验证签名和计算签名密钥 x_s 的时间之和为 $3(q_s + q_h + 2)T_M$.

根据引理 8, 12 和定理 13^[10], 攻击者 A 在时间 $t' \leq 120686q_h t / \varepsilon$ 内能对同一个消息构造两个有效的签名的概率为 $\varepsilon'' \geq 1/9$, 所以, 我们的方案中解决离散对数的概率是 $\varepsilon' \geq 1/9$, 时间为 $t' \leq 120686q_h t / \varepsilon + 3(q_s + q_h + 2)T_M$, 其中 $\varepsilon \geq 10(q_h + q_s)(q_s + 1)/2^l$.

定理 2 在随机预言机模型下, 提出的方案能够在 $(t, q_{\text{sign}}, q_{\text{verify}})$ 适应性选择消息攻击下实现消息的保密性.

证明 挑战者产生系统参数、代理签名密钥和公钥对 $(x_s, Y = x_s P)$, m 个指定接收人的私钥和公钥对 (x_{vj}, y_{vj}) , 并把 $y_{vj} (j = 1, 2, \dots, m)$ 发送给 A . 然后, 挑战者建立 Sign 问询表 $S\text{-list}$, 初始时空.

问询: 攻击者 A 可以能动的发动下列问询:

1) 签名问询: A 发送 (y_{vj}, m) 给挑战者, 并问询签名 oracle. 挑战者查找 $S\text{-list}$, 如果在 $S\text{-list}$

中存在 (y_{vj}, m) , 则返回 (R_1, R_3) 给 A ; 否则, 挑战者随机选择 $a \in Z_q^*, R_1 = aP, R_2 = ay_{vj}, c = a + x_s H(R_1 \| y_{vj} \| R_2 \| m), R_3 = E_{R_2}(c \| m_w \| y_0 \| r \| y_d \| m)$; 然后把 $(y_{vj}, m, r, m_w, c, H, R_1, R_3, R_2)$ 记入 S -list, 并返回 (R_1, R_3) 给 A .

2) 验证问询: A 发送 (y_{vj}, R_1, R_3) 给挑战者, 并问询验证 oracle. 挑战者查找 S -list, 如果在 S -list 中存在 (y_{vj}, R_1, R_3) , 则返回 1 给 A ; 否则返回 0.

挑战: 攻击者 A 选择指定接收人 V_1, V_2 和两个相同长度的消息 m_0 和 m_1 , 把 m_0, m_1, y_{v1} 和 y_{v2} 发送给挑战者. 挑战者公平的扔硬币 $\lambda \leftarrow \{0, 1\}$, 并针对 m_λ 和 $y_{v\lambda}$ 产生一个签名 $\sigma^\lambda = (R_1^\lambda, R_3^\lambda)$, 返回给 A .

攻击者 A 能发动新的问询, 但是不能针对 m_0 和 m_1 发动签名问询.

猜测: A 输出一位 λ' , 如果攻击者能猜出正确的位, 表明他能知道签名的消息.

因为 A 不知道 $x_{v\lambda}$, 因此不能从 R_3^λ 中恢复出所需要的参数, 因此他无法决定 m_0 和 m_1 中哪一个正确的. 在签名 oracle 问询中, 因为 a 是 Z_q^* 中随机选取的, 所以 $\Pr[\sigma^\lambda = \sigma'] = \Pr[a = a'] = 1/q$ 这样, 攻击者 A 的优势是可忽略的.

定理 3 在随机谰言机模型下, 提出的方案能够在 $(t, q_{\text{sign}}, q_{\text{verify}})$ 适应性选择消息和选择指定接收人的攻击下实现不可验证性.

证明 证明过程与定理 2 的证明类似, 略.

定理 4 攻击者不可能把提出的方案转换成公开可验证签名.

证明 如果攻击者能把提出的方案转换成公开可验证签名, 那么他与定理 2 矛盾. 因为可转成公开可验证签名, 则攻击者就知道签名消息 m .

3.2 提出协议的安全性证明

本节首先给出安全协议的定义, 继而证明所提方案的安全性依赖于计算 Diffie-Hellman (CDH) 问题假设.

我们采用 CK 安全模型来证明新提出协议的安全性. 关于该模型的详细情况参见文献 [11,12], 这里我们只对其作简要介绍.

在 CK 安全模型中, 协议执行过程被刻画成谰言机 $\Pi_{i,j}^s$ (指协议的参与者 i 与 j 之间的第 s 次会话), 敌手的能力被刻画成对谰言机 $\Pi_{i,j}^s$ 的如下 4 种问询, 即 $\text{Send}(\Pi_{i,j}^s, M)$ 问询 (以下简称 Send 问询)、 $\text{Reveal}(\Pi_{i,j}^s)$ 问询 (以下简称 Reveal 问询)、 $\text{Corrupt}(i)$ 问询 (以下简称 Corrupt 问询) 和 $\text{Test}(\Pi_{i,j}^s)$ 问询 (以下简称 Test 问询). 在回答该问询前, 新鲜的谰言机 $\Pi_{i,j}^s$ 首先随机选取 $b \in \{0, 1\}$, 若 $b = 0$, 则它返回最终的会话密钥 sk , 否则, 返回一个随机值.

协议的安全性被刻画成挑战者 A 与敌手 E 间的两阶段游戏. 在第一阶段, 允许 E 对随机谰言机 $\Pi_{i,j}^s$ 进行多项式次的上述 4 种问询. 在第二阶段, 敌手 E 可选择某个新鲜谰言机 $\Pi_{i,j}^s$ 进行 Test 问询, 但不允许对谰言机 $\Pi_{i,j}^s$ 进行 Reveal 问询, 也不允许 E 对 j 进行 Corrupt 问询. 最后敌手 E 输出对会话密钥的猜测 b' , 若 $b' = b$, 则敌手 E 攻击成功. 敌手 E 攻击成功的优势定义为 $\text{Adv}(k) = |\Pr[\text{Succ}^E(k)] - 1/2|$.

定义 2 假设一个协议满足下列 4 个条件, 那么它是安全的认证密钥交换协议:

1) 在一个良性的敌手面前, 两个随机谰言机 $\Pi_{i,j}^s$ 和 $\Pi_{j,i}^t$ 接受持有相同的会话密钥 sk , 这个密钥在密钥空间 $\{0, 1\}^l$ 上统一随机分配 (l 是协议的安全参数).

2) 如果两个未腐化的谰言机 $\Pi_{i,j}^s$ 和 $\Pi_{j,i}^t$ 有匹配的会话, 那么这两个谰言机接收并持有相同的会

话密钥 sk .

3) 敌手没有做任何 Reveal 查询.

4) 令 $\Pr[\text{Succ}^E(k)]$ 是敌手 E 在测试查询中赢得游戏的概率, 则有 $\text{Adv}(k) = |\Pr[\text{Succ}^E(k)] - 1/2|$, $\text{Adv}(k)$ 是可忽略的.

定理 5 假设 CDH 问题在计算上是困难的且 Hash 函数 H, h 是独立的随机谰言, 那么提出的认证协议是可证安全的.

证明 假设存在一个概率多项式时间 (PPT) 敌手 E , 它有一个不可忽略的优势 $\varepsilon(k)$ 攻破我们的协议, 可以利用 E 作为子程序来构造一个 CDH 困难问题的解决者 A .

设 (P, x_iP, y_jP) 是一个给敌手 A 解决 CDH 困难问题的实例, A 的目标是计算出 x_iy_jP .

A 从身份集合 $\{\text{ID}_1, \text{ID}_2, \dots, \text{ID}_n\}$ 中随机选择两个用户 $\text{ID}_i, \text{ID}_j \in \{\text{ID}_1, \text{ID}_2, \dots, \text{ID}_n\}$, 令 E 与每个用户交互的次数是 $t(t = 1, 2, \dots, m)$. A 试图猜测被 E 选择的 Test 查询. 如果 E 选择第 t 会话作为 Test 会话, 那么 A 将在此游戏中获胜. A 完成 E 的实验并随机选择所有用户的临时私钥 (除 $\Pi_{i,j}^s$ 和 $\Pi_{j,i}^t$ 外). A 记录所用用户的公钥并随机回答 H 和 h 查询. 对谰言机 $\Pi_{i,j}^s$ 和 $\Pi_{j,i}^t$ 之外的对所有的 Send、Reveal 和 Corrupt 查询, A 作为一个诚实的游戏者来回答上述查询. 如果 E 对用户 ID_i 或用户 ID_j 作 Corrupt 查询, 则 A 放弃. 当 E 对 $\Pi_{i,j}^s$ 或 $\Pi_{j,i}^t$ 作 Send 查询时 (不是计算值 $sk = h(x_iy_jP)$), 因为 A 不知道 x_iy_jP 的值, 所以 A 随机选择一个 sk' 来代表 sk . 最后, 如果 E 对 $\Pi_{i,j}^s$ 或 $\Pi_{j,i}^t$ 发送 Reveal 查询, 那么 A 用它猜想的价值 sk' 作应答则不计算 $h(x_iy_jP)$.

条件 1 和 2: 从定义 2 可知, 对谰言机 $\Pi_{i,j}^s$ 和 $\Pi_{j,i}^t$ 存在一个良性的敌手 E , 当未腐化的谰言机 $\Pi_{i,j}^s$ 和 $\Pi_{j,i}^t$ 有匹配的会话时, 这两个谰言机接受并持有相同的会话密钥 $SK = h(x_iy_jP)$.

条件 3: 如果 $\text{Adv}(k)$ 是可以忽略的, 那么我们可以构造一个新的敌手以不可忽略的概率成功解决 CDH 困难问题.

假定敌手 E 随机选择一些新鲜谰言 $\Pi_{i,j}^s$ 作 Test 查询, 这种谰言被持有的密钥形式为 $h(x_iy_jP)$, 令 F 是这样的事件: h 是以前被 E 带着输入 x_iy_jP 作过 Test 查询, 但并没有对谰言机 $\Pi_{i,j}^s$ 和 $\Pi_{j,i}^t$ 作过查询. 如果 $\text{Adv}(k)$ 是可以忽略的, $\eta(k)$ 表示一些不可忽略的函数, 那么 $\Pr[\text{Succ}^E(k)] = 1/2 + \eta(k)$,

$$\Pr[\text{Succ}^E(k)] = \Pr[\text{Succ}^E(k)|F]\Pr[F] + \Pr[\text{Succ}^E(k)|\bar{F}]\Pr[\bar{F}].$$

因为 h 是随机谰言的且 $\Pi_{i,j}^s$ 是新鲜的谰言, 所以有 $\Pr[\text{Succ}^E(k)|\bar{F}] = 1/2$.

可得 $1/2 + \eta(k) \leq \Pr[\text{Succ}^E(k)|F]\Pr[F] + 1/2$.

即 $\Pr[\text{Succ}^E(k)|F] \geq \eta(k)$, $\Pr[F] \geq \eta(k)$, 它们都是可忽略的.

令 $\Pr[F] = \varepsilon(k)$ ($\varepsilon(k)$ 是不可忽略的函数). E 能够用来构造另一个不同的敌手 A , A 能以不可忽略的概率成功解决 CDH 困难问题.

令 P 是循环群 G 的一个生成元, 用户和公共云 VC 各自随机选择一个元素 $x_i \in Z_q^*, y_j \in Z_q^*$, A 必须猜想 x_iy_jP , 它是一个 CDH 困难问题的实例.

在作临时私钥查询 (ephemeral key reveal($\Pi_{i,j}^s$)) 时, 如果它是在用户 ID_i 和 ID_j 之间的第 t 次会话, 则 A 输出“失败”. 对所有的关于 ID_j 的会话, Ephemeral Key Reveal($\Pi_{i,j}^s$) 不被处理 (事件 I). 在其他情况下, A 返回它在回答 Send 查询时选择的对应的各用户的临时私钥. 在作测试查询 (test query) 时, 如果它不是用户 ID_i 和 ID_j 之间的第 t 次会话, 则 A 输出“失败”(事件 II). 对所有的其他情况查询, A 返回一个随机值.

在 A 模拟期间, 如果事件 I 和事件 II 没有发生, 则它不会中止. E 会保持预期的测试会话 (test session) 是新鲜的. 因此, 如果 E 选择用户 ID_i 和 ID_j 之间的第 t 次会话作为“测试会话”, 那么事件 I 和

表 1 认证传输量比较

Table 1 Authentication communication cost comparison

	The proposed scheme	Juang et al. scheme ^[8]
Round	1	2
Communication cost	1188 bit	3056 bit

事件 II 不会发生. 如果事件 I 和事件 II 没有发生, 则不会输出“失败”.

如果 A 不能处理 E 发起一个临时私钥显示查询 (ephemeral key reveal query), 那么事件 I 发生.

$$\Pr[I] = m(n-1)/(mn(n-1)), \Pr[\bar{I}] = 1 - \Pr[I] = (n-1)/n.$$

如果 E 没有选择被 A 预期的会话作为“测试会话”, 那么事件 II 发生. $\Pr[\bar{II}] = 1/(mn(n-1))$, $\Pr[\text{Succ}^A(k)|F] = \Pr[\bar{I} \wedge \bar{II}] = 1/mn^2$, $\Pr[\text{Succ}^A(k)] \geq \Pr[\text{Succ}^A(k)|F]\Pr[F] = \varepsilon(k)/mn^2$.

由于我们假设的 $\varepsilon(k)$ 是不可忽略的, A 能带着 $\Pr[\text{Succ}^A(k)] = \varepsilon(k)/mn^2$ 的概率解决 CDH 困难问题也是不可忽略的, 这与 CDH 假设是相矛盾的. 因此, 不存在一个概率多项式时间的敌手能以不可忽略的优势来攻击我们的认证协议. 所以, 我们的认证协议在随机谰言模型下是可证安全的.

4 效率比较

在本节, 给出提出的方案与 Juang 等^[8]方案的计算量、认证轮数和传输效率比较.

4.1 传输效率比较

我们定义消息的发送与反馈作为一轮, 因此提出的方案只需一轮认证, 而 Juang 等的方案需要二轮论证. 在提出的方案中, 用户的请求消息需要传输 (R_1, R_3) 共 $164+768=932$ bit, 因为 $R_3 = E_{R_2}(C\|m_w\|y_u\|t\|r\|R)$ 中 (R, r, y_u) 是椭圆曲线中的点, 其长度各为 164 bit, 时间 t 为 6 bit, c 是签名为 160 bit, 代理证书 m_w 为 100 bit, 因此需要用 3 次 256 bit 对称加密算法加密来计算 R_3 , 其输出长度为 768 bit. 公有云反馈消息需要传输 (R_4, V) 共 $164+256=420$ bit. 所以, 总共传输 1188 bit.

Juang 等^[8]方案中, 第一次需要传输 $(N_1, Q, r_H, R, S_U, y_U, y_{ur}, ID_H)$, 共需要 $256+64+164+164+160+164+164+64=1100$ bit; 第二次需要传输 $[N_1\|Q\|y_U\|n_2]_{KVH}$, 因为 $N_1\|Q\|y_U\|n_2$ 的长度为 $256+64+164+64=548$ bit, 因此需要用 3 次 256 bit 对称加密算法加密, 其输出长度为 768 bit; 第三次需要传输 $[h(N_1\|Q\|y_{U_{new}}\|n_2), C_1, l]_{KVH}$, 因为被加密参数的长度是 $256+256+256=768$ bit, 因此需要用 3 次 256 bit 对称加密算法加密, 其输出长度为 768 bit; 第四次需要传输 $([h(N_1\|Q\|y_{U_{new}}\|n_2), n_2])$ 需要 $256+164=420$ bit. 所以, 总共传输 3056 bit.

从表 1 可知, 提出的方案的认证过程的轮数和传输量优于 Juang 等^[8]方案.

4.2 计算量比较

设 T_m 是椭圆曲线上的点乘计算时间, T_e 是对称加密/解密操作时间, T_h 是 Hash 操作时间, T_i 是计算乘法逆元的时间, 其余的计算量与这些相比, 可以忽略.

在提出的方案中, 在用户注册阶段, 需要 5 次椭圆曲线上的点乘计算, 4 次 Hash 操作; 在认证阶段, 需要 8 次椭圆曲线上的点乘计算, 5 次 Hash 操作, 3×3 次对称加密/解密操作. 总共需要 $13T_m + 9T_h + 9T_e$.

表 2 计算量比较
Table 2 The computation cost comparison

	The proposed scheme	Juang et al. scheme ^[8]
Registration	$5T_m + 4T_h$	$6T_m + 2T_h$
Authentication	$8T_m + 5T_h + 9T_\varepsilon$	$7T_m + 7T_h + (nT_h) + (n+6)T_\varepsilon + 2T_i$
Total	$13T_m + 9T_h + 9T_\varepsilon$	$13T_m + (n+9)T_h + (n+6)T_\varepsilon + 2T_i$

Juang 等^[8] 方案中, 需要 6 次椭圆曲线上的点乘计算, 2 次 Hash 操作; 在认证阶段, 需要 7 次椭圆曲线上的点乘计算, 7 次 Hash 操作, 2 次计算乘法逆元, $n+2 \times 3$ 次对称加密/解密操作, 同时需要预计算 n 次 Hash 值. 总共需要 $13T_m + 9T_h + (nT_h) + (n+6)T_\varepsilon + 2T_i$.

从表 2 可知, 提出的方案略优于 Juang 等的方案.

5 结束语

针对 Juang 等提出的第一个应用于云计算的基于代理签名的认证协议中存在的若干不足, 提出了新的可证安全的可转换代理签密方案, 该方案与已有方案相比, 能确保签名人身份的保密性, 并给出了安全性证明, 这是第一个可证明安全的可转换代理签密方案. 基于该方案, 提出了一种新的可应用于云计算的一轮认证协议, 它具有保护用户隐私性、抗抵赖性、不需要更新用户的公钥以及私有云与访问的公有云不需要预先共享秘密等优点. 而且, 新协议效率优于 Juang 等的方案, 并在随机谕言机模型下证明了安全性.

参考文献

- 1 Wu J Y, Fu J Q, Ping L D, et al. Study on the P2P cloud storage system. Act Electr Sinic, 2011, 39: 1100–1107 [吴吉义, 傅建庆, 平玲娣, 等. 一种对等结构的云存储系统研究. 电子学报. 2011, 39: 1100–1107]
- 2 Weiss A. Computing in the clouds. Network, 2007, 11: 16–25
- 3 Identity verification and data protection in cloud computing: the best implementation scheme for trusted environment. <http://www.rsa.com> [云计算中的身份识别和数据保护: 建立可信任环境的最佳实施方案. <http://www.rsa.com>]
- 4 Rahnema M. Overview of the GSM system and protocol architecture. IEEE Commun Magaz, 1993, 31: 92–100
- 5 Lee W B, Yeh C K. A new delegation-based authentication protocol for use in portable communication systems. IEEE Trans Wirel Commun, 2005, 4: 57–64
- 6 Lee T F, Hwang T, Chang S H, et al. Enhanced delegation-based authentication protocol for PCSs. IEEE Trans Wirel Commun, 2009, 8: 2166–2171
- 7 Wang R C, Juang W S, Lei C L. A privacy and delegation-enhanced user authentication protocol for Portable Communication Systems. Int J Ad Hoc Ubiquit Comput, 2010, 6: 183–190
- 8 Juang W S, Chiu J Y, Chang H Y. A secure and efficient delegation-based authentication scheme in public clouds. In: The 1st Cross-Straits Conference On Information Security, Hangzhou, 2011. 96–102
- 9 Nicolosi A, Krohn M, Dodis Y, et al. Proactive two-party signatures for user authentication. In: Proceedings of the Network and Distributed System Security Symposium, San Diego. 2003
- 10 Pointcheval D, Stern J. Security arguments for digital signatures and blind signatures. J Crypt, 2000, 13: 361–396
- 11 Chen L, Chen Z, Smart N P. Identity-based key agreement protocols from pairings. Int J Inf Sec, 2007, 6: 213–241
- 12 Canetti R, Krawczyk H. Analysis of key-exchange protocols and their use for building secure channels. In: Advances in Cryptology EUROCRYPT'01. London: Springer-Verlag, 2001. 453–474

Provably secure authentication protocol based on convertible proxy signcryption in cloud computing

XIE Qi^{1,2*}, WU JiYi^{2,4}, WANG GuiLin³, LIU WenHao¹, CHEN DeRen⁴ & YU XiuYuan¹

1 School of Information Science and Engineering & Institute of Cryptology and Its Applications, Hangzhou Normal University, Hangzhou 310036, China;

2 Key Lab of E-Business and Information Security, Hangzhou Normal University, Hangzhou 310036, China;

3 University of Wollongong, School of Computer Science & Software Engineering, Wollongong NSW 2522, Australia;

4 College of Computer Science, Zhejiang University, Hangzhou 310027, China

*E-mail: qixie68@yahoo.com.cn

Abstract Mutual authentication between the user and the public cloud is essential requirement for the user to access the public cloud in cloud computing. In 2011, Juang et al. proposed a first authentication scheme based on proxy signature. The advantage of the scheme is that the user only needs to register on his home service cloud (HSC), and can pass through the authentication of the public cloud with the help of his HSC. However, their scheme has three weaknesses: 1) the user's HSC needs to update the user's public key in each session to protect the user's privacy; 2) HSC may suffer from network jam when many users in the same HSC need to register on different public clouds simultaneously; and 3) a secret key should be shared between HSC and visiting cloud. To overcome these weaknesses, a provably secure convertible proxy signcryption for privacy preserving is proposed. Based on this scheme, a novel one-round authentication protocol is proposed, which the user only needs to register on his HSC, and can pass through the authentication of the visiting cloud without the help of his HSC. On the other hand, the proposed protocol can provide some nice properties, such as user privacy protection, non-repudiation, without updating the user's public key, and secret key does not have to be shared between HSC and visiting cloud. In addition, the proposed scheme is provably secure in the random oracle model, and is more efficient than Juang et al.'s scheme.

Keywords cloud computing, authentication protocol, convertibility, proxy signature, signcryption, provably secure

XIE Qi was born in 1968. He received his Ph.D. degree in applied mathematics from Zhejiang University, Hangzhou in 2005. Currently, he is a Professor at School of Information Science and Engineering in Hangzhou Normal University. His research interests are digital signatures, authentication and key exchange protocols, etc. Dr. Xie is a senior member of Chinese Association for Cryptologic Research.

WU JiYi was born in 1980. He received Ph.D. degree in computer science and technology from Zhejiang University, Hangzhou in 2011. Currently, he is an Associate Professor at Hangzhou Normal University. His research interests include services computing, trust and reputation. Dr. Wu is a senior member of China Computer Federation (CCF).

WANG GuiLin is currently a Senior Lecturer in the School of Computer Science and Software Engineering, University of Wollongong, Australia. He received his Ph.D. degree in computer science from Institute of Software, Chinese Academy of Sciences, in 2001. His main research interests include the analysis, design, and applications of digital signatures and security protocols.

LIU WenHao was born in 1974. He received the Ph.D. degree in information security from University of Electronic Science and Technology of China, Chengdu in 2010. Currently, he is a Senior Researcher at Information Security and Cryptography. His research interests include cloud computing security, mobile and wireless security, network security, and cryptography. Dr. Liu is a member of ACM, CCF and IACR professional members.