一种低运算量 Ad-hoc 网络匿名路由协议

刘方斌 张 琨 张 宏

(南京理工大学计算机学院 南京 210094)

摘 要 Ad-hoc 网络节点运算能力差,能量有限且移动速度高,而公钥运算量大,能耗高,运算周期长,所以公钥运算不适合于 Ad-hoc 网络。已提出的匿名路由协议却含有大量的公钥运算。为了降低公钥运算量,将双线性对和零知识证明应用于匿名路由协议中,提出一种新的匿名路由协议——低运算量的 Ad-hoc 网络匿名路由协议,该协议大幅降低了公钥运算量。实验结果表明,该协议建立匿名路由所消耗的能量少,时延低。

关键词 双线性对,零知识证明,低能耗,低时延,匿名路由,Ad-hoc 网络

中图法分类号 TN918 **文献标识码** A

Low Computational Load Anonymous Routing Protocol for Ad-hoc Networks

LIU Fang-bin ZHANG Kun ZHANG Hong

(Institute of Computer Science, Nanjing University of Science and Technology, Nanjing 210094, China)

Abstract Nodes in Ad-hoc networks are limited in energy, have poor computational ability and move fast, and public key encryptions have heavy computational load, consume a lot of energy and have long computational cycle time, so public key encryptions are adapted for the Ad-hoc networks. The proposed anonymous routing protocols have a lot of public key encryptions. To reduces public key encryptions, we applied bilinear pairing and Zero Knowledge Proofs into anonymous routing protocol, and proposed a new anonymous routing protocol——an low computational load anonymous routing protocol for Ad-hoc networks(LCAR), which reduces public key encryptions heavily. Our analysis and simulation study verify that our protocol is much better than existing anonymous routing protocols on the aspects of energy efficiency and end-end delay.

Keywords Bilinear pairing, Zero knowledge proofs, Energy efficiency, Low end-end delay, Anonymous routing, Ad-hoc networks

1 引言

移动 Ad-hoc 网络由于不需要基础设施支持并具有自组织、自管理等特点,因此在军事战场、抢险救灾等环境中得到广泛的应用。但由于 Ad-hoc 网络固有的弱点,如采用无线信号作为传输介质,无线信号的无向性等,使得敌人可以根据监听到的路由信息分析出源和目的节点身份,确定它们的物理位置,从而实施物理打击或摧毁节点。为了使通信双方不被暴露,人们提出了多种匿名路由协议,如 SDAR[1],ANODR[2],MASK[3]和 AnonDSR[4],但它们引入了大量的公钥运算,从而消耗了大量的能量以及增加了路由建立的时间。为了降低路由建立的能量消耗和时延,必须尽量减少公钥运算。本文提出一种新的匿名路由协议,它将零知识证明及双线性对应用于匿名路由协议中,大幅减少了公钥运算量。

2 基于双线性对的匿名密钥协商

2.1 双线性对介绍

令 G_1 为由 P 生成的循环加法群,阶为 q,G_2 是具有相同阶 q 的循环乘法群,a,b 是 Z_q^* 中的元素,设 G_1 和 G_2 这两个群中的离散对数问题是困难问题,双线性对是指满足下列性质的一个映射 $\hat{e}:G_1\times G_1\rightarrow G_2$:

(1)双线性性:对所有 $P,Q\in G_1$,$\hat{e}(aP,bQ)=\hat{e}(P,Q)^{ab}$。

(2)非退化性:存在 $P,Q\in G_1$,使得 $\hat{e}(P,Q)\neq 1$。

(3)可计算性:对所有的 $P,Q\in G_1$,存在有效的算法计算 $\hat{e}(P,Q)$。

双线性映射可以通过有限域上的超椭圆曲线上的 Tate 对或者 Weil 对来构造[5]。

2.2 匿名密钥协商

令 G_1 为由 P 生成的循环加法群,阶为 q,G_2 是具有相同阶 q 的循环乘法群,$\hat{e}:G_1\times G_1\rightarrow G_2$ 为一个双线性映射。私钥生成中心(Private key generator, PKG)随机地选择一个主密钥 $s\in Z_q^*$,计算公钥 $P_{pub}=sP$,并选择一个安全的哈希函数 $H_1:\{0,1\}^*\rightarrow G_1^*$ 和一个密钥导出函数 g。PKG 公开系统参数 $\{G_1,G_2,q,\hat{e},P,P_{pub},H_1,g\}$[5]。

到稿日期:2010-12-19 返修日期:2011-02-23 本文受国家自然科学基金(90718021,61003210),江苏省自然科学基金(SBK201022379)资助。

刘方斌(1985-),男,博士生,主要研究方向为 Ad-hoc 网络安全及路由,E-mail:nj_lfb@163.com;张 琨 女,副教授,硕士生导师,主要研究方向为信息安全;张 宏 男,教授,博士生导师,主要研究方向为信息安全、数据挖掘。

对任意用户 A，其身份为 ID_A，PKG 计算其私钥 $S_A=sP_A$ 并把此私钥安全地发送给 A，P_A 为 A 的公钥，且 $P_A=H_1(ID_A)$。对另一用户 V，其公私钥对为 (P_V,S_V)。V 与 A 协商一会话密钥而不暴露 V 与 A 的身份，其过程如下：

(1)V 任选一个随机数 r_V，并将 r_VP_V 广播出去。

(2)A 接收到 r_VP_V 后，选择一个随机数 r_A，并计算

$$k_A=H(\hat{e}(r_VP_V,P_{pub})^{r_A})$$

将 r_AP 交给 V(此时 A 不知道 V 的身份)。

(3)V 收到 r_AP 后，计算

$$k_V=H(\hat{e}(r_VS_V,r_AP))$$

用户 V 与 A 即建立一共享密钥 $k_{V,A}=g(k_A)=g(k_V)$，但皆未泄漏自己的身份，即是 V 与 A 亦不知道对方的身份。

2.3 协议分析

(1)由于不能从 r_VP_V 及 r_AP 中分析出 V 与 A 的身份，因此 V 发送 r_VP_V 给 A，A 发送 r_AP 给 V，皆未泄漏自己的身份。

(2)密钥协商(2)中，A 在不知道 V 身份的情况下，发送 r_AP 给 V 的实现将在下文匿名路由协议的路由应答阶段阐述。

(3)k_A 与 k_V 相等，原因如下：

$$\begin{aligned}k_A&=H(\hat{e}(r_VP_V,P_{pub})^{r_{V,A}})=H(\hat{e}(r_VP_V,sP)^{r_{V,A}})\\&=H(\hat{e}(P_V,P)^{r_Ar_Vs})=H(\hat{e}(r_VsP_V,r_AP))\\&=H(\hat{e}(r_VS_V,r_AP))=k_V\end{aligned}$$

3 基于零知识证明的陷门构造方案

已提出的匿名路由方案中，陷门的构造有两种方法：

(1)使用公钥构造陷门，主要包括 SDDR[1]、ANODR[2]、SDAR[3]等匿名路由协议，这些匿名路由协议能够很好地实现通信双方身份、关系的匿名性，但在建立匿名路由的过程中因解陷门而引入了大量的公钥运算，消耗大量的节点能源，增加了路由建立时延。

(2)使用对称密钥构造陷门，主要包括 AnonDSR[4]、ARM[6]、ASR[7]等匿名路由协议，虽然这类协议解陷门不需要公钥运算，减少了部分公钥运算量，但需要其它协议预先在源和目的节点建立会话密钥，致使开销增加、实时性降低。

下面使用零知识证明构造陷门，此方法未引入公钥运算，亦不需要预先在源和目的节点间建立会话密钥。

3.1 基于零知识证明的陷门构造方案

零知识证明，即证明者 P 向验证者 V 证明某个论断是正确的，但不向 V 提供任何有用的信息[8]。基于零知识证明的思想，Feige 等人提出了 Feige-Fiat-Shamir 身份识别协议[8]，该协议中的计算量远小于公钥运算量[8]。借助于 Feige-Fiat-Shamir 身份识别协议，我们提出一种新颖的陷门构造方案。

Feige-Fiat-Shamir 身份识别协议是节点 P 向节点 V 证明知道它自己的秘密身份。从相反的方向考虑，V 想向 P 证明它知道 P 的公开身份，而不泄漏 P 的身份信息，亦可表述为：V 想寻找 P，而不对外泄露 V 和 P 的身份。

基于零知识证明的陷门构造方案如下：

有一可信任中心秘密地选取形式为 $4r+3$ 的两个大素数 p、q，使得 $n=pq$ 是计算上难分解的，然后公布 n 作为所有用户的模。P 的秘密身份由小于 n 的 k 个数 $c_1,c_2,\cdots,c_k$ 组成，P 的公开身份为 $d_1,d_2,\cdots,d_k$，且

$$\gcd(c_i,n)=1$$

$$d_ic_i^2\equiv\pm1(\bmod n) \quad (1)$$

$$d_i<n \quad 1\leqslant i\leqslant k$$

在初始状态下，V 知道公开的 n 和 P 的公开身份 $d_1,d_2,\cdots,d_k$。

(1)V 随机选择子集 $S=\{s_1,s_2,\cdots,s_j\}\in\{1,2,\cdots,k\}$ 及 r，令 x 为 $r^2 \bmod n$，其中 S、x 公开，r 保密；

(2)V 使用它所要寻找的目的 P 的公开身份计算出

$$W_d=\prod_{i=1}^{j}d_{s_i} \bmod n$$

$$y=rW_d \bmod n$$

并把 (y^2,S,x) 当作陷门广播出去；

(3)任意节点 P' 接收到陷门后，利用自己的秘密身份计算出

$$W_c'=\prod_{i=1}^{j}c_{s_i}' \bmod n$$

并验证

$$x\equiv y^2W_c'^4(\bmod n) \quad (2)$$

(4)若式(2)成立，则说明 P' 就是 P，即节点 V 成功寻找到节点 P。

在此过程中，未泄漏 V 和 P 的任何身份信息，而且计算量远少于公钥运算量。

3.2 陷门构造方案的分析

(1)假定分解因子问题是一个难题，因此要求式(1)成立，否则立即可以分解出 n 的因子 p 和 q[9]。

(2)因为不知道 n 的因子计算模 n 的平方根，这与分解因子问题计算难度是相同的，所以不能够从 P 的公开身份推断出 P 的秘密身份[9]。同样道理，也不能从 x 推断出 r。

(3)如果源节点 V 寻找的目的节点就是 P'，则式(2)应该成立，因为

$$y^2W^4\equiv r^2W_d^2W_c^4\equiv r^2(W_dW_c^2)^2\equiv r^2\equiv x \bmod x$$

成立。

(4)为了降低 P 成功欺骗 V 的概率，Feige-Fiat-Shamir 身份识别协议需要多次执行，而本协议与 Feige-Fiat-Shamir 身份识别协议的验证过程是相反的，即 P 执行验证过程，不存在 P 欺骗 V 的情况，所以不需要多次执行本协议。

(5)为了不能够从 y 推断出 P 的公开身份，我们在 W_d 前乘以一个随机数 r，且此随机数应该保密。

4 匿名路由协议

我们提出一种新的匿名路由协议，它在保证匿名性、安全性的前提下，可大幅减少公钥运算量。

构造如下形式的匿名路由请求包和路由应答包：

$$RREQ=[ARREQ,seqnum,r_XP_X,onion_X,tr_{dest},E_{K_f}(seqnum),E_{SK}(M_S)]$$

$$RREP=[ARREP,onion_{X-1},r_{X-1,X}P,E_{K_{X-1,X}}(seqnum,K'_f,N_{X-1,X})]$$

且

$$tr_{dest}=\{Verify_{dest},E_{PK_{dest}}(ID_{dest},SK,K_f)\}$$

$$Verify_{dest}=(y^2,x,S)$$

$$onion_X=E_{K_X}(N_X,(onion_{X-1}))$$

$M_S=(ID_{src}, ID_{dest}, SK, K_f, PL, P, Sign_{src})$

$Sign_{src}=H_2(ID_{src}, ID_{dest}, SK, y^2, x, S, K_f, PL, P)$

符号声明：

ARREQ，ARREP：匿名路由请求、应答的标志；

ID_{dest}：目的节点身份标识；

ID_{src}：源节点的身份标识；

$seqnum$：全局唯一序列号，用于标示此 RREQ；

r_X：节点 X 生成的随机数；

y^2、x、S：零知识证明中的相关参数；

PK_{dest}：目的节点公钥；

SK：源和目的节点协商的会话密钥；

K_f：源节点生成的一次性对称密钥；

K_X：节点 X 生成的对称密钥；

N_X：节点 X 生成的假名；

$N_{X-1,X}$，$K_{X-1,X}$：由节点 X 生成的与节点 $X-1$ 会话使用的一次性假名及密钥；

P，PL：路由包填充的内容及长度；

K_f'：目的节点解密 RREQ 包所得到的 K_f；

P：群 G_1 的生成元。

4.1 路由请求阶段

源节点生成并广播如上 RREQ 数据包。任意节点 X 接收到 RREQ 数据包后，通过 $seqnum$ 查看是否接收过此 RREQ 数据包。若接收过，则丢弃之；否则记录下 $r_X P_X$，tr_{dest} 及 $E_{K_f}(seqnum)$，并生成此次匿名路由的假名 N_X 及对称密钥 K_X。用 K_X 加密 N_X 及 RREQ 中的 $onion_{X-1}$，生成 $onion_X$，并用 $onion_X$ 替代 RREQ 中的 $onion_{X-1}$，之后广播新的 RREQ。广播完成后，通过解陷门 tr_{dest} 来验证自己是否是目的节点，若不是，则丢弃 tr_{dest}，若是，则通过解密 tr_{dest} 可以得到源节点生成的会话密钥 SK，用 SK 解密 $E_{SK}(M_S)$，验证接收到的信息没被修改后，生成 RREP 包，并广播之。路由请求阶段的过程如图 1 所示。

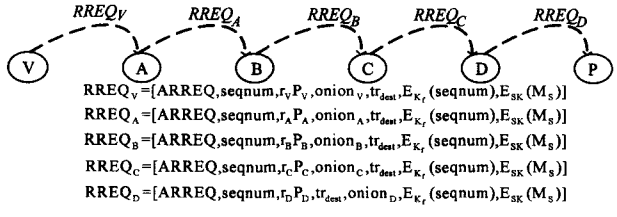

图 1 路由请求示意图

4.2 路由应答阶段

源节点 P 生成随机数 $r_{D,P}$ 及假名 $N_{D,P}$，并使用 $r_D P_D$ 生成与节点 D 的会话密钥 $K_{D,P}$，最后生成如下 RREP 包，并广播之：

$$RREP=[ARREP, onion_D, r_{D,P}P, E_{K_{D,P}}(seqnum, K_f', N_{D,P})]$$

任意节点 X 接收到 RREP 包后，使用自己的对称密钥 K_X 解密 RREP 数据包的第二部分，即 $onion$，并查看解密结果的第一部分是否为自己的假名 N_X，若不是，则丢弃此 RREP 包，若是，则记录下 $r_{X,X+1}P$ 和 $E_{K_{X,X+1}}(seqnum, K_f', N_{X,X+1})$ 并执行如下步骤：

(1)节点 X 利用 $r_{X,X+1}P$ 生成与节点 X+1 的会话密钥 $K'_{X,X+1}$，根据第一节的双线性对知识，可知 $K'_{X,X+1}=K_{X,X+1}$。使用 $K_{X,X+1}$ 解密 $E_{K_{X,X+1}}(seqnum, K_f', N_{X,X+1})$，得到 $seqnum$，K_f' 以及 $N_{X,X+1}$。

(2)验证 $E_{K_f'}(seqnum)$ 是否等于 $E_{K_f}(seqnum)$，若相等，则说明此 RREP 包来源于合法的目的节点，否则丢弃之。

(3)生成随机数 $r_{X-1,X}$ 及假名 $N_{X-1,X}$，且用 $r_{X-1,X}P$ 替换 RREP 中的 $r_{X,X+1}P$，$N_{X-1,X}$ 替换 $N_{X,X+1}$，用解密后的 $onion_{X-1}$ 替代 $onion_X$，之后广播新的 RREP 包。

路由应答阶段示意图如图 2 所示。

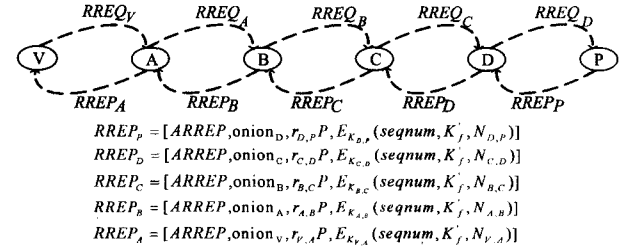

图 2 路由应答示意图

路由应答结束后，形成一条如图 3 所示的匿名路径，且各节点在本地维护表 1 所列的路由表。

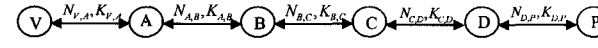

图 3 匿名路径

表 1 节点的路由表

节点 $X-1$、X 共享的会话密钥及假名	节点 X、$X+1$ 共享的会话密钥及假名	被 K_f 加密的序列号	路由请求序列号
$K_{X-1,X}N_{X-1,X}$	$K_{X,X+1}N_{X,X+1}$	$E_{K_f}(seqnum)$	$seqnum$

若是源节点，则在路由表中还应该保存目的节点 ID 及会话密钥 SK；若是目的节点，则还应该保存源节点 ID 及会话密钥 SK。

4.3 数据传输阶段

源节点 V 生成如下数据包：

$$DATA=[N_{V,A}, E_{K_{V,A}}(E_{SK}(datas))]$$

A 节点接收到数据包后，根据 $N_{V,A}$ 到路由表中查找到相应的会话密钥 $K_{V,A}$，解密后，再用 A 与 B 的会话密钥 $K_{A,B}$ 加密 $E_{SK}(datas)$，并用与 B 共享的假名 $N_{A,B}$ 替代 $N_{V,A}$，即生成数据包：$[N_{A,B}, E_{K_{A,B}}(E_{SK}(datas))]$，并广播之。如此反复，直到目的节点 P 接收到 D 发来的数据包：$[N_{D,P}, E_{K_{D,P}}(E_{SK}(datas))]$。$P$ 解密得到 $E_{SK}(datas)$ 后，再使用与源节点的会话密钥 SK 即可恢复出数据。数据传输阶段如图 4 所示。

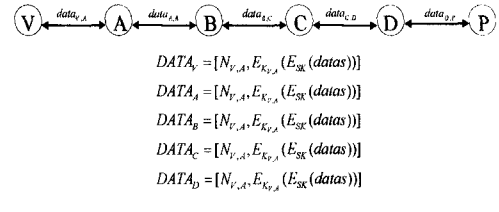

图 4 数据包传输过程

为了提高安全性，可在传输数据包时更新相邻节点的路由假名，即使用如下形式数据包：

$$DATA=[N_{V,A}, E_{K_{V,A}}(N'_{V,A}E_{SK}(datas))]$$

$N'_{V,A}$ 是由源节点 V 生成的用于 V 与 A 之间的新的路由假名。当 A 节点接收到数据包后，使用 $N'_{V,A}$ 替代路由表中的

$N_{V.A}$。

5 协议分析

5.1 低能耗、低时延分析

由于 Ad-hoc 网络节点大多使用电池供电，能量有限，且运算能力差，移动速度高，而公钥运算量大，能耗高，运算时间长，因此应当尽量减少匿名路由协议中的公钥运算量，以降低能量消耗，缩短路由建立时延。我们主要通过以下两个途径降低公钥运算量：

(1)基于双线性对的匿名密钥协商

假设网络中节点数为 N，路径长度为 L，每个节点有 M 个邻居。

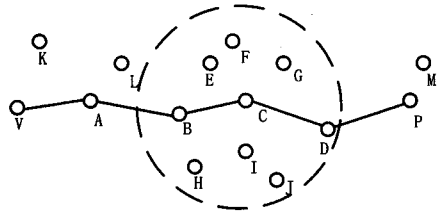

图 5 节点分布示例

如果使用公钥协商会话密钥，如 SDAR[1]，ANODR[2]，则会导致大量的公钥运算。SDAR[1]使用源节点产生临时公钥协商源和中间节点的会话密钥，此方法导致源节点每发出一个 RREQ 数据包，就使得网络中每个节点进行一次公钥运算。SDAR[1]中每发出一个 RREQ 数据包，因协商密钥而导致的公钥运算量为：N。ANODR[2]使用中间节点产生的临时公钥协商密钥，其运算量相对少些，但仍导致路径上各节点的邻居节点进行公钥运算。如图 5 所示，因 C 节点在路径上，所以在 RREP 阶段其所有邻居节点(图中虚线圈中的节点)都要进行一次公钥解密运算。ANODR[2]中每发出一个 RREQ 数据包，因协商密钥而导致的公钥运算量为：$L*(M+1)$。

本协议在 RREP 阶段先使用 *onion* 判断节点是否为路径上节点，只有路径上的节点才进行双线性对运算，所以每发出一个 RREQ 数据包，双线性对运算量为：$2*L$。这远少于使用公钥运算协商密钥所导致的运算量。

(2)基于零知识证明的陷门构造方案

在 SDAR[1]，ANODR[2]等路由协议中皆使用目的节点公钥构造陷门，导致每发出一个 RREQ 包，网络中所有节点都要进行一次公钥解密，以验证自己是否是目的节点。它们每发出一个 RREQ 数据包，因解陷门而导致的公钥运算量为：N。

本协议使用零知识证明构造陷门，网络中节点在解陷门时只需进行少许运算。零知识证明运算量比公钥运算量低[8]，且可以通过减少 S 集合中元素个数，进一步减少运算量，所以其运算量相对于公钥运算可以忽略。本方案因解陷门而导致的公钥运算量为：0。

AnonDSR[3]使用安全参数建立协议，先建立源和目的节点的会话密钥，再进行匿名路由的建立。但安全参数建立协议暴露了源节点和目的节点的身份，降低了匿名性，且增加了能量消耗和路由建立的时延。

本协议与已提出的匿名路由协议公钥运算量的对比如表 2 所列。

表 2 公钥运算量比较

	LCAR	ANODR	SDAR	AnonDSR	EARP[10]
目的节点匿名性	零知识证明构造陷门，可以实现匿名	使用公钥构造陷门；可以实现匿名性	使用公钥构造陷门；可以实现匿名性	安全参数建立协议暴露目的节点	使用公钥构造陷门；可以实现匿名性
暴露给邻居节点	否	否	暴露	否	否
公钥运算量	L	M*L*P/2 或者 M*L*P/2+N	3*N'+2*L	N	N'+L
匿名路由请求的先决条件	不需要	不需要	需要先建立可信系统	需要先建立安全参数	假设源和目的节点已共享会话密钥；需要建立可信系统

P：ANODR 中每个节点平均拥有的一次性公钥个数；
N'：SDAR 及 EARP[10]中达到某一可信级别的节点个数。

5.2 安全性分析

(1)源节点、目的节点以及中间节点的匿名性

路由请求包中，节点使用随机数 r_X 乘以公钥 P_X，所以 P_X 未暴露节点身份；tr_{dest} 中的 $Verify_{dest}$ 亦未暴露节点身份，这由零知识证明保证；tr_{dest} 中目的节点的身份被目的节点公钥加密，所以亦未暴露目的节点身份。路由应答过程中未使用与节点身份相关的信息，所以未暴露节点身份。

(2)源和目的节点的不可关联性

由于 RREQ 是全网洪泛的且 RREP 的每一部分在每一跳过程中都是改变的，因此无法从 RREQ 及 RREP 数据包中找出源和目的节点的关联性，又因为源和目的节点身份都是匿名的，因此源和目的节点是不可关联的。

(3)主动攻击和被动攻击

目的节点可以通过检查 RREQ 数据包的签名查看数据包是否被修改过。每次使用不同的 K_f、$seqnum$ 及会话密钥 SK，可以防止对 RREQ 包任何部分的重放攻击。

若有节点假冒目的节点做出路由应答，则中间节点可以通过验证 $E_{K_f'}(seqnum)$是否等于 $E_{K_f}(seqnum)$来判断。

由于 RREQ 的第六、七部分及 tr_{dest} 的第二部分被加密，tr_{dest} 的第一部分安全性由零知识证明保证，r_X 乘以公钥 P_X，因此恶意节点无法从 RREQ 数据包中窃听任何有用的信息。

RREP 的第四部分被加密，第三部分的安全性由双线性对保证，所以恶意节点亦无法从 RREP 数据包中窃听任何有用的信息。

(4)时间攻击

Jiejun Kong 等人提出的利用 dummy 包[2]的方法可以很好地防范时间攻击。

(5)DOS 攻击

本协议虽不能完全防止 DOS 攻击，但由于消耗能量少，可以大幅降低 DOS 攻击的危害。

6 仿真实验

实验中我们从建立路由的能量消耗、路由建立时延、数据包发送成功率及控制包比例等方面，对比了本协议与已提出的匿名路由协议。(1)能量消耗：建立一条匿名路由所消耗的能量；(2)建立路由时延：从源节点发送路由请求包至接收到第一个路由应答包的时间；(3)数据包成功率：源节点发送的

数据包数与目的节点成功接收的数据包数的比例；(4)控制包比例：一个节点发送路由控制包数与此节点发送数据包数的比例，此比例取所有节点的平均值。

我们使用 NS-2[11]实验仿真平台。NS-2 是目前国际上非常流行的用于 TCP、路由等方面的仿真平台。仿真环境如下：仿真场景设置为 2000m×2000m，初始阶段 200 个节点平均分布于仿真区域中，节点有效传输距离为 300m，物理层使用 two-way ground model，链路层协议使用 802.11 DCF，节点移动模型为 random-way point model，节点移动速率为 5m/s，节点停顿时间为 60s，节点产生数据包的速率为 100 bytes/min，节点发送数据包的速率为 11Mb/s，仿真时间为 3600s，源和目的节点在所有节点中随机选取。所有数据皆是 5 次实验的平均值。

公钥加密使用密钥长度为 1024 位的 RSA 算法，对称加密使用密钥长度为 128 位的 AES/Rijndael 算法，哈希函数使用 160 位的 SHA-1 算法。

6.1 能量消耗对比

图 6(a)对比各匿名路由协议能量消耗随仿真时间的变化情况；图 6(b)是 LCAR 能量消耗的详细情况。SDAR 和 EARP 需要维护可信系统，且使用公钥构造陷门及协商会话密钥，所以其消耗能量最多。AnonDSR 与 ANODR 能量消耗比 EARP 少，且由于 ANODR 在路由应答阶段建立会话密钥，所以其能量消耗比 AnonDSR 少。由于 LCAR 使用零知识证明构造陷门，且在路由应答阶段使用双线性对协商会话密钥，因此其在所有匿名路由协议中能量消耗最少。

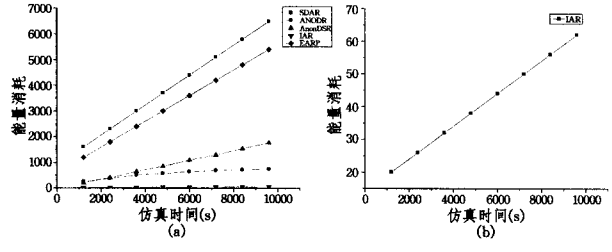

图 6 能量消耗随仿真时间的变化情况

图 7(a)对比各匿名路由协议能量消耗随节点移动速度的变化情况；图 7(b)是 LCAR 能量消耗的详细情况。SDAR 和 EARP 所耗能量依然是最多的，LCAR 依然最少，且所有的匿名路由协议随节点移动速度的增加，能量消耗的变化率也越大。由于 LCAR 的公钥运算量比其他匿名路由协议少，因此能量变化速率最小。

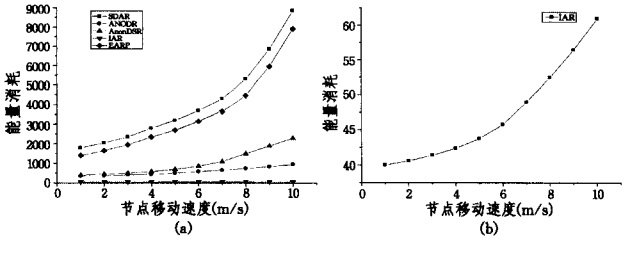

图 7 能量消耗随节点移动速度的变化情况

图 8(a)对比各匿名路由协议能量消耗随网络规模的变化情况；图 8(b)是 LCAR 能量消耗的详细情况。所有协议随网络规模的增大，能量消耗都在增加，但 LCAR 的能量消耗始终是最少的，主要原因是 LCAR 的公钥运算量最少。

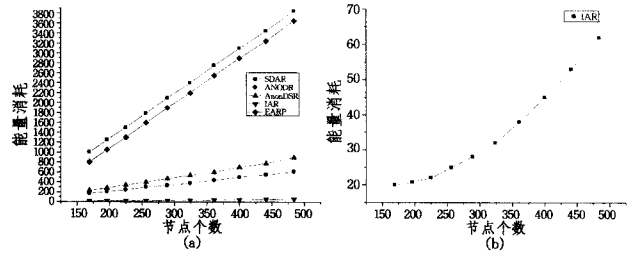

图 8 能量消耗随网络规模的变化情况

6.2 路由建立时延对比

图 9 显示 SDAR 的平均路由建立时延最长，而 LCAR 最短，因为公钥运算所需时间长，而 LCAR 的公钥运算量最少。由于 AnonDSR 在匿名路由请求前要执行安全参数建立协议，因此时延相对增加。

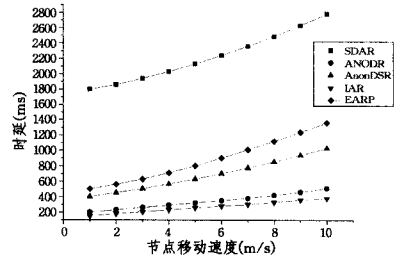

图 9 路由建立时延

6.3 数据包成功率

图 10 描述了源节点发送数据包的成功率。随着节点移动速度的增加，各个路由协议的数据包发送成功率都在下降，其中 SDAR 下降最快而 LCAR 最慢。由于公钥运算时间长，导致路由的建立和维护更加困难，因此公钥运算量大的路由协议成功率下降得快。SDAR 在路由请求阶段引入的公钥运算量最多，所以其成功率下降最快；而 LCAR 在路由建立过程中，公钥运算量最少，所以其成功率下降最慢。

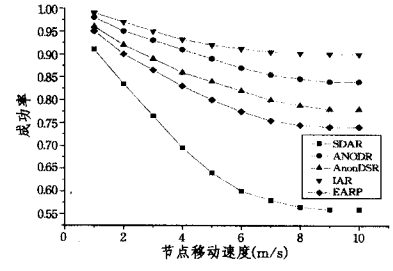

图 10 数据包发送成功率

6.4 控制包比例

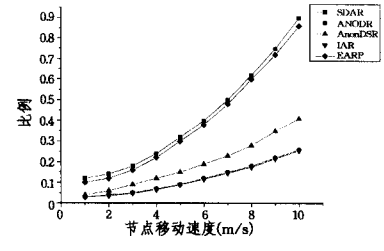

图 11 控制包比例

图 11 显示随节点移动速度的变化，控制包与数据包比例的变化情况。SDAR 与 EARP 的比例非常高，原因是它们皆需要维护可信系统，而随着节点速度的增加，可信系统维护成本加大，控制包增多。AnonDSR 的控制包比例比 EARP 少，

但比 ANODR 及 LCAR 多，因 AnonDSR 在匿名路由请求前需要安全参数建立协议。LCAR 和 ANODR 的控制包比例相对最少。

结束语 Ad-hoc 网络节点运算能力差，能量有限且移动速度高，但已提出的匿名路由协议含有大量的公钥运算，运算量大，能量消耗大且路由建立时延长。为了降低能量消耗与路由建立时延，我们提出了一种低运算量的匿名路由协议。它将零知识证明及双线性对应用到匿名路由建立过程中，大幅降低匿名路由过程中的公钥运算量，从而降低了能量消耗和路由建立时延。

现在的双线性映射主要通过有限域上的超椭圆曲线上的 Tate 对或者 Weil 对来构造，所以下一步工作将对椭圆曲线进行研究，以提高双线性对的运算效率。

参 考 文 献

[1] Boukerche A, El-Khatib K, Xu Li, et al. SDAR: A Secure Distributed Anonymous Routing Protocol for Wireless and Mobile Ad-hoc Networks[C]// 29th Annual IEEE International Conference on Local Computer Networks(LCN'04). Tampa, Florida, USA, November, 2004

[2] Kong Jie-jun, Hong Xiao-yan, Gerla M. An Identity-Free and On-Demand Routing Scheme against Anonymity Threats in Mobile Ad-hoc Networks[J]. IEEE Transactions on Mobile Computing, Frequency, 2007, 6: 888-902

[3] Zhang Yan-chao, Liu Wei, Fang Yu-guang. MASK: Anonymous On-Demand Routing in Mobile Ad-hoc Networks[J]. IEEE Transactions on wireless communications, 2006, 5(9)

[4] Song Rong-gong, Korba L, Yee G. AnonDSR: Efficient Anonymous Dynamic Source Routing for Mobile Ad-Hoc Networks [C]// Proceedings of the 2005 ACM Workshop on Security of Ad-hoc and Sensor Networks. Alexandra, Virginia, USA, January 2005

[5] 许春香，李发根，聂旭云，等. 现代密码学[M]. 成都：电子科技大学出版社，2008：135-139

[6] Seys S, Preneel B. ARM: Anonymous Routing Protocol for Mobile Ad hoc Networks[J]. International Journal of Wireless and Mobile Computing, 2009, 3: 145-155

[7] Zhu Bo, Wan Zhi-guo, Kankanhalli M S, et al. Anonymous Secure Routing in Mobile Ad-Hoc Networks[C]// Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN'04). Tampa, Florida, USA, November 2004

[8] Feige U, Fiat A, Shamir A. Zero Knowledge Proofs of Identity [C]// Proceedings of the nineteenth annual ACM symposium on Theory of computing. New York, USA, 1987

[9] 何德全，肖国镇，卿斯汉，等. 安全协议[M]. 北京：清华大学出版社，2005：215-217

[10] Li Xiao-qing, Li Hui, Ma Jian-feng, et al. An Efficient Anonymous Routing Protocol for Mobile Ad-hoc Networks, Information Assurance and Security 2009[C]// IAS'09. Fifth International Conference on. vol. 2, Aut. 2009: 287-290

[11] McCanne S, Floyd S. Advances in Network Simulation [EB/OL]. http://www.isi.edu/nsnam/. , July 2010

(上接第 33 页)

结束语 目前，基于 DWT-SVD 的水印方案大多都是将水印信息嵌入到奇异值中，也出现了一些利用 SVD 分解的正交矩阵第一列相邻系数关系稳定的特性来嵌入水印信息的一些算法，但这些算法通过调制相邻系数的关系来嵌入水印信息，虽然通过阈值调整来平衡算法的透明性和鲁棒性，获得了较高的 PSNR，但不可避免地会造成严重的局部失真，影响视觉效果。本文利用 SVD 分解的正交矩阵第一列系数值的稳定性，在量化阈值 T 的控制下嵌入水印信息，并根据正交矩阵向量系数间的制约关系调整其它系数值，保证嵌入信息的稳定性。实验结果表明，本文提出的算法在保证较好的透明性的前提下对各种攻击具有较强的鲁棒性，特别对 JPEG 压缩具有优异的鲁棒性，水印提取过程无须原始图像，具有极强的实用性。

参 考 文 献

[1] 黄达人，刘九芬，黄继武. 小波变换域图像水印嵌入对策和算法[J]. 软件学报，2002，13(7)：1290-1297

[2] 刘瑞祯，谭铁牛. 基于奇异值分解的数字图像水印方法[J]. 电子学报，2001，29(2)：168-171

[3] Liu Rei-zhen, Tan Tie-niu. An SVD-based watermarking scheme for protecting rightful ownership[J]. IEEE Trans. Multimedia, 2002, 4(1): 121-128

[4] Zhang Xiao-ping, Li Kan. Comments on An SVD-Based Watermarking Scheme for Protecting Rightful Ownership[J]. IEEE Transactions on multimedia, 2005, 7(2): 593-594

[5] 赵星阳，孙继银. 一类基于奇异值分解的图像水印算法伪验证分析[J]. 计算机应用，2010，30(2)：517-520

[6] Mohammad A A, Alhaj A, Shaltaf S. An improved SVD-based watermarking scheme for protecting rightful ownership[J]. Signal Processing, 2008, 88(9): 2158-2180

[7] Bhatnagar G, Raman B. A new robust reference watermarking scheme based on DWT-SVD[J]. Computer Standards & Interfaces, 2009, 31: 1002-1013

[8] Bao P, Ma X. Image adaptive watermarking using Wavelet domain singular value decomposition[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2005, 15(1): 96-102

[9] Chang Chin-chen, Tsai P, Lin Min-hui. SVD-based digital image watermarking scheme[J]. Pattern Recognition Letters, 2005, 26 (10): 1577-1586

[10] 张建伟，鲍政，王顺凤. 图像小波域分块奇异值分解的自适应水印方案[J]. 中国图象图形学报，2007，12(5)：811-818

[11] Chung K-L, Yang Wei-ning, Huang Yong-hua, et al. On SVD-based watermarking algorithm[J]. Applied Mathematics and Computation, 2007, 188(1): 54-57

[12] Fan Ming-quan, Wang Hon-xia, Li Sheng K. Restudy on SVD-based watermarking scheme[J]. Applied Mathematics and Computation, 2008, 203(2): 926-930

[13] 黄松，张伟，陈军，等. 一个基于 DWT 的自适应数字水印算法[J]. 计算机科学，2006，33(7)：155-157