基于门限的移动自组网动态地址配置方案 *

易双凤[1], 钱海峰[1], 周 渊[2]
(1. 华东师范大学 计算机科学技术系, 上海 200241; 2. 国家计算机网络应急技术处理协调中心, 北京 100029)

摘 要: 在对抗性的 MANETs 中,由于存在恶意节点,节点的地址自配置无法有效执行,为了对抗恶意节点的相关安全攻击,提出了基于门限的动态地址配置方案(Threshconf)。该方案主要通过门限签名为地址配置协议提供了访问控制安全机制,保证地址自配置协议在对抗性的 MANETs 中有效执行;同时与其他几种地址配置方案进行了比较分析。

关键词: 移动自组网; 动态地址配置; 恶意节点; 门限签名; 访问控制; Threshconf
中图分类号: TP393 **文献标志码**: A **文章编号**: 1001-3695(2010)08-3090-06
doi:10.3969/j.issn.1001-3695.2010.08.075

Dynamic address configuration based on threshold in mobile Ad hoc network

YI Shuang-feng[1], QIAN Hai-feng[1], ZHOU Yuan[2]
(1. *Dept. of Computer Science & Technology, East China Normal University, Shanghai* 200241, *China*; 2. *National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing* 100029, *China*)

Abstract: Due to the vicious actions of malicious nodes in an adversarial network, address auto-configuration couldn't be carried out with effect. To thwart corresponding security attacks about address auto-configuration in MANETs, this paper presented a dynamic address configuration based on threshold in an adversarial MANETs, referred to Threshconf. Threshconf with access control constructed by using threshold signature could prevent against correlative networks and guarantee the protocol could be executed effectively in hostile networks. Finally, also discussed the performance and security of Threshconf.

Key words: mobile Ad hoc network; dynamic address configuration; malicious node; threshold signature; access control; Threshconf

移动自组网(MANETs)是由对等的移动节点组成的一个临时性的多跳自组织无线网络。网络节点的地址配置是实现网络互通的关键技术。由于固有特性,如开放的无线介质、无中心控制和基础设施等,传统的有线网络中的地址配置方案无法直接应用于 MANETs。此外,MANETs 易遭受安全攻击,尤其是存在恶意节点的环境中,安全问题更为严重。因此,必须要有适合 MANETs 的安全地址配置方案。

近十年来,许多 MANETs 地址配置方案被提出[1,2],如 MANETconf[3]、DACP[4]、Buddyconf[5]。但大多的地址配置方案都假设网络中的节点可信任,不存在恶意节点。在安全方面,由于门限密码将秘密进行多方共享,可应用于多个成员共同管理资源的场景,许多门限安全方案被提出,提供访问控制等安全服务。本文探讨存在恶意节点的 MANETs 中的地址配置方案,即对抗性 MANETs 中的地址配置方案,研究可提供访问控制的地址配置协议,其主要困难在于如何提供适当的安全,使得地址配置协议可对抗相应的网络攻击,但同时又能保证地址配置协议的性能。因此,提出了基于门限的地址自动配置(简记为 Threshconf),使得地址配置协议既可对抗相应的网络攻击,又尽量保持高性能。

1 背景及相关工作

现有一些解决方案在地址配置过程中不支持安全性,且与 MANET 地址自动配置的安全性相关研究较少,提出的解决方案也较少[6~10]。一个地址配置方案应处理网络初始化、节点加入、节点离开(正常和非正常)、网络分割和网络合并[1,2]。

1.1 不支持安全性的地址配置方案

下面主要介绍和分析现有的三个不支持安全性的地址配置方案,即 MANETconf[3]、DACP[4]、Buddyconf[5]。

在 MANETconf 地址配置中,网络的每个节点都维护一张已分配的地址列表。当新节点加入时,其某一邻居节点根据存储地址列表选取一个可用地址。但为了避免同时加入网络的节点地址冲突,仍需执行冲突检测。为了更新各节点维护的地址列表,节点需定期广播其保存的地址列表。每个节点维护一个网络 ID,当节点收到不同的网络 ID 时,可判定网络合并,而当收不到所有节点的地址消息时,即可判定网络分割。

在文献[4],DACP 引入了动态的地址机构(address authority,AA),用于维护网络的状态信息,如已分配的 IP 等。AA 是动态地选取网络中的一个节点担任的。新加入的节点随机地自配置一个 IP 和一个候选 IP,进行主动重复地址检测。若未发现地址重复,则在 AA 处登记。AA 周期性广播带有网络标志的广告消息,当节点一段时间没有收到广告消息,则认为被分割,需要重新选择 AA,并且重新注册地址;若 AA 收到其他网络的 AA 发送的广告,则说明发生了网络合并,则两个网络

收稿日期: 2009-12-29; **修回日期**: 2010-03-01 **基金项目**: 国家自然科学基金资助项目(60703004);国家教育部博士点基金资助项目(20070269005)

作者简介:易双凤(1986-),女,江西宜春人,硕士研究生,主要研究方向为网络安全(sherry_1@live.cn);钱海峰(1977-),男,副教授,博士,主要研究方向为信息安全与密码学;周渊(1972-),男,高级工程师,博士,主要研究方向为信息安全、软件确保.

的 AA 负责检测重复地址,然后通过泛洪通知发生地址重复的节点,重复地址的节点重新选择地址。

Buddyconf 方法的基本思想是每个节点管理一个未分配的 IP 地址集。新加入的节点广播消息,寻找已配置的邻居节点,选定其中一个作为代理。代理节点为新加入的节点配置一个 IP 地址,同时将其管理的未分配的 IP 地址集的一半交由新节点管理。节点离开时,将分配的 IP 和管理的 IP 地址集返回给网络中的节点。网络维护一个网络 ID 来检测分割和合并。节点需周期性地泛洪,进行同步过程来检测节点地址间的泄漏。

以上三个协议中,MANETconf 和 DACP 都基于地址冲突检测(duplicate address detection,DAD),与先验式路由协议能很好兼容,两者的维护开销、通信开销和配置时延都很高;Buddyconf 无须冲突检测,通信开销和配置时延相对较低。三者都需周期性地广播,进行网络同步,而 Buddyconf 只与邻居节点进行广播,通信开销相对较低,且能较好地适应网络的扩展性。在 MANETconf 和 Buddyconf 中,每个节点执行相同的协议,无须选举节点执行不同的操作,新节点的地址配置依赖已在网络的邻居节点。DACP 需选举 AA,AA 负载较大,且当 AA 失效时,损失较大;新节点是主动进行地址配置,不依赖其他节点。

1.2　安全的地址配置方案

如表 1 所示,存在六种与地址配置相关的攻击,即地址欺骗攻击、地址耗尽攻击、冲突地址攻击、假冲突地址攻击、Sybil 攻击和流量过载 DoS 攻击[11,12]。现有的支持安全性的地址配置方案可分为基于自验证[6]、质询响应[7,9]、信任模型[10]、在线验证/证书[7,8]四类。

表 1　与地址配置相关的攻击

攻击	说明
地址欺骗	恶意节点伪装成一个已分配的或空闲的 IP 地址。当伪装成一个已分配的 IP 地址时,可窃听或截断被伪装节点的所有消息。当伪装一个空闲的 IP 地址加入网络时,可获取更多的消息,从而发起主动攻击,如 DoS 攻击
地址耗尽	恶意节点申请尽可能多的地址而耗尽地址空间,将地址分配给不存在的节点。当地址空间耗尽后,其他新加入的节点就无法配置地址
冲突地址	恶意节点分配一个已用的地址给新加入的节点,使得网络中出现冲突地址
假冲突地址	在冲突地址检测时,恶意节点假称候选的 IP 地址已分配,使得新节点重新选址一个 IP 地址
Sybil	一个 Sybil 节点捏造一个新的身份或窃取一个合法节点的身份,从而获得多个 IP 地址,方便发动各种攻击
DoS	恶意节点可反复地同时发送地址配置协议的消息,如同时发送大量的地址请求消息,为不同的候选 IP 地址发送重复地址检测消息,导致网络的流量过载

在文献[6],每个节点随机产生一个公私钥对后,使用一个单陷门的哈希函数将节点的地址与其公钥进行绑定,即使用哈希函数和公钥产生其地址。为了不产生重复地址,需要进行重复地址检测。该方案可抵抗地址耗尽攻击和假冲突地址攻击,但只处理了节点加入网络,没有对网络初始化、节点离开网络以及网络分割/合并进行相应处理。

在文献[9],假设每一个节点拥有一个事先分配的密钥,使用单陷门哈希函数、时戳和该密钥来为地址配置协议提供安全性。该协议的通信开销和计算开销都相对较小,但安全性很弱,一旦事先分配的密钥泄漏,恶意节点就可进行各种攻击了。

在文献[7],新节点需与 K 位一跳邻居进行互验证,K 大于或等于某个阈值。通过验证后,从一跳邻居中选择一位进行地址分配。其缺点是,无法抵抗耗尽攻击。某个新节点通过互验证阶段后,可同时向多位一跳邻居进行地址分配请求。

通过以上方案的分析并考虑相应的安全攻击,在对抗性 MANETs 中,一个较佳的地址配置方案应满足以下要求:a)地址配置不存在冲突,网络中不存在重复地址;b)节点在网络时,分配一个 IP 地址;当节点离开网络时,IP 地址需回收;c)当没有可用的 IP 地址时,新的节点不能加入网络;d)能处理网络分割和网络合并;e)可抵抗相应的网络攻击;f)地址配置的通信开销和配置开销应尽可能小;g)对路由协议具有较好的兼容性。

2　基于门限的地址配置方案

2.1　GDH 门限签名方案

门限签名是秘密共享技术与数字签名技术相结合的一种密码体制,它使群体中的某些成员或给定的子集可代表整个群体来签名。自 1990 年 Desmedt 等人[13]提出门限签名后,许多门限方案被相继提出[14~16]。2001 年,Boneh 等人[15]提出了基于双线性对的签名方案,称为短签名方案,具有签名长度短、计算效率和通信效率高的优点。Boldyreva[14]在 2003 年提出的基于 GDH 签名的门限签名方案,以其具有通信量和计算量小的特点比较引人注目,本文采用该门限签名方案。

GDH 门限签名基于双线性对。令 G_1 和 G_2 分别为阶数是素数 q 的加群和乘群,g 为 G_1 群的生成元。一个双线性对 $\hat{e}:G_1\times G_2\to G_2$ 满足下列性质:

a)双线性性。$\hat{e}(aP,bP')=\hat{e}(P,P')^{ab}$,对所有 $P,P'\in G_1$ 和所有 $a,b\in Z$。

b)非退化性。若 $\hat{e}(P,P')=1,\forall P'\in G_1$,则 P 是群 G_1 的单位元。

c)可计算性。存在有效算法可以计算 $\hat{e}(P,P'),\forall P,P'\in G_1$。

令 G 是一个阶为素数 q 的 GDH 群,g 为其生成元。$[\{0,1\}^*\to G^*]$ 为一哈希函数族。其中任意一个哈希函数均可将任意长度的字符串映射到群 G^*,从中随机选择 H。令 I 为 (g,q,H)。GDH 门限签名方案由以下算法 (K,SS,VS,SG,VG) 组成:

a)分布式密钥生成算法 $K(I)$。通过 (n,t) - DKG[17] 由 n 个成员共同生成私钥 sk 和相对应的公钥 $Y\leftarrow G^{sk}$。每个成员都不知道私钥 sk,只持有一个私钥分量 ss_i,并公布其私钥分量的一个承诺 $C_i=g^{ss_i}$,但当至少 t 个成员联合其私钥分量,可恢复出密钥 sk。如成员 $1\sim t$ 联合其私钥分量时,$sk=\sum_{i=1}^{t}ss_i\times L_i$。其中 $L_i=\prod_{j=1,j\neq i}^{t}\frac{j}{j-i}$,是公开的参数。计算,返回公钥 $pk=Y$、L_i 和各私钥分量的承诺 C_i。

b)部分签名算法 $SS(I,ss_i,M)$。计算 $\sigma_{ss_i}=H(M)^{ss_i}$,返回 (M,σ_{ss_i})。

c)验证部分签名算法 $VS(M,C_i,\sigma_{ss_i})$。若 $\hat{e}(g,\sigma_{ss_i})=\hat{e}(C_i,H(M))$,则签名合法,返回 1,否则签名不合法,返回 0。

d)合成群签名算法 $SG(I,\sigma_{ss_i},M)$。计算 $\sigma=\prod_{i=1}^{t}(\sigma_{ss_i})^{L_i}$,返回 (M,σ)。

e)验证群签名 $VG(M,I,\sigma)$。若 $e(g,\sigma)=e(Y,H(M))$,则签名合法,返回 1,否则返回 0。

2.2　假设与符号

1)门限签名

a)(t,l),采用(t,l)门限方案。在本文中l等于网络节点数n,至少t个群密钥分量持有者可恢复群密钥,或生成合法的群签名。$t \leqslant l$,且t固定,大小不会太大。

b)$(q,g,G_0,G_1,H,\hat{e})$,网络的安全参数。可由网络设计者(不参与网络)提供,发布在知名网站上,也可由网络中的节点通过广播协商。G_0,G_1是阶为大素数q的群,g为群G_0的生成元;$H:\{0,1\}^* \rightarrow z_q^*$是一个抗碰撞的哈希函数;$\hat{e}:G_0 \times G_0 \rightarrow G_1$为一个安全的双线性对[15]。

c)(Gpk,Gsk),网络的群公私钥对,由进行网络中初始化的节点通过 Pedersen 提出的可验证的秘密共享 VSS[18] 方案共同生成的,$Gpk = g^{Gsk}$。

d)L_i,$L_i = \prod_{j=1,j\neq i}^{t} \frac{j}{j-i}$,是公开的参数。

e)(ss_i, g^{ss_i})。节点P_i所持有的群私钥分量ss_i及相应的验证公钥g^{ss_i}。当门限为t时,$Gsk = \sum_{i=1}^{t} ss_i \times L_i$。

f)$\mathrm{sig}_{ss_i}(m)$。节点P_i对消息m的部分群签名,$\mathrm{sig}_{ss_i}(m) = H(m)^{ss_i} (\mathrm{mod}\ p)$。

g)$V(g,g^{ss_i},H(m),\mathrm{sig}_{ss_i}(m)) \stackrel{?}{=} 1$。验证节点$P_i$生成的部分群签名。若$\hat{e}(g,\mathrm{sig}_{ss_i}(m)) = \hat{e}(g^{ss_i},H(m))$,则验证成功,结果为 ture;否则结果为 false。

h)$\mathrm{sig}_{Gsk}(m)$。由至少t个成员共同生成的对消息m的合法群签名。当门限为t时,$\mathrm{sig}_{Gsk}(m) = \prod_{i=1}^{t} (\mathrm{sig}_{ss_i}(m))^{L_i}$。

i)$V(m,Gpk,\mathrm{sig}_{Gsk}(m)) \stackrel{?}{=} 1$。验证对消息$m$的群签名,若签名合法,则返回 true,否则返回 false。

j)GMC_i。节点P_i的群证书,$\mathrm{GMC}_i = H(\mathrm{IP}_i,\mathrm{ID}_i)^{Gsk}$,对节点$P_i$的一个群签名。当节点$P_i$加入网络进行地址配置时,网络中至少有$t$个节点是共同颁发的。

2)公钥数字证书 PKC

a)(pk_i,sk_i),节点P_i公私钥对,$pk_i = g^{sk_i}$。假设事先存在一个 CA,为每个节点颁发公钥数字证书 PKC_i,包括节点标志 ID_i。每个节点拥有一对公钥pk和私钥sk分别用于消息验证和签名,私钥不被别的节点获取。

b)$Sig_{sk_i}(m,r)$。节点P_i对消息m和随机量r的签名。

c)$\sigma_{sk}(m)/\sigma_{sk}$。用$sk$对消息$m$签名,$\sigma_{sk}(m) = H(m)^{sk}$。

d)$v_{pk}(\sigma,m)$。用pk验证签名,当$\hat{e}(\sigma,g) = \hat{e}(pk,H(m))$时,通过验证,返回 1,否则返回 0。

e)S,两个节点共同的对称密钥。节点P_i和P_j的对称密钥$S_{i,j} = (pk_j)^{sk_i} = (pk_i)^{sk_j} = (g)^{sk_i \times sk_j}$[19]。

f)$E_S(m)$。对称加密消息m,密钥为S。

g)$D_S(m)$。对称解密消息m,密钥为S。

3)IP 地址和网络 ID

a)IP。本文考虑独立的 MANETs,不与外界网络连接,因此所有的 IP 地址均可用来进行地址配置。每个节点P_i分配惟一的 IP,并关联一个生存期 ttl。

b)$B_{\mathrm{IP\_available}}$,每个节点负责管理网络中的部分可用地址集。$B_{\mathrm{IP_available}}$中每个 IP 地址与该节点的 IP 地址的差值称为距离,不大于$B_{\mathrm{IP_available}}$的大小 size_B。每个节点的 IP 地址 IP_i 在 $B_{\mathrm{IP\_available}} \cup \{\mathrm{IP}_i\}$ 中值最大或最小,最大时,记 flag = 1,最小时,记 flag = 0。因此无须记录每个可用 IP 地址,只记录$B_{\mathrm{IP_available}}$的大小和 flag,即 $B_{\mathrm{IP\_available}} = (\mathrm{size}_B, \mathrm{flag})$。

c)$\mathrm{List}_{\mathrm{allocalted}}$。每个节点也维护一张已占用的 IP 地址列表 $\mathrm{list}_{\mathrm{allocated}}$。$\mathrm{List}_{\mathrm{allocated}}$中每一项的格式如下:

$$(\mathrm{IP}_i, \mathrm{ID}_i, \mathrm{GMC}_i, B_{\mathrm{IP\_available}})$$

d)Netid。网络 ID,是全局惟一的标志符。

e)$a \oplus b$。a和b进行异或运算。

2.3 网络初始化

网络初始化由一个信任节点 TD 和其他至少t个节点共同完成,其过程类似于 Shamir 提出的秘密共享方案[20]。完成网络初始化后,TD 不需要了。TD 和$n-1(n>t)$个节点共同初始化网络。首先,TD 随机选择 Netid、网络群密钥Gsk和一个次数为$t-1$的多项式$f(z)$,$f(z)$满足$f(0) = Gsk$,并计算相应的群公钥$Gpk = g^{Gsk}$。同时 TD 将所有的可用地址均分成n个连续的地址集 $\mathrm{block}_j (j = 1,2,\cdots,n)$,每个集合大小为 $\mathrm{size}_{\mathrm{block}}$,然后,每个节点$P_i (i = 1,2,\cdots,n-1)$向 TD 请求加入网络。TD 验证节点$P_i$的公钥数字证书$PKC_i$后,将地址集 block_i 中值最小的地址 IP_i 分配给节点P_i,block_i 中剩下的可用地址由节点P_i管理,即节点P_i负责管理的可用地址集 $B_{\mathrm{IP\_available}} = (\mathrm{size}_{\mathrm{block}} - 1, \mathrm{flag} = 0)$,并为节点$P_i$计算其持有的群密钥分量$ss_i = f(\mathrm{ID}_i)$和群证书 $\mathrm{GMC}_i = H(\mathrm{IP}_i, \mathrm{IDM}_i)^{Gsk}$。

2.4 节点加入网络

假设网络中已有n个节点。一个新节点P_{n+1}获得 Netid 后,P_{n+1}与其已在网络中的邻居节点$P_i (i = 1,2,\cdots,n)$共同协作进行地址配置,步骤如下:

a)节点P_{n+1}随机选择一个 IP 地址,作为临时源地址 $\mathrm{IP}_{\mathrm{source}}$,然后广播地址请求 AREQ。AREQ 的格式如下:

$$\langle \mathrm{AREQ}, \mathrm{IP}_{\mathrm{source}}, r, \sigma_{sk_{n+1}} = H(\mathrm{IP}_{\mathrm{source}}, r)^{n+1}, \mathrm{PKC}_{n+1} \rangle$$

其中:r是随机数,用于抵抗重放攻击。

同时启动申请加入计时器 $\mathrm{timer}_{\mathrm{AREQ}}$。若 $\mathrm{timer}_{\mathrm{AREQ}}$超时后,未接收到至少$t$份合法的地址应答消息 AREP,则重发若干次,直到收到回复,否则停止操作。

b)当已在网络中接收到 AREQ 的节点P_i以一定的概率做如下操作响应。首先验证P_{n+1}的数字证书,若验证失败,丢弃该消息并停止操作;然后验证 $v_{pk_{n+1}}(\sigma_{sk_{n+1}}(\mathrm{IP}_{\mathrm{source}}, r)) \stackrel{?}{=} 1$,若不等,丢弃该消息并停止操作;接着检查 $B_{\mathrm{IP\_available}}$ 是否为空,若 $B_{\mathrm{IP\_available}}$ 为空,丢弃该消息并停止操作;否则,从 $B_{\mathrm{IP\_available}}$ 中选定与 IP_i 距离最大的 IP 地址 $\mathrm{IP}_{\mathrm{tentative}}$ 后,返回地址应答消息 AREP。AREP 的格式如下:

$$\langle \mathrm{AREP}, \mathrm{IP}_{\mathrm{tentative}}, E_{s_{j,n+1}}(\mathrm{size}, \mathrm{flag}', r'), \sigma_{sk_i}, \mathrm{PKC}_i, \mathrm{IP}_i, \mathrm{GMC}_i \rangle$$

其中:$\mathrm{flag}' = 1 \oplus \mathrm{flag}$;当 size_B 为奇数时,$\mathrm{size} = \mathrm{size}_B/2$,否则 $\mathrm{size} = \mathrm{size}_B/2 - 1$;$\sigma_{sk_i} = H(\mathrm{IP}_{\mathrm{tentative}}, \mathrm{size}, r')^{sk_i}$。同时启动计时器 $\mathrm{timer}_{\mathrm{AREP}}$。$\mathrm{Timer}_{\mathrm{AREP}}$超时前,不再为其他节点进行地址配置,直到计时器超时或收到关于 $\mathrm{IP}_{\mathrm{tentative}}$ 的 AACK 消息。若计时器超时前,收到关于 $\mathrm{IP}_{\mathrm{tentative}}$ 的 AACK 消息,停止计时,更新 $B_{\mathrm{IP\_available}}$ 和 $\mathrm{list}_{\mathrm{allocated}}$。

1(AREQ) P_{n+1}广播:

$$P_{n+1} \xrightarrow{\langle \mathrm{AREQ}, \mathrm{IP}_{\mathrm{source}}, r, \sigma_{sk_{n+1}}, \mathrm{PKC}_{n+1} \rangle} \{P_1, \cdots, P_n\}$$

2(AREP)每个 $P_i (i \in [1,t'])$ 且 $t \leqslant t' \leqslant n$)选定试探地址 $\mathrm{IP}_{\mathrm{tentative}}$ 后,返回给 P_{n+1}:

$$P_{n+1} \xleftarrow{\langle \mathrm{AREP}, \mathrm{IP}_{\mathrm{tentative}}, E_{S_{i,n+1}}(\mathrm{size}, \mathrm{flag}', r'), \sigma_{sk_i}, \mathrm{PKC}_i, \mathrm{IP}_i, \mathrm{GMC}_i \rangle} \{P_1, \cdots, P_{t'}\}$$

3(AACK) P_{n+1}选择试探地址 $\mathrm{IP}_{\mathrm{tentative}}$ 后,从 t'个应答者选择t个,将其 ID 组成

SL_{n+1},发送:

$$P_{n+1} \xrightarrow{\langle \text{AACK}, \text{IP}_{n+1}, \text{size}, \text{flag}', r'', SL_{n+1}, \sigma_{sk_{n+1}} \rangle} \{P_1, \cdots, P_t\}$$

4 (AACKREP)每个 P_j:

①计算部分群签名 $\sigma_{ss_j} = H(\text{IP}_{n+1}, \text{ID}_{n+1})^{ss_j}$

②计算隐藏的私钥分量 $\tilde{x}^j_{n+1} = ss_j \cdot L_j(\text{ID}_{n+1})$

③加密 σ_{ss_j}和 $\tilde{x}^j_{n+1}$

$$P_{n+1} \xleftarrow{\langle \text{AACKP}, E_{S_{n+1},j}(\sigma_{ss_j}, \tilde{x}^j_{n+1}), r''', \sigma_{sk_j} \rangle} \{P_1, \cdots, P_{t'}\}$$

5 P_{n+1}

①计算群证书 $\text{GMC}_{n+1} = \sum_{j=1}^{t} \sigma_{ss_j} \cdot L_j(0)$

②计算私钥分量 $ss_{n+1} = \sum_{j=1}^{t} \tilde{x}^j_{n+1}$

c)P_{n+1}接收到 $t'(t \leqslant t' \leqslant n)$份 AREP,验证各个 AREP 和解密消息,选择其中的相应 size 值最大的一个 $\text{IP}_{\text{tentative}}$作为其地址 IP_{n+1},则对应的 $B_{\text{IP\_available}} = (\text{size}, \text{flag}')$,并且从 AREP 发送者中选择 t 个节点,这些节点的 ID_j 组成一个列表 SL_{n+1},然后广播地址确认消息 AACK。AACK 的格式如下:

$$\langle \text{AACK}, \text{IP}_{n+1}, \text{size}, \text{flag}', r'', SL_{n+1}, \sigma_{sk_{n+1}} \rangle$$

其中:$\sigma_{sk_{n+1}} = H(\text{IP}_{n+1}, \text{size}, \text{flag}', r'', SL_{n+1})^{sk_{n+1}}$,$r''$为随机数。为了描述简单,令 SL_{n+1}由$(\text{ID}_1, \text{ID}_2, \cdots, \text{ID}_t)$组成。

d)SL_{n+1}中的节点 P_j 收到 AACK 后,首先根据 $\text{list}_{\text{allocated}}$的记录检查 IP_{n+1}、size 和 flag′的正确性,若不正确,丢弃消息并停止操作;然后验证 $\sigma_{sk_{n+1}}$,若验证失败,丢弃消息并停止操作。接着计算部分群签名:

$$\sigma_{ss_j} = H(\text{IP}_{n+1}, \text{ID}_{n+1})^{ss_j}$$

将私钥分量按如下公式进行隐藏:

$$\tilde{x}^j_{n+1} = ss_j \times L_j(\text{ID}_{n+1})$$

其中:$L_j(\text{ID}_{n+1}) = \prod_{i=1, j \neq j}^{t} \frac{\text{ID}_{n+1} - \text{ID}_i}{\text{ID}_j - \text{ID}_i}$,发送确认应答 AACKP 给 P_{n+1}。AACKP 格式如下:

$$\langle \text{AACKP}, E_{s_{j,n+1}}(\sigma_{ss_j}, \tilde{x}^j_{n+1}), r''', \sigma_{sk_j} \rangle$$

其中:$\sigma_{sk_j} = H(E_{s_{j,n+1}}(\sigma_{ss_j}, \tilde{x}^j_{n+1}), r''')^{sk_j}$,$r'''$为随机数。

e)节点 P_{n+1}接收到 AACKP 后,验证σ_{sk_j}并解密,得到$(\sigma_{ss_j}, \tilde{x}^j_{n+1})$。计算 $\text{GMC}_{n+1} = \sum_{j=1}^{t} \sigma_{ss_j} \times L_j(0)$,$ss_{n+1} = \sum_{j=1}^{t} \tilde{x}^j_{n+1}$。最后用群公钥验证 GMC_{n+1}。

至此加入了网络,最后,将其地址配置信息进行全网泛洪。

2.5　节点离开网络

节点离开网络有两种情形:a)节点发送了地址释放消息 ADD_CLEAN 后离开网络;b)节点离开网络,但没有发送地址释放消息。当节点离开网络,广播地址释放消息 ADD_CLEAN。ADD_CLAEN 格式如下:

$$\langle \text{ADD\_CLEAN}, \text{IP}_i, \text{GMC}_i, \text{size}_B, \text{flag}, r, \sigma_{sk_i} \rangle$$

其中:$\sigma_{sk_i} = H(\text{IP}_i, \text{GMC}_i, \text{size}_B, r)^{sk_i}$。

节点 P_j 收到地址释放消息后,验证签名。若签名合法,将 IP_i 从 $\text{list}_{\text{allocated}}$中清除。假设 P_j 负责管理的可用地址集为 $B_{\text{IP\_available}}' = (\text{size}_B', \text{flag}')$。若 $\text{flag} \oplus \text{flag}' = 1$ 且 P_i 和 P_j 负责管理可用地址集相邻,P_j 就接管节点 P_i 负责的可用地址集,$B_{\text{IP\_available}}' = (\text{size}_B' + \text{size}_B + 1, \text{flag}')$。

网络中的节点 P_i 处理其他节点突然离开的过程如图 1 所示。为了发现节点突然离开网络,网络中的节点 P_i 不仅需周期性地与其邻居节点通信,还需周期性地与在其 $\text{list}_{\text{allocated}}$中节点 $P_j(j \neq i)$进行通信,P_i 管理的可用 IP 地址集与 P_j 管理的可用地址集是连续的,且满足:当 flag = 1,P_i 比 P_j 管理的可用地址集中的地址值都大;当 flag = 0,P_i 比 P_j 管理的可用地址集中的地址值都大。P_i 通过发送 HELLO 消息与 P_j 进行通信。HELLO 格式如下:

$$\langle \text{HELLO}, \text{IP}_j, \text{IP}_i, \text{ID}_i, \text{GMC}_i, E_{s_{i,j}}(\text{flag}, \text{size}_B, r), \sigma_{sk_i} \rangle$$

其中:$\sigma_{sk_i} = H(\text{GMC}_i, \text{flag}, \text{size}_B, r)^{sk_i}$,同时启动一个计时器$\text{timer}_{\text{HELLO}}$。

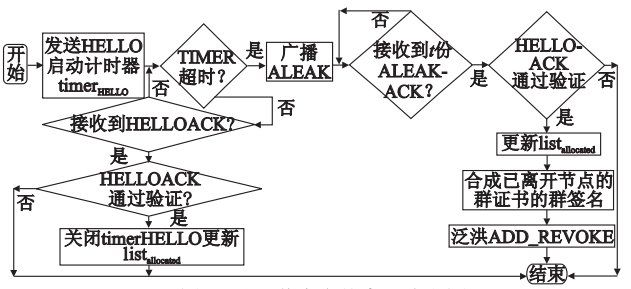

图1　处理节点突然离开流程图

若 $\text{time}_{\text{HELLO}}$超时前,$P_i$ 接收到 P_j 的应答消息 HELLOACK,则关闭 $\text{timer}_{\text{HELLO}}$,更新 $\text{list}_{\text{allocated}}$的 P_j 项,此次询问完成。HELLOACK 格式如下:

$$\langle \text{HELLOACK}, \text{IP}_j, \text{ID}_j, \text{GMC}_j, E_{s_{i,j}}(\text{flag}, \text{size}_B, r), \sigma_{sk_j} \rangle$$

其中:$\sigma_{sk_j} = H(\text{GMC}_j, \text{flag}, \text{size}_B, r)^{sk_j}$。

若 $\text{time}_{\text{HELLO}}$超时,$P_i$ 广播一个地址泄露信息 ALEAK,发起对节点 P_j 的群证书撤销的群签名。ALEAK 的格式如下:

$$\langle \text{ALEAK}, \text{IP}_i, \text{IP}_j, \text{GMC}_j, r'', \sigma_{sk_i} \rangle$$

其中:$\sigma_{sk_i} = H(\text{IP}_j, \text{GMC}_j, r'')^{sk_i}$。

当网络中 P_i 的邻居节点 P_l 接收到消息 ALEAK 后,也发送 HELLO 消息试图与 P_j 通信。若 P_l 未收到节点 P_j 的应答消息,发送给 P_i 一个地址泄露应答消息 ALEAKACK。ALEAKACK 格式如下:

$$\langle \text{ALEAKACK}, \text{GMC}_j, \text{sig}_{ss_l}(\text{GMC}_j), r, \sigma_{sk_l} \rangle$$

其中:$\sigma_{sk_l} = H(\text{GMC}_j, \text{sig}_{ss_l}(\text{GMC}_j), r)^{sk_l}$。

当 P_i 收到另外 $t-1$ 份部分群签名后,合成 $\text{sig}_{Gsk}(\text{GMC}_j)$并验证。若通过验证,接管 P_j 负责管理的可用地址,将 P_j 从 $\text{list}_{\text{allocated}}$中清除并加入可用地址集中,将地址撤销消息 ADD_REVOKE 全网泛洪。ADD_REVOKE 的格式如下:

$$\langle \text{ADD\_REVOKE}, \text{IP}_j, \text{ID}_j, \text{GMC}_j, \text{sig}_{Gsk}(\text{GMC}_j) \rangle$$

网络中的其他节点接收到 ADD_REVOKE,更新 $\text{list}_{\text{allocated}}$后就完成了节点 P_j 突然离开网络造成的地址泄露。

2.6　网络分割与合并

为简化描述,考察当前节点数为 n 的网络分割为两个子网络,只有当两个网络的(Netid,*Gpk*)相同时才进行合并。将网络中 IP 地址最小的节点 ID 与 Netid 一起形成一个二元组$(\text{Netid}, \text{ID}_{\text{min_IP}})$,记为 UUID,用于区分两个不同的网络。当 IP 地址最小的节点离开网络时,UUID 才会变化。

1)网络分割　一个网络分割为两个子网络时,若子网络的节点数不少于 t 时,当做大量节点同时离开网络处理;若子网络的节点数少于 t 时,需重建网络或加入其他的网络。

2)网络合并　Netid_i、Netid_j 是同一个网络分割后的成员节点数不少于 t 的两个子网络,令 P_i 是 Netid_i 中的节点,P_j 是 Netid_j 中的节点。当 P_i 和 P_j 可相互直接通信时,交换其 UUID。如果接收到的 UUID 不同时,就检测到了网络合并。

网络合并时，P_i 和 P_j 相互交换各自的 $list_{allocated}$ 后，P_i 在 $Netid_i$ 中泛洪 P_j 的 $list_{allocated}$，P_j 在 $Netid_j$ 中泛洪 P_i 的 $list_{allocated}$。每一个节点最终将其本身的 $list_{allocated}$ 与接收的 $list_{allocated}$ 进行合并。网络合并时，会出现地址冲突和冲突的可用地址集，即两个节点同时使用一个 IP 地址和两个节点负责管理的可用地址集有交集。节点 P_i 和 P_j 地址冲突时分为两种情况。一是只发生了地址冲突，两者负责管理的地址集不存在交集。这时只需其中一个节点将其管理的可用地址集中最大的或最小的 IP 地址自配置为其 IP 地址即可。另一种是地址和可用地址集都发生了冲突。管理的可用地址集小的节点保持其 IP 不变，并选择其管理的可用地址集中与其 IP 地址差值最大的 IP 地址，将这个 IP 地址配置另一个节点，同时将其管理的可用地址集分一半给另一个节点管理。

3 安全性与性能分析

假设 N 为节点数目；T 为平均一跳时延；D 为网络直径；r 为重试次数；d 为节点平均度数；h 为平均跳数；$o(\ )$ 表示开销或时延；S 为一次签名的时间；V 为一次验证签名的时间；E 为一次对称加或解密的时间；t 为门限。表 2 给出了与地址配置方案的安全性和性能比较。

表 2 配置方案特性比较

配置方案	MANET-conf	DACP	Buddy-conf	Wang[6]	Cavalli[7]	Taghiloo[9]	Thresh-conf
通信开销	$o(r\times N^2)$	$o(r\times N^2)$	$o(r\times h\times N)$	$o(r\times N^2)$	$o(r\times h\times N)$	$o(r\times h\times N)$	$o(h\times N)$
配置时延	$o(r\times D\times T)$	$o(r\times D\times T)$	$o(r\times h\times T)$	$o(r\times D\times T)$	$o(r\times h\times T)$	$o(r\times h\times T)$	$(6S+(t'+t)\times(2V+E)+4V+2E+6T)\times r$
全网的周期消息	有	有	无	–	无	有	无
是否依赖路由协议	是	否	否	–	否	无	否
地址欺骗攻击	无保护	无保护	无保护	vulnerable	vulnerable	可抵抗	可抵抗
冲突地址攻击	无保护	无保护	无保护	vulnerable	vulnerable	vulnerable	可抵抗
假冲突地址攻击	无保护	无保护	无保护	可抵抗	vulnerable	可抵抗	可抵抗
DoS 攻击	无保护	无保护	无保护	vulnerable	vulnerable	vulnerable	vulnerable
Sybil 攻击	无保护	无保护	无保护	vulnerable	vulnerable	vulnerable	可抵抗
地址耗尽攻击	无保护	无保护	无保护	vulnerable	可抵抗	vulnerable	可抵抗
处理网络分割/合并	是	是	是	无	是	无	是

3.1 安全性分析

本方案中的安全性基于已证明安全的 GDH 门限签名方案和求解离散对数问题。如果攻击者想获得网络中的节点的部分群密钥，则等价于求解离散对数问题，因此攻击者无法冒充网络中的节点而获得相应权限，或从群公钥中求得群私钥。

本方案分布式地进行地址分配与网络维护，不需要可信中心，符合移动自组网不存在中心节点的要求，避免了可信中心被攻破、整个网络信息暴露的后果，同时能有效抗击内部的恶意节点行为。在此方案中，即使攻击攻破了 $t-1$ 个节点或 $t-1$ 个内部恶意节点进行联合，也不能对网络信息造成影响。本方案可有效抵抗以下攻击，分析如下：

a）地址欺骗。在本方案中，每一个节点拥有数字证书，其中私钥是安全的，则恶意节点无法通过与网络中的合法节点进行相互验证获得群证书，将 IP 地址、ID 和网络进行绑定，无法伪装成一个空闲的 IP 地址。即使恶意节点通过其他的方式获得了一个已在网络中的合法节点的群证书，由于不知道相应的私钥，生成合法的身份验证消息就等价于求解离散对数问题。

b）冲突地址攻击。只有当已在网络中的 t 个恶意节点联合时才能成功地分配一个已用的地址，发动冲突地址冲突。假设网络中有 n 个节点，存在 m 个恶意节点，新节点有 $k(k\geqslant t)$ 个邻居节点，则冲突地址攻击成功的最大概率为

$$\sum_{i=t}^{k}(C_m^i\times C_{n-m}^{k-1}\times C_i^t)/(C_n^k\times C_k^t)$$

c）假冲突地址攻击。新节点加入网络时，无须进行冲突地址检测，所以不存在假冲突地址攻击。

d）重放攻击。本方案的每条消息会与一个随机数相关，即使较早的消息被截获，由于每次选择的随机数不同，使得攻击者实施重放攻击达不到预期效果。

e）DoS 攻击。MANETs 中消息是通过节点之间进行转发而进行多跳之间的通信。本方案中的任何一条消息只有在通过消息验证后才进行转发。为了防止在网络中的恶意节点反复地同时发送合法消息，只需限定转发速率。当一个恶意节点反复地同时发送不合法消息时，只对其一跳邻居节点有影响，而一跳邻居节点以一定的概率对其消息进行响应。

f）Sybil 攻击和地址耗尽攻击。由于群证书将 IP 地址、用户 ID 和网络进行了绑定，而攻击者无法知道其他节点的私钥，无法捏造一个新的身份或窃取一个合法节点的身份而申请多个 IP 地址。

3.2 性能分析

当一个新节点加入网络时，新节点进行计算的时间为 $3S+(t'+t)\times(2V+E)+V$。其中 $t'(t'\geqslant t)$ 是响应其地址请求消息的一跳邻居节点数，其参与地址配置的一跳邻居节点的计算时间为 $3(S+V)+2E$，进行通信的时间为 $6T$，因此配置时延为 $(6S+(t'+t)\times(2V+E)+4V+2E+6T)\times r$；一个新节点配置后需将地址配置信息进行全网泛洪，其通信开销的复杂度为 $o(h\times N)$，但无须周期性全网泛洪。为了抵抗相应攻击，加入了访问控制安全机制。群密钥的生成与分发、相邻节点对消息进行认证和合成群签名以及计算散列函数都需要大量的开销。但要做到移动自组网的安全，这些开销几乎是不可避免的。

4 结束语

本文提出了基于门限的地址配置方案。考虑了网络中存在的相关安全攻击，用门限签名构造了访问控制安全机制，使得地址配置协议可以在对抗性 MANETs 中正常执行，并同时保持较高的性能。门限 t 是重要的网络参数，影响着地址配置的效率与提供的安全性。本文提供的方案中，门限 t 是一个固定值，没有考虑变化的门限方案。作为本方案的扩展，可以与上下文感知系统结合，改变门限 t 的大小，从而增强地址配置方案与网络拓扑的适应性。

参考文献:

[1] BERNARDOS C, CALDERON M, MOUSTAFA H. Survey of IP address autoconfiguration mechanisms for MANETs, draft-bernardos-manet-autoconf-survey-03[R]. 2008.

[2] BACCELLI E. address autoconfiguration for MANET: terminology and problem statement, Draft-bernardos-manet-autoconf-survey-04 [R]. 2008.

[3] NESARGI S, PRAKASH R. MANETconf: configuration of hosts in a mobile Ad hoc network[C]// Proc of the 21st Annual Joint Conference of IEEE Computer and Communications Societies. New York: IEEE, 2002:1059-1068.

[4] SUN Yuan, BELDING-ROYER E M. Dynamic address configuration in mobile Ad hoc network[R]. Santa Barbara, USA: Computer Science Department, UCSB, 2003.

[5] MOSHIN M, PRAKASH R. IP address assignment in a mobile Ad hoc network[C]// Proc of Military Communication Conference. New York: IEEE, 2002:856-861.

[6] WANG Pan, REEVES D S, NING Peng. Secure address auto-configuration for mobile Ad hoc networks[C]// Proc of the 2nd Annual International Conference on Mobile and Ubiquitous Systems: Networks and Services. 2005:519-522.

[7] CAVALLI A, ORSET J M. Secure hosts auto-configuration in mobile Ad hoc networks[C]// Proc of the 24th International Conference on Distributed Computing Systems. 2004:809-814.

[8] LANGER A, KÜHNERT. Security issues in address autoconfiguration protocols an improved version of the optimized dynamic address configuration protocol[EB/OL]. (2007). http://archiv. tu-chemnitz. de/pub/2007/0049/data/paper_odacp. pdf.

[9] TAGHILOO M, TAJAMOLIAN M, DEHGHAN M, *et al*. Virtual address space mapping for IP auto-configuration in MANET with security capability[C]//Proc of International Conference on Advanced Infocomm Technology. New York: ACM Press, 2008.

[10] HU Sheng-lan, MITCHELL C J. Improving IP address autoconfiguration security in MANETs using trust modeling[C]// Proc of the 1st International Conference on Mobile Ad hoc and Sensor Networks. 2005:83-92.

[11] TAGHILOO M, TAGHILOO, DEHGHAN M. A survey of secure address auto-configuration in MANET[C]// Proc of the 10th IEEE Singapore International Conference on Communication Systems. 2006.

[12] ABDELMALEK A, FEHAM M, TALEB-AHMED A. On recent security enhancements to autoconfiguration protocols for MANETs real threats and requirements[J]. International Journal of Computer Science and Network Security, 2009, 9(4): 401-407.

[13] DESMEDT Y, FRANKEL Y. Threshold cryptosystems[C]// Proc of the 9th Annual International Cryptology Conference. Berlin: Springer, 1990:307-315.

[14] BOLDYREVA A. Efficient threshold signature, multi-signature and blind signature schemes based on the Gap-Diffe-Hellman-group signature scheme[C]//Proc of the 6th International Conference on Practice and Theory in Public Key Cryptography. Berlin: Springer-Verlag, 2003: 31-46.

[15] BONEH D, LYNN B, SHACHARN H. Short signatures from the weil pairing[C]//Proc of the 7th International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2001: 514-532.

[16] 黄梅娟, 张建中. 一种安全的门限群签名方案[J]. 计算机应用研究, 2006, 23(6):116-117.

[17] PEDERSEN T P. Noninteractive and information-theoretic secure verifiable secret sharing[C]//Advances in Cryptology-CRYPTO. Berlin: Springer, 1991:129-140.

[18] PEDERSEN T P. A threshold cryptosystem without a trusted party [C]//Proc of Workshop on the Theory and Application of Cryptographic Techniques. Berlin: Springer, 1991: 522-526.

[19] RESCORLA E. IEEE RFC 2631, Diffie-Hellman key agreement method[S]. 1999.

[20] SHAMIR A. How to share a secret [J]. Communication of ACM, 1979, 28(1):612-613.

(上接第 3085 页)

参考文献:

[1] IEEE 802.11e-WG, wireless LAN medium access control (MAC) and physical layer (PHY) specification: medium access control (MAC) quality of service (QoS) enhancements[S]. [S. l.]: IEEE, 2005.

[2] ROMDHANI L, NI Qiang, TURLETTI T. Adaptive EDCF: enhanced service differentiation for IEEE 802.11 wireless Ad hoc networks [C]// Proc of the IEEE WCNC. New Orleans, LA: IEEE, 2003: 1373-1378.

[3] NI Qiang. Performance analysis and enhancements for IEEE 802.11e wireless networks[J]. IEEE Network Magazine, 2005, 19(4): 21-27.

[4] OH B J, CHEN Chang-wen. Energy efficient H.264 video transmission over wireless Ad hoc networks based the adaptive 802.11e EDCA MAC protocol[C]// Proc of on IEEE ICME. 2008:1389-1392.

[5] LEE B-H, LAI H-C. Analysis of adaptive control scheme in IEEE 802.11 and IEEE 802.11e wireless LANs[J]. IEICE Trans on Communications, 2008, E91-B (3): 862-870.

[6] FENG Zheng-yong, WEN Guang-jun, ZOU Zi-xuan, *et al*. RED-TXOP scheme for video transmission in IEEE 802.11e EDCA WLAN [C]// Proc of the IEEE ICCTA. Beijing: IEEE, 2009:371-375.

[7] LEE J Y, LEE H S, MA J S. Model-based QoS parameter control for IEEE 802.11e EDCA[J]. IEEE Trans on Communications, 2009, 57(7): 1914-1918.

[8] IRANTHA S, ABEYSEKERA B A, MATSUDA T. Dynamic contention window control mechanism to achieve fairness between uplink and downlink flows in IEEE 802.11 wireless LANs[J]. IEEE Trans on Communications, 2008, 7(9): 3517-3525.

[9] NS2 network simulator[CP/OL]. [2009]. http: //www. isi. edu/nanam/ns/.

[10] KE Zhi-heng, SHIEH C. An evaluation framework for more realistic simulations of MPEG video transmission[J]. Journal of Information Science and Engineering, 2008, 24(2): 425-440.

[11] WIETHOELTER S, EMMELMANN M, HOENE C. TKN EDCA model for ns-2, TKN-06-003[R/OL]. (2006-06)[2009]. http://www. tkn. tu-berlin. de/ research/802.11e ns2.