一种基于神经网络权值同步的TinySec协议密钥更新方案

蔡家楣 严杜鹃 陈铁明
(浙江工业大学软件学院 杭州 310023)

摘 要 一种新的神经网络模型通过输出互学习可实现内部权值同步,将该模型用于安全密钥协商具有计算耗费低、通信量少等特点。在介绍权值同步模型的基础上,结合传感器网络安全协议TinySec的密钥更新问题,提出一种基于神经网络权值同步的轻量级密钥更新方案,有效解决了运行TinySec协议的节点密钥文件更新,增强了TinySec协议安全性。

关键词 传感器网络,神经网络,权值同步,密钥更新

Weight Synchronization Neural Network-based Key Update Scheme for TinySec Protocol

CAI Jia-mei YAN Du-juan CHEN Tie-ming
(College of Software Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract A new neural network-based model can synchronize its weight vectors by output-based mutual learning. Secure key agreement schemes based on such weight synchronization model is lower consuming as for computation and communication. Based on the weight synchronization model, a novel lightweight key agreement protocol was proposed using for key update of TinySec protocol. The proposed scheme can efficiently update the key file content for TinySec running nodes, which can improve the TinySec security level.

Keywords Sensor network, Neural network, Weight synchronization, Key update

1 引言

Berkeley大学开发的TinySec协议是无线传感器网络上第一个完全实现链路层安全的体系结构,已集成在TinyOS[1,2]系统上。TinySec不支持密钥管理,但可为应用层密钥管理模型提供安全传输服务。TinySec采用RC5,Skipjack等对称算法实现节点间的访问控制、数据完整性、数据保密性等安全服务,成为当前应用最广的传感网络安全协议。

TinySec作为一个研究平台用来测试和估算高水平安全信息包,但无密钥更新机制。它采用统一分发密钥的手段,部署前,每个节点均被分配以相同的密钥,节点之间通信一直使用该密钥进行加密、解密、认证以及密钥的协商和更新。该方法存在很大的风险性,一旦某个节点被俘,则密钥信息泄漏,危及整个网络的安全。因此,设计WSN上的密钥更新机制相当有必要。

2 TinySec协议介绍

TinySec协议由美国加州大学伯克利分校的Chris Karlof, Naveen Sastry, David Wagner 3位专家为无线传感器网络量身订造,是一种链路层加密机构。它依赖密码源语,设计使用了8位IV,并且使用密码分组链(CBC),其中默认的CBC算法为Skipjack,满足网络资源限制和安全要求[3]。

TinSec支持两种不同的安全选项:

(1)认证加密(TinySec-AE)模式。TinySec加密数据负载和使用MAC认证包。这个MAC由加密数据和信息包头计算得出。

(2)仅有认证(TinySec-Auth)模式。TinySec使用MAC认证整个的包,但是数据负载是没有经过加密的。

TinySec的核心是一种有效的分组密码机构,它使用一个单对称密钥在一组传感器节点间共享。在传输一个消息包之前,每个节点会首先加密数据并应用消息认证码(MAC)保护数据的完整性。接收者使用MAC来认证消息包在传输过程中没有被修改,然后解密消息。该协议采用最简单的加密机制——网络共享密钥。它提供一个基本安全等级、最大利用率和最小化的配置。任何被认证的节点可以与任何其他被认证的节点交换信息,所有的通信都进行加密处理。信息未经认证的节点将被拒绝。密钥分配相对来说简单,节点在发布之前与共享密钥装载一起。

这种方案的优点是计算复杂度低,由于网络中只有一个密钥,因此很容易增加新的节点;不过,缺点是网络的安全性较差,TinySec密钥机制无法抵御节点捕获攻击。如果对手危及单一节点或者获取了密钥,那么它就可以窃听通信,并在网络中任何地方注入信息。针对节点捕获的威胁,我们需要一个良好的密钥更新机制。

到稿日期:2008-05-15 本文受国家自然科学基金(60773115),国家863技术专题计划项目(2006AA10Z235),浙江省自然科学基金(Y106290),浙江省科技厅计划项目(2007C21008)资助。

蔡家楣(1946—),男,教授,研究领域为信息安全、软件工程;严杜鹃(1984—),女,硕士研究生,研究方向为网络信息安全;陈铁明(1978—),男,讲师,在职博士,研究方向为安全协议、信息安全,E-mal:tmchen@zjut.edu.cn。

3 神经网络权值同步模型

神经密码学的设想由意大利神经网络学家 Lauria 教授在 1990 年提出[7]，被国内外学者跟踪研究，已取得不少研究成果。最近研究表明，两个神经网络互相学习具备权值同步的特性[5,8,9]，利用该同步特性可进一步研究密码协议的应用，进而开辟了研究神经网络密码协议的新方法[10-12]。我们引入的基于神经网络互学习模型的密钥协商协议[4]，是利用该权值同步特性来设计的。

神经网络权值同步模型分单层神经网络互学习模型和多层奇偶安全模型。由于单层模型协商密钥的安全性完全依赖于输入向量的保密性，不能直接用于安全密钥协商，因此我们引入了树型奇偶机模型 TPM(Tree Parity Machine)[6]，基本结构如图 1 所示。它是一个含多个隐藏单元的多层树型神经网络，该模型通过增加隐含节点屏蔽单层神经网络节点的直接输出信息，有效保障了权值同步学习的安全性。

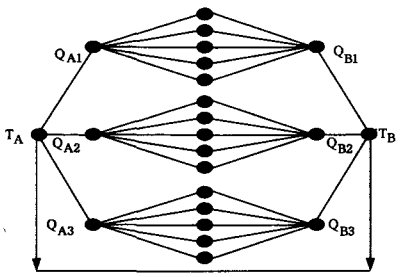


图 1 树型奇偶机(Tree Parity Machine)模型

TPM 模型是离散互学习模型，记输入个数为 N，W 为正交规范化的 N 维权向量，σ 为取值 $+1$ 或 -1 的神经元输出。权值向量 W 的取值限制在整数区间 $[-L,L]$ 上(权值边界 L 为一正整数)；输入向量 X 的取值为 $+1$ 或 -1 的二值序列，网络的最终输出为关于所有隐含层输出位的乘积，即满足：

$$\tau_{A(B)}=\prod_{i=1}^{K}\sigma_{A_i(B_i)}$$

假设 TPM 模型包含 K 个隐含单元，每个隐含单元拥有 N 维随机输入向量，第 k 个单元的输出为 $\sigma^k(t)$，则最终的输出为

$$\tau_{A(B)}(t)=\prod_{K=1}^{K}\sigma^k_{A(B)}(t)=\prod_{j=1}^{K}sign\left(\sum_{i=1}^{N}w_{A_i^j(B_i^j)}x_{A_i^j(B_i^j)}\right)$$

其中，$sign()$ 函数定于如下：

$$sign(X)=\begin{cases}+1, X>=0\\-1, X<0\end{cases}$$

TPM 模型初始时每个隐含层都是相同的高斯输入 $X_A^k=X_B^k$，且每一次相互学习后 X 的值随机变换，但始终保持 $X_A^k=X_B^k$。W_A^k，W_B^k 分别为神经网络的权值向量，且每一步仅在神经网络最终输出位相等时更新，即 $\tau_A=\tau_B$：，对于 K 个单层神经网络的一种权值更新规则设置如下[6]：

$$W^k_{A(B)}(t+1)=W^k_{A(B)}(t)+\rho^k_{A(B)}(t)X^k_{A(B)}(t)$$

其中，$\rho^k_{A(B)}=\begin{cases}\sigma^k_{A(B)}, & \tau^k_{A(B)}=\tau_A=\tau_B\\0, & \text{其它}\end{cases}$，$k\in\{1,2,\cdots,K\}$。

例如，当 $K=3$ 时，满足 $\Gamma_A=\Gamma_B=-1$ 的 $\sigma^1_{A/B}$，$\sigma^2_{A/B}$，$\sigma^3_{A/B}$ 有如下 4 种可能的情形：$(+1,+1,-1)$、$(+1,-1,+1)$、$(-1,+1,+1)$、$(-1,-1,-1)$。因此，第三方攻击者无法确定权值更新的内部神经节点，无法和正常通信的双方在相同时间内实现权值的同步。

基于 TPM 互学习模型设计的密钥协商方案，其实现过程为：两个神经网络在初始向量相等的情况下，随机产生各自的权值向量，并通过判断输出位是否相等来更新各自的权值向量，学习双方神经网络执行有限次输出位交互后，可使各自的权向量快速达到同步，再以该同步的权向量为参数，将其映射成一致的密钥，即可作为交互双方的交换密钥，具有快速、高效、安全等特点。基本过程如图 2 所示，具体分析过程可参考文献[4]。

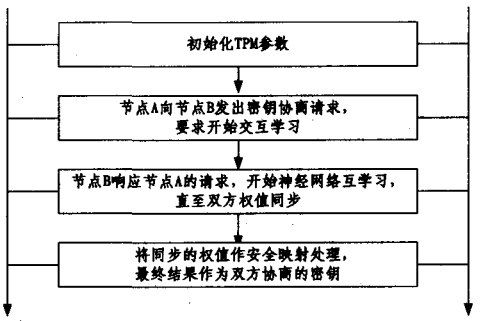


图 2 基于 TPM 神经网络互学习的密钥协商基本模型

4 新型密钥更新方案

4.1 方案设计

针对 TinySec 密钥机制的缺陷，我们引入基于神经网络互学习模型的密钥协商协议 TPM，提出一种新型密钥更新方案来实现 TinySec 密钥的更新其流程如图 3 所示。模型主要思想为下述 6 点(假设 n 个传感器节点)：

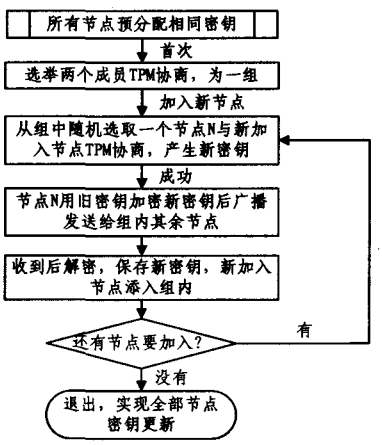


图 3 新型密钥更新方案流程

(1)所有节点预配置一个相同的共享密钥；

(2)推举两个节点 N_1，N_2 执行基于神经网络互学习的密钥协商协议，协商出共同的密钥，各自存放在密钥文件中，这两个节点成立一个组；

(3)新加入一个节点时，从组中随机选取一个节点与之进行 TPM 协商，产生新的共享密钥；

(4)然后，选出的节点使用旧共享密钥加密新共享密钥后，采用组播的方式传送给组中其它节点；

(5)组内其他节点收到消息后，用原先保存的密钥解密，获取最新的协商密钥。组中节点密钥更新完毕，保持同步状

态。

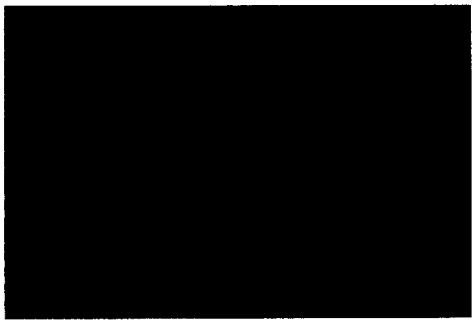
(6)新加入点 N_i,重复第(3)步,直至最后一个节点 N_n 执行完 TPM 协议,将其添加到组内,全部密钥更新完毕。

4.2 方案实现与测试

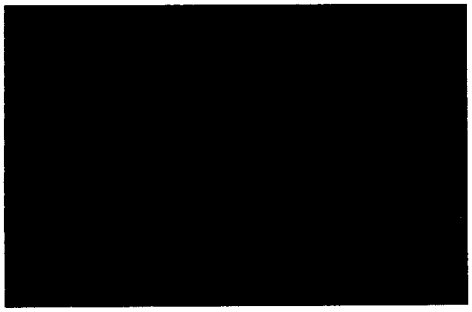
在每个传感器节点上部署协议程序开发包,通过 TinyOS 仿真平台实现。这里我们假设 3 个节点的情况,0,1,2 号节点。第一步,0 节点与 1 节点实现基于 TPM 的密钥协商,各自生成长度为 20 个字节的密钥文件,再经 md5 哈希函数转化成符合 TinySec 长度为 16 个字节的密钥:0.k 和 1.k。结果截图如下:

首先,0 节点与 1 节点协商出来的密钥为 430b455-d0295fc1e7ec94f62e53b057c5821188e,经 MD5 哈希后变为 4d7fdf01b2fcbfc450b87a0dbe024d3d,即为两个节点新密钥,各自保存在自己的密钥文件中。

当 2 节点加入要求更新密钥时,选择 0 节点与之再一次密钥协商。同样地,密钥保存在 0.k 和 2.k 文件里。接着 0 节点用旧的密钥加密新产生的密钥并发送到 1 节点,1 节点收到后用旧密钥解密,获取新密钥,保存在密钥文件里。



2 节点与 0 节点新协商出来的密钥为:ce337d738c301-a8683eccc6e86cbdda1



接下来,Before encrypting:ce337d738c301a8683eccc-6e86cbdda1,0 节点将该密钥加密发送给 1 节点,采用三重 Des 加密算法,密钥为之前的共享密钥 4d7fdf01b2fcbfc45-0b87a0dbe024d3d,得出 After encrypting:即为产生的密文,1 节点收到后用共享密钥解密,获得最新密钥 ce337d738c301a8683eccc6e86cbdda1。至此,3 个节点密钥更新完毕。

推广到 n 个节点的情况,可根据前面提出的方案仿真实现。通过单个节点逐一加入协商,生成新的共享密钥,同时更新组内其他节点密钥,直至最后一个节点也协商完成,则全部密钥得以更新。

结束语 TinySec 是一个易扩展的安全协议,支持高层通信协议。基于神经网络的密钥协商方案具有计算耗费低、通信量少等特点,适用于资源严格受限的无线传感器网络。本文提出的基于神经网络的密钥协商方案不依赖于传统公钥算法的大数运算,执行速度快、耗费资源少,可有效解决 TinySec 协议密钥快速更新。下一步的研究工作将集中在利用神经网络模型解决传感器网络组密钥的快速更新问题。

参 考 文 献

[1] http://www.tinyos.net

[2] 孙利民,李建中,陈渝,等.无线传感器网络[M].北京:清华大学出版社,2005

[3] Karlof C,Sastry N,Wagner D. TinySec:A Link Layer Security Architecture for Wireless Sensor Networks//Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems. Nov. 2004

[4] 陈铁明,蔡家楣.基于神经网络互学习模型的密钥协商协议[J].计算机研究与发展,2006,43(8)

[5] Kinzel W,Kanter I. Interacting neural networks and cryptography [J]. Advances in Solid State Physics,2002,42:383

[6] Volkmer M. Authenticated tree parity machine key exchange [OL]. 2004. iacr. org/2004/204. ps. gz

[7] Lauria F E,Neurocrytology O,Caianiello E R,et al. Parallel Architectures and Neural Networks // Third Italian Workshop [A]. Singapore,1990:337-343

[8] Metzler R,Kinzel W. Interacting neural networks. Phys. Rev.,2000,E 62:2555

[9] Kanter I,Kinzel W. The theory of neural networks and cryptography//Proceedings of the XXII Solvay Conference on the Physics of Communication. 2002:631

[10] Rosen-Zvi M,Klein E,Kanter I,et al. Mutual learning in a tree parity machine and its application to cryptography. Phys. Rev.,2002,E 66:066135

[11] Ruttor A, Kinzel W, Shacham L, et al. Neural cryptography with feedback. Phys. Rev.,2004,E 69:046110

[12] Chen T-M, Cai J-M. A Novel Remote User Authentication Scheme Using Interacting Neural Network // ICNC2005. LNCS3610. Berlin:Springer-Verlag,2005:1117-1120