第 36 卷 第 1 期 计 算 机 工 程 2010 年 1 月
Vol.36 No.1 Computer Engineering January 2010
·安全技术· 文章编号：1000—3428(2010)01—0170—04 文献标识码：A 中图分类号：TP393

基于分簇的 Ad Hoc 网络密钥协商协议

张小彬，韩继红，王亚弟，刘 敏
(解放军信息工程大学电子技术学院，郑州 450004)

摘 要：以节点的公钥证书为基础，基于椭圆曲线密码体制提出一种分簇结构的 Ad Hoc 网络会话密钥协商协议，对协议的安全性和效率进行分析。该协议满足普遍认可的密钥协商安全要求，可抵抗中间人攻击、重放攻击、消息伪造攻击等多种攻击，有效地降低终端的计算、存储能力需求，减少了协商过程的通信开销。

关键词：Ad Hoc 网络；椭圆曲线密码体制；分簇；密钥协商

Clustering-based Key Agreement Protocol in Ad Hoc Network

ZHANG Xiao-bin, HAN Ji-hong, WANG Ya-di, LIU Min
(Institute of Electronic Technology, PLA Information Engineering University, Zhengzhou 450004)

【Abstract】 Based on the nodes' public key certificates, this paper proposes a clustering-based session key agreement protocol in Ad Hoc networks using the Elliptic Curve Cryptography(ECC). It analyzes the security and efficiency of the protocol. The protocol satisfies the common admissive security demands in key agreement. It can resist the man-in-the-middle attack, replay attack, message forge attack and so on, the requirement of computation and storage in mobile terminal is reduced, and the communication burden in the process is lessened effectively.

【Key words】 Ad Hoc network; Elliptic Curve Cryptography(ECC); clustering; key agreement

Ad Hoc 网络是一组带有无线收发装置的移动终端组成的多跳临时性自治系统。与传统网络不同的是，它可以在没有或不便利用现有网络基础设施的情况下，通过移动节点间的相互协作构建起一个移动通信网络，具有组网迅速灵活、系统抗毁性强等优点，在军事、救灾、商业等领域有着广阔的应用前景。Ad Hoc 网络具有动态变化的网络拓扑、受限的无线传输带宽、移动终端的局限性、分布式控制等特征，这些特征使其更易遭受被动窃听、主动入侵、拒绝服务等攻击。因此，Ad Hoc 网络节点间的安全通信面临极其严峻的挑战，而进行安全通信的前提就是要在通信双方之间安全地建立会话密钥。因此，建立会话密钥对于保障 Ad Hoc 网络的通信安全至关重要。

1 相关工作

密钥建立的方式一般有密钥分发、密钥预分发、密钥协商 3 种。在 Ad Hoc 网络中，每个移动节点的身份对等，没有单个可信任的节点，因此，集中式密钥分发机制并不适合 Ad Hoc 网络。文献[1]提出密钥预分配方案，每个节点从密钥服务器得到一个随机密钥对的集合，当 2 个节点需要进行安全通信时，就检查是否共享同一密钥，如果没有，就利用拥有共享密钥的中间节点来建立会话密钥。此方案需要密钥服务器的参与。另外，节点预分配密钥的数量直接影响着系统的安全性和可用性。文献[2]基于分簇的网络结构和可验证的秘密分享提出一个密钥协商方案，用接收方的公钥加密交换的 Diffle-Hellman 值、发送方和接收方信息，由接收方所在簇的簇首在簇内广播交换的信息，隐藏了消息的接收方。该方案在每次协商时接收方所在簇的所有节点都需要进行计算以确认消息的接收方，导致计算量较大。文献[3]通过在 Diffle-Hellman 密钥交换协议上结合哈希密钥链认证提出了一个对密钥建立方案，该方案具有较好的安全性，但节点间需要进行 3 次交互且指数运算的次数较多，计算和通信开销较大。文献[4]提出一个基于身份的分簇结构的会话密钥协商协议，但协议中将会话密钥的哈希值直接在通信线路上传递，使得该协议存在安全弱点。

2 预备知识

椭圆曲线上的密码体制分别由 Neal Koblitz 和 Victor Miller 提出，它利用有限域上的椭圆曲线有限群代替基于离散对数问题密码体制中的有限循环群。

(1)有限域 $GF(p)$ 上的椭圆曲线通常用 E 表示，是对于固定的 a，b，满足形如方程：$y^2 \equiv x^3 + ax + b(\mathrm{mod}\ p)$ 的所有点 (x,y) 的集合，外加一个零点或无穷远点 O。其中，大素数 $p \geqslant 3$，$a,b \in GF(p)$，且有 $4a^3 + 27b^2(\mathrm{mod}\ p) \neq 0$。令 P 为椭圆曲线 E 上的点，则使得 $nP = O$ 的素数 n 称为点 P 的阶。

(2)椭圆曲线离散对数问题：已知椭圆曲线 E 和点 P，随机生成一个整数 d，容易计算 $Q = d \times P$，但给定 Q 和 P 却很难计算 d。

(3)椭圆曲线密码体制(Elliptic Curve Cryptography, ECC)的主要优点是：ECC 的安全性基于椭圆曲线离散对数问题的难解性，160 bit 的 ECC 密钥就可以达到 1 024 bit 的 RSA 密钥的安全强度。因此，ECC 具有存储、计算效率高和通信带

作者简介：张小彬(1982－)，男，硕士研究生，主研方向：计算机网络安全；韩继红，教授；王亚弟，教授、博士生导师；刘 敏，硕士研究生

收稿日期：2009-06-05 **E-mail**：zhxbnhjd@sina.com

宽节约等方面的优势。

3 网络结构及相关假设

3.1 网络结构

Ad Hoc 网络一般有 2 种网络结构:平面结构和分级结构。平面结构的网络中所有节点是对等的,路由维护的开销较大,不具有很好的扩展性。而在分级结构中,节点按照分簇算法划分成多个簇,每个簇都由一个簇首和多个簇成员组成。低级簇的簇首节点之间形成高一级的网络。分级结构的 Ad Hoc 网络如图 1 所示。

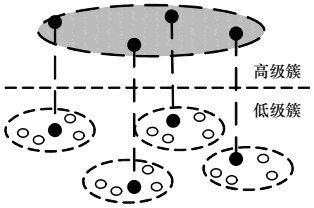

图 1 分级结构的 Ad Hoc 网络

分级结构的最大优点是可扩充性好,路由开销要比平面结构小,能够最优化网络带宽和网络管理。因此,本文考虑分簇结构的 Ad Hoc 网络密钥协商。

3.2 相关假设

相关假设如下:

(1)假设簇首节点在簇内周期性地广播簇内成员名单。

(2)假设簇首节点间周期性地交换本簇簇成员名单。

(3)假设已有一个基于节点可信度的分簇算法来对网络分簇。

(4)假设网络中的每个合法节点都拥有认证权威机构颁发的数字证书,网络中的节点不存储其他节点的证书。

4 基于分簇的密钥协商协议

在分簇结构的 Ad Hoc 网络中,节点间的通信可以分为簇内节点的对等通信和簇间节点的对等通信 2 类。簇内节点的对等通信包括低级簇簇内节点的对等通信和高级簇簇内节点的对等通信。对处于同一簇内的节点,密钥协商双方直接进行。对处于不同簇的节点间的协商,由于簇首具有较高的可信度,可以借助簇首来完成。

4.1 符号定义

协议的符号定义如下:

ID_i: 节点 i 的唯一标识

P: 椭圆曲线上阶为 n 的基点

x_i: 节点 i 的私钥, $x_i \in [2, n-2]$

Y_i: 节点 i 的公钥, $Y_i = x_i P$

CID_i: 簇 i 的唯一标识

CH_i: 簇 i 中簇首的唯一标识

$K_{i,j}$: 节点 i 和节点 j 共享的会话密钥, $K_{i,j} = K_{j,i}$

N_i: 节点 i 选取的新鲜因子

r_i: 节点 i 选取的随机数, $r_i \in [2, n-2]$

Q_i: 节点 i 在密钥协商时的贡献值

R_i: 随机数 r_i 的一个盲化值

S_i: 节点 i 计算得到的签名消息项

$E_{K_{i,j}}(m)$: 使用对称密钥 $K_{i,j}$ 加密明文 m

H_1: 强单向哈希函数, $H_1: \{0,1\}^* \to \{0,1\}^r, r \in Z_n^*$

H_2: 强单向哈希函数, $H_2: \{0,1\}^* \to Z_n^*$

$Certi$:由离线的权威认证机构颁发给节点 i 的数字证书,证书基于椭圆曲线密码体制。 $Certi = \{ID_i, T_i, Y_i, e_i, (D_i, S_i)\}$,其中 T_i 为签发时间和有效期, e_i 为消息 $ID_i \| T_i \| Y_i$ 经过哈希运算后的摘要, (D_i, S_i) 为 e_i 签名后的整数对。

4.2 簇内节点对等通信的密钥协商

假定同一簇内的节点 A 和 B 想要建立一个会话密钥,会话由节点 A 发起。协议执行过程如图 2 所示。

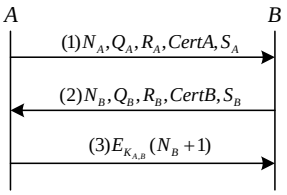

图 2 簇内节点的对密钥协商

具体过程如下:

(1)节点 A 产生一个随机数 $r_A \in [2, n-2]$,计算 $Q_A = r_A P$, $R_A = r_A H_2(r_A) P$, $S_A = r_A H_2(r_A) + x_A H_2(N_A, ID_B, Q_A, R_A) \bmod n$,向 B 发送 $M_1 = \{N_A, Q_A, R_A, CertA, S_A\}$。

(2)节点 B 收到 M_1 后,验证 $CertA$ 的有效性,若有效,则从中取出 A 的公钥 Y_A 验证 $R_A = S_A P - Y_A H_2(N_A, ID_B, Q_A, R_A)$ 是否成立,若不成立,则终止此次会话,否则,节点 B 产生一个随机数 $r_B \in [2, n-2]$,计算 $Q_B = r_B P$, $R_B = r_B H_2(r_B) P$, $S_B = r_B H_2(r_B) + x_B H_2(N_A + 1, ID_A, Q_B, R_B) \bmod n$,向 A 发送 $M_2 = \{N_B, Q_B, R_B, CertB, S_B\}$,同时节点 B 计算会话密钥,

$$K_{A,B} = H_1(r_B Q_A + x_B Y_A) = H_1(r_B r_A P + x_B x_A P)$$

(3)节点 A 收到 M_2 后,验证 $CertB$ 的有效性,若有效,从中取出 B 的公钥 Y_B 验证 $R_B = S_B P - Y_B H_2(N_A + 1, ID_A, Q_B, R_B)$ 是否成立,若不成立,则终止此次会话,否则,计算 $K_{A,B} = H_1(r_A Q_B + x_A Y_B) = H_1(r_A r_B P + x_A x_B P)$,并向节点 B 发送 $M_3 = E_{K_{A,B}}(N_B + 1)$。

(4)节点 B 解密消息 M_3 并进行确认,若确认通过,则双方就建立了共享的会话密钥。

由低级簇的簇首组成高一级簇,因此,簇首之间的会话密钥协商方法同低级簇的簇内协商过程相似。

4.3 簇间节点对等通信的密钥协商

在簇内会话密钥协商的基础上,簇间节点借助簇首来建立会话密钥。簇首贡献自己选择的随机数,该随机数的安全传递则是通过簇内已建立的会话密钥来保证。假定节点分别属于不同的簇 CID_1, CID_2,对应的簇首分别为 CH_1, CH_2。

A, B 之间想要建立一个共享的会话密钥,会话由节点 A 发起。协议执行过程如下:

(1)节点 A 产生一个随机数 $r_A \in [2, n-2]$,计算 $Q_A = r_A P$, $R_A = r_A H_2(r_A) P$, $S_A = r_A H_2(r_A) + x_A H_2(N_A, ID_B, Q_A, R_A) \bmod n$,得到 $MsgA$,并使用 K_{A,CH_1} 加密 CH_1, ID_B和N_A,得到 $Msg1$。节点 A 向 CH_1 发送消息 $M_1 = \{ID_A, CID_1, CH_1, MsgA, Msg1\}$。

(2)簇首 CH_1 收到节点 A 发送来的 M_1 后,使用 K_{A,CH_1} 对 $Msg1$ 进行解密,得到 CH_1, ID_B和N_A。然后检查解密得到的 CH_1 和 N_A,若错误,终止此次会话,否则,产生一个随机数 r_1,使用 K_{CH_1,CH_2} 加密 CID_1, ID_A, ID_B, r_1和N_{CH_1} 得到 $Msg2$,并根据解密得到的 ID_B 向节点 B 所在簇簇首 CH_2 发送 $M_2 = \{CID_1, CH_1, CID_2, CH_2, MsgA, Msg2\}$。

(3)簇首 CH_2 收到 CH_1 发送来的 M_2 后，使用 K_{CH_1,CH_2} 对 $Msg2$ 进行解密，并检查解密得到的 CID_1 和 ID_A，若错误，终止此次会话，否则，产生一个随机数 r_2，使用 K_{B,CH_2} 加密 $CID_1, ID_A, ID_B, r_1, r_2$ 和 N_{CH_2} 得到 $Msg3$，并根据解密得到的 ID_B 向节点 B 发送 $M_3=\{CID_2, CH_2, ID_B, MsgA, Msg3\}$。

(4)节点 B 收到 CH_2 发来的 M_3 后，首先对 $MsgA$ 中的 $CertA$ 进行验证，通过后，再从中取出 A 的公钥 Y_A 验证 $R_A=S_AP-Y_AH_2(N_A, ID_B, Q_A, R_A)$ 是否成立，若成立，使用 K_{B,CH_2} 对 $Msg3$ 进行解密并检查所得 ID_A 和 ID_B，若正确，产生一个随机数 $r_B\in[2,n-2]$，计算 $Q_B=r_BP$，$R_B=r_BH_2(r_B)P$，$S_B=r_BH_2(r_B)+x_BH_2(N_A+1, ID_A, Q_B, R_B) \bmod n$，得到 $MsgB$，并使用 K_{B,CH_2} 加密 $CID_1, ID_A, CH_2, N_{CH_2}+1$，向 A 发送 $M_4=\{ID_B, CID_2, CH_2, MsgB, Msg4\}$，同时节点 B 计算会话密钥，$K_{A,B}=H_1(r_Br_1r_2Q_A+x_BY_A)=H_1(r_Br_1r_2r_AP+x_Bx_AP)$。如果此步中的 3 项验证有任一项未通过，则终止此次会话。

(5)簇首 CH_2 收到节点 B 发送来的 M_4 后，使用 K_{B,CH_2} 对 $Msg4$ 进行解密，然后检查解密得到的 CH_2 和 $N_{CH_2}+1$，若错误，终止此次会话，否则，使用 K_{CH_1,CH_2} 加密 $CID_2, ID_B, ID_A, r_1, r_2, N_{CH_1}+1$ 得到 $Msg5$，并根据解密得到的 CID_1 和 ID_A，向节点 A 所在簇簇首 CH_1 发送 $M_5=\{CID_2, CH_2, CID_1, CH_1, MsgB, Msg5\}$。

(6)簇首 CH_1 收到簇首 CH_2 发送来的 M_5 后，使用 K_{CH_1,CH_2} 对 $Msg5$ 进行解密，并检查解密得到的 CID_2 和 $N_{CH_1}+1$，若错误，终止此次会话，否则，使用 $K_{CH_1,A}$ 加密 $CID_2, ID_B, ID_A, r_1, r_2, N_A+1$ 得到 $Msg6$，并根据解密得到的 ID_A，向节点 A 发送 $M_6=\{CID_1, CH_1, ID_A, MsgB, Msg6\}$。

(7)节点 A 收到簇首 CH_1 发送来的 M_6 后，首先对 $MsgB$ 中的 $CertB$ 进行验证，若验证通过，则从证书中取出节点 B 的公钥 Y_B 验证 $R_B=S_BP-Y_BH_2(N_A+1, ID_A, Q_B, R_B)$ 是否成立，若成立，再使用 K_{A,CH_1} 对 $Msg6$ 进行解密，并检查解密得到的 ID_A, ID_B 和 N_A+1 的正确性，若正确，节点 A 计算会话密钥 $K_{A,B}=H_1(r_Ar_1r_2Q_B+x_AY_B)=H_1(r_Ar_1r_2r_BP+x_Ax_BP)$。然后使用 $K_{A,B}$ 加密 N_B+1 得到 M_7，将 M_7 发送到节点 B。如果前面 3 项验证有任一项未通过，则终止此次会话。

(8)节点 B 解密消息 M_7 并进行确认，若确认通过，则双方建立起了共享的会话密钥。

簇间节点的对密钥协商的执行过程如图 3 所示。

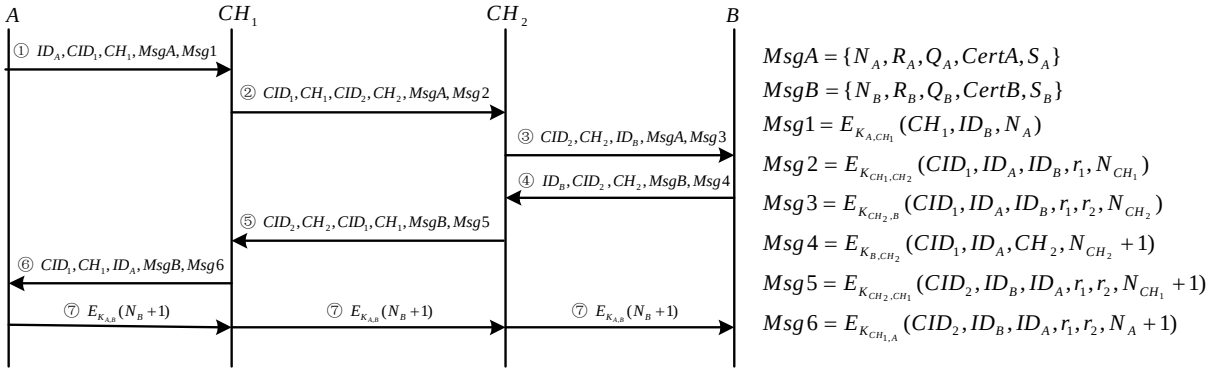

图 3　簇间节点的对密钥协商

5　协议分析

5.1　安全性分析

本文中的密钥协商协议的计算安全性是基于椭圆曲线上的离散对数难题和安全单向杂凑函数，下面对协议的安全性进行分析，协议满足现在普遍认可的密钥协商安全要求：

(1)密钥控制安全

协议的参与者(或攻击者)不能预先选择或预知会话密钥。在本文协议中，协商的会话密钥计算中包括了双方每次产生的随机数 r_A 和 r_B，在簇间节点协商中还包括簇首选择的随机数 r_1 和 r_2，协议的参与者(或攻击者)无法预知这些随机数，因此，本协议满足密钥控制安全性。

(2)已知会话密钥安全

某次会话密钥的泄漏不会影响到其他轮协商的会话密钥的保密性。协议中，每次会话密钥的计算都包含新的随机数，而且采用了抗碰撞的哈希函数，因此，会话密钥之间是独立的，一个密钥被破解不会影响到另一个密钥的安全性。

(3)向前安全性

协议参与者的长期私钥泄漏不会影响到这之前所协商的密钥的安全性。在簇内协商协议中，即使攻击者获得参与者 A 或 B 的长期私钥，甚至同时得到了两者的私钥，其也很难获得之前 A, B 协商的会话密钥，这是因为协商的会话密钥中包含节点 A 和 B 产生的随机数 r_A 和 r_B，这 2 个随机数并不直接在网络中传输，在网络中传输的是 Q_A, R_A 和 Q_B, R_B，由椭圆曲线上离散对数难题可知，攻击者很难由 Q_A, R_A 和 Q_B, R_B 推出 r_A 和 r_B。在簇间协商协议中，簇首节点选择的随机数 r_1 和 r_2 是利用簇内已建立的会话密钥加密发送的，攻击者也无法得到。另外，如果攻击者想从 $r_AH_2(r_A)=S_A-x_AH_2(N_A, ID_B, Q_A, R_A) \bmod n$ 中计算出 r_A 也是不可行的，由于 $H_2(r_A)$ 是强单向函数。因此攻击者也就无法计算出会话密钥，协议具有向前的安全性。

(4)密钥泄露的伪装攻击安全

敌手破解用户的长期私钥后不能够伪装成别人来欺骗被破解的用户。在协议中，假设节点 A 的私钥被攻击者 C 破解，攻击者 C 窃听到节点 B 发送的消息，然后伪装成节点 B 向节点 A 发送该信息。但由于其无法获得节点 A 选择的随机数 r_A，仍然无法计算出正确的会话密钥。在协议的最后一步确认中伪装将被识破，因此攻击者无法伪装成别人来欺骗用户 A，协议满足密钥泄露的伪装攻击安全。

(5)未知密钥协商攻击安全

用户不能被欺骗去和未知的第三者协商密钥。簇内协商协议中，节点 *B* 收到节点 *A* 发来的信息后，首先验证消息中的证书 *CertA* 是否是有效，然后从中取出公钥来验证签名项，如果正确，表明该消息确实是 *A* 发送过来的且所签名的消息项没有被修改过，而签名中包含了收方 *B* 的标识，因此，还可以确定该消息的接收方确实是 *B*，从而完成对 *A* 的认证。同样，节点 *A* 通过对 *B* 发送的消息的验证完成对节点 *B* 的认证。簇间协商协议中，节点 *A* 和 *B* 之间利用与簇内协商相似的过程分别对 *MsgB* 及 *MsgA* 进行验证，完成双向的身份认证；而节点 *A, B* 与簇首之间，簇首与簇首之间则通过簇内已建立的会话密钥来完成双向的身份认证。因此，本文中的协议可以防止这种欺骗的发生。

除满足以上要求外，本文中的协议还能抵抗篡改攻击，在协议中，假定攻击者想要找到一个 S_A 来使随后的消息验证通过，则攻击者先要选择一个随机数 r_A，计算 $Q_A = r_A P$，$R_A = r_A H_2(r_A)P$，然后，必须找到 d 使其满足验证等式 $dP = R_A + Y_A H_2(N_A, ID_B, Q_A, R_A)$，但是找到 d 等价于解决椭圆曲线离散对数难题，从而攻击者很难伪造一个有效的消息来欺骗消息的接收者。另外，在簇间协商协议中，攻击者没有收发双方之间共享的会话密钥，其无法伪造消息中的加密项来使收方的验证通过。因此协议能够抵抗篡改攻击。

此外，本文协议中加入了新鲜因子，收发双方通过对新鲜因子的验证可以有效地防止攻击者利用旧的消息发起的重放攻击。由于本文协议是双向认证的，因此，还能够抵抗中间人攻击。

5.2 执行效率分析

本文从计算量和通信量来对协议执行效率进行分析。

5.2.1 计算量

计算量的大小可以通过协议中主要运算的总的执行时间来表示。各种运算的时间符号定义如表 1 所示。

表 1 符号定义

符号	定义
T_{MUL}	执行一次模乘运算的时间
T_{EXP}	执行一次模指数运算的时间
T_{ADD}	执行一次模加运算的时间
T_{EC-MUL}	执行一次椭圆曲线上乘法运算的时间
T_{EC-ADD}	执行一次椭圆曲线上加法运算的时间
T_{CERT}	执行一次证书验证的时间
T_{ASYM}	执行一次非对称加解密运算的时间
T_{SYM}	执行一次对称加解密运算的时间
T_{MAP}	执行一次 map-to-point 哈希运算的时间
T_H	执行一次普通哈希运算的时间
T_I	执行一次求逆运算的时间
T_M	执行一次取模运算的时间

以簇内密钥协商协议为例给出参与方的计算量，由于通信双方的计算量相同，因此以参与方 *A* 来统计，协议中参与方 *A* 计算量主要包括 3 个部分：(1)计算 Q_A, R_A 和 S_A，计算量为 $2T_{EC-MUL} + 2T_H + 2T_{MUL} + T_{ADD}$。(2)验证 *B* 的证书及 *B* 发送来的消息项，计算量为 $T_{CERT} + 2T_{EC-MUL} + T_H + T_{EC-ADD}$。(3)会话密钥的计算及密钥确认消息的对称加密运算，$2T_{EC-MUL} + T_{EC-ADD} + T_H + T_{SYM}$。因此，节点 *A* 总的计算量为 $6T_{EC-MUL} + 2T_{EC-ADD} + 2T_{MUL} + T_{ADD} + 4T_H + T_{CERT} + T_{SYM}$。对于 T_{CERT}，假设证书中签名算法使用的是椭圆曲线数字签名标准 ECDSA，则对证书进行验证时的计算量为 $T_H + T_I + 2T_{MUL} + 2T_{EC-MUL} + T_M$。为了便于计算量的比较，基于

文献[5-6]给出一些运算操作转换到模乘运算的关系如下：

$T_{EXP} \approx 240T_{MUL}$

$T_{EC-MUL} \approx 29T_{MUL}$

$T_{EC-ADD} \approx 0.12T_{MUL}$

$T_{SYM} \approx 4T_{MUL}$

$T_H \approx 0.23T_{MUL}$

其中，相对于 $T_{MUL}, T_{ADD}, T_I, T_M, T_{EC-ADD}$ 都可以忽略不计。

根据上述转换关系，节点 *A* 总的计算量约为 $8T_{EC-MUL} + 2T_{EC-ADD} + 4T_{MUL} + 5T_H + T_{SYM} \approx 241T_{MUL}$。在簇间密钥协商协议中，节点 *A* 的计算量比簇内协商时多了 2 次对称加解密运算，簇首需要进行 4 次对称加解密运算。采用上面的方法对文献[2-4]中的协议的计算量进行统计，其中，文献[3]中的协议未采用分簇结构，同时每次协商时需要执行的哈希次数不固定，因此，未计入执行哈希运算的计算量。协议计算量比较如表 2 所示。

表 2 本文协议与其他协议的计算量比较

协议方案	簇内协商	簇间协商	
	节点 *A* 或节点 *B*	节点 *A* 或节点 *B*	簇首节点
文献[3]方案	$972T_{MUL}$	$972T_{MUL}$	$972T_{MUL}$
文献[2]方案	-	$480T_{MUL} + 2T_{ASYM}$ (1)	0
文献[4]方案	$484T_{MUL} + T_{MAP}$ (2)	$488T_{MUL} + T_{MAP}$ (2)	$16T_{MUL}$
本文协议	$241T_{MUL}$	$249T_{MUL}$	$16T_{MUL}$

其中，(1)表示接收方所在簇内节点都需要进行 2 次非对称加解密运算，且 T_{ASYM} 远大于 T_{SYM}；(2)表示 T_{MAP} 比 T_H 要大得多。

通过比较可以看出，本文协议的计算量明显优于其他几个协议，具有较高的运行效率。

5.2.2 通信量

本文中协议通信量用协议运行中产生的消息数来表示。对于簇内的密钥协商协议，在协商阶段，通信双方仅需要进行 1 次交互，共 2 条信息；在确认阶段，发起方需要发送 1 条确认消息到接收方。因此，簇内密钥协商总的通信量为 3。在簇间密钥协商时，通信双方及簇首生成并转发的消息数为 7，因此，总的通信量为 7。本文中的协议具有较小的通信量。

6 结束语

针对 Ad Hoc 网络通信安全需求及特点，本文提出一种采用 ECC 的基于分簇的密钥协商协议，用于在簇内和簇间建立通信双方的会话密钥，解决了移动 Ad Hoc 网络节点之间需要保密通信的问题。提出的协议能够满足现在普遍认可的密钥协商安全要求，并能够抵抗重放攻击、中间人攻击、消息仿造攻击等，具有较好的安全性。协议采用椭圆曲线密码体制、单向哈希函数等来协商会话密钥，减少了通信带宽，并且不需要进行复杂的指数运算，从而减轻了移动终端的计算负担，降低了终端的存储要求，可以较好地适用于资源受限的移动 Ad Hoc 网络。

（下转第 180 页）