普适计算环境中基于身份的密钥管理方案

孙 凌 田 源
(河南商业高等专科学校 郑州 450044)

摘 要 针对普适环境中的密钥管理问题,利用椭圆曲线加法群设计了一种新的基于身份的密钥管理方案。新方案利用门限秘密共享机制构建了分布式密钥生成中心,并设计了私钥更新、主密钥分量更新和会话密钥协商策略。与现有基于身份密钥管理方案相比,新方案具有更强的安全性和更高的执行效率。

关键词 普适计算,密钥管理,基于身份的密码体制,门限机制
中图法分类号 TP393 **文献标识码** A

Identity-based Key Management Scheme in Pervasive Computing Environments

SUN Ling TIAN Yuan
(Henan Business College, Zhengzhou 450044, China)

Abstract Considering the key management in pervasive computing environments, this paper proposed a novel identity-based key management scheme from additive elliptic curve group. It employs the secret-sharing technique to construct distributed private key generators, and designs the methods of updating private key, updating host-key shares and negotiating session key. It can achieve higher security requirements and has higher efficiency compared with available identity-based schemes.

Keywords Pervasive computing, Key management, Identity-based cryptography, Threshold mechanism

1 引言

普适计算[1]是继主机计算和桌面计算之后发展起来的新的计算模式。在普适计算环境中,用户可以随时随地获得服务,并且普适环境能够通过感知用户信息自动地向用户提供智能化服务。为了实现上述功能,普适环境中分布着大量的嵌入式设备(如传感器)。这些设备计算和存储能力有限,通常以无线形式组成网络。这些特点使构成普适环境的节点极易受到各种攻击。因此,设计适合普适环境的密钥管理方案,确保普适环境通信安全已成为亟待解决的问题之一。

普适环境中的设备通常具有有限的计算和存储能力,这要求密钥管理方案在保证安全性的同时具有较高的执行效率。文献[2,3]分别提出了普适环境中的密钥预分配方案,在预分配方案中,每个节点预先存储一定数量的密钥信息,网络运行中,每个节点与相邻节点通信,通过简单的交互和计算就可以建立通信密钥。预分配方案通信和计算开销较小,但是安全性较差,当单个节点被攻破后,会造成密钥信息的泄露,从而影响其它节点的安全。基于公钥体制的密钥管理方案[4]具有较强的安全性,但是传统的公钥体制需要复杂的证书管理过程,计算和通信开销较大,不适合资源受限的普适计算环境。文献[5]利用基于身份公钥密码体制[6]提出了一种普适环境下的认证和密钥管理方案,方案中用户的公钥由身份信息计算得到,不需要证书对身份和公钥进行绑定,避免了传统公钥体制中复杂的证书管理过程,提高了执行效率,并具有较强的安全性。但是该方案需要设置一个集中的密钥生成中心,当多个节点请求密钥更新时,由于无线链路带宽有限会造成单点失效问题。文献[7]结合基于身份公钥体制和(n,k)门限原理提出了一种分布式密钥管理方案,方案介绍了分布式密钥生成中心的构建方法,并设计了私钥更新和会话密钥协商策略。该方案避免了由于无线链路带宽有限造成的单点失效问题,并且增强了系统容侵能力。但是,在私钥更新的过程中,私钥份额以明文形式传送,恶意节点可以获得私钥份额并构造出完整私钥。

现有基于身份的密钥管理方案虽然在效率上优于基于证书的密钥管理方案,但是同普适环境对执行效率的要求相比,仍存在计算开销大的问题。本文利用椭圆曲线加法群设计了一种新的基于身份的分布式密钥管理方案,并详细设计了私钥更新、主密钥分量更新和会话密钥协商策略。与现有基于身份密钥管理方案相比,该方案具有更强的安全性和更高的执行效率。

2 普适计算环境中基于身份的密钥管理方案设计

在基于身份的密码体制中,需要有一个可信的密钥生成中心 PKG(Private Key Generator)利用系统主密钥为用户生成私钥和更新私钥。在普适计算环境下,设置这样一个集中的 PKG 存在以下问题。首先,普适环境中的节点之间以无线

到稿时间:2012-09-20 返修时间:2013-01-02 本文受河南省高等学校青年骨干教师资助计划基金项目(201104)资助。
孙 凌(1976-),女,硕士,副教授,主要研究方向为无线网络安全技术,E-mail:sunling19@126.com;田 源(1980-),女,硕士,讲师,主要研究方向为网络安全技术、数据处理。

连接，链路带宽有限，当多个节点请求 PKG 更新私钥时，会出现网络拥塞而造成单点失效和拒绝服务。另外普适环境中节点具有脆弱性，容易被恶意实体攻破，一旦 PKG 被攻破，主密钥就会被恶意实体获得，整个网络就会暴露。针对上述问题，借鉴现有密钥管理方案中的方法，利用门限秘密共享机制将密钥生成中心的功能分散到多个节点上，可以避免单点失效并增强系统的容侵能力。具体方法是设置一个集中式的密钥生成中心 PKG 和由多个节点构成的分布式密钥生成中心 D-PKGs(Distributed Private Key Generators)。PKG 持有完整的主密钥，负责完成系统初始化和节点初始私钥生成，初始化完成后，PKG 销毁主密钥。每个 D-PKGs 节点持有一个主密钥分量，门限个 D-PKGs 节点可以恢复主密钥。在网络运行中，D-PKGs 节点负责为网络节点提供在线私钥更新服务。为了防止恶意实体通过累积门限个主密钥分量来重构主密钥，本文设计了主密钥分量更新策略，由 PKG 定期对 D-PKGs 节点的主密钥分量进行更新。

本文提出的密钥管理方案由 4 个关键部分组成：系统建立、节点私钥更新、主密钥分量更新和会话密钥协商。

2.1 系统建立

设网络包含 N 个节点，节点集合为 $\Psi(|\Psi|=N)$，ID 为整个网络的节点身份标识集合。对于任意节点 A，其拥有全网唯一的身份标识 $ID_A\in ID$。在系统初始化时，PKG 做如下设置：

(1)选择椭圆曲线 $E(F_p)$ 上的 q 阶循环加法群 G_1，G_1 的生成元为 P。随机选择 $s\in Z_q^*$ 作为系统主密钥，系统公钥为 $P_{pub}=sP\in G_1$。定义以下安全哈希函数：$H_1:\{0,1\}^*\rightarrow Z_q^*$，$H_2:G_1\rightarrow Z_q^*$，$H_3:G_1\rightarrow\{0,1\}^*$。

(2)选择 n 个节点构成 D-PKGs 节点集合 $\Omega(|\Omega|=n,\Omega\subset\Psi)$，建立初始主密钥共享多项式：

$$f(x)=(s+\sum_{i=1}^{t-1}a_ix^i)\bmod q\ (a_i\in Z_q^*)$$

对于任意 D-PKGs 节点 $V\in\Omega$，PKG 为其计算初始主密钥分量 s_V 和主密钥分量验证参数 W_V：

$$s_V=f(H_1(ID_V))\bmod q$$

$$W_V=s_VP$$

所有 D-PKGs 节点的主密钥分量验证参数数组为 $W=\{s_VP\in G_1\,|V\in\Omega\}$。PKG 将 W 向全网公开，任意节点可利用 W_V 来验证交互方是否为合法 D-PKGs 节点。PKG 将 s_V 安全地发送给节点 V，t 个 D-PKGs 节点联合即可重构 $f(x)$，通过计算 $f(0)=s$ 恢复主密钥。

(3)$phase$ 是与节点更新次数相关的非零字符串，用来生成节点在不同时间段内的公钥。初始时刻，$phase=phase_0$，并向全网公开 $phase_0$。PKG 通过以下步骤为节点 A 生成初始私钥 K_A^{-1}：

PKG 随机选择 $r_A\in Z_q^*$，计算 $R_A=r_AP$，$D_A=r_A+sH_1(ID_A\|phase)$，初始私钥为 $K_A^{-1}=(D_A,R_A)$，通过安全信道将 K_A^{-1} 发送给 A。

A 通过检验等式 $R_A+H_1(ID_A\|phase)P_{pub}=D_AP$ 是否成立来验证私钥的有效性，A 安全地保存部分私钥 D_A。A 的初始公钥为 $K_A=H_1(ID_A\|phase)$。

在网络运行过程中，节点每次更新私钥前取 $phase=phase+1$。

(4)在网络运行中，节点更新一次私钥至少需要向 t 个 D-PKGs 节点广播请求信息。由于普适环境中的节点以无线形式连接，当大量节点请求更新私钥时，会造成网络拥塞。为避免这种现象，采用分批更新机制。PKG 设置节点私钥更新周期为 w，将所有节点分成 m 次更新，PKG 设置一个首次更新时间集合 $T_u=\{iw/m\,|\,i=1,2,\cdots,m\}$。定义映射：$g:ID\rightarrow T_u$，要求 g 将节点身份标识均匀地映射到 T_u。对于成员节点 A，在 $t_A=g(ID_A)$ 时进行第一次私钥更新，以后每隔 w 时间申请一次私钥更新。PKG 向全网公开 w、T_u 和 g。

初始化完成后，PKG 选择 $r_{PKG}\in Z_q^*$，计算 $R_{PKG}=r_{PKG}P$，$s_{PKG}=r_{PKG}+s$，PKG 删除 s、r_{PKG} 以及主密钥共享多项式 $f(x)$，将 s_{PKG} 作为证明自己身份的签名密钥，然后系统公开参数 $\{P,P_{pub},G_1,H_1,H_2,H_3,W,phase_0,w,T_u,g,R_{PKG}\}$。

2.2 节点私钥更新

假设在系统时间 τ 时刻，节点 A 需要将当前私密 K_A^{-1} 更新为 $K_{A\_new}^{-1}=(D_{A\_new},R_{A\_new})$，更新过程如下：

Step1 节点 A 利用公开参数计算得到下一个周期内的公钥 K_{A_new}(t_A 为节点 A 的首次更新时间，k 为已完成更新次数)：

$$t_A=g(ID_A)$$

$$k=(\tau-t_A)/w$$

$$K_{A\_new}=H_1(ID_A\|phase_0+k+1)$$

Step2 节点 A 随机选择 $y\in Z_q^*$，计算 $Y=yP$，$h=H_1(Y\|ID_A\|R_A)$，计算签名 $z=y+D_Ah$，向网络中至少 t 个 D-PKGs 节点广播私钥更新请求消息：

$$REQ_{update}=\{K_{A\_new},R_A,Y,z,ID_A\}$$

Step3 收到请求消息的 D-PKGs 节点 V 验证节点 A 是否为已撤销节点。若 A 已被撤销，则结束算法；否则，计算 $h=H_1(Y\|ID_A\|R_A)$，计算 A 的前公钥 K_A，验证签名是否成立：

$$zP=Y+h(R_A+K_AP_{pub})$$

若不成立，则算法结束；否则继续运行。

Step4 节点 V 随机选择 $x_V\in Z_q^*$，计算节点 A 的私钥信息 $M_V=x_V+s_VK_{A\_new}$，$X_V=x_VP$，对部分私钥 M_V 加密得 $U=M_V\oplus H_2(x_VY)$，向节点 A 发送更新应答消息：

$$RES_{update}=\{U,X_V\}$$

Step6 节点 A 收到节点 V 的更新应答消息后，利用已掌握的 y，通过下式解密得到 V 签发的部分私钥：$M_V=U\oplus H_2(yX_V)$。

Step7 节点 A 取出节点 V 的验证参数 W_V，验证 $M_VP=X_V+K_{A\_new}W_V$ 是否成立，若成立，则保存 M_V 和 X_V；否则认为节点 V 不合法，丢弃该节点签发的私钥信息。

Step8 节点 X 在收到 t 个通过验证的私钥信息后，利用 Lagrange 内插法计算新私钥：$K_{A\_new}^{-1}$。

$$\begin{aligned}D_{A\_new}&=\sum_{V\in\Lambda}\lambda_V(0)M_V\\&=\sum_{V\in\Lambda}\lambda_V(0)(x_V+s_VK_{A\_new})\\&=\sum_{V\in\Lambda}\lambda_V(0)x_V+\sum_{V\in\Lambda}\lambda_V(0)s_VK_{A\_new}\\&=r_A'+sK_{A\_new}R_{A\_new}=\sum_{V\in\Lambda}\lambda_V(0)X_V=\sum_{V\in\Lambda}\lambda_V(0)x_VP\\&=r_A'P\end{aligned}$$

式中，$\lambda_V(0)$ 为插值系数，Λ 为通过验证的 t 个 D-PKGs 节点集合。

2.3 主密钥分量更新

为了防止门限个主密钥分量被攻击者掌握而造成主密钥

泄露，系统需对各 D-PKGs 节点的主密钥分量进行定期更新。主密钥分量更新要保证原主密钥不变。主密钥分量更新基于以下原理。

定理 1 隐藏秘密为 s 的 (n,t) 门限多项式 $f(x)$ 与隐藏秘密为 0 的 (n,k) $(0<k\leqslant t\leqslant n)$ 门限多项式 $\xi(x)$ 之和依然是隐藏秘密为 s 的 (n,t) 的门限多项。

推论 1 隐藏秘密为 s 的秘密份额 $s_i(i=1,2,\cdots,n)$ 与隐藏秘密为 0 的秘密份额 $x_i(i=1,2,\cdots,n)$ 之和 s_i+x_i 依然是隐藏秘密为 s 的秘密份额。

定义 1 称隐藏秘密为 0 的 (n,k) 门限多项式 $\xi(x)$ 为累加多项式，称对应的密钥分量为累加密钥分量。

由上述结论可以看出，只要在初始主密钥分量的基础上加上一个累加多项式产生的累加主密钥分量就可以实现主密钥分量更新，同时保证原主密钥不变。由于构造累加多项式不需要原主密钥参与，因此 PKG 虽然删除了主密钥 s，但仍然可以完成主密钥分量更新。在更新过程中，PKG 利用 s_{PKG} 对累加主密钥分量进行签名来证明其正确性。

设当前主密钥共享多项式为 $f(x)$。PKG 随机选取次数为 $k-1$ 的累加多项式 $(0<k\leqslant t\leqslant n)$：

$$\xi(x)=\left(\sum_{i=1}^{k-1}b_i x^i\right) \bmod q \ (b_i\in Z_q^*)$$

对于任意 D-PKGs 节点 $V\in\Omega$，利用上式为其计算累加主密钥分量 s_V' 和累加主密钥分量验证参数 W_V'：

$$s_V'=\xi(H_1(ID_V)) \bmod q$$

$$W_V'=s_V'P$$

累加主密钥分量验证参数数组为 $W'=\{s_V'P\in G_1 \mid V\in\Omega\}$。上述计算过程可离线进行。

具体更新步骤如下：

Step 1 请求更新节点 V 选择随机数 $y\in Z_q^*$，计算 $Y=yP$，$h=H_1(Y\parallel ID_V\parallel R_V)$，计算签名 $z_V=y+D_Vh$，向 PKG 发送更新请求消息：

$$REQ_{update}=\{R_V,Y,z_V,ID_V\}$$

Step 2 PKG 收到请求消息后，检查节点 V 是否为已撤销的节点。若是，则算法结束；否则，计算 $h=H_1(Y\parallel ID_V\parallel R_V)$，并根据 ID_V 计算节点 V 当前公钥 K_V，验证签名是否正确：

$$z_VP=Y+h(R_V+K_VP_{pub})$$

若不成立，则算法结束；否则继续执行。

Step 3 PKG 选取随机数 $x\in Z_q^*$，对累加主密钥分量 s_V' 加密得 $u=(H_2(xY)+s_V') \bmod q$，计算 $X=xP$，$h=H_1(u\parallel ID_{PKG}\parallel R_{PKG}\parallel X)$，计算签名 $z_{PKG}=x+s_{PKG}h$。向节点 V 返回更新应答消息 RES_{update}：

$$RES_{update}=\{X,u,z_{PKG}\}$$

Step 5 节点 V 收到更新应答消息 RES_{update} 后，计算 $h=H_1(u\parallel ID_{PKG}\parallel R_{PKG}\parallel X)$，验证签名 $z_{PKG}P=X+h(R_{PKG}+P_{pub})$ 是否成立。若不成立，则丢弃 RES_{update}；若成立，则利用 X,u 和仅由自己掌握的 y，通过下式解密得到累加主密钥分量：

$$s_V'=(u-H_2(yX)) \bmod q=(u-H_2(xyP)) \bmod q$$
$$=(u-H_2(xY)) \bmod q$$

节点 V 计算新的主密钥分量：$s_{V\_new}=s_V+s_V'$。

Step 6 当所有结点完成更新后，PKG 向全网广播：

$$Q_{Broadcast}=\{W',T_{update},sig_{PKG}\}$$

式中，W' 是累加主密钥分量验证参数数组，T_{update} 为主密钥分量的启用时间，sig_{PKG} 是 PKG 对 W'，T_{update} 的签名。

Step 7 任何节点收到消息后，验证签名是否正确，若正确，则计算新的主密钥分量验证参数：

$$W_{V\_new}=W_V+W_V'=s_VP+s_V'P=(s_V+s_V')P=s_{V\_new}P$$

当系统时钟 $\tau=T_{update}$ 时，所有节点启用新验证参数数组 W'。

2.4 会话密钥协商

若在系统时间 τ 时刻，节点 A 与节点 B 需要建立会话。会话密钥协商过程如下：

Step 1 A 随机选择 $a\in Z_q^*$，计算 $T_A=aP$，$h=H_1(T_A\parallel ID_A\parallel R_A)$，计算签名 $z_A=a+D_Ah$，向 B 发送消息：

$$\{T_A,z_A,ID_A,R_A\}$$

Step 2 B 收到消息后，根据 ID_A 计算 A 的当前公钥 K_A，计算 $h=H_1(T_A\parallel ID_A\parallel R_A)$，验证签名是否成立：

$$z_AP=T_A+h(R_A+K_AP_{pub})$$

若不成立，则算法结束；否则继续执行。

Step 3 B 随机选择 $b\in Z_q^*$，计算 $T_B=bP$，$h=H_1(T_B\parallel ID_B\parallel R_B)$，计算签名 $z_B=b+D_Bh$，向 A 发送消息：

$$\{T_B,z_B,ID_B,R_B\}$$

B 计算会话密钥：$k_{BA}=H_3(bT_A)$。

Step 4 A 收到消息后，根据 ID_B 计算 B 的当前公钥 K_B，计算 $h=H_1(T_B\parallel ID_B\parallel R_B)$，验证签名是否成立：

$$z_BP=T_B+h(R_B+K_BP_{pub})$$

若不成立，则算法结束，否则 B 计算会话密钥：$k_{AB}=H_3(aT_B)$。

通过上述过程，节点 A、B 建立了共同的会话密钥：$k_{BA}=H_3(bT_A)=H_3(abP)=H_3(aT_B)=k_{AB}$。

3 方案安全性分析

3.1 系统主密钥安全

本方案可防止主密钥被攻击者或普通节点获得，主密钥安全由椭圆曲线离散对数问题(ECDLP)的难解性来保证，主要体现在以下方面。

(1)攻击者无法由系统公开参数计算出主密钥。系统中与主密钥有关的公开参数有：G_1 的生成元 P、系统公钥 $P_{pub}=sP$、主密钥分量验证参数数组 W。攻击者无法利用 P 和 $P_{pub}=sP$ 计算出 s，因为这需要解决椭圆曲线离散对数问题。攻击者利用 P 和 $P_{pub}=sP$ 根据双线性对的性质可以计算出：$e(P,P_{pub})=e(P,sP)=e(P,P)^s$，但是攻击者通过 $e(P,P)^s$ 计算 s 同样需要解决 ECDLP。攻击者利用门限个主密钥分量更新参数 $W_V=s_VP$ 可以重构出：$\sum_{V\in\Lambda}\lambda_V(0)W_V=\sum_{V\in\Lambda}\lambda_V(0)s_VP=sP$，但是由 sP 计算 s 同样面临解决 ECDLP。

(2)不诚实的普通成员节点无法利用自己的私钥 $K_A^{-1}=(D_A,R_A)$ 计算出 s，因为若要根据 $D_A=r_A+sH_1(ID_A\parallel phase)$ 计算 s，首先要利用 $R_A=r_AP$ 计算出 r_A，同样面临解决 ECDLP。

(3)攻击者无法通过攻破 PKG 来获得主密钥。初始化完成后，PKG 销毁了 s、r_{PKG} 以及主密钥共享多项式 $f(x)$，只保存了 s_{PKG} 作为证明自己身份的签名密钥。攻击者即使攻破了 PKG 结点，也无法由 $s_{PKG}=r_{PKG}+s$ 计算出 s，因为攻击者首先要利用 $R_{PKG}=r_{PKG}P$ 计算 r_{PKG}，同样面临解决 ECDLP。

(4)攻击者也可通过欺骗干扰或监听 D-PKGs 节点失误的方式,并通过累积获取门限个主密钥分量来重构主密钥。但本文方案提供的周期性主密钥分量更新机制可大大降低这种攻击成功的可能性。

由上述分析可以看出,该方案可以很好地保证主密钥的安全性,因此系统具有较强的容侵能力。

3.2 私钥和主密钥分量更新安全

节点 A 进行私钥更新的过程中,满足机密性和认证性。

(1)机密性。在私钥更新过程中,D-PKGs 节点 V 利用随机数 x_V 和请求消息中的 Y,对产生的部分私钥 M_V 进行加密传送。攻击者可以获得的信息有:$Y=yP$、$X_V=x_VP$ 和 $U=M_V\oplus H_1(x_VY)$。由 Y 和 X_V 计算 x_VY 是困难的,所以攻击者无法解密 U 获得 M_V。

(2)认证性。D-PKGs 节点通过验证 $zP=Y+h(R_A+K_AP_{pub})$ 来确定请求节点 A 的合法性。这是因为:

$$\begin{aligned}zP&=(y+D_Ah)P=yP+D_AhP\\&=yP+h(r_A+sK_A)P=yP+h(r_AP+sK_AP)\\&=Y+h(R_A+K_AP_{pub})\end{aligned}$$

攻击者虽然可以选择 $y'\in Z_q^*$,计算 $Y'=y'P$,但是因得不到 D_A 而不能伪造签名 $z'=y'+D_Ah'$,无法通过 D-PKGs 节点的验证。

更新请求节点 A 利用 D-PKGs 节点 V 的主密钥分量验证参数 W_V 通过验证 $M_VP=X_V+K_{A\_new}W_V$ 来确定部分私钥 M_V 的正确性。对于不掌握主密钥分量 s_V 的攻击者,虽然可以选择 $x'_V\in Z_q^*$,伪造 $X_V'=x_V'P$,但是无法伪造 $M_V'=x_V'+s_VK_{A\_new}$,所以不能通过节点 A 的验证。

在主密钥分量更新过程中,同样提供了类似的安全机制,保证了主密钥分量的机密性和不可伪造性。

3.3 会话密钥协商安全

本方案中的会话密钥协商协议满足以下安全特性。

(1)已知会话密钥安全 Known-Key Security(KKS):本方案中协商参与者每次协商都会生成不同的会话密钥,一个会话密钥的泄漏不会影响其他会话密钥的安全性。如果节点 A 与节点 B 的会话密钥 k_{AB} 泄露,攻击者仅可在本次交互中冒充 A 与 B 通信或是冒充 B 与 A 通信,但不会影响到 A 与 B 间或与其他节点间的其它会话的安全。

(2)前向保密性 Forward Security(FS):指一个实体或多个实体长期私钥的泄漏不会影响以前建立的会话密钥的安全。本方案中,对于已掌握节点 A 私钥 K_A^{-1} 的攻击者,因无法获得与旧会话密钥有关的 a 和 b,而无法计算出旧的会话密钥。对于已经掌握系统主密钥的攻击者(能够得到任意一方的私钥),同样无法计算出旧的会话密钥。所以协议提供了完善的前向保密性和主密钥前向保密性。

(3)抗密钥泄露伪装 Key-Compromise Impersonation Resilience(KCIR):指攻击者若获得某节点 A 的长期私钥,则仅能向其它节点冒充 A,而无法向 A 冒充其它节点。本方案中,密钥协商双方需要对产生的密钥协商参数进行签名,已掌握节点 A 私钥的攻击者无法在协议中假冒 B,因为他不知道 B 的私钥,无法计算出密钥协商参数 T_B 的正确签名。

(4)密钥控制 Key-Control(KC):本方案中,会话密钥由双方选取的随机数 a 与 b 共同决定,节点 A 与 B 都不能单独控制密钥的生成,即不能强制会话密钥是一个预先选择的值。

同文献[7]中方案相比,本方案具有更强的安全性。因为本方案在私钥更新过程中,D-PKGs 对更新后的私钥份额加密传送,保证了私钥份额的机密性。而文献[7]在私钥更新过程中,私钥份额以明文形式传送,恶意节点可以获得私钥份额并构造出完整私钥。

4 方案效率分析

本节将对方案的计算开销进行分析,并与文献[7]的方案进行比较,因为两个方案都是基于身份的分布式密钥管理方案。由于系统建立只在初始化时运行一次,并且可以离线进行,对系统运行效率影响较小,另外文献[7]的方案不涉及主密钥份额更新,因此仅分析私钥更新和会话密钥协商两个子协议的计算开销。所考虑的运算类型包括双线性对运算(P)、映射到 G_1 上点的 Map-to-Point 哈希运算(MtP)、G_1 上点乘运算(gM)、G_1 上点加运算(pA)。相对于上述运算,其它运算的开销(如有限域上的模乘、模加运算以及简单哈希操作)可忽略不计。

本文方案同文献[7]方案的详细计算开销情况如表 1 所列。本文方案的私钥更新过程中,不考虑 Step 1 和 Step 8 的计算开销,因为这两步可在更新前和更新后离线进行。私钥更新过程中应答节点计算开销是指网络中全部 N 个成员节点更新完毕后,单个 D-PKGs 节点的平均计算开销。

表 1 计算开销

方案	子协议	请求节点计算开销	应答节点计算开销
本文方案	私钥更新	(1+3t)gM+tpA	(Nt/n)(5gM+2pA)
	会话密钥协商	5gM+2pA	5gM+2pA
文献[7]方案	私钥更新	2tP	$(\frac{Nt}{n})$(gM+2P)
	会话密钥协商	1MtP+3gM+2P	1MtP+3gM+2P

由文献[8、9]可知,双线性对运算、Map-to-Point 哈希运算和 G_1 上点加运算的计算开销分别是 G_1 上点乘运算的 10 倍、1 倍和 1/10。根据上述比率和表 1 中的数据可以计算出本文方案两个子协议相对于文献[7]方案,在计算效率上提高的百分比,如表 2 所列。

表 2 计算效率提高百分比

子协议	请求节点计算效率提高百分比	应答节点计算效率提高百分比
私钥更新	84.5%	75.2%
会话密钥协商	78.3%	78.3%

由表 2 数据可以看出,相对于文献[7]方案,本文方案两个子协议减少了 70%以上的计算开销。这主要是由于在密钥管理过程中,本方案不需要进行复杂的双线性对运算,只需要进行点乘和点加操作。本方案由于具有较小的计算开销,因此更加适合在资源受限的普适计算环境中应用。

结束语 密钥管理是确保普适环境安全的关键问题。本文利用椭圆曲线加法群设计了一种新的基于身份的密钥管理方案。与现有同类方案相比,本方案在保证安全性的基础上,减少了 70%以上的计算开销。本方案同时具有安全性和效率上的优势,更加适合在普适环境下应用。

参 考 文 献

[1] 徐光祐,史元春,谢伟凯. 普适计算[J]. 计算机学报,2003,26(9):1042-1050

(下转第 131 页)

从图2—图5还可以看出，在嵌入率一定时，PSNR随着量化步长D的增加而减小，因为量化步长增加，水印嵌入深度加深，量化误差增大，图像的视觉质量随之降低，PSNR随之减小；在量化步长一定时，PSNR随着嵌入率的增加而减小，因为嵌入率增加说明在相同的载体中嵌入的水印数据量增加，被量化调制像素增加，从而量化误差增大，PSNR减小；PSNR一定时，量化步长随着嵌入率的增加而减小，嵌入率增加，为了保证一定的PSNR，水印嵌入深度要减小，量化步长要减小。

结束语 量化调制方法因其嵌入信息多、计算复杂度低，在数字水印和信息隐藏领域得到了广泛应用。但是无论是隐藏信息还是嵌入水印都要保证嵌入后载体的视觉质量，量化调制方法中影响载体视觉质量的参数有量化步长、水印数据量和量化系数，水印数据量和量化系数往往是事先确定的，而量化步长对应于嵌入深度，是可以调节的。目前确定量化步长值的方法是通过反复试验，在实际应用中进行大批量嵌入时是不现实的，针对这个问题本文对含水印图像的PSNR和量化步长的定量关系进行了估算，在水印量固定时，根据嵌入后载体的PSNR要求，可以直接计算出量化步长值，无需反复试验。本文虽仅以奇偶量化调制方法为例进行研究，但研究结果可以很容易扩充到其它量化调制方法。本文仅对空域像素这种量化系数进行了研究，实际上量化调制方法还经常用于其它量化系数，如DCT系数、DWT系数等，我们下一步的工作是针对这些量化系数对量化步长与视觉质量之间的定量关系进行研究。

参 考 文 献

[1] Subramanyam A, Emmanuel V, Sabu, et al. Robust Watermarking of Compressed and Encrypted JPEG2000 Images[J]. IEEE Transactions on Multimedia, 2012, 14(3): 703-716

[2] 凌洁，刘琚，孙建德，等. 基于视觉模型的迭代AQIM水印算法[J]. 电子学报，2010，38(1)：151-155

[3] 邓艺，赵险峰，冯登国. 基于非均匀DCT的量化索引调制隐写[J]. 电子与信息学报，2010，32(2)：323-328

[4] 胡青，龙冬阳. 基于DWT-SVD的奇异向量量化水印算法[J]. 计算机科学，2011，38(11)：30-33

[5] Chen B, Wornell G W. Provably robust digital watermarking [C] // Proceedings of SPIE. Vol. 3845 of Multimedia Systems and Applications II, San Jos'e, USA, 1999: 43-54

[6] Chen B, Womell G. Quantization index modulation: a class of provably good methods for digital watermarking and information embedding [J]. IEEE Transactions on Information Theory, 2001, 47(4): 1423-1443

[7] Chen B, Wornell G W. Quantization index modulation methods for digital watermarking and information embedding of multimedia [J]. Journal of VLSI Signal Processing Systems, 2001, 27(1): 7-33

[8] 肖俊，王颖. 扩展变换抖动调制水印算法中投影向量的研究[J]. 中国图象图形学报，2006，11(12)：1799-1805

[9] 肖俊，王颖，李象霖. 带失真补偿的抖动调制水印算法中的补偿因子研究[J]. 电子学报，2007，35(4)：786-790

[10] 李雷达，郭宝龙，武晓钥. 一种新的空域抗几何攻击图像水印算法[J]. 自动化学报，2008，34(10)：1235-1242

[11] Zhu Xin-shan. Image-adaptive Spread Transform Dither Modulation Using Human Visual Model [C]// Proceedings of International Conference on Computational Intelligence and Security, Lecture Notes in Artificial Intelligence (LNAI) 4456. Springer-Verlag, 2007: 913-923

[12] 肖筱南. 新编概率论与数理统计[M]. 北京：北京大学出版社，2008：141-142

[13] Tsai M J, Yu K Y, Chen Y Z. Joint wavelet and spatial transformation for digital watermarking [J]. IEEE Transactions on Consumer electronics, 2000, 46(1): 241-245

(上接第127页)

[2] Park J S, Sadi M G, Kim D S, et al. A Novel Pairwise Key Predistribution Scheme for Ubiquitous Sensor Network [C] // MADNES 2005. Singapore, 2005: 2-13

[3] Kausar F, Hussain S, Park J H, et al. A Key Distribution Scheme Preventing Collusion Attacks in Ubiquitous Heterogeneous Sensor Networks[C] // IFIP International Federation for Information Processing 2007. Taipei, Taiwan, 2007: 745-757

[4] Ge H. An Efficient Key Management Scheme for Pervasive computing[C] // IEEE International Conference on Multimedia. Irvine, USA, 2005: 657-661

[5] Moon J S, Park J H, Lee D G. Authentication and ID-Based Key Management Protocol in Pervasive environment [J]. Wireless Personal Communications, 2009, 50(3): 221-233

[6] Shamir A. Identity Based Cryptosystems and signature schemes [C]// Proc. CRYPTO'84. New York, USA, 1984: 47-53

[7] Sun H, Zheng X F, Deng Z Q. An Identity-based and Threshold Key Management Scheme for Ad hoc Networks[C]// IEEE International Conference on Networks Security Wireless Communications and Trusted Computing. Wuhan, 2009: 520-523

[8] Koblitz P, Menezes P, Vanstone S. The state of elliptic curve cryptography[J]. Designs, Codes and Cryptography, 2000, 19: 173-193

[9] Barreto P, Kim H, Lynn B, et al. Efficient algorithms for pairing-based cryphosystems [C] // CRYPTO2002. California, USA, 2002: 354-368