容迟容断网络中一种分布式的安全数据分发机制

焦亚洲[①] 金志刚*[②] 舒炎泰[①]
[①](天津大学计算机科学与技术学院 天津 300072)
[②](天津大学电子信息工程学院 天津 300072)

摘 要：与传统网络不同，容迟容断网络(Delay/Disruption Tolerant Networks, DTN)中大部分时间不存在端到端路径，传统的基于中心服务器的各种安全机制在 DTN 中并不适用，DTN 中的数据分发也面临着同样的问题。该文提出了一种完全分布式的安全数据分发机制，采用分布式的基于身份的认证机制，无需中心私钥生成器 (Private Key Generator, PKG)的存在，并且通过门限机制和分类数据名称到分类密钥的映射，节点只需与任意门限个邻居节点通信即可获得所需分类数据的密钥。分析和仿真实验表明，该机制在保证安全性的前提下，与基于移动密钥服务器的方案相比，其密钥获取效率大大提高，非常适合 DTN 这种环境。

关键词：容迟容断网络；分布式安全机制；数据分发；门限机制；基于身份
中图分类号：TP393; TP309 **文献标识码**： A **文章编号**：1009-5896(2011)07-1575-07
DOI: 10.3724/SP.J.1146.2010.01364

A Distributed Secure Data Dissemination Mechanism for Delay/Disruption Tolerant Networks

Jiao Ya-zhou[①] Jin Zhi-gang[②] Shu Yan-tai[①]
[①](*Department of Computer Science and Technology*, *Tianjin University*, *Tianjin* 300072, *China*)
[②](*Department of Electronics and Information Engineering*, *Tianjin University*, *Tianjin* 300072, *China*)

Abstract: Different from traditional networks, there is often no contemporaneous end-to-end link between the source and destination in Delay/Disruption Tolerant Networks (DTN). So traditional security mechanism that based on central server is not suitable to DTN, and data dissemination in DTN faces the same challenge. This paper proposes an absolute distributed secure data dissemination mechanism for DTN. The mechanism adopts a distributed identity-based authenticated method, which is no need for a central Private Key Generator (PKG). Besides, depending on threshold cryptography and the mapping from category name to category key, the node only needs to communicate with random neighbor nodes whose number is no less than a certain threshold, and then it can acquire the data category key. Analytical and simulation results show this mechanism can guarantee the security requirements and greatly improve the efficiency of obtaining keys compared to method based on mobile key server, so it suits DTN very well.

Key words: Delay/Disruption Tolerant Networks (DTN); Distributed secure mechanism; Data dissemination; Threshold cryptography; Identity-based

1 引言

在一些新出现的无线网络中，如传感器网络、卫星和星际网络、车载网络等，由于节点移动、分布稀疏、射频关闭等多种原因，网络多数时候都处于不连通的状态，通常将具有这类特征的网络称为容迟/容断网络 (Delay/Disruption Tolerant Networks, DTN) [1]。与传统网络相比，DTN 具有长延迟、高误码率及频繁断路等特性，所以针对传统 Internet 和无线 Ad hoc 网络设计的数据分发算法并不适用于 DTN。由于 DTN 中大部分时间不存在端到端路径，节点都以存储-携带-转发的模式传输数据，每个要转发的数据都会在中间节点上存储较长时间，这样一来，内容存储便成了 DTN 的核心服务[2]，因此 DTN 中的数据分发一般采用基于内容的发布/订阅模型，在该模型下，网络中的节点可分为 3 种角色：订阅节点、发布节点和中间节点，数据按其内容被分为不同的类别，发布和订阅节点根据自己的兴趣来选择相应的类别进行发布和订阅，中

2010-12-13收到，2011-04-08改回
国家863计划项目(2007AA01Z220)，国家自然科学基金(61072063)和天津市应用基础及前沿技术研究计划(08JCYBJC14200)资助课题
*通信作者：金志刚 zgjin@tju.edu.cn

间节点则负责对收到的订阅请求和数据进行转发，如图 1 所示，本文所提的分布式安全机制也是基于该模型。

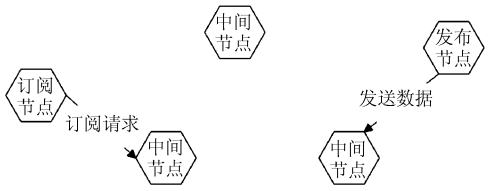

图 1 DTN 中的数据分发模型

近些年来，针对DTN的特殊环境，一些研究人员对其各方面的安全问题进行了研究，Seth等人[3]认为传统的的公钥体制由于使用了中心认证机构，因此并不适合DTN，而采用了适合于DTN的基于身份的加密机制(Identity-Based Cryptography, IBC)。Asokan等人[4]认为采用IBC机制可以有效降低对中心服务器和网络连通性的要求，并能够提供更好的机密性。Su等人[5]针对DTN的特点，利用二分Hash树和基于身份的环签名方法提出了一种匿名认证协议。Li等人[6]使用了相遇票据的方法来保证节点间的相遇数据不会被伪造，并以此为基础设计了对抗DTN中黑洞攻击的方法。Lu等人[7]利用容迟车载网络的社会性特点来安放路边设施，并通过相应的隐私保护措施来应对数据包分析和跟踪攻击。

不过，对于 DTN 中安全数据分发机制的研究还不多，Chuah 等人[8]提出了一种以数据为中心的安全方案，该方案为不同类别的数据设置不同的密钥，节点只有在得到相应的密钥后才能加密或解密各个类别的数据，但是在该机制中仅仅依靠一个移动密钥服务器(Mobile Key Server, MKS)来对密钥进行分发，这不仅会造成单点失效问题，而且由于 DTN 所固有的长延迟和频繁断接等特性，普通节点并不一定能及时地与 MKS 取得联系，这会大大降低密钥分发的效率。Mukherjee 等人[9]针对稀疏 Ad hoc 网络提出了从组标识到组密钥的映射方法，但并未涉及新加入节点如何装载初始密钥的问题，按照其所采用的单变量多项式，新加入节点要想装载初始密钥的分片，授权节点间需要多次交互才可以完成[10]，这一点并不适合于 DTN。

本文提出的分布式安全数据分发机制采用分布式的基于节点身份的认证机制，新加入节点只需与门限个节点通信即获得其身份 ID 对应的私钥，无需中心私钥生成器 (Private Key Generator, PKG)的存在。同时，通过将分类数据的名称映射为节点装载的初始密钥的组合，节点只需与任意门限个邻居节点通信即可获得所需分类数据的密钥，授权节点采用双变量多项式来为请求节点计算密钥分片，授权节点间无需交互。该机制完全以分布式的方式运行，在保证安全性的前提下，大大提高了节点获取密钥的效率，非常适合于 DTN。

2 分布式的安全数据分发机制

2.1 系统初始化

鉴于 IBC 机制的优点，本文采用更适合 DTN 的 IBC 机制，在 IBC 机制中，每个节点具有唯一的身份 ID，该 ID 可以是节点的 MAC 地址、邮箱地址等具有全局唯一性的标识。同时，在本文中，假定每个节点都有一个能证明其身份的身份证书，该证书由权威机构颁发。

IBC 机制的系统参数包括：一个大素数 q，两个阶为 q 的群 G_1 和 G_2 以及双线性映射 $e: G_1 \times G_1 \to G_2$。设 P 为 G_1 的一个生成元，系统的主密钥为 s，则系统的公钥 $P_{\text{pub}} = sP$。每个节点的公钥由该节点的 ID 产生，例如节点 i 的 ID 用 ID_i 表示，其对应的公钥为 $P_i = H_1(\text{ID}_i)$, $H_1: \{0,1\}^* \to G_1$ 为 Hash 函数，用于将节点的 ID 映射为 G_1 中的点，节点 i 的私钥 $d_i = sP_i$。IBC 机制的系统参数$(G_1, G_2, e, P, P_{\text{pub}}, H_1)$为公开参数，而系统的主密钥 s 为秘密值。

在传统的 IBC 机制中，需要一个中心 PKG 利用主密钥 s 来为每个节点计算私钥，但依靠一个中心 PKG 的方法会造成单点失效以及中心节点不可达等问题[11]，这在 DTN 中效率会非常低，因此本文采用门限密码方案来实现主密钥 s 在各个节点间的共享，传统的 Shamir 体制[12]采用一个基于单变量多项式的秘密共享方案，但单变量多项式在节点的准入控制方面，需要授权节点间的多次交互才能完成[10]，这对于 DTN 来讲并不合适，因此本文采用不需要授权节点间交互的基于双变量多项式的门限密码体制，该双变量多项式是有限域 GF(q)上的，其表达式如下所示：

$$f(x,y) = \sum_{\alpha=0}^{t-1} \sum_{\beta=0}^{t-1} f_{\alpha\beta} x^{\alpha} y^{\beta} \tag{1}$$

式(1)中的 x 和 y 为变量，$f_{\alpha\beta}$ 为常数项系数，t 为门限值。

系统由一组初始节点进行初始化，初始化在一个封闭的环境中进行，初始节点间相互信任，并通过安全信道传递消息，设这组初始节点为$\{N_1, N_2, \cdots, N_l\}$，初始化时应保证网络中初始节点的个数 $l \geq t$。

该组中的每一个节点 $i\,(1 \leq i \leq l)$选择一个有限域 GF(q)上形如式(1)的双变量多项式：

$$f^i(x,y)=\sum_{\alpha=0}^{t-1}\sum_{\beta=0}^{t-1}f_{\alpha\beta}^i x^\alpha y^\beta \tag{2}$$

其中 $f_{\alpha\beta}^i$ 是由 N_i 随机选择的常数项系数。

令 $h_i=H_2(\mathrm{ID}_i)$，$H_2:\{0,1\}^*\rightarrow \mathrm{GF}(q)$为 Hash 函数，用于将任意字符串映射为有限域 GF(q)中的元素，H_2 为所有网络节点所共知。

每一个节点 i 向该组的其它节点发送请求[REQ-$f_i^j(x,h_i)$,ID_i]，请求一个由 h_i 计算得来的单变量多项式，如节点 $j(1\le j\le l$ 且 $j\ne i)$为节点 i 计算的单变量多项式如下所示：

$$f_i^j(x,h_i)=\sum_{\alpha=0}^{t-1}\sum_{\beta=0}^{t-1}f_{\alpha\beta}^j x^\alpha h_i^\beta \tag{3}$$

当节点 i 收到其它节点发回的单变量多项式后，就可以计算出一个属于自己的主密钥 s 的共享多项式。

$$\begin{aligned}s_i(x)&=\sum_{j=1}^{l}f_i^j(x,h_i)\\&=\sum_{\alpha=0}^{t-1}\sum_{\beta=0}^{t-1}(f_{\alpha\beta}^1+f_{\alpha\beta}^2+\cdots+f_{\alpha\beta}^l)x^\alpha h_i^\beta\end{aligned} \tag{4}$$

令共享主密钥 s 时系统采用的双变量多项式为

$$\begin{aligned}f(x,y)&=\sum_{\alpha=0}^{t-1}\sum_{\beta=0}^{t-1}(f_{\alpha\beta}^1+f_{\alpha\beta}^2+\cdots+f_{\alpha\beta}^l)x^\alpha y^\beta\\&=\sum_{\alpha=0}^{t-1}\sum_{\beta=0}^{t-1}f_{\alpha\beta}x^\alpha y^\beta\end{aligned} \tag{5}$$

则 $s_i(x)=f(x,h_i)$，属于 i 的主密钥 s 的分片为

$$s_i=s_i(0)=f(0,h_i)=s+\sum_{\beta=1}^{t-1}f_{s\beta}h_i^\beta \tag{6}$$

式(6)中的 $f_{s\beta}$ 为相应的常数项系数。

利用同样的方法，每个初始节点通过选择不同的常数项系数也可以得到 Q 个初始密钥 $K=\{K_1,K_2,\cdots,K_Q\}$的共享多项式和属于自己的一份分片，这 Q 个初始密钥在构建分类数据密钥时使用，其密钥标识符可以用$\{I_1,I_2,\cdots,I_Q\}$来表示。共享密钥 K_r $(1\le r\le Q)$时系统采用的双变量多项式用 $f_r(x,y)$ 表示，则节点 i 得到的密钥 K_r 的共享多项式 $s_{r,i}(x)$ 和分片 $s_{r,i}$ 分别为

$$s_{r,i}(x)=f_r(x,h_i)=\sum_{\alpha=0}^{t-1}\sum_{\beta=0}^{t-1}f_{r\alpha\beta}x^\alpha h_i^\beta \tag{7}$$

$$s_{r,i}=f_r(0,h_i)=k_r+\sum_{\beta=1}^{t-1}f_{r\beta}h_i^\beta \tag{8}$$

式(7)和式(8)中的 $f_{r\alpha\beta}$ 和 $f_{r\beta}$ 是相应的常数项系数，其中的门限值可以与生成主密钥 s 时的相等，也可以不相等，在本文中为叙述方便，统一以 t 表示。

然后节点 i 向其它节点发送请求[REQ-d_i,ID_i]，请求其它节点为自己计算私钥 d_i 提供分片信息。其它节点如节点 j 收到请求后将消息[REP-d_i,s_jP_i]发回给 i，当节点 i 收到 t 个或以上个返回消息后，就可以计算出自己的私钥：

$$d_i=\sum_{j=1}^{t}s_jP_i\prod_{q=1,q\ne i}^{t}\frac{-h_q}{h_j-h_q}=sP_i \tag{9}$$

2.2 新节点的加入

初始化完成以后，系统将运行在一个开放的环境中，为了防止传递的消息被窃听或篡改，必须对消息进行加密和签名，当有新的节点加入时，新节点也需要获得其身份 ID 对应的私钥，以便在以后对其发送的消息进行签名。

令节点 m 为申请加入的节点，m 将消息[REQ-d_m,ID_m]广播给与其相遇的邻居节点，当其邻居节点 i 收到此消息后，节点 i 和 m 根据 Diffie-Hellman 协议产生此次交互的共享密钥，步骤如下：

首先节点 i 和 m 分别选择一个大数 w 和 z，然后节点 i 用其私钥 d_i 对包含有(n，g，$g^w \bmod n$)的消息进行签名后发送给 m，消息中的 n 和 g 为 Diffie-Hellman 协议的两个公开参数，为所有节点所共知。

m 在收到消息后，用 i 的公钥 P_i 对消息的签名进行验证，验证通过后向 i 发送包含有($g^z \bmod n$)的消息，这样 m 和 i 就得到了此次交互的共享密钥 $k_{mi}=g^{wz} \bmod n$，之后 m 用 k_{mi} 将其身份证书加密后发送给 i，节点 i 在验证之后对 m 进行授权，用其私钥 d_m 对消息[REP-d_m,s_iP_m,$s_i(h_m)$]进行签名并用 k_{mi} 加密后发送给 m。

上述的 Diffie-Hellman 密钥交换协议具有认证功能，可以防止中间人攻击。因为节点 i 发送的消息是用其私钥签名的，其它节点不可能有 i 的私钥，所以首先节点 m 可以验证 i 的身份。然后，在生成共享密钥 k_{mi} 之后，m 需要向 i 提供其身份证书才能获得相应的私钥分片，即便之前是其它节点拦截了双方传送的消息，获得了 k_{mi}，但是由于它不能提供有效的身份证书，它也得不到相应的私钥分片。

当 m 得到 t 个节点的提供的 s_iP_m 之后，就可以按式(9)计算出其私钥 d_m，然后可以计算 $e(P_{\mathrm{pub}},P_m)$与 $e(P,d_m)$，若两者相等，则 d_m 正确，若不相等，则可以再选择 t 个节点(可以与原来的 t 个有重复)进行计算，直到得到正确的 d_m。与此同时，m 也可以发现哪些节点提供了错误的分片。

节点 m 还能根据收到的 t 个 $s_i(h_m)$ 计算出属于它的一个共享多项式：

$$\begin{aligned}s_m(x)&=\sum_{i=1}^{t}\prod_{j=1,j\ne i}^{t}\frac{x-h_j}{h_i-h_j}s_i(h_m)\\&=\sum_{i=1}^{t}\prod_{j=1,j\ne i}^{t}\frac{x-h_j}{h_i-h_j}f(h_i,h_m)=f(x,h_m)\end{aligned} \tag{10}$$

则属于 m 的主密钥分片 s_m 为

$$s_m = s_m(0) = f(0,h_m) = s + \sum_{\beta=1}^{t-1} f_{s\beta} h_m^{\beta} \tag{11}$$

这样，节点 m 也可以对在其后加入的节点进行授权，因为如果 PKG 的功能只由一组初始节点来承担的话，随着网络规模的扩大，这组初始节点可能会变的不可达，因此授权节点的增多会大大增加新加入节点获取密钥的效率。

类似地，节点 m 也可以请求得到初始密钥 K_r ($1 \le r \le Q$)的共享多项式 $s_{r,m}(x)$ 并计算出相应的密钥分片 $s_{r,m}$。

$$\begin{aligned} s_{r,m}(x) &= \sum_{i=1}^{t} \prod_{j=1,j\neq i}^{t} \frac{x-h_j}{h_i-h_j} s_{r,i}(h_m) \\ &= \sum_{i=1}^{t} \prod_{j=1,j\neq i}^{t} \frac{x-h_j}{h_i-h_j} f_r(h_i,h_m) = f_r(x,h_m) \end{aligned} \tag{12}$$

$$s_{r,m} = s_{r,m}(0) = f_r(0,h_m) = k_r + \sum_{\beta=1}^{t-1} f_{r\beta} h_m^{\beta} \tag{13}$$

对于 $s_m(x)$ 和 $s_{r,m}(x)$ 的正确性，可以按照文献[10]中所述的方法进行验证，在此不再赘述。

2.3 分类数据密钥的获取

在本文中为每个分类数据设置不同的密钥，各个分类密钥可以按文献[8]的方法采用一个 MKS 来存储和管理，但是在 DTN 中该方法效率很低；也可以让网络中的节点装载各个分类数据密钥的一份分片，请求节点只需得到特定分类的 t 个分片即可计算出该分类数据的密钥，但是如果分类较多，则各个节点存储的密钥分片数将会非常多，这会占用节点大量的存储空间，因此本文通过将分类名称映射为节点装载的初始密钥的方法来解决此问题。

假设节点 m 为想获得某分类数据密钥的节点，它将消息[REQ- K_{cn}, ID_m ,category name||time]用其私钥 d_m 签名后向与其相遇的邻居节点广播，消息中的 K_{cn} 表示所请求的分类数据密钥，符号||表示字符串的串联，time 表示该密钥有效期对应时间段的起始时间，在本文中，每个分类数据在不同的时间段会有不同的密钥，另外，本文假定所有节点的时间是同步的，并且节点不能请求未来时间段的分类密钥。

当 m 的某一邻居节点 A 收到该消息后首先用 P_m 对 m 的签名进行验证，然后 A 核实节点 m 是否有权访问该类数据(具体的访问政策取决于具体的应用，在本文中不再进一步讨论)，如果核实通过，则 A 计算字符串 category name||time 的 Hash 值：

$$H_{cn}=H_3(\text{category name}||\text{time}) \tag{14}$$

式(14)中的 H_3 表示 Hash 函数，可以是常用的 MD5 或 SHA-1 等，为网络中所有节点所共知。然后 A 接着计算 H_{cn}^{j} ($j=2,3,\cdots,k$)，H_{cn}^{j} 表示 H_{cn} 的第 j 重 Hash 值，如 $H_{cn}^{2}=H_3(H_{cn})$。然后取这 k 个 Hash 值的最低 L ($L=[\ln_2 Q]$)个有效位，分别用 $R(H_{cn},1)$, $R(H_{cn},2),\cdots,R(H_{cn},k)$ 来表示，因为是取 Hash 值的最低 L 个有效位，所以 R 代表了序列$\{1,2,\cdots,Q\}$的一个置换，这样就可以得到 $I_{R(H_{cn},1)}$, $I_{R(H_{cn},2)},\cdots,$ $I_{R(H_{cn},k)}$ 这 k 个初始密钥的标识符，也就是将数据类别的名称映射成了节点装载的初始密钥，然后 A 计算

$$W(H_{cn},A) = \sum_{i=1}^{k} s_{R(H_{cn},i),h_A} H_{cn}^{i} \tag{15}$$

接着节点 A 用 d_A 对消息[REP- K_{cn},ID_A,$W(H_{cn},A)$]签名，并用 P_m 加密后发送给 m。

将式(13)代入式(15)可以得到如下所示的多项式：

$$W(H_{cn},A) = \sum_{i=1}^{k} H_{cn}^{i} \left(K_{R(H_{cn},i)} + \sum_{\beta=1}^{t-1} f_{R(H_{cn},i),\beta} h_A^{\beta} \right) \tag{16}$$

当 m 收到 t 个节点的回应后，它就可以从 t 个如式(16)的多项式中解得所请求分类的密钥：

$$K_{cn} = \sum_{i=1}^{k} H_{cn}^{i} K_{R(H_{cn},i)} \tag{17}$$

此后，节点 m 可以分别计算两组节点(每组 t 个，两组节点可以有重复)提供的分片，以验证所得到的 K_{cn} 的正确性，若两组得出的 K_{cn} 相等，则说明 K_{cn} 是正确的，若不相等，则可以继续计算第 3 组，直到得到两组相等的 K_{cn}。在此过程中，节点 m 也能够发现哪些节点提供了错误的分片，从而可以找出这些恶意节点。

2.4 恶意节点的发现和密钥的定期更新

在 2.2 节和 2.3 节中节点在获得所请求密钥的分片时，可以进行相应的验证，从而可以发现哪些恶意节点提供了错误的分片，当发现恶意节点时，可以对其进行举报，将其节点号在网络中进行广播。

获得分类数据密钥的节点在发布数据时需用该密钥对数据进行加密，同时，也需要用其私钥对数据的摘要进行签名，没有经过签名的数据将不会被中间节点转发。如果有节点被发现在发布垃圾数据或者病毒数据，则该节点也将会被举报。

同时，网络中的每个节点维护一个恶意节点列表，每收到 1 次对某一节点的举报，就将该节点在恶意节点列表上的记录加 1，若某节点的恶意记录超过了一定的阈值，此后将忽略其提供的密钥分片。当然为了防止有节点进行恶意举报，规定每个节点只能对某一特定节点进行一次举报，重复的举报将会被忽略。

同时，本文的分类数据密钥采用定期更新的策

略，如 2.3 节的式(14)所示，在计算每一个分类数据的密钥时都关联一个时间值，同一分类在不同时间段的密钥是不同的，这样每隔一段时间，想发布或订阅某分类数据的节点就必须重新获取相应的密钥。而当某节点记录的某一特定节点被举报次数超过一定阈值后，将拒绝为其提供相应的密钥分片，因此定期更新密钥的机制可以将有恶意行为的节点及时剔除出网络。

3　安全性与效率分析

3.1 安全性分析

首先，在初始化时，初始节点间处在封闭的环境中，并通过安全信道来获得属于自己的主密钥分片和共享多项式，因此初始化阶段是安全的；而在新加入节点获取私钥时，由于采用了具有认证功能的 Diffie-Hellman 协议，也只有能够提供身份证书的节点才可以获得其 ID 对应的私钥分片和共享多项式，因此本文所提的安全机制在私钥的获取方面是安全的。

在获取分类数据密钥时，不管是请求节点还是提供密钥分片的节点都需用其私钥对发送的消息进行签名，因此提供错误分片的恶意节点将会被及时发现。同样，在发布数据时，节点也需要用其私钥对数据摘要进行签名，因此发布垃圾数据或是病毒数据的恶意节点也会被及时发现，从而这些恶意节点能够尽快被剔除出网络之外。

此外，由于在计算分类数据密钥时，需要将分类名称映射为 k 个初始密钥，这样，如果一个节点请求分类数据密钥数大于 k 个，则从理论上来讲，该节点有可能得到由 k 初始密钥组成的 k 个多项式，这样该节点就可以从这 k 个多项式中解出 k 个初始密钥，但是文献[9]的分析表明这种可能性非常小，尤其是在初始密钥数 Q 较大的情况下。

本文所提方案的安全强度取决于门限机制的安全强度，当小于 t 个节点被妥协时，系统可以保持安全，当大于或等于 t 个节点被妥协，并且这些被妥协节点合谋的话，IBC 机制中的主密钥 s 和初始密钥将会暴露。

一旦主密钥 s 或初始密钥暴露，则可以按照系统初始化时的方法由一组节点对系统进行重建。在重建时，重建节点间用自己的私钥来对发送的消息进行签名并用另一方的公钥加密传递的消息，无需安全信道的存在，通过重建可以将有恶意行为的节点排除在网络之外。

3.2 计算复杂性分析

在初始化阶段，每个节点需要 $(l-1)t$ 次模乘运算来为其它节点计算共享多项式，同时需要一次模加运算来计算自己的共享多项式，此后当计算私钥时则需要一次拉格朗日插值运算。

对于新加入的节点，在获取其私钥、主密钥 s 的共享多项式以及初始密钥的共享多项式时各需要一次拉格朗日差值运算。同时，在利用 Diffie-Hellman 协议生成共享密钥时，需要 $O(t)$ 次的模指数运算，然后用生成的共享密钥进行 $O(t)$ 次的加密和解密运算。在接收到授权节点的返回消息时，它需要进行 $O(t)$ 次验证运算，在验证私钥正确性时需要进行两次对运算。而每个授权节点需要进行模指数运算、签名运算、用共享密钥进行的加密和解密运算各一次，同时需要一次点乘运算和 $O(t)$ 次的模乘运算。

在获取分类密钥时，对于请求节点来讲，它需要通过高斯消去来解得 K_{cn}，其计算量为 $O(t^3)$ 次模乘运算，而对于每一个授权节点则需要 $O(t)$ 次模乘运算。同时，请求节点需要一次签名运算和 $O(t)$ 次验证运算，而授权节点需要签名运算和验证运算各一次。授权节点需要用请求节点的公钥进行一次加密运算，而请求节点则需用其私钥进行 $O(t)$ 次解密运算。

此外，在上述 3 个过程中，请求和授权节点都需要若干次的 Hash 运算。

3.3 通信复杂性分析

通信复杂性取决于网络连通的程度，如果节点比较密集，则请求节点只需一次广播即可得到 t 个授权，但是对于 DTN，网络的连通程度很低，所以请求节点可能需要多次广播才可以，最坏时需要 $O(t)$ 次的单播才能得到 t 个授权，而对于授权节点则只需一次单播即可。

4　仿真实验

接下来本文采用ns-2仿真工具对本文所提的分布式安全机制与文献[8]的基于MKS的方案在获取分类密钥的延迟和成功率方面进行比较，由于运算代价的延迟远小于通信延迟，因此，模拟时我们不考虑协议的运算代价对性能的影响。

仿真场景如下：50个节点随机分布在大小为 2000×2000 m^2的仿真区域内，节点按照移动模型(Random Way Point, RWP)运动，每个节点的移动速度服从0到20 (m/s)上的均匀分布。仿真共进行10次，每次仿真进行1500 s。仿真分布式机制时，每次仿真随机选择10个节点作为密钥请求节点，在仿真基于MKS的方案时，MKS节点从50个节点中任意选择，仿真时通过调节节点的通信范围来模拟网络的连通程度，仿真结果取10次仿真的平均值。

图2给出了节点获取分类密钥的平均延迟随节

点通信范围的变化关系，其中平均延迟表示从请求节点开始请求到其获得分类密钥所用的时间，在计算平均延迟时只考虑成功获得分类密钥的节点。从图中可以看出，在节点的通信范围较低时，网络连通程度也较低，这时在采用分布式机制时，由于节点只需与任意门限个节点通信即可获得所需密钥，而在基于MKS的方案中，请求节点只有遇到MKS节点时才能获得所需密钥，因此在这种情况下，分布式机制的平均延迟大大低于基于MKS的方案，如在节点通信范围为10 m，门限值t=3时，前者比后者的平均延迟低了91.6%。而随着节点通信范围的增加，基于MKS的方案的平均延迟有所降低，但是分布式机制仍然大大优于基于MKS的方案，如在节点通信范围为200 m，门限值t=10时，分布式机制仍比基于MKS的方案的平均延迟低了82.1%。在采用分布式机制时，不同的门限值对平均延迟也有一定的影响，门限值较低时，节点需要通信的节点较少，则平均延迟较低，随着门限值增高，则平均延迟变大，但是其安全性也相对较高。

图3给出了节点获取密钥的成功率随节点通信范围的变化关系，密钥获取成功率表示请求分类密钥的节点中最终获得所请求密钥的节点所占的比例。从图中可以看出，在网络的连通程度较低时，基于MKS的方案的密钥获取成功率非常低，尤其是在节点通信范围为10 m时，其密钥更新成功率几乎为0，而此时，即便是门限值较大的分布式方案，其密钥获取成功率仍然很高，如门限值t=10时的密钥获取成功率仍然达到了82%。随着网络连通程度的增加，在基于MKS的方案中，节点的密钥获取成功率增加的较快，但始终跟分布式机制有较大差距。

5 结束语

本文提出的分布式安全数据分发机制完全以分布式的方式运行，在认证时，采用分布式的基于节点身份的认证机制，新加入节点只需与门限个节点通信即获得其身份 ID 对应的私钥，无需中心 PKG 的存在。同时，通过将分类数据的名称映射为节点装载的初始密钥的组合，节点也只需与任意门限个邻居节点通信即可获得所需分类数据的密钥，而在这两个过程中，授权节点都采用双变量多项式来为请求节点计算密钥分片，授权节点之间无需交互。分析和仿真实验表明，该机制在保证安全性的前提下，大大提高了节点获取密钥的效率，非常适合DTN 这种环境。下一步我们将研究更加高效的重建整个安全系统的机制以及进一步细化对恶意节点的惩罚策略。

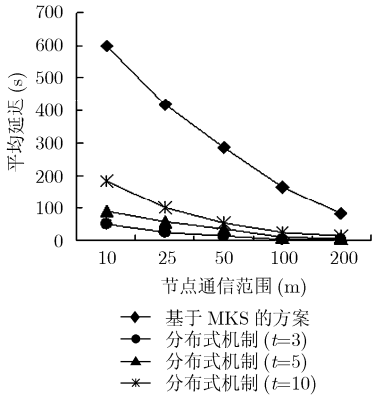

图2 获取密钥的平均延迟随节点通信范围的变化关系

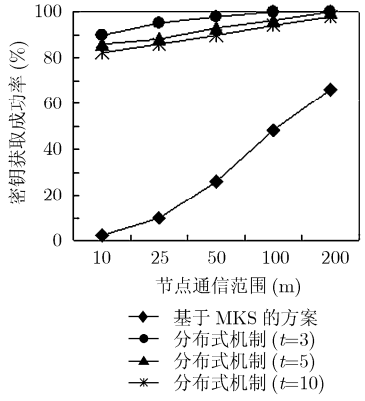

图3 密钥获取成功率随节点通信范围的变化关系

参 考 文 献

[1] Fall K. A delay tolerant networking architecture for challenged internet[C]. Proceedings of IEEE SIGCOMM 2003, Karlsruhe, 2003: 27–34.

[2] 熊永平, 孙利民, 牛建伟等. 机会网络[J]. 软件学报, 2009, 20(1): 124–137.
Xiong Yong-ping, Sun Li-min, and Niu Jian-wei, *et al*.. Opportunistic networks[J]. *Journal of Software*, 2009, 20(1): 124–137.

[3] Seth A, Hengartner U, and Keshav S. Practical security for disconnected nodes[C]. Proceedings of the First Workshop on Secure Network Protocols(NPSec), Boston, 2005: 31–36.

[4] Asokan N, Kostiainen K, and Ginzboorg P, *et al*.. Applicability of identity-based cryptography for disruption-tolerant networking[C]. Proceedings of MobiSys Workshop on Mobile Opportunistic Networking(MobiOpp), San Juan, 2007: 52–56.

[5] Su Ren-wang and Cao Zhen-fu. An efficient anonymous authentication mechanism for delay tolerant networks[J].

Computers and Electrical Engineering, 2010, 36(3): 435–441.

[6] Li Feng, Wu Jie, and Srinivasan A. Thwarting blackhole attacks in distruption-tolerant networks using encounter tickets[C]. Proceedings of IEEE INFOCOM 2009, Rio de Janeiro, 2009: 2428–2436.

[7] Lu Rong-xing, Lin Xiao-dong, and Shen Xue-min. SPRING: a social-based privacy-preserving packet forwarding protocol for vehicular delay tolerant networks[C]. Proceedings of IEEE INFOCOM 2010, San Diego, 2010: 1–9.

[8] Chuah M and Metgzer R. Secure data retrieval system (SEDAR) for DTNs[C]. Proceedings of IEEE Milcom 2008, San Diego, 2008: 1–7.

[9] Mukherjee A, Gupta A, and Agrawal D P. Distributed key management for dynamic groups in MANETs[J]. *Pervasive and Mobile Computing*, 2008, 4(4): 562–578.

[10] Saxena N, Tsudik G, and Yi J H. Efficient node admission and certificateless secure communication in short-lived MANETs[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2009, 20(2): 158–170.

[11] Daza V, Morillo P, and Ràfols C. On dynamic distribution of private keys over MANETs[J]. *Electronic Notes in Theoretical Computer Science*, 2007, 171(1): 33–41.

[12] Shamir A. How to share a secret[J]. *Communications of the ACM*, 1979, 22(11): 612–613.

焦亚洲：男，1982 年生，博士生，研究方向为容迟容断网络、网络与信息安全.

金志刚：男，1972 年生，教授，博士生导师，研究方向为网络与信息安全、网络性能评价.

舒炎泰：男，1942 年生，教授，博士生导师，研究方向为计算网络、CIMS.