移动自组网中基于多跳步加密签名函数签名的分布式认证

熊 焰,苗付友,张伟超,王行甫
(中国科学技术大学计算机科学技术系,安徽合肥 230026)

摘 要: 移动自组网 Manet(Mobile Ad Hoc Network)是一种新型的无线移动网络,由于其具有网络的自组性、拓扑的动态性、控制的分布性以及路由的多跳性,所以,传统的安全机制还不能完全保证 Manet 的安全,必须增加一些新的安全防范措施. 本文探讨了 Manet 所特有的各种安全威胁,提出了一种基于多跳步加密签名函数签名的安全分布式认证方案,即将移动密码学与(n, t)门槛加密分布式认证相结合,并采用了分布式容错处理算法和私钥分量刷新技术以发现和避免攻击者假冒认证私钥进行非法认证以及保护私钥分量和认证私钥不外泄.

关键词: 移动自组网络; 移动密码学; 门槛加密机制; 分布式认证
中图分类号: TP393 **文献标识码**: A **文章编号**: 0372-2112 (2003) 02-0161-05

Secure Distributed Authentication Based on Multi-Hop Signing with Encrypted Signature Functions in Mobile Ad Hoc Networks

XIONG Yan, MIAO Fu-you, ZHANG Wei-chao, WANG Xing-fu
(*Department of Computer Science and Technology of University of Science and Technology of China, Hefei, Anhui 230026, China*)

Abstract: Mobile Ad Hoc Network (Manet) is a new kind of wireless mobile network. Because of its self-organization, dynamic topology, distributed control and multi-hop routing, traditional security mechanisms are not sufficient to secure Manet. Therefore, some new security mechanisms must be used. Security threats faced by Manet are studied, and a secure distributed authentication scheme based on multi-hop signing with encrypted signature functions is proposed. Here mobile cryptography and (n, t) threshold cryptography distributed authentication are combined, and distributed fault tolerance processing algorithm and private key share refreshing technology are used to detect and prevent attackers from violating authentication using digital signature and protect private key shares and authentication private key from leaking.

Key words: mobile Ad Hoc network; mobile cryptography; threshold cryptography; distributed authentication

1 引言

移动自组网络是一种新型的无线移动网络,它由一组带有无线收发器的移动终端所组成的一个无基站多跳步临时性自治网络系统. 它主要应用在军事战场、抗洪救火、无法布线等特殊和紧急环境下,并具有一般通信网络所没有的一些特点:(1)网络的自组性;(2)拓扑的动态性;(3)带宽的有限性;(4)控制的分布性;(5)路由的多跳性;(6)安全的不可靠性等等. 在国外,Manet 的研究已成为一个热门课题,吸引了大批科研人员的研究兴趣,并取得了许多令人瞩目的研究成果. 国内科研院所也先后对 Manet 进行了研究和探讨,其范围从协议、算法的理论研究到实验系统的具体实现,为我国加快移动自组网络的研究步伐,尽快加入世界先进行列奠定了基础.

由于 Manet 是一种特殊无线网络,所以,它不仅面临着传统网络所受到的安全威胁,而且,还面临着自身一些特殊的安全威胁,主要表现在以下 3 个方面:(1)攻击者很容易通过无线链路窃听、假冒、篡改和转发在其上传输的信息;(2)由于移动结点有可能漫游到敌对环境中(例如敌战区),所以,恶意攻击不仅来自网络的外部,而且,也有可能来自内部被攻破的移动结点;(3)由于 Manet 拓扑结构和移动结点数目不断变化(移动结点不断加入和离开 Manet),所以,移动结点之间的信任关系也经常改变,特别是检测出被攻破移动结点以后,其他移动结点对它们的信任也随之消失.

此外,动态路由在 Manet 中占据着极为重要的位置,也是 Manet 通信的基础. 因此,动态路由的安全与否直接关系到整个 Manet 是否能够安全正常运行. Manet 所面临的三个方面的安全威胁在动态路由中具体表现为二类攻击:第一类攻击来自外部即攻击者通过注入错误路由信息、恶意转发过时路由信息以及将 Manet 人为地分成多个假不连通的区域或者通过产生大量的重发和无效的路由信息大大加重 Manet 中的通信负载;第二类也是更为严重的攻击来自内部被攻破的移动结

收稿日期:2002-05-28;修回日期:2002-07-30
基金项目:安徽省自然科学基金(No. 00043105)

点即被攻破移动结点能够广播不正确的路由信息给其他移动结点而无法被发现和识别.

采用传统的公钥密码体制、数字签名、包序列以及单向 Harsh 函数(如 MD5)相结合的方法既可以保证 Manet 中动态路由信息和数据传输的保密性,又可以识别动态路由信息和数据的来源,此外,还可以防范"重播攻击"和"填充攻击"(Padding Attack),然而,它需要一个信任的认证中心来提供密钥管理服务.

由于 Manet 是一个无中心的无线网络,因而没有一个移动结点是值得信任的,所以,不仅单个认证中心的崩溃将造成整个网络无法获得认证,而且更为严重的是,被攻破认证中心的私钥可能会泄露给攻击者,攻击者可以使用其私钥来签发错误的证书来假冒一个移动结点或取消一个证书,致使 Manet 完全失去了安全性. 如果采用多个相同的认证中心拷贝也存在严重的问题:增加了被攻击的目标,任一个认证中心拷贝被攻破,则整个 Manet 就失去了安全性.

鉴于此,我们提出了 (n, t) 门槛加密分布式认证即由 Manet 中 n 个移动结点中任意 t 个移动结点通过合作共同提供认证服务(其中,n 个移动结点中的每一个,我们称其为一个认证结点,每一个认证结点具有认证私钥的一部分,称之为私钥分量)以克服单个认证中心以及多个认证中心拷贝所产生的问题,并使最多 $t-1$ 个不具有 Byzatine 行为的被攻破认证结点无法联合生成合法认证私钥进行非法认证. 然而,(n, t) 门槛加密分布式认证要求将多个认证结点的私钥分量逐步集中起来以便共同生成认证私钥,这将产生一个严重的安全问题:如果一个认证结点被攻破,那么,在这个认证结点上集中的私钥分量包括它自己的私钥分量都会被攻击者通过计算来获得,特别是第 t 个认证结点被攻破以后,情况将变得更糟,认证私钥或者用于生成认证私钥的 t 个私钥分量将被攻击者获得,这样,攻击者可以用窃取的 t 个私钥分量生成认证私钥或者能够直接用窃取的认证私钥进行非法认证. 为此,我们在移动密码学的基础上采用一个多跳步加密签名函数签名的方案来实现 (n, t) 门槛加密分布式认证以防止私钥分量和认证私钥的泄露. 此外,对于 $m(m <= (n-1)/3)$ 个具有 Byzatine 行为的被攻破认证结点,我们利用分布式容错处理技术中 Lamport 解决 Byzatine 将军问题的协同一致算法来发现它们,并将其从 n 个认证结点中清除. 对于循环移动攻击 n 个认证结点,我们仍采用 Lidong Zhou[9] 的私钥分量刷新技术使攻击者很难同时获得 t 个认证结点的私钥分量进行联合认证.

因此,我们所采用的基于多跳步加密签名函数签名的 (n, t) 门槛加密分布式认证方法不仅能够容忍一定数量的认证结点崩溃,而且还能够有效防范内部被攻破认证结点对分布式认证的干扰,此外,还避免了私钥分量和认证私钥的泄露. 因此,它弥补了用传统网络安全机制(公钥密码体制、数字签名、包序列以及单向 Harsh 函数)解决 Manet 安全性的不足.

2 关于拉各朗日插值公式的(n, t)门槛加密分布式认证算法

由于在 Manet 中不存在任何一个值得信任的移动结点且每一个移动结点随时都有可能移动到所有其他移动结点通信范围之外(如果是认证中心的话,那么就无法提供认证服务),所以,如果只采用一个认证中心或多个相同认证中心拷贝进行认证,那么,它们就很容易被攻破. 攻击者就会盗用被攻破认证中心的认证私钥进行非法认证. 因此,我们可采用 (n, t) 门槛加密机制进行分布式认证. 其基本思想就是将 Manet 中的 n 个移动结点作为一个认证整体,其中每一个移动结点也称之为一个认证结点,由这 n 个认证结点中的任意 t 个协同合作生成认证私钥以提供分布式认证,而任意不超过 $t-1$ 个认证结点无法共同生成认证私钥,这样,既可以克服单个认证中心和多个相同认证中心拷贝所产生的问题,同时,又能够防范少于 t 个被攻破认证结点联合起来生成认证私钥进行恶意地认证. 整个认证只有一对认证公钥 SPK 和认证私钥 SSK(可由 RSA 算法生成),它知道所有移动结点的公钥,每一个移动结点都可以发送查询和更新请求给这个认证整体以获得其他移动结点的公钥以及更新自己的公钥. 所有的移动结点都信任由认证私钥 SSK 签发的证书. 每一个认证结点都知道认证公钥 SPK,并与普通移动结点一样也都有一对公钥 PKi 和私钥 SKi 以保证认证结点之间路由信息和数据信息交换的安全性. 此外,认证私钥 SSK 并不存放在任何一个认证结点中(也就是说它并不存在),而是采用拉各朗日插值公式将认证私钥以 n 个分量形式(我们称之为私钥分量)分别存放在 n 个认证结点中,这 n 个私钥分量形成一个私钥分量向量 $(k_1, \cdots, k_i, \cdots, k_n)$,认证私钥 SSK 可由私钥分量向量 $(k_1, \cdots, k_i, \cdots, k_n)$ 中任意 t 个私钥分量按拉各朗日插值公式生成. 基于拉各朗日插值公式的认证私钥分存和生成过程如下:

令 $GF(q)$ 是一有限域,$q > n$. 随机选取一组 $a_1, a_2, \cdots, a_{t-1} \in GF(q)$,与认证私钥 SSK 一起形成多项式函数:

$$F(x) = SSK + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1}$$

设 α 是 $GF(q)$ 域的本原元素,将 $\alpha^1, \cdots, \alpha^i, \cdots, \alpha^n$,分别代入 $F(x)$,得到 $k_i = F(\alpha^i)$ $(i = 1, 2, \cdots, n)$,分别保存到认证结点 i 中. 不失一般性,设认证结点 $1, \cdots$,认证结点 $j, \cdots$,认证结点 t 分别提供各自的私钥分量 k_j 以及各自的 α^i,由拉各朗日插值公式(这里的加,减,乘,除都是在 $GF(q)$ 上进行的,特别是除法,实际是乘以分母的逆元素):

$$F(x) = SSK + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1}$$

$$= P(x) = \sum_{j=1}^{t} k_j \prod_{i \neq j} (x - \alpha^i) / (\alpha^j - \alpha^i)$$

当 $x = 0$ 时,$SSK = F(0) = p(0) = \sum_{j=1}^{t} k_j \prod_{i \neq j} (\alpha^i) / (\alpha^j - \alpha^i)$

$$= \sum_{j=1}^{t} k_j \beta_j ; \left(\text{记 } \beta_j = \prod_{i \neq j} \alpha^i / (\alpha^j - \alpha^i)\right)$$

因此,由任意 t 个认证结点上的私钥分量能够生成认证私钥 SSK,而不足于 t 个认证结点则无法生成认证私钥 SSK.

由上,我们可形成如下基于拉各朗日插值公式的 (n, t) 门槛加密分布式认证算法:

(1) 当任意一个移动结点 m 需要认证时,它就向一个认证结点(可以随机选择,例如,认证结点 1)发出申请,认证结点 1 收到申请后,计算:$S_1 = k_1 \beta_1$,并将 S_1 发送给下一个合作

认证结点(可以随机选择,例如,认证结点 2);

(2)认证结点 2 收到 S_1 后,计算 $S_2 = S_1 + k_2\beta_2$,并将 S_2 发送给下一个合作认证结点(可随机选择,如认证结点 3);

……

(3)认证结点 i 收到 S_{i-1} 后,计算 $S_i = S_{i-1} + k_i\beta_i$,并将 S_i 发送给下一个合作认证分量(可随机选择,如认证结点 $i+1$);

……

(4)认证结点 t 收到 S_{t-1} 后,计算 $S_t = S_{t-1} + k_t\beta_t$,用认证公钥 PSK 来验证是否是认证私钥 SSK,若不是,则本次生成认证私钥失败,将本认证结点 t 作为认证结点 1,重新开始合作生成认证私钥 SSK;否则,用 S_t 作为认证私钥 SSK 签发证书,并将其发送给移动结点 m.

3　发现和清除具有 Byzatine 行为的认证结点

在被攻破的认证结点中,有一类被攻破认证结点具有 Byzatine 行为,即它们总是提供不同假私钥分量来破坏认证私钥的生成.因此,我们采用分布式容错处理技术中 Lamport 解决 Byzatine 将军问题的协同一致算法[6]来发现 $m(m <= (n-1)/3)$ 个具有 Byzatine 行为的认证结点并将它们从认证中清除出去.检测具有 Byzatine 行为认证结点的 Lamport 协同一致算法一共有三步:第一步,认证结点交换各自的私钥分量;第二步,认证结点将收到的私钥分量组成一个私钥分量向量;第三步,认证结点交换所形成的私钥分量向量,每一个认证结点都得到一组私钥分量向量,并检测该组私钥分量向量中的 Byzatine 错误以发现具有 Byzatine 行为的认证结点.由于每一个认证结点的私钥分量是机密的,不允许其他认证结点知道,所以,每一个认证结点 i 先用自己的公钥 PK_i 对其进行加密,然后,将其作为其私钥分量的替身发送给其他认证结点.每一个认证结点记录所发现的具有 Byzatine 行为的认证结点以便将它们从下次联合认证中排除出去.

4　基于多跳步加密签名函数签名的安全分布式认证

如前所述,基于 (n,t) 门槛加密分布式认证能够防止不足于 t 个被攻破认证结点(即不具有 Byzatine 行为认证结点)联合起来生成认证私钥 SSK,但是该方案本身存在着一个严重的安全隐患:从基于拉各朗日插值公式的 (n,t) 门槛加密分布式认证算法中,我们可以看出,如果认证结点 i 被攻破,那么,在其上的 i 个私钥分量 $k_1, \cdots, k_i$ 就会被攻击者很容易地计算出来,更为严重的是,算法中的认证结点 t 被攻破,则攻击者计算出的将是能够生成认证私钥 SSK 的 t 个私钥分量 $k_1, \cdots, k_i, \cdots, k_t$ 或者直接获得已生成的认证私钥 SSK.

为了保证任意一个认证结点 i 在合作签名时不暴露自己的私钥分量 k_i,特别是认证私钥 SSK,我们在 Sander 等人[10]提出的移动密码学基础上,提出了一种多跳步加密签名函数签名的方法,并将其应用到基于拉各朗日插值公式的 (n,t) 门槛加密分布式认证算法中.Sander 将移动密码学定义为"研究网络中与移动可执行代码信息安全性相关的数学方法",其中一个重要的概念就是用加密函数计算(Computing with Encrypted Functions)的协议简称 CEF 协议.CEF 协议的基本思想是:如果结点 A 上的函数 f 要传到结点 B 上将其数据 x 作为输入进行计算,那么,该函数 f 首先必须转换成某种其他函数(即对函数 f 进行加密) $E(f)$,其次,$E(f)$ 传送到结点 B 上将其数据 x 作为输入进行计算,最后,计算的结果传回结点 A 并进行解密得到 $f(x)$.CEF 协议的一个重要应用就是不可拆分数字签名(Undetachable Digital Signature),即保证一个源结点上的签名函数能够传送到目的结点上进行数字签名而不暴露源主机的私钥,其方法是用加密签名函数进行签名.然而,Sander 的不可拆分数字签名方法只适应源——目的两个结点之间用加密签名函数签名的情况,而 t 个认证结点在合作生成认证私钥时每一个私钥分量需要经过一个或多个认证结点进行相应的认证私钥生成计算,因此,我们在基于拉各朗日插值公式的 (n,t) 门槛加密分布式认证算法中,采用了一个多跳步加密签名函数签名的方案以确保在每一个认证结点上其他认证结点的私钥分量以及认证私钥不被泄露.我们在 RSA 算法的基础上利用指数函数来生成加密的签名函数从而隐藏了认证私钥 SSK,基本原理如下:

假设 d 是由 RSA 算法生成一对私钥和公钥,随机选择一个数 h,计算:

$$m = h^d \bmod n$$

此处的 n 是由 RSA 算法选用的,而非前文所提到的 (n,t) 门槛加密机制中的 n.

由上,可以生成签名函数:

$$f_{\text{signed}}(.) = m^{(.)} \bmod n$$

这样隐藏了真正的签名函数 $s(.) = (.)^d \bmod n$,我们称其为加密的签名函数,对于一个输入 x 即被签名信息(例如一个证书),计算:

$$z = f_{\text{signed}}(x) = m^x \bmod n = (h^d)^x \bmod n = (h^x)^d \bmod n$$

所得 z 即为 x 的加密签名,检验该签名的正确性,需先计算:

$$y = h^x \bmod n$$

再用公钥 e 验证等式 $y = z^e \bmod n$ 是否成立即可.

实际上在 RSA 加密算法下,因为 $z = f_{\text{signed}}(x) = (h^x)^d \bmod n = y^d \bmod n = s(y)$,所以,等式 $y = z^e \bmod n$ 直接检验的是隐藏的签名函数 $s(.)$ 作用于 y 的签名($z = s(y) = y^d \bmod n$),而由于 y 与 x 一一对应($y = h^x \bmod n$),因而也间接检验了加密签名函数 $f_{\text{signed}}(.)$ 作用于 x 的签名($z = f_{\text{signed}}(x)$).

下面给出了我们采用加密签名函数签名方法对基于拉格朗日插值公式的 (n,t) 门槛加密分布式认证算法的修改:

(1)设移动结点 C 需要对某信息的认证,就随机选取一个数 h,将 $(h, id\_C, req\_C, i=1)$ 发给第一个认证结点(其中,i 为计数器,id_C 是提出申请的移动结点名,req_C 是其认证请求),不妨设为认证结点 1.认证结点 1 收到 $(h, id\_C, req\_C, i=1)$ 后,比较计数器 i 和 t 的大小,如果 $i < t$:

计算:$i = i + 1$;

$m_1 = h^{h_1\beta_1} \bmod n$;

$m = m_1$;

生成函数:$f_{\text{signed}}(.) = m^{(.)} \bmod n = h^{k_1\beta_1(.)} \bmod n$

将$(id_C, req_C, h, m, f_{signed}(\cdot), i)$发给下一个随机选取的认证结点,不妨设为认证结点 2;

(2) 认证结点 2 收到$(id_C, req_C, h, m, f_{signed}(\cdot), i)$后,比较计数器 i 和 t 的大小,如果 $i < t$:

计算: $i = i + 1$;

$m_2 = h^{k_2\beta_2} \bmod n$;

$m = mm_2 = m_1 m_2 = h^{k_1\beta_1 + k_2\beta_2} \bmod n$

生成函数: $f_{signed}(\cdot) = m^{(\cdot)} \bmod n = h^{k_1\beta_1 + k_2\beta_2(\cdot)} \bmod n$

同样将$(id_C, req_C, h, m, f_{signed}(\cdot), i)$发给下一个随机选取的认证结点;

……

i:认证结点 i 收到$(id_C, req_C, h, m, f_{signed}(\cdot), i)$后,比较计数器 i 和 t 的大小,如果 $i < t$:

计算: $i = i + 1$;

$m_i = h^{k\beta_i} \bmod n$;

$m = mm_i = m_1 m_2 \cdots m_{i-1} m_i = h^{k_1\beta_1 + k_2\beta_2 \cdots k\beta_i} \bmod n$

生成函数: $f_{signed}(\cdot) = m^{(\cdot)} \bmod n$

$= h^{(k_1\beta_1 + k_2\beta_2 \cdots k\beta_i)(\cdot)} \bmod n$

继续将$(id_C, req_C, h, m\ f_{signed}(\cdot), i)$发给下一个随机选取的认证结点;

……

t:认证结点 t 收到$(id_C, req_C, h, m, f_{signed}(\cdot), i)$后,比较计数器 i 和 t 的大小,此时, $i = t$:

计算: $i = 1$;

$m_t = h^{k\beta_t} \bmod n$;

$m = mm_t = m_1 m_2 \cdots m_{t-1} m_t = h^{(k_1\beta_1 + k_2\beta_2 + \cdots + k\beta_t)} \bmod n$

生成函数: $f_{signed}(\cdot) = (m_1 m_2 \cdots m_t)^{(\cdot)} \bmod n$

$= h^{(k_1\beta_1 + k_2\beta_2 + \cdots + k\beta_t)(\cdot)} \bmod n$

$= h^{SSK(\cdot)} \bmod n$ (其中 $SSK = \sum_{j=1}^{t} k_j\beta_j$ 为认证私钥)

生成所需要的加密签名:认证结点 t 输入 x,(例如证书,它取决于用户的认证请求 req_C),

计算: $z = f_{signed}(x) = h^{SSK(x)} \bmod n$

$= (h^x)^{SSK} \bmod n$

这样就得到了 x 的加密签名 z,并且在生成加密签名函数过程中,认证结点 t 无需使用认证私钥 SSK(实际上,任一个认证结点都不存放认证私钥 SSK).

认证结点 t 对 z 进行检验:

计算: $y = h^x \bmod n$

因为在 RSA 算法下, $z = y^{SSK} \bmod n$,所以,如果 $y = z^{PSK} \bmod n$ (其中, PSK 为认证公钥)成立,说明是正确的,否则,重新开始合作生成加密的签名函数;

$t + 1$:认证结点 t 在得到正确的加密签名 z 后,将(x, z)发给请求认证的移动结点 C,移动结点 C 通过下列步骤来检验其真实性:先计算 $y = h^x \bmod n$ (此处的 h 即 C 在这次申请中提交给认证结点的随机数 h),再检验等式 $y = z^{PSK} \bmod n$ 是否成立以此来判断所得到认证信息 x 的真伪.

算法的安全性讨论:

(1) 如果有一个认证结点试图得到认证私钥 SSK,那么,它主要有两种方法:一是得到 t 个认证结点的私钥分量,它可以通过收集每一个(h, m_i)来解方程 $h^{ki} = m^i \bmod n$ 来得到 t 个私钥分量,但其难度相当于对 RSA 算法的选择明文攻击.此外,传送过程中携带的是若干个 m_i 的乘积 m,给单个认证结点收集 m_i 带来很大难度. 二是如果得到完整签名函数的认证结点 t 被攻破,则由于加了密的签名函数并不暴露认证私钥,所以,它只能通过破译 RSA 算法而得到 SSK,因此,本方法的安全性在相当程度上依赖于 RSA 算法的安全性.

(2) 本方法还可以有效地防止攻击者的重复认证攻击:移动结点 C 在每次发出认证请求时都随机选取一个数 h,当攻击者将所收集的旧认证信息(x, z)发给移动结点 C 时,移动结点 C 计算 $y = h_1^x \bmod n$ (h_1 是当前请求所选择的随机数),而 $z = (h^x)^{SSK} \bmod n$ 所包含的 h 是以前请求所选择的随机数,显然 $h \neq h_1$,所以, $y \neq z^{PSK} \bmod n$,从而移动结点 C 可以识别出签名是否伪造.

5 防范循环移动攻击的私钥分量刷新技术

当 n 个认证结点遭到循环移动攻击时,则我们采用私钥向量的同形特性进行私钥分量刷新. 同形特性是:如果私钥向量$(K_1^1, \cdots, K_n^1)$和$(K_1^2, \cdots, K_n^2)$分别能生成私钥 SSK_1 和 SSK_2,那么$(K_1^1 + K_1^2, \cdots, K_n^1 + K_n^2)$能生成私钥 $SSK_1 + SSK_2$. 令 $SSK_2 = 0$,则可以得到能生成 SSK_1 的新私钥向量.

下面证明基于拉格朗日插值公式的认证私钥分存方案满足同形特性:

设 $F(x) = SSK_1 + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1}$;

$G(x) = SSK_2 + b_1 x + b_2 x^2 + \cdots + b_{t-1} x^{t-1}$

对一组$(\alpha_1, \alpha_2, \cdots, \alpha_n)$,将 $\alpha_1, \alpha_2, \cdots, \alpha_n$ 分别代入 $F(x)$,得到私钥分量:

$K_1^1 = F(\alpha_1) = SSK_1 + a_1\alpha_1 + a_2\alpha_1^2 + \cdots + a_{t-1}\alpha_1^{t-1}$;

……

$K_i^1 = F(\alpha_i) = SSK_1 + a_1\alpha_i + a_2\alpha_i^2 + \cdots + a_{t-1}\alpha_i^{t-1}$;

……

$K_n^1 = F(\alpha_n) = SSK_1 + a_1\alpha_n + a_2\alpha_n^2 + \cdots + a_{t-1}\alpha_n^{t-1}$;

由拉格朗日插值公式:

$$F(x) = P(x) = \sum_{j=1}^{t} K_j^1 \prod_{i \neq j} \frac{x - \alpha_i}{\alpha_j - \alpha_i};$$

$$SSK_1 = F(0) = P(0) = \sum_{j=1}^{t} K_j^1 \beta_j;$$

同理,将 $\alpha_1, \alpha_2, \cdots, \alpha_n$ 分别代入 $G(x)$,得到:

$$K_i^2 = G(\alpha_i) = SSK_2 + b_1\alpha_i + b_2\alpha_i^2 + \cdots + b_t\alpha_i^{t-1};\ i = 1, 2, \cdots, n$$

$$G(x) = Q(x) = \sum_{j=1}^{t} K_j^2 \prod_{i \neq j} \frac{x - \alpha_i}{\alpha_j - \alpha_i};$$

$$SSK_2 = G(0) = Q(0) = \sum_{j=1}^{t} K_j^2 \prod_{i \neq j} \frac{-\alpha_i}{\alpha_j - \alpha_i} = \sum_{j=1}^{t} K_j^2 \beta_j$$

由上可得: $K_1^1 + K_1^2 = SSK_1 + SSK_2 + (a_1 + b_1)\alpha_1 + (a_2$

$$+ (b_2)\alpha_1^2 + \cdots + (a_{t-1} + b_{t-1})\alpha_1^{t-1};$$

……

$$K_i^1 + K_i^2 = SSK_1 + SSK_2 + (a_1 + b_1)\alpha_i + (a_2 + b_2)\alpha_i^2 + \cdots + (a_{t-1} + b_{t-1})\alpha_i^{t-1};$$

……

$$K_n^1 + K_n^2 = SSK_1 + SSK_2 + (a_1 + b_1)\alpha_n + (a_2 + b_2)\alpha_n^2 + \cdots + (a_{t-1} + b_{t-1})\alpha_n^{t-1}$$

令函数 $H(x) = F(x) + G(x) = SSK_1 + SSK_2 + (a_1 + b_1)x + (a_2 + b_2)x^2 + \cdots + (a_{t-1} + b_{t-1})x^{t-1}$. 因为 $K_i^1 + K_i^2 = H(\alpha_i)$ $(i = 1, 2, \cdots, n)$,所以,由拉格朗日插值公式得到:

$$H(x) = \sum_{j=1}^{t}(K_j^1 + K_j^2)\prod_{i\neq j}(\alpha^i - x)/(\alpha^i - \alpha^j)$$

当 $x = 0$ 时, $SSK_1 + SSK_2 = H(0) = \sum_{j=1}^{t}(K_j^1 + k_j^2)\prod_{i\neq j}(\alpha^i)/(\alpha^i - \alpha^j) = \sum_{j=1}^{t}(K_j^1 + K_j^2)\beta_j$;

即 $(K_1^1 + K_1^2, \cdots, K_n^1 + K_n^2)$ 能生成私钥 $SSK_1 + SSK_2$,满足同形特性.

私钥分量刷新过程:每一个认证结点 i 随机产生能生成私钥为 0 的私钥向量 $(K_{i1}, \cdots, K_{in})$,每一个私钥分量 kij 通过安全链路发送给认证结点 j,每一个认证结点 j 收集到 $K_{1j}, \cdots, K_{nj}$ 后,计算新的私钥分量 $K_j' = K_j + \sum K_{ij}$.

6　相关及进一步工作

目前,在国内,华中理工大学电子与信息工程系与西安电子科技大学 ISN 国家重点实验室分别提出了一种分布式移动通信身份认证密钥分发协议和一种移动通信中的密钥托管方案,他们只是分别对移动通信网的分布式密钥分发以及密钥托管进行了研究,并未采用 (n, t) 门槛加密机制由多个认证结点彼此协作提供分布式认证. 而在国外只有美国 Cornell 大学 Lidong Zhou 等人[9]首先提出了利用 (n, t) 门槛加密机制来增强 Manet 认证的安全性,并采用私钥分量刷新技术防范攻击者对 n 个认证结点进行循环移动攻击,然而,他们采用的是基于 (n, t) 门槛加密机制的集中式认证算法,并未用移动密码学中加密函数计算的方法(即加密签名函数签名的方法)来保护私钥分量和认证私钥不泄露以及利用 Lamport 分布式协作一致算法来发现和清除被攻破具有 Byzatine 行为的认证结点. 此外,在他们的论文中既未用一个具体的认证私钥分存方法来实现 (n, t) 门槛加密机制,也未用具体形式的认证私钥分量来证明私钥分量的同形特性.

进一步的工作:(1)在我们采用加密签名函数签名的基于拉各朗日插值公式的 (n, t) 门槛加密分布式认证算法中使用了计数器 i,一旦被攻击者篡改,就无法生成正确的签名函数,我们拟采用移动密码学中用加密数据计算的方法来加以解决;(2)如果在联合生成加密的签名函数过程中有认证结点提供了错误的私钥分量,那么,在第 t 个认证结点上就无法形成正确的加密签名函数,由于我们采用基于拉各朗日插值公式的认证私钥分存方法不具备自动纠错功能,只能重新从头开始生成加密签名函数,因此,我们将利用可纠错编码(如 RS 码)来分存认证私钥,这样在合作生成认证私钥时,就可以自动检测并纠正一定数目的错误私钥分量,无须从头重新开始生成加密签名函数.

7　结论

本文采用基于拉各朗日插值公式的认证私钥分存方法实现了 (n, t) 门槛加密分布式认证. 为了保证分布式认证的安全性,利用了分布式容错处理中 Lamport 分布式协作一致算法以及私钥分量刷新技术,并在移动密码学的基础上提出了一种多跳步加密签名函数签名的方法用以增加 (n, t) 门槛加密分布式认证的安全性. 因此,本文的安全分布式认证不仅能够容忍一定数量的认证结点崩溃,而且还能够有效防范内部被攻破认证结点对分布式认证的干扰,此外,还避免了私钥分量和认证私钥的外露,弥补了传统网络安全机制(公钥密码体制、数字签名、包序列以及单向 Hash 函数)解决 Manet 安全性的不足.

参考文献:

[1] ZJ Haas, Parlman. The performance of query control schemes for zone routing protocol [J]. ACM/ Trans. Net, 2001, 9(4): 427 - 438.

[2] D B Johnson, D A Maltz. Dynamic Source Routing in Ad-Hoc Wireless Networks [M]. Mobile Comp. 1996. 153 - 81.

[3] F Stajano, R Andersson. The resurrecting duckling: Security issues in Ad-hoc wireless networks [A]. Security Protocols, 7th International Workshop Proceedings [C]. LNCS Springer-Verlag, 1999. 102 - 105.

[4] R Gennaro, et al. Robust threshold DSS signatures [A]. Advances in Cryptography, Proc. Eurocrypt '96 [C]. Springer-Verlag, 1996.

[5] Y Frakel, et al. Proactive RSA [A]. Advances in Cryptography, Proc. crypt '97 [C]. Springer-Verlag, 1997. 440 - 454.

[6] Lamport, et al. The byzatine generals problem [J]. ACM Trans On Programming Languages and Systems, 1982, 4: 382 - 401.

[7] Y Desmedt, S Jajodia. Redistributing Secret Shares in New Access Structures and Its Application [R]. George Masion University, 1997.

[8] RFC1321. The MD5 Message Digest Algorithm [S].

[9] Lidong Zhou, Zygmunt J Hass. Securing Ad Hoc networks [J]. IEEE Network, 1999: 24 - 29.

[10] Tomas Sander, Christian Tschudin. Protecting mobile agents against malicious hosts [R]. Forthcoming LNCS, 1998.

[11] 曹鹏,等. 一种分布式移动通信身份认证密钥分发协议 [J]. 华中理工大学学报, 1999, 27(8): 95 - 97.

[12] 孙晓蓉,等. 移动通信中的密钥托管方案 [J]. 电子学报, 1999, 27(4): 137 - 139.

作者简介:

熊　焰　男,1960 年 8 月生于安徽合肥,博士、副教授,研究方向为分布式处理、移动计算、移动通信、计算机网络及信息安全. Email: yxiong@ustc.edu.cn.