移动自组网中一种基于集群的Byzantine节点检测机制

刘 洋 俞能海 冯现磊
(中国科学技术大学多媒体计算与通信教育部-微软重点实验室 合肥 230027)

摘 要 该文主要介绍了移动自组网中一种基于集群方法的 Byzantine 错误检测机制。并结合 CBRP，提出了一种适合移动自组网的内部出错节点清除算法。通过算法分析证明了所提出的算法可以显著减少清除具有 Byzantine 错误节点时所需的消息数目，降低了网络负载，有效提高了移动自组网的安全性和可信度。

关键词 Byzantine 将军问题，移动自组网，集群，安全
中图分类号：TP309 **文献标识码**：A **文章编号**：1009-5896(2006)12-2386-04

A Cluster Based Solution to BGP on Mobile Ad hoc Network

Liu Yang Yu Neng-hai Feng Xian-lei
(*MOE-Microsoft Key Lab. of Multimedia Computing and Communication, Hefei* 230027, *China*)

Abstract In this paper, a novel scheme to detect Byzantine fault on mobile Ad hoc network is presented. A fault-tolerant algorithm is introduced combined with CBRP, which improves the security and reliability of manet(mobile Ad hoc network). Algorithm analysis shows the scheme can reduce the messages complexity effectively.

Key words Byzantine generals problem, manet, Cluster, Security

1 引言

移动自组网(mobile Ad hoc network, manet)是一种新型的无线多跳网络。网络中的移动节点均由无线链路连接，而且没有固定的基础结构(infrastructure)。manet还具有能量和带宽有限、路由多跳、拓扑多变、无中心等特点。因此设计高效、安全和可信的manet有着十分重要的意义。以往针对manet安全问题的研究主要集中在分布式认证[1]和分布式密钥管理[2]等抵抗外部攻击这一方面，而针对manet中如何发现并清除内部叛变节点以及被攻破节点问题的研究相对较少。

在研究分布式系统容错处理技术时，Lamport最早在文献[3]中把叛变节点清除问题归结为Byzantine将军问题(Byzantine Generals Problem, BGP)，并提出一种基本的解决算法Oral Message算法，可以用来清除 $n(n\ge 4)$ 个节点中 $t(t\le(n-1)/3)$ 个叛变节点，当 n 和 t 比较大时这种算法的复杂度非常高。文献[4−7]对Lamport的算法提出了改进，一方面降低了运算复杂度，另一方面使之适应更普遍的网络情况；文献[6，8]还研究了在不可靠链路环境下解决BGP的方法；Wang等在文献[8]中最早将BGP引入到manet中，并提出了一种在无线移动环境下解决BGP的基本方案。上述方案在节点数目增多时，同样存在着运算时间和消息传输量随之剧增的问题，如何有效降低解决Byzantine将军问题的算法复杂度一直是这一研究领域的热点。

层次(hierarchical)网络能够很好地解决大规模网络中由于节点数目增多而带来的路由表庞大和难于管理的问题。Bergano等人在文献[10]将hierarchical clustering routing引入manet中。将一个规模相对较大的manet划分成若干个相对较小的集群(cluster)，能够形成一个两层的结构，在此基础上再划分集群则可以形成多层的结构，利用层次结构信息可以显著减少路由和网络管理以及安全机制的复杂度，从而提高网络使用效率[11]。集群在manet中也是一个非常热门的课题，已经发表了很多研究manet中集群划分算法和群首选取算法的文献，基于集群的路由协议[12,13]和安全机制[14]也经常出现在近期的论文中。在对这些文献调研的基础上，本文作者提出一种基于集群的解决Byzantine将军问题的算法，该算法的基本思想是：首先将整个manet划分成若干个彼此相邻且规模较小的集群，然后通过在每个集群中分布式地执行Oral Message算法来检测网络中具有Byzantine错误的节点。

本文后面部分的结构如下，在第 2 节集中说明文中涉及的术语或缩写，并简要介绍一下基于集群的路由协议CBRP(Cluster Based Routing Protocol)和 BGP；然后在第 3 节中将讨论本算法所需的必要假设，并对 manet 中基于 cluster 的 Byzantine 将军问题进行讨论和分析；在第 4 节中，结合 CBRP 的两层网络结构提出一种基于 cluster 的 Byzantine 节点检测算法；在第 5 节将对算法的正确性给出必要的证明；在最后一节我们对本文做一个简要的总结并介绍下一步的工作。

2 符号说明和 CBRP，BGP 简介

2.1 符号说明

Cluster_i表示第 i 个集群；CH_i为Cluster_i的群首(Cluster header)，每个集群中群首是唯一的；GW为网关(gateway)节点，Cluster之间通过GW节点通信；CM为Cluster中的成员(member)节点，不担任GW和CH；m_i 为Cluster_i中节点总数；t_i 为 Cluster_i 中Byzantine节点总数；$|C|$ 为网络中

2005-05-08 收到，2005-09-19 改回

Cluster 总数；T 为 Byzantine 节点总数，有 $T=\sum t_i$；BGP 为 Byzantine 将军问题(byzantine generals problem)；$\mathrm{OM}(t)$ 为 Oral Message 算法，参数 t 表示 Byzantine 节点数目为 t。

2.2 CBRP简介

CBRP是一种为manet设计的，基于集群的路由协议。它首先通过分布式的执行改进的min-ID集群算法，将自组网划分为一些直径为 2-hop的cluster，cluster之间彼此相邻也可以有重叠[12]。cluster内部节点之间的路由通过CH上保存的CM的信息动态发现和执行；在进行跨cluster的路由时，由于已经将节点划分到各个集群中，利用等级结构可以有效地简化路由表和减少路由请求的传输。

CBRP 要求节点定期或者通过条件触发发送 HELLO MESSAGE 来更新自己的 neighbor table 和 cluster adjacency table，从而获知 2−3 hop 范围内的当前网络拓扑结构。有路由请求时，源节点只需向所有的 CH 广播路由请求，而不是像 DSR(动态源路由)那样向所有的节点广播路由请求。因而减少了路由请求报文的数目，减少了网络负载。

2.3 BGP简介

BGP 可以如下描述：在一个由 n 个节点组成的网络中，存在 t 个错误节点，错误节点总是会阻止正常节点达成一致，这些节点称为具有 Byzantine 错误的节点(以下简称 Byzantine 节点)，为表达方便，本文称这种阻止正常节点达成一致的行为为 Byzantine 行为。因此需要研究一种通信协议使得：(1) 正确的节点能够对某一行动达成一致；(2) 少数 Byzantine 节点不能阻止正确节点达成这样的一致。

Lamport最早提出了Oral Message算法来解决BGP。该算法可以在 $(3t+1)$ 个节点中检测出不超过 t 个Byzantine节点。算法执行时：一个起始节点向其它节点发送message，其它节点接收message后，作为起始节点再递归地相互转发message，最后每个节点对收到的消息集合执行 majority( ) 来达成一致。具体算法请见参考文献[3]。考虑到Byzantine节点有能力伪造消息，Lamport在同一篇文章中又提出了Signed Message 算法。为了简单起见而又不失一般性，本文使用最基本的Oral Message算法。

3　问题分析

$\mathrm{OM}(t)$ 算法是一个递归过程，当节点总数很大时，需要发送和处理的消息数目非常巨大，比如要在 n 个节点中检测出 $t(t<n/3)$ 个Byzantine节点，一个节点需要发送的消息数目为：$(n-1)\cdot(n-2)\cdots(n-t-1)$[3]。如果能将 n 控制在一个比较小的范围内，则可以大大减少消息数量。

集群技术可以将 n 个节点的 manet 划分为 $|C|$ 个节点数目近似相等的若干个 cluster，m_i 大小近似为 $n/|C|$，这样就能将每个节点发送的消息数量为 $O\left(n/|C|\right)^{t_i+1}$，参见第 5 节定理 2 的证明。而且在 cluster 内部 CM 之间，CM 到 CH 或者 CM 到 GW 的距离最远为 2-hop，多数消息可以直接发送至目的节点，而且执行 $\mathrm{OM}(t)$ 算法时所发送的消息被局限在 cluster 内部，不涉及到跨 cluster 这样的长距离路由，因此对整个网络的影响非常小。以上是我们方案的基本思想，为了简单起见，下面将结合出错节点数目，针对一个两层的网络拓扑结构对这个方案进行详细的讨论和分析。

(1) 当 $t<\min(m_i/3)$ 时　在这种情况下，网络中的 Byzantine 节点数目较少，因为对于任何一个 cluster，m_i 和 t_i 满足：

$$t_i \le t < \min(m_i/3) \le m_i/3$$

因此，无论该 cluster 中有多少 Byzantine 节点，都可以通过在 cluster 中直接执行一次 $\mathrm{OM}(t_i)$ 算法，达到清除 Byzantine 节点的目的。

(2)至少存在一个cluster$_i$，其中 $t_i > m_i/3$，但是这样的cluster总数小于 $|C|/3$ 时　如果Byzantine节点比较集中地分布在几个cluster中，会使得某个cluster$_i$内部Byzantine节点数目 $t_i > m_i/3$。这时如果沿用(1)中的方法，在每个cluster中分别执行 $\mathrm{OM}(t_i)$ 算法，那么在不符合节点数目关系的cluster$_i$中就不能达到清除Byzantine节点的目的，反之Byzantine节点还有可能联合起来将合法的节点清除。针对这种情况，我们提出：可以首先在每个cluster中选择一个节点，然后通过在CH之间进行跨cluster的 $\mathrm{OM}(t)$ 实现一种检测机制。为了说明这一方法，需首先讨论cluster的Byzantine行为这一概念。

对于一个节点，它是否具有 Byzantine 行为这一事件是确定的，也就是说它要么具有 Byzantine 行为，要么不具有 Byzantine 行为；而对于一个 cluster，因为它由多个节点组成，所以对它行为的判定需要由那些参与检测的节点决定。

定义 1　Cluster 行为。对于一个由多个节点组成的 cluster，它的行为是指在跨 cluster 进行 $\mathrm{OM}(t)$ 算法时，被检测出的行为。它是否具有 Byzantine 行为与 cluster 中参与 $\mathrm{OM}(t)$ 算法的节点的行为有关。

定义 2　Byzantine cluster。如果在一次跨cluster $\mathrm{OM}(t)$ 算法执行后，cluster$_i$的某个节点被检测出具有Byzantine行为，那么cluster$_i$就具有Byzantine行为。否则cluster$_i$视为正常的cluster。

从上面的定义可以看出，一个含有 Byzantine 节点的 cluster 存在被误判为合法 cluster 的可能，但是一个不含有 Byzantine 节点的 cluster 绝不会被误判为 Byzantine cluster。

为了简化协议，选择 CH 节点进行跨 cluster 的 $\mathrm{OM}(t)$ 算法，因此一个 cluster 的行为由 CH 节点决定。在集群形成(cluster formation)过程中每个 cluster 的 CH 是根据节点属性或者链路等信息动态地选择(视所选取的 cluster formation 算法)，cluster 内部任何一个节点都有可能被选择为 CH。考虑一个总节点数目为 m_i 的 cluster，假设内部有 t_i 个 Byzantine 节点，那么这个 cluster 在进行检测时成为 Byzantine cluster 的概率 P 等于 Byzantine 节点被选做 CH 的概率。再假设每个节点具有相同的概率被选做 CH，则有 $P=t_i/m_i$，对于

$t_i > m_i/3$ 的 cluster，$P>1/3$，也就是至少有 1/3 的概率检测出这个 cluster 具有 Byzantine 行为(如果错误节点之间协作把正确节点剔除，则 P 会逐渐趋近 1)。而对于 $t_i < m_i/3$ 的 cluster，$P<1/3$，(当剔除 Byzantine 节点后，P 逐渐趋近 0)。因此我们的方案是基于概率的。如果在一次跨 cluster 的 $\mathrm{OM}(t)$ 算法执行之后，检测出某个 cluster 具有 Byzantine 行为，则可以判定该 cluster 中 Byzantine 节点数目已经超过了 $m_i/3$，这样的 cluster 已经不能正常的使用，该 cluster 应该从整个网络中剔除。这种方法的代价是同时会剔除一部分正常节点，但是正常节点可以在跨 cluster 的 $\mathrm{OM}(t)$ 算法执行后通过身份认证机制重新加入网络。而被判定为具有 Byzantine 行为的节点如果要再次加入网络则需要通过相对更加严格的认证。

我们可以证明(参见第 5 节)，当网络中总节点数目 N 和 Byzantine 节点总数 T 满足 $T \le (1/3)^2 N$ 时，总能够保证网络中 Byzantine cluster 的数目小于 cluster 总数的三分之一，即能保证跨 cluster 的 $\mathrm{OM}(t)$ 算法能够正确执行。

(3)当Byzantine cluster数目超过 $|C|/3$ 时　在跨cluster执行 $\mathrm{OM}(t)$ 算法时，有一个重要的问题需要考虑，因为我们的方案是基于基本的 $\mathrm{OM}(t)$ 算法的，如果Byzantine cluster的数目超过了 $|C|/3$，则不满足 $\mathrm{OM}(t)$ 算法正确执行的基本假设，因此就不能通过这种方法正确检测出来。

4 算法描述

综合上述分析，我们现在结合 CBRP 提出一种基于集群的 $\mathrm{OM}(t)$ 算法。CBRP 将网络划分为一个两层的结构，因此这里提出的算法也是两层结构。这种算法有两种执行顺序。

一种是自顶向下的顺序，也就是先执行跨 cluster 的 $\mathrm{OM}(t)$ 算法，然后再在每个 cluster 内部执行 $\mathrm{OM}(t)$。这种算法仅在 Byzantine 节点相对较多且集中在 $|C|/3$ 个 cluster 内时才能够正确有效地的执行。当 Byzantine 节点分布较分散的时候，由于有一定的概率产生 Byzantine cluster 数目超过 $|C|/3$ 这种情况，会使得跨 cluster $\mathrm{OM}(t)$ 算法失效，这种方法的优点在于如果事先已知网络满足前面的假设，可以通过跨 cluster 的 $\mathrm{OM}(t)$ 算法快速定位并找出出错的 cluster，避免全部节点都参与执行 $\mathrm{OM}(t)$，降低网络负载和节点运算量；另一种则先在每个cluster内部执行 $\mathrm{OM}(t)$ 算法，然后再执行跨 cluster 的 $\mathrm{OM}(t)$ 算法。这样在第一步就可以把那些 $t_i < m_i/3$ 的cluster中的Byzantine节点清除，剩下的cluster要么是 $t_i > m_i/3$ 的cluster，要么是不含Byzantine节点的cluster，这样可以保证跨cluster的 $\mathrm{OM}(t)$ 算法能够顺利执行。因为不能预知cluster$_i$中 t_i 的具体值，因此在cluster内部执行 $\mathrm{OM}(t)$ 算法时需要按最坏的情况处理，即在每个cluster分布式执行 $\mathrm{OM}\left(\lfloor (m_i-1)/3 \rfloor\right)$ 算法，对于跨cluster的情况则需要执行 $\mathrm{OM}\left(\lfloor |C|/3 \rfloor\right)$。另外，在执行跨cluster的 $\mathrm{OM}(t)$ 算法时，我们约定选取ID最小的CH作为起始节点，这样做可以防止重复执行跨cluster的 $\mathrm{OM}(t)$ 算法。算法描述如下：

(1)对于cluster$_i$ //执行Cluster内部的 $\mathrm{OM}(t)$ 算法；

(a)CH$_i$ 向它的 neighbor table 中所有以 CH$_i$ 作为 Host Cluster的节点发送初始消息 v；　//为了简化算法，选取CH作为起始节点；

(b)Cluster$_i$的CM$_i$收到CH$_i$的消息 v 后，以 v 为初始值，执行 $\mathrm{OM}\left(\lfloor (|m_i-1|/3) \rfloor\right)$。向cluster内其它 m_i-2 个CM发送消息 v，如果没有收到则发送默认消息。

(c)对于CM$_j$，如果 $j \ne i$，那么使用 v_i 作为CM$_j$从CM$_i$收到的消息值，如果没有收到这样的消息，就使用默认值。然后结点CM$_j$用值 $\mathrm{majority}(v_i,\cdots,v_{n-1})$ 作为CH$_i$发送的消息。

(d)Cluster$_i$中的CM$_j$通过最终收到的 v_i，判断并记录哪些节点具有Byzantine行为，拒绝与该节点的数据连接。(如果CH$_i$被清除，根据cluster formation算法重新选出一个CH$_i$)

(e)CH$_i$向manet中所有其它CH广播自己状态(已经完成对cluster$_i$内部Byzantine节点清除)

(2)当 ID 最小的 CH 收到其它所有 CH 在第(e)步的消息后，生成一个初始消息，同其它的 CH 执行 $\mathrm{OM}\left(\lfloor (|C|-1)/3 \rfloor\right)$。检测 Byzantine cluster。//执行跨 cluster 的 $\mathrm{OM}(t)$ 算法；

(3)对于所有的CH$_i$

(a)如果CH$_i$是Byzantine节点，根据cluster formation算法在正常节点中随机选择一个作为新的CH$_i$；

(b)CH$_i$通知cluster$_i$内的GW节点，断开与Byzantine cluster的连接。//清除Byzantine节点。

在执行第 2 步时，至少有 33%的概率能够检测出Byzantine集群，但这仍然比较低，可以作如下改进：当CH$_i$收到一个跨cluster的 $\mathrm{OM}\left(\lfloor (|C|-1)/3 \rfloor\right)$ 消息时，它随机转发给cluster$_i$中一个CM节点，由该CM代表CH$_i$执行跨cluster的 $\mathrm{OM}\left(\lfloor (|C|-1)/3 \rfloor-1\right)$ 算法，并对该消息的加上本cluster的标识用于区分，而不再使用节点标识来区分消息。

在执行跨 cluster 的 $\mathrm{OM}\left(\lfloor (|C|-1)/3 \rfloor\right)$ 算法时每个 cluster 大约需要处理的消息数目为：

$$\mathrm{num} = (|C|-1)\cdot(|C|-2)\cdots\left(\lfloor 2|C|/3-1 \rfloor\right)$$

其中只要有一条消息出错都可以认为该 cluster 中存在 Byzantine 节点，一条消息是否出错的概率为 p，假设发送每条消息这一事件是统计独立的，则该 cluster 不被检测出来的概率为 $p_{\mathrm{esc}} = (1-p)^{\mathrm{num}}$。因此，在执行跨 cluster 的 $\mathrm{OM}\left(\lfloor (|C|-1)/3 \rfloor\right)$ 算法时，成功检测出一个 Byzantine 集群的概率为 $P = 1-p_{\mathrm{esc}} = 1-(1-p)^{\mathrm{num}}$。又因为在这样的 cluster 中 $p>1/3$，所以 $P = 1-(1-p)^{\mathrm{num}} > 1-(2/3)^{\mathrm{num}}$。比如，当 $|C|=4$ 时，$\mathrm{num}=3$，则 $P>0.701$，而当 $|C|=5$ 时，$P>0.992$，当 $\mathrm{num}\to\infty$ 时 $P\to 1$。

5 相关证明

定理 1　若 Byzantine 节点总数 T 和网络中节点总数 N 满足 $T \le (1/3)^2 N$，则 Byzantine cluster 的数目小于 cluster 总

数的 1/3。

证明 假设在一个有 N 个节点的自组网中，存在 T 个 Byzantine 节点，如果在集群形成过程中共生成了 $|C|$ 个 cluster，cluster 平均大小为 $\bar{m}$，则 $N=\bar{m}|C|$，假设有 k 个 Byzantine cluster，则有 k 个 cluster 中 Byzantine 节点数目不小于该 cluster 中总节点数目的 1/3。要想获得最多的 Byzantine cluster，必须尽量利用每个 Byzantine 节点，而且每个 cluster 中 Byzantine 节点数目尽量要小，因此 T 个 Byzantine 节点最多产生 $3T/\bar{m}$ 个 Byzantine cluster，因此有 $k<|C|/3 \Leftrightarrow 3T/\bar{m}<|C|/3 \Leftrightarrow T<\bar{m}|C|/9=(1/3)^2 N$。　证毕

定理 2　如果采用本算法，则每个节点需要发送的消息数目为 $O\left(N/|C|\right)^{N/3|C|+1}$。

证明 每个节点发送的消息数目主要取决于执行 cluster$_i$ 内部的 $\mathrm{OM}(t_i)$ 算法时所发送的消息数目，从第 3 节的分析可知，这部分的消息数目为 $O\left(\left(N/|C|-1\right)\cdot\left(N/|C|-2\right)\cdots\left(N/|C|-t_i-1\right)\right)$ 因为 t_i 不能预先得知，故在执行cluster内部的 $\mathrm{OM}(t_i)$ 时，选取 $t_i=m_i/3=N/3|C|$，所以每个节点需要发送的消息数目为 $O\left(\left(N/|C|-1\right)\cdot\left(N/|C|-2\right)\cdots\left(N/|C|-N/3|C|-1\right)\right)=O\left(N/|C|\right)^{N/3|C|+1}$。　证毕

如果采用基本 $\mathrm{OM}(t)$ 算法，则每个节点需要发送的消息数目为 $O\left(N^{T+1}\right)$。因此本算法显著减少了每个节点需要发送的消息数目。

6　结束语

本文提出了一种在 manet 中使用集群技术来解决 BGP 的基本算法。结合 CBRP 详细阐述了两层网络拓扑结构下的检测算法以及这种算法的优势；本文还讨论了算法执行所需的条件以及在执行跨 cluster 的 $\mathrm{OM}(t)$ 算法时的检测概率，采用这种跨 cluster 的 $\mathrm{OM}(t)$ 算法，至少有 33%的概率能够检测出具有 Byzantine 行为的集群，若采用改进机制，通过恰当的选取集群大小和数量，则检测概率可以趋于 1。算法分析证明，采用基于集群的 $\mathrm{OM}(t)$ 算法时每个节点所发送消息数目为 $O\left(N/|C|\right)^{N/3|C|+1}$，远小于采用基本的 $\mathrm{OM}(t)$ 时的 $O\left(N^{T+1}\right)$。

为了提高算法的效率和稳健性还有许多工作需要做，比如如何有效地划分集群使得每个集群大小近似相等，这一问题对本文介绍的算法有比较大的影响；其次，还可结合 Byzantine 节点的概率分布情况进一步讨论它与总节点数目的关系，使得本算法能够适应更多的网络情况；最后，由于本算法基于最基本的 $\mathrm{OM}(t)$ 算法，因此还有很大的优化余地。这些都将作为我们进一步的研究工作。

参 考 文 献

[1] Zhou L, Haas Z J. Securing Ad hoc networks. *IEEE Network*, 1999, 13(6): 24−30.

[2] Luo H Y, Lu S W. Ubiquitous and robust authentication services for Ad hoc wireless networks. Technical Report TR-200030, Dept. of Computer Science, UCLA. 2000.

[3] Lamport L, Shostak R, Pease M. The Byzantine generals problem. *ACM Trans. on Programming Languages and Systems*, 1982, 4(3): 382−401.

[4] Dolev D, Strong H R. Authenticated algorithms for Byzantine agreement. *SIAM Journal of Computation*, 1983, 12(4): 656−666.

[5] Dolev D, Reischuk R, Strong H R. Early stoping in Byzantine agreement. *Journal of the ACM*, 1990, 37(4): 720−741.

[6] Siu Hin-Sing, Chin Yeh-Hao, Yang Wei-Pang. Byzantine agreement in the presence of mixed faults on processors and links. *IEEE Trans. on Parallel and Distributed Systems*, 1998, 9(4): 335−345.

[7] Galil Z, Mayer A, Yung Moti. Resolving message complexity of Byzantine agreement and beyond. 36th Annual Symposium on Foundations of Computer Science, 1995: 724−733.

[8] Burmester M, Desmedt Y G. Secure communication in an unknown network with Byzantine faults. *Electronics Letters*, 1998, 34(8): 741−742.

[9] Wang S C, Yang W P, Cheng C F. Byzantine agreement on mobile Ad-hoc network. IEEE International Conference on Networking, Sensing and Control, 2004, vol.1: 52−57.

[10] Bergano M R, *et al*.. System design specification for mobile multimedia wireless network (MMWN) (draft), DARPA project DAAB07-95-C-D156, Oct.1996.

[11] Atsushi Iwata, Ching C C, Pei, G Y, Gerla M, Chen T W. Scalable routing strategies for Ad hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, 1999, 17(8): 1369−1379.

[12] Jiang M L, Li J Y, Tay Y C. Cluster based routing protocol (CBRP) fuctional specification, draft-ietf-manet-cbrp-spec-01.txt, Work in progress, Aug. 1999.

[13] Poosarla R, Deng Hongmei, Ojha A, Agrawal D P. A cluster based secure routing scheme for wireless Ad hoc networks. IEEE International Conference on Performance, Computing, and Communications, 2004: 171−175.

[14] Bechler M, Hof H. A cluster-based security architecture for Ad hoc networks, INFOCOM 2004, Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, 7−11 March, 2004, vol.4: 2393−2403.

刘　洋：　男，1981 年生，硕士生，研究方向为无线自组织网络.

俞能海：　男，1964 年生，教授，研究方向为宽带无线网络与多媒体通信.

冯现磊：　男，1981 年生，硕士生，研究方向为无线自组织网络.