一个可扩展的匿名通信协议

陆天波　方滨兴　孙毓忠　程学旗
（中国科学院计算技术研究所软件研究室，北京 100080）
（中国科学院研究生院，北京 100039）
E-mail：lutianbo@software.ict.ac.cn

摘　要　对 Internet 上的许多应用来说，匿名显得越来越重要。然而，目前的 Internet 协议并不具有隐藏通信端地址的功能。该文在分析现有匿名技术的基础上，提出了一个可扩展的 P2P 匿名通信协议 WonGoo。WonGoo 通过分层加密和随机转发取得了强匿名和高效率，它在延长匿名路径的同时减少了消息头部开销，提高了可扩展性。文中还分析了 WonGoo 的匿名性。WonGoo 提供与应用独立的实时双向匿名通信服务。

关键词　匿名　分层加密　随机转发　P2P

文章编号 1002-8331-(2005)07-0015-03　**文献标识码** A　**中图分类号** TP393

A Scalable Protocol for Anonymous Communication

Lu Tianbo　Fang Binxing　Sun Yuzhong　Cheng Xueqi
(Software Division, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080)
(Graduate School of the Chinese Academy of Sciences, Beijing 100039)

Abstract: Anonymity is increasingly important for many networked applications.Yet current Internet protocols provide no support for masking the identity of communication endpoints.This paper describes WonGoo, a peer-to-peer, scalable communication protocol that provides strong anonymity and high efficiency with layered encryption and random forwarding.It reduces the message overhead and adds the covert path length.This paper also analyzes the anonymity provided by this protocol.WonGoo provides application independent, real-time, and bi-directional anonymous communications that are resistant to both eavesdropping and traffic analysis.

Keywords: anonymity, layered encryption, random forwarding, P2P

1　引言

最近十来年，Internet 迅猛发展，极大地改变了人们的通信和生活方式。在 1990 年，除了科研领域的人之外，很少有人听说过 Internet；而在 1998 年，全世界 Internet 上的用户则多达 1.13 亿，2002 年 9 月上升到 6.056 亿[1]。然而，Internet 在给人们带来便利的同时，又给人们的隐私带来了很大的威胁。作为保护用户隐私的一种方法，匿名技术在最近二十多年得到了很快的发展，引起了人们的足够重视。研究者们已经作了很多工作[2~11]，希望能够解决 Internet 上的匿名问题。该文将研究基于 MIX[5]和 Crowds[9]的匿名方法，其原因在于到目前为止，这两个协议由于其实用性而被认为是解决 Internet 上匿名问题的最有前景的方法。

在基于 MIX 的协议中[2,3,6~10]，发送方必须知道一条确定的匿名路经，然后对消息进行分层加密。通常，路径越长，匿名性就越好[12]；但同时消息头部开销就越大，这极大地影响了匿名系统的效率，限制了其可扩展性。Crowds 通过随机转发，而不是分层加密取得匿名。因此，Crowds 不存在消息头部开销问题；而且，发送者只需知道匿名路径上的第一个中间节点。尽管 Crowds 的效率很高，但它不能抵抗较强的攻击，如被动全局攻击。如何取得匿名和效率的平衡是该文所要解决的问题。

WonGoo 是一个基于 MIX 和 Crowds 的可扩展 P2P 协议，用于低延迟匿名通信。它通过分层加密和随机转发取得强匿名性和高效率。WonGoo 通过延长匿名路径，同时减少消息头部开销而取得好的可扩展性。作为一个传输层的协议，它提供发送者匿名、接收者匿名和关系匿名[13]。

假定 WonGoo 的攻击者可以观察部分网络通信流；可以产生、修改、删除和延迟通信流；可以操作自己的 MIXes；可以控制网络中的部分 MIXes。然而，攻击者不能破坏协议所采用的加密技术。

2　背景与基本概念

MIX 的概念是由 David Chaum 于 1981 年在他的论文[5]中提出的，它的基本思想是通过使用中间节点来变换和混杂来自多个用户的消息，使窃听者无法确定输入消息和输出消息的对应关系，从而无法跟踪某条消息的传输路径，发现“谁跟谁通讯”的事实。

MIX 节点是指对邮件进行处理的一台计算机，MIX 系统由一个或多个 MIX 节点（MIXes）构成；用户在发送消息之前必须确定一条消息传输的路径，并获得该路径上所有 MIX 节点的公钥；随后构造如下消息：

基金项目：国家自然科学基金（编号：60273016）；国家 863 高技术研究发展计划基金（编号：2001AA142110）
作者简介：陆天波（1977-），男，博士研究生，主要研究方向：网络信息安全、分布式计算。方滨兴（1960-），男，研究员，博士生导师，主要研究方向：网络信息安全、并行计算。孙毓忠，男，研究员，主要研究方向：计算机系统结构。程学旗，男，研究员，主要研究方向：网络信息安全。

$M_0=K_0(R_0,M)$

$M_1=K_1(R_1,M_0,A_0)$

$M_2=K_2(R_2,M_1,A_1)$

...

$M_n=K_n(R_n,M_{n-1},A_{n-1})$

其中 K_n 到 K_1 是从用户到最终接收者的路径上所有 MIX 节点的公钥，A_i 和 R_i 分别是地址和随机填充数；下标 0 对应了最终接收者。

消息 M_n 被发送给路径上的第一个 MIX 节点，它用自己的私钥解密后将消息 M_{n-1} 发送给 A_{n-1} 指定的下一个 MIX 节点；该过程被每个收到消息的 MIX 节点执行，直到接收者 A_0 获得消息 M。可以看出，在未掌握 K_i 的情况下，是无法发现 M_i 和 M_{i-1} 的对应关系的。同时，为了隐藏输入输出消息的时序对应关系，节点还维护了一个消息缓冲池，并仅在缓冲池满时将消息乱序输出，必要时会产生掩饰流(Dummy Traffic)。

Chaum 的论文主要目的是提出一种解决电子邮件匿名性的方法，但后来的研究者已经把 MIX 扩展成一种通用的通信匿名性保护技术，其中影响较大的有美国海军研究实验室(The Naval Research Laboratory，NRL)主持的 Onion Routing[8]，零知识系统开发的用于隐私保护(主要是 Web 浏览)的 Freedom[3]，德国得累斯顿技术大学开发的用于匿名 Web 访问的 Web MIXes[2]，MIT 研究的网络层匿名协议 Tarzan[6]和瑞士联邦学院研究的应用层匿名协议 MorphMix[10]。它们的共同弱点是消息头部开销大，因而带宽利用率低，可扩展性不好。

Crowds[9]的目的是为用户提供匿名 Web 浏览，它使得用户能够匿名地从 Web 服务器取回信息而不对服务器和第三方泄漏用户自己的信息，如 IP 地址、域名等；其思想是“混在人群中”，意思是把自己隐藏在群体中。Crowds 把 Web 用户组织成一个称为 crowd 的群体，crowd 代表用户执行 Web 交易。crowd 中的用户用一个驻留在用户机器上的称之为 jondo 的代理表示。用户的请求通过本地的 jondo 随机地转发给另一个 crowd 成员或者直接提交给终端服务器。当一个 crowd 成员收到另一个 crowd 成员送来的请求时，它会随机选择是把该请求转发给下一个 crowd 成员，还是把该请求提交给终端服务器。这样，Web 服务器和其他的 crowd 成员以及第三方观察者就不能确定该请求是由谁发起的。随后的所有请求及服务器端的应答都是沿着这条路径传递的。Crowds 采用随机转发，没有过多的消息头部开销，因而其效率高，可扩展性好。但是，Crowds 协议并不能抵御通信流分析，它设计的主要目的是抵制 Web 服务器获得访问者的真实地址，对于拥有监听网络通信流能力的攻击者，它并不能维护足够的匿名性。

3 WonGoo 协议

这里提出一个新的基于 MIX 和 Crowds 的匿名通信协议 WonGoo。称发起匿名通信的一方为发送方(initiator)，用 I 表示，接受匿名通信的一方为接收方(responder)，用 R 表示。用 P_i 或 Q_i 表示节点 i。特别地，也用 P_{k+1} 表示接收方。而且，利用 K_x 表示节点 X 的公钥，$K(M)$ 表示利用密钥 K 对消息 M 进行加密。还采用了跳跳加密(link by link encryption)以抵抗较强的攻击。该文将交替使用词语‘对等点’和‘节点’。WonGoo 协议如下：

步骤 1：发送者 I 利用算法 $Fixed\text{-}Path(I,R)$(见图 2)创建一条固定的 MIX 路径 $I\text{-}P_1\text{-}P_2,\ldots,P_i,\ldots,P_k\text{-}R$。假设原始消息为 M_0，发送者 I 利用 $P_i(i=1,2,\ldots,k+1)$的密钥创建消息 M_1 并把它发送给 P_1。

$M_i=K_i(R_0,M_0)$

...

$M_i=K_i(R_i,M_{i+1},S_i,P_{i+1})$

...

$M_1=K_1(R_1,M_2,S_1,P_2)$

M_i 是指对等点 P_i 收到的消息，R_i 是随机填充数。S_i 是一个用于随机转发的参数，随后将说明其含义。

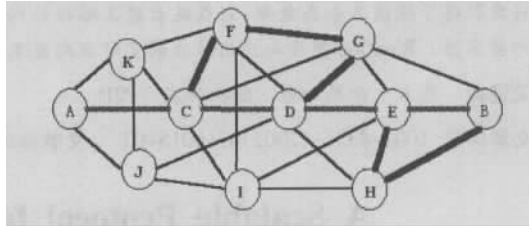

图 1 WonGoo 协议的一个实例

A 是发送方，B 是接收方；$A\text{-}C\text{-}D\text{-}E\text{-}B$ 是一条固定的 MIX 路径，$A\text{-}C\text{-}F\text{-}G\text{-}D\text{-}E\text{-}H\text{-}B$ 是一条 WonGoo 路径，其中 C，D 和 E 是由 Fixed Path 选择算法确定的，F，G 和 H 是随机选择的

Procedure Fixed-Path(P_{src},P_{dest})

1.P_{src} 选择匿名路径长度 k；

2.P_{src} 从 P_{dest} 的 IP 地址中抽取出前 24 位。例如，假设 P_{dest} 的地址为 159.226.41.180，则 P_{src} 抽取出 159.226.41，称之为 temp_IP；

3.P_{src} 随机选择匿名路径上的第一个节点 P_1，随后，$P_i(i=1,2,\ldots,k-1)$根据 temp_IP 随机选择下一个节点 P_{i+1}。并且，P_i 把 temp_IP 发送给 P_{i+1}。结果就形成了一条匿名路径 $P_{src}\text{-}P_1\text{-}P_2,\ldots,P_i,\ldots,P_k\text{-}P_{dest}$，使用 temp_IP 而不是 P_{dest}，原因在于使用 P_{dest} 将会把 P_{dest} 暴露给节点 $P_i(i=1,2,\ldots,k)$，结果降低了匿名性。

图 2 Fixed Path 选择算法

步骤 2：对等点 P_i 对收到的消息 M_i 进行解密并获得匿名路径上的下一个对等点 P_{i+1} 的地址。在 MIX 中，P_i 将把消息 M_{i+1} 直接转发给节点 P_{i+1}。然而，在 WonGoo 中，情况不是这样的。如果 $S_i\leq 0$，则节点 P_i 把消息 M_{i+1} 直接转发给节点 P_{i+1}；如果 $S_i>0$，那么 $S_i=S_i-1$，并且，P_i 进行随机选择以决定是否把 M_{i+1} 直接转发给 P_{i+1}。结果 P_i 以随机产生的概率 p_f 把 M_{i+1} 直接转发给 P_{i+1}，以概率 $1-p_f$ 创建消息 M_{i+1}'并把它转发给 Q_j(Q_j 是由 P_i 从其邻居中随机挑选的)。

$M_{i+1}'=(M_{i+1},S_i-1,P_{i+1})$

步骤 3：Q_j 收到消息 M_{i+1}'，并获得下一个节点 P_{i+1} 的地址。随后 Q_j 执行与 P_i 类似的操作。如果 Q_j 不是直接把消息 M_{i+1} 转发给 P_{i+1}，那么它创建消息 M_{i+1}''并把它转发给一个从其邻居表中随机选定的节点 Q_{j+1}。注意 $Q_j(j=1,2,\ldots)$仅仅进行跳跳加密操作，没有分层加密操作，而 P_i 执行了这两类加密操作。

$M_{i+1}''=(M_{i+1},S_i-2,P_{i+1})$

步骤 4：最后，M_0 经过节点 $P_1,P_2,\ldots,P_i,\ldots,P_k$ 到达 R，在发送方 I 和接收方 R 之间形成了一条 WonGoo 路径：

$$I\text{-}P_1\text{-}Q_1^1\text{-}Q_2^1,\ldots,Q_{s_1}^1\text{-}P_2,\ldots,P_i\text{-}Q_1^i\text{-}Q_2^i,\ldots,Q_{s_i}^i\text{-}P_{i+1},\ldots,P_k\text{-}R$$

随后的所有消息以及从接收方返回的消息都沿着这条路

径传递。

然而，随机转发可能会导致路径过长，这样，延迟就比较大。而且由于 P2P 系统的动态性，路径越长，其失效的概率就越大，因此，必须控制路径长度。该文通过发送者 I 解决这个问题。当 I 创建消息时，它在每一层填充进一个数 S_i，S_i 表示消息 M_{i+1} 在对等点 P_i 和 P_{i+1} 之间可经过的最多节点数。根据 S_i，P_i 决定是否把消息 M_{i+1} 直接转发给 P_{i+1}。

4 性能分析

这里主要比较 WonGoo 与 MIX 的匿名性。长的 MIX 路径的主要问题是过多的消息头部开销。因此，第一代 Onion Routing 规定所有路径长度均为 5 跳(hops)；Freedom 的最长路径为 3 跳。在 WonGoo 中，通过随机转发延长了匿名路径，同时减少了消息头部开销。这里将主要比较通过 Fixed Path 选择算法选定的节点与随机选定的节点之间的抗攻击性差异。

假设 L_1 是 $Alice(I)$ 和 $Bob(R)$ 之间的一条 WonGoo 路径，其形成过程如第二部分中所描述，P_i 是通过 Fixed Path 选择算法确定的，而 Q_j^i 是通过随机转发确定的，因此 P_i 和 Q_j^i 的功能是有差异的。L_2 是一条与 L_1 相同的 MIX 路径，P_i 和 Q_j^i 都是通过Fixed Path 选择算法确定的，因此 P_i 和 Q_j^i 具有相同的功能，这与 L_1 中的是不一样的。L_3 也是一条 MIX 路径。笔者的目标是比较 L_1，L_2 和 L_3 的匿名性。M_1，M_2 是 L_1 和 L_2 中的消息，它们具有相同的大小。

$L_1(L_2)$：$I-P_1-Q_1^1-Q_2^1,\dots,Q_{s_1}^1-P_2,\dots,P_i-Q_1^i-Q_2^i,\dots,Q_{s_i}^i-P_{i+1},\dots,P_k-R$

L_3：$I-P_1-P_2,\dots,P_i,\dots,P_k-R$

定理：攻击者不能分辨出 WonGoo 路径上的节点 P_i 和 Q_j^i，除非他已经控制了 P_i 和 Q_j^i。

证明：

(1)设攻击者只能被动地观察通信流。Q_j^i 收到 M_1 和 M_2 后，对 M_2 进行分层解密并转发，然而对 M_1 只进行转发。由于采用了跳跳加密，M_1 在经过 Q_j^i 的前后其形式是不一样的，这样攻击者就无法分辨出 M_1 和 M_2，即攻击者不能分辨出 L_1 中的 P_i 和 Q_j^i。因此在这种情况下，L_1 和 L_2 具有相同的匿名性。

(2)假设 $Q_j^i(1\leq j<S_i)$ 是一个被攻击者所控制的节点，则 Q_j^i 知道 M_1 是一个以概率转发的 WonGoo 消息，M_2 是一个 MIX 消息。在 L_1 中，Q_j^i 知道 Q_{j+1}^i 和 P_{i+1}，其中 Q_{j+1}^i 是由 Q_j^i 随机选择的，P_{i+1} 包含在 Q_j^i 所收到的消息中。然而，在 L_2 中，Q_j^i 只知道节点 Q_{j+1}^i。因此，L_1 对攻击者提供了更多的信息，结果是 L_1 的匿名性降低了。如果，P_i 和 P_{i+1} 之间只有一个节点 Q_1^i，或者 $Q_{s_i}^i$ 是一个被攻击者控制的节点，则 L_1 与 L_2 具有相同的匿名性。

(3)如果 P_i 被攻击者控制，则在 L_1 中，P_i 知道 Q_1^i 和 P_{i+1}，而 L_2 中的 P_i 只知道 Q_1^i，结果是 L_1 的匿名性降低了。

(4)假设除了 P_k 以外，L_1 和 L_3 中的 $P_i(i\neq k)$ 都被攻击者控制(如果 P_k 也被控制了，则匿名性就不存在了)。在这种情况下，L_1 和 L_3 具有相同的匿名性，尽管 L_1 可能比 L_3 长很多，因为攻击者根据在 P_i 处所获得的信息就可以跟踪消息了，而不用去管中间可能经过的 Q_j 节点。在其他情况下，L_3 将比 L_1 提供更多的信息给攻击者，前提是 L_1 和 L_3 中的 P_i 所处条件相同，原因在于 L_3 中的 Q_j 提高了攻击者获取信息的难度。因此，L_1 提供的匿名性不比 L_3 少。

为了对匿名进行评估，Garvish 和 Gerdes 在文献[14]中根据破坏匿名所需要联合起来的节点数定义了匿名复杂度。

定义：系统的匿名复杂度定义为不能破坏系统匿名的最大联合节点数。N 匿名，用 $OA(N)$ 表示，意味着必须至少联合 $N+1$ 个节点，才能破坏系统的匿名性。

根据这种定义，WonGoo 的匿名复杂度介于 $OA(k-1)$ 和 $OA(k-1+\sum_1^k\sum_1^{S_i}m)$ 之间，因为某些节点可能会被攻击者控制。而 MIX 的匿名复杂度为 $OA(k-1)$，与 MIX 相比，延长了匿名路径，同时减少了消息头部开销。这增强了 WonGoo 的匿名性，提高了其效率。另外，WonGoo 由于采用了分层加密，因而，与 Crowds 相比，它能抵抗更强的攻击。文献[7]建议了一种称之为 Inter-Mix 的方法，Inter-Mix 与该文的方法类似，但是它在加长匿名路径的同时，也增加了消息头部开销，这限制了它的效率和可扩展性。

5 结论与展望

WonGoo 综合了 Crowds 和 MIX 的思想，是一个强匿名、高效和可扩展的 P2P 协议。通过分层加密和随机转发，WonGoo 取得了匿名和效率的折中(Tradeoff)。与 MIX 相比，WonGoo 减少了消息头部开销，提高了效率，增强了可扩展性；与 Crowds 相比，WonGoo 能够抵抗更强的攻击。

笔者将进一步研究兼顾效率和匿名性的评估机制，以对 WonGoo 和 MIX 等相关协议进行评估，原因是目前的匿名评估机制[13]只考虑了匿名性，而忽视了效率，这样的评估方法是不全面的。而如何解决 WonGoo 的低层物理拓扑与上层逻辑拓扑之间的匹配(match)问题将是下一步工作中的又一个富于挑战的问题。(收稿日期：2004 年 12 月)

参考文献

1.http://www.nua.ie/surveys/how_many_online/

2.BERTHOLD O,FEDERRATH H.Web MIXes:A System for Anonymous and Unobservable Internet Access[C].In:International Workshop on Design Issues in Anonymity and Unobservability,Berkley,USA:Springer-Verlag,2001

3.BOUCHER P,SHOSTACK A,GOLDBERG I.Freedom Systems 2.0 Architecture.http://www.freedom.net/info/whitepapers/Freedom_System_2_Architecture.pdf,2000-12

4.CHAUM D.The dining cryptographers problem:Unconditional sender and recipient untraceability[J].Journal of Cryptology,1988;1(1):65~75

5.CHAUM D.Untraceable electronic mail,return addresses and digital pseudonyms[J].Communications of the ACM,1981;24(2):84~88

6.FREEDMAN M J,MORRIS R.Tarzan:A Peer-to-Peer Anonymizing Network Layer[C].In:the Proceedings of the 9th ACM Conference on Computer and Communications Security(CCS-02),2002-11

7.GüLCü C,TSUDIK G.Mixing E-mail with BABEL[C].In:Network and Distributed Security Symposium(NDSS 96),1996:2~16

8.REED M,SYVERSON P,GOLDSCHLAG D.Anonymous Connections and Onion Routing[J].IEEE Journal on Selected Areas in Communi-

(下转 29 页)

要进行时序关联规则挖掘时，首先要确定最小支持度 min_sup 和最小置信度 min_conf。时序关联规则的目的就是找出达到满足最小支持度 min_sup 和最小置信度 min_conf 的所有序列规则。它由两个过程实现这一步：首先是找出 U 中所有支持度不小于 min_sup 的序列，这样的项集称为频繁序列，比如设 min_sup×U 为 2，表 2 中的 bI_1d 就有 2 个，时间序列 bI_1d 就是频繁序列。第二步是用获得的频繁项集产生所有置信度不小于 min_conf 的序列关联规则。

查找频繁序列的处理上计算量非常大，一般要采用逐层搜索迭代的方法来找出频繁项集。下面 TimeSeq_Apriori 算法实现了这一过程，首先扫描序列数据库，直接得到候选的 1-项集 C_1，进而根据最小支持度 min_sup 获得所有的频繁 1-序列 L_1。然后根据 L_1 查找频繁 2-序列 L_2，依次进行下去，最终 C_k 通过函数 *apriori_gen*(L_{k-1})得到，从而根据支持度 min_sup 得到最大频繁序列 L_k。

应用到时间序列的 Apriori 算法和常规的算法区别在于候选集 C_k 的产生方法，当 L_1 产生以后，要通过联接 L_1 得到 C_2，这要考虑所有的时间间隔。下面的算法 *TimeSeq_apriori_gen*(L_{k-1}, *TI*)说明了这一问题：比如(b)和(c)属于 L_1，同时 $TI=\{I_0,I_1,I_2\}$，这样得到(b,I_0,c)，(b,I_1,c)，(b,I_2,c)，(c,I_0,b)，(c,I_1,b)，(c,I_2,b)候选序列，设($e_1\&_1e_2\&_2\cdots\&_{k-1}e_k$)是在 L_k 中一个频繁序列，($e_1\&_1e_2\&_2\cdots\&_{k-2}e_{k-1}$)和($e_2\&_2\cdots e_{k-1}\&_{k-1}e_k$)必须是频繁的，因为所有包括($e_1\&_1e_2\&_2\cdots\&_{k-1}e_k$)的都要包括上述子序列，采用类似的方式就可根据 L_{k-1} 得到 C_k，最后得到 L_k。

2.4 算法具体实现

TimeSeq_Apriori 算法实现如下：

```
{ L1=find_1-frequent_item(S);
  For(k=2;Lk-1≠Φ;k++)
{ Ck=TimeSeq_apriori_gen(Lk-1,TI);
  For each sequence s∈S
    { Ck=subseq(Ck,s)
  For each cadidate c∈Cs
    c.count++;
}
Lk={ c∈Ck|c.count≥min_sup}
}
return ∪Lk;
}
```

生成候选集的 *TimeSeq_apriori_gen*(L_{k-1},*TI*)具体实现如下：

```
{ Procedure apriori_gen(Lk-1,TI)
  For each sequence l1∈Lk-1 {
    For each sequence l2∈Lk-1 {
    If(k=2)then {
      For each time_interval i∈TI {
        c=L1×i×L2;
        add c to Ck;
        } }
        else
        { if(l1[3]=l2[1]∧l1[4]=l2[2]∧…∧l1[(k-1)*2-1]=l2[(k-1)*2-
3]∧l1[(k-1)*2]=l2[(k-1)*2-2]
    then { c=l1×l2;
    add c to Ck;}}}
    return Ck;
    }
```

3 开发实例

该文以某石油天然气公司作为应用背景，以几种关键设备作为实例，采集了历年实际运行的生产数据，对数据进行处理以后，存入数据库中，训练相应参数形成时序关联规则模型。

所开发的信息系统采用 Oracle 公司的 Jdevelop 工具，运行支持环境为 Java2.1.3。后台采用实时数据库(infoplus)系统采集现场数据，然后经数据处理存入采用 Oracle 9i(9.0.2)for Windows 数据库。Web 服务器采用包含于 Oracle 9iAS 内的 Oracle Container for J2EE(OC4J)for Windows。

4 结论

该文采用时间序列分析对某企业的大量历史数据进行分析，首先用模糊理论和常规方法对数据进行处理，找出偏离常规运行状态但未到报警界限的参数点，然后采用时间窗对参数离散处理，划分时间间隔得到时间序列数据库。然后对传统的 Apriori 算法进行改进，提出了基于关联规则的时间序列分析算法并编程实现，最终得到了按次序排列且有时间间隔的异常参数点对设备故障影响的规则，为企业设备运行时的故障监控提供了理论依据。(收稿日期：2005 年 1 月)

参考文献

1.丁祥武.序列模式的一种模型及其挖掘[J].中南民族学院学报(自然科学版)，1999，18(2)：44~48

2.Jiawei Han，Micheline Kamber.数据挖掘：概念与技术[M].机械工业出版社，2001：237~251

3.杨泽民，陈莉，范权润.加权关联规则的并行挖掘算法[J].计算机工程与应用，2003，39(8)：192~193

4.YenLiang Chena，Mei-Ching Chiang，Ming-Tat Ko.Discovering time-interval sequential patterns in sequence databases[J].Expert Systems with Applications，2003，25：343~354

5.Sergey Brin，Rajeev Rastogi.Mining Optimized Gain Rules for Numeric Attributes[J].IEEE Transactions on Knowledge and Data Engineering，2003，15(2)：324~338

(上接 17 页)

cations，1998，16(4)：482~494

9.MICHAEL K REITER，AVIEL D Rubin.Crowds：Anonymity for Web Transactions[J].ACM Transactions on Information and System Security，1998，1(1)：66~92

10.RENNHARD M，PLATTNER B.Introducing MorphMix：Peer-to-Peer based Anonymous Internet Usage with Collusion Detection[C].In：the Proceedings of the Workshop on Privacy in the Electronic Society，2002-11

11.SHERWOOD R，BHATTACHARJEE B，SRINIVASAN A.P5：A protocol for scalable anonymous Communication[C].In：2002 IEEE Symposium on Security and Privacy，2002：58~70

12.GUAN Yong，FU Xinwen，BETTATI Rand ZHAO Wei.An Optimal Strategy for Anonymous Communication Protocols[C].In：the Proceedings of the 22nd IEEE International Conference on Distributed Computing Systems(ICDCS 2002)

13.PFITZMANN A，KÖHNTOPP M.Anonymity，Unobservability，and Pseudonymity-A Proposal for Terminology.Draft v0.14，http：//www.freehaven.net/anonbib/papers/Anon_Terminology_v0.14.pdf，2003-05

14.GAVISH B，GERDES J H.Anonymous mechanisms in group decision support systems communication[J].Decision Support Systems，23(4)：297~328