

一种新的网络中层管理者方案

张建伟¹, 邹华胜²

(1. 南京气象学院数学系, 南京 210044; 2. 中国煤炭经济学院, 烟台 264005)

摘要: 提出一种新的基于 SNMPv2 的中层管理者方案以弥补 M2M MIB 的不足, 其轮询方法和报警机制优于现有同类产品, 且具有更好的安全性和互操作性。

关键词: 简单网络管理协议(SNMP); 管理信息库(MIB); 中层管理者
中图分类号: TP393 **文献标识码:** A

随着基于 TCP/IP 协议的 Internet 的广泛使用, 其网管技术愈来愈受到重视。IAB 于 1988 年提出简单网络管理协议(SNMP)^[1], 目的是为基于 TCP/IP 协议的网管系统提供标准的开发和运行平台。因 SNMP 具有简单、实用、灵活、占用资源少等特点, 十分适合互连网络管理。IETF 在 1993 年颁布了 SNMP 第二版 SNMPv2, 其中引入了分布式的管理方法, 其框架中做了八项重要的改进^[5,6]。然而, SNMPv2 的中层管理者只具备有限的监控能力^[2,7], 且缺乏对管理者信息处理任务的负载平衡, 这是 SNMPv2 对中层管理者功能定义的缺陷。文献[2]针对 SNMPv2 的不足, 开发了一个基于 SNMPv2 的中层管理者 SubManager, 但 SubManager 设计得不完善, 只具备简单的信息加工和存储能力, 缺乏报警、事件处理和过滤机制。作为补充, 本文给出一个较完善的基于 SNMPv2 的中层管理者设计方案, 取名 MLManager。它以 SNMPv2 为蓝本, 根据当前网管系统的需要, 从 1) 增强安全性与兼容性; 2) 使用动态轮询机制, 进一步提高带宽利用率; 3) 提高报警质量; 4) 能独立承担数据收集、加工和监控代理的任务等方面来改进现有网管系统的互操作性, 弥补现有商用中层管理者产品的不足^[3,4]。

1 MLManager 的总体结构

图 1 是 MLManager 的总体结构。其中 ML-MIB 由轮询控制表、历史表、阈值表、报警表等 8 个表和 2 个变量(eventtimer、maxtime) 组成。本文按 SNMPv2-SMI 定义和命名 ML-MIB 的变量。图 2 显示了 ML-MIB 在整个 MIB 树结构中所处的位置。下面分别介绍各模块的功能及模块与相关表的关系。限于篇幅, 文中略去有关算法及相关表的变量说明。

1.1 总控模块

总控模块由一个计时器程序和一个“客户/服务器”模式中的服务器进程^[10-12]组成。服务器进程随系统启动而启动, 一旦启动就处于等待状态, 并连续不断地测试下列两个条件, 若有一成立, 则作相应处理。

(1) 在某一 UDP 端口上侦听, 检查是否有报文到达, 若有, 则激活过滤模块进程去鉴别报

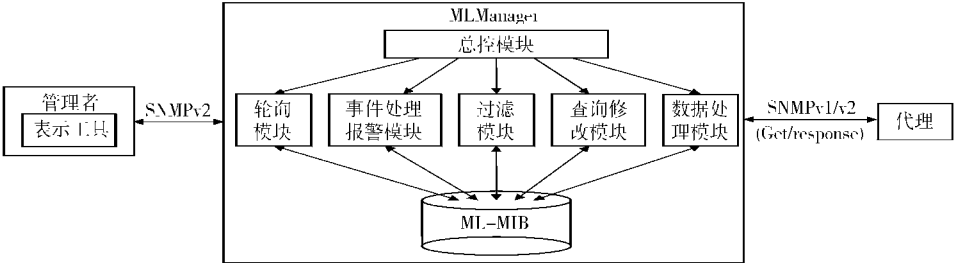


图 1 MLManager 的总体结构

Fig. 1 General structure of a MLManager

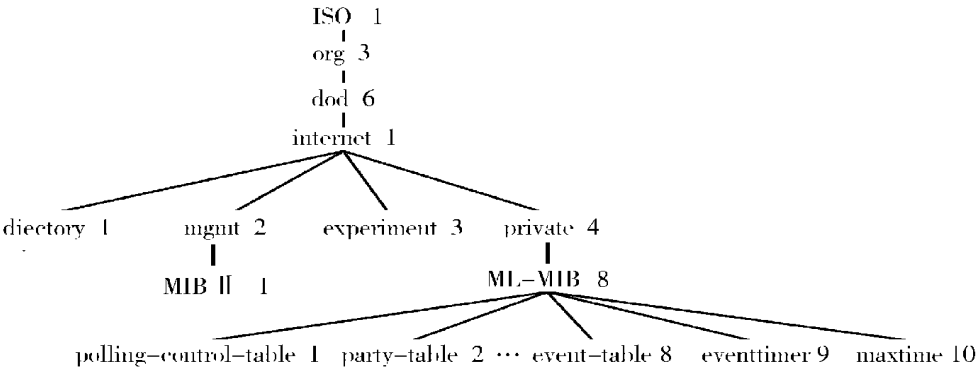


图 2 ML-MIB 在 MIB 树中的位置

Fig. 2 Position of ML-MIB in a MIB tree

文身份。

(2) 查询合法报文缓冲区, 检查合法报文队列是否为空, 若不为空, 则根据队首报文类型作如下处理: 1) 若报文类型为 trap 或 report, 则通知事件处理报警模块(进程)取走队首报文; 2) 若报文类型为 response, 则通知数据处理模块(进程)取走队首报文; 3) 若报文类型为 inform/set, 则通知查询修改模块(进程)取走队首报文。

计时器程序的任务是指示轮询时间和指示检查轮询结果值时间。

指示轮询时间: 在 polling-control-table 中已为每个代理 agentID 设置了一个计时器变量 timer, 其单位为 s, 最大值为相应代理的 maxInterval(tag= 0)或 minInterval(tag= 1)。当系统初始工作时, 计时器程序将所有代理的计时器变量均置 0, 并连续不断地检查每个代理计时器变量的值。若某个代理 agentID 的计时器变量 timer 已达到其最大值 maxInterval(tag= 0)或 minInterval(tag= 1), 则计时器程序立即激活轮询模块进程对代理 agentID 作轮询, 并同时置该代理的 timer 为 0, 重新开始计时。如此反复。

指示检查轮询结果值时间: 在 ML-MIB 中, 有两个特别的变量: eventtimer、maxtime, 它们的类型均为整型, 单位为 s。它们的关系为: maxtime 用于记录检查轮询结果值的时间间隔, 由系统启动时预先设置。eventtimer 在系统启动时也从 0 开始计时, 若达到最大值 maxtime, 则计时器程序激活事件处理报警模块进程检查事件匹配条件和类型结果值, 并同时置 event-timer 为 0, 重新开始计时。如此反复。

图 3 为总控模块与各功能模块和相关表(polling-control-table)的关系。图 4 为 Polling-control-table 的结构, 其中各项含义从略。

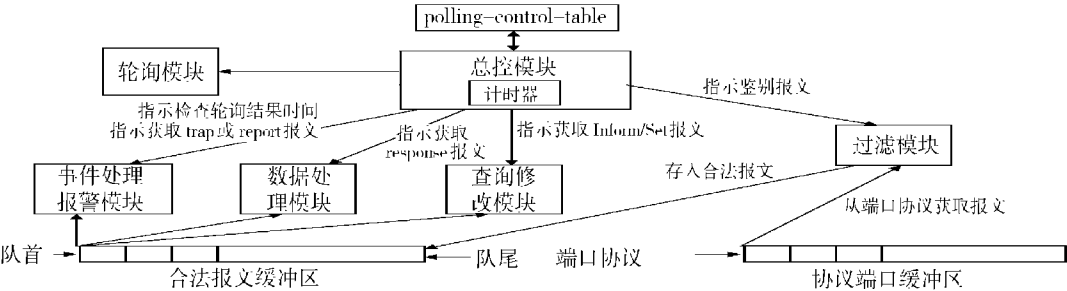


图 3 总控模块与各功能模块和相关表的关系

Fig. 3 The relations between the control-general module, functional modules, and the associated polling-control-table

pagentID	pstatus	pindex	timer	tag	maxInterval	minInterval	
resulttableSize	responsetableSize	eventtableSize	variable1	variable2	...		

图 4 Polling-control-table 的结构

Fig. 4 The structure of a polling-control-table

1.2 轮询模块

模块功能为在总控模块计时器程序的控制下,按 polling-control-table 的设置分别对各代理作轮询。在正常情况下,对各代理的轮询间隔为 maxInterval。若某代理对上一轮询无响应,则对该代理的轮询控制间隔立即减小为 minInterval,相应的时间窗也自动调整为最大值 maxtimeWindow。其目的在于放宽延迟控制,增加轮询次数,以避免代理的应答报文由于通信线路拥挤而被安全机制过滤掉,从而导致事件处理报警模块“误报”代理故障。若一个停止应答的代理发来响应报文,则对该代理的轮询间隔立即恢复正常。轮询算法从略。

图 5 是轮询模块与相关表的关系。图 6 给出 response-state-table 和 party-table 的结构。

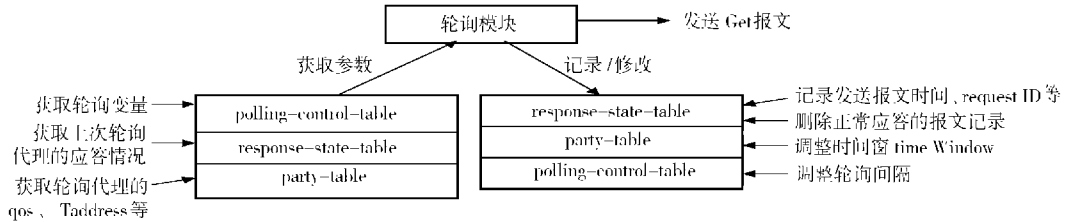


图 5 轮询模块与相关表的关系

Fig. 5 The relation between the polling module and the associated tables

1.3 过滤模块

模块功能为鉴别报文的身份。对于支持 SNMPv2 的管理者和代理使用安全协议^[8]鉴别报文;对于支持 SNMPv1 的代理使用团体方案鉴别报文。由于 M LManager 使用安全协议^[8]鉴别报文身份,因此它必须保持与被管代理的时间同步,在处理管理者报文与代理报文时,算法上有些差异。算法从略。图 7 是过滤模块与相关表(party-table、privilege-table)的关系。图 8 是 privilege-table 的结构。

1.4 数据处理模块

负责处理各代理对 M LManager 的轮询响应报文。数据处理模块是一个调度进程,它与各

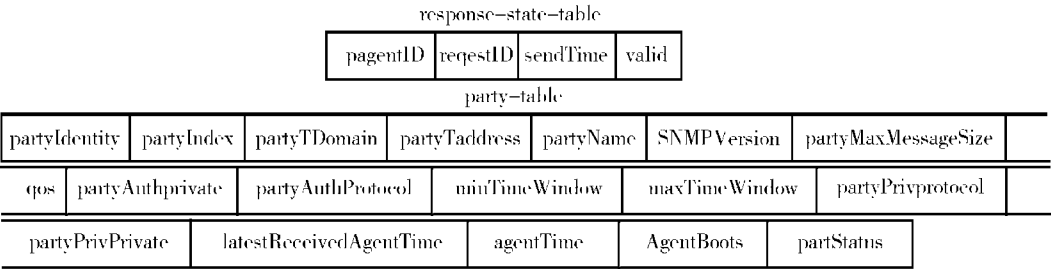


图 6 Response-state-table 及 part-table 的结构

Fig.6 Structures of response-state-table and part-table

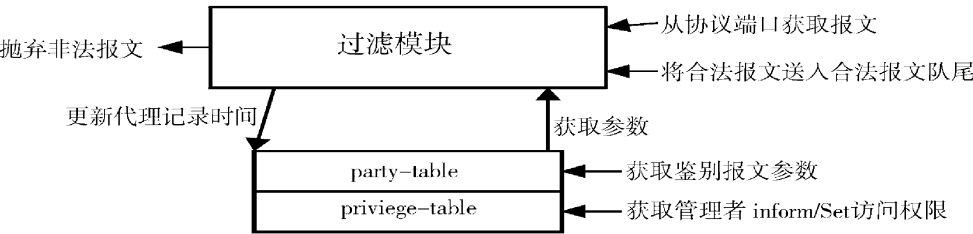


图 7 过滤模块与相关表的关系

Fig.7 The relation between the filter module and the associated tables

pvmanagerID	pvstatus	pvagentID1	pvagentID2	...	pvagentIDn
-------------	----------	------------	------------	-----	------------

图 8 privilege-table 的结构

Fig.8 The structure of a privilege-table

管理者(站)预先配置(或远程装入)的多个子处理模块一起协作完成对不同代理响应报文的处理任务。其中,每个代理的响应报文对应一个子处理模块,这些子处理模块平时驻留于外存,当它负责处理的报文到达时,才被数据处理模块调入内存,因此占用内存资源较少。更重要的是,采用这种设计方法,软件易于维护和扩展新的功能。若需要修改或增加对某代理响应报文的子处理模块时,只需修改或增加相应于该代理的子处理模块和控制参数。图 9 是数据处理模块与相关表(response-state-table、history-table)的关系。图 10 是 history-table 的结构。

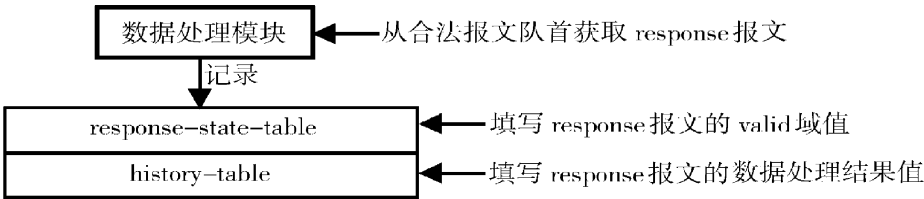


图 9 数据处理模块与相关表的关系

Fig.9 The relation between data-processing module and the associated tables

1.5 查询修改模块

查询修改模块的功能有两部分。(1) 允许上层管理者按 privilege-table 许可的权限访问

hagentID	moduleID	hstamptime	hstatus	value1	value2	...
----------	----------	------------	---------	--------	--------	-----

图 10 History-table

Fig. 10 History-table

ML-MIB 中指定代理在 history-table、response-state-table、event-log-table 中的历史所有值,以便了解某代理在某期间的运行状态,并负责产生响应报文。(2) 允许上层管理者按 privilege-table 许可的权限,修改指定代理在 polling-control-table、history-table、threshold-table、alarm-table、party-table 中的参数值,并负责产生响应报文。需要注意的是,MLManager 在产生响应报文时,必须按管理者在 party-table 中要求的 qos 进行。图 11 展示查询修改模块与相关表的关系。图 12 给出了 threshold-table、event-log-table、alarm-table 的结构。

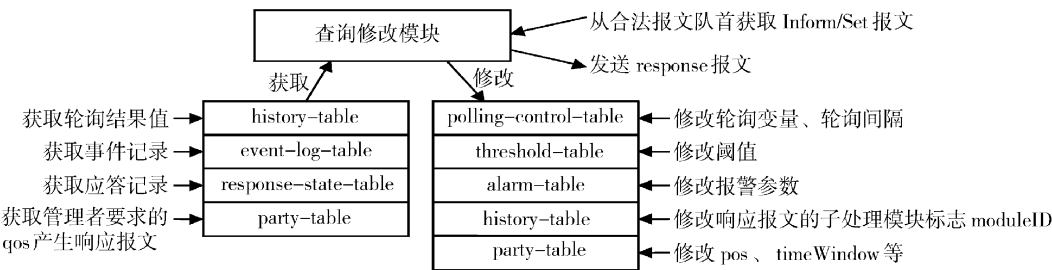


图 11 查询修改模块与相关表的关系

Fig. 11 The relation between the inquiring/modifying module and its associated tables

threshold-table					
tagentID	tstatus	minthreshold d1	maxthreshold d1	minthreshold d2	maxthreshold d2

event-log-table		
eagentID	eventtime	eventtype

alarm-table					
aagentID	astatus	amanagerID 1	amanagerID 2	...	amanagerIDm

图 12 Threshold-table、event-log-table、alarm-table 的结构

Fig. 12 The structures of the threshold-table、event-log-table、alarm-table

1.6 事件处理报警模块

报警机制是网管系统设计中的一个难点和关键点。建立高质量的报警机制,有助于管理者(站)及早、准确地发现隐患点,把网络故障消除在发生之前。本文的报警机制采用以下两个方法来控制报警的数量,以提高报警质量。(1) 使用RMON 的滞后机制(hysteresismechanism)^[9]来控制报警的数量,即在每个时段内只向管理者报告重要的、不重复的事件。(2) 报警时考虑代理间的互联关系,以提高报警质量。

在图 13 所示的网管系统中,若路由器故障,则路由器后的所有代理均会对中层管理者的轮询停止应答。若不考虑代理间的互联关系,则中层管理者会向管理者发出多个报警。考虑代理间互联关系的报警机制只报告路由器故障,而把路由器后的所有代理报告为状态未知。后一种报警机制更有助于管理者准确地诊断网络故障,因而更加合理。

依据代理间互联关系进行报警的关键是建立 MLManager 到需轮询代理的路由图。文献

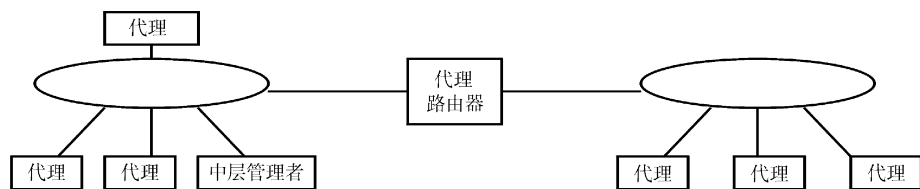


图 13 一个网管系统

Fig. 13 Schemetic of a network MLManager system

[13] 介绍了利用信宿 IP 地址建立网络拓扑的方法, 该方法实现的关键是利用 traceroute(探索路由程序) 获取从信源通往信宿的全部路由。

由于 MLManager 和 MLManager 需轮询代理的地址存放于 party-table 中, 是已知的, 构造 MLManager 到各代理路由图的方法是以 MLManager 为信源, 分别以各代理的 IP 地址为信宿, 利用 traceroute 获取从信源通往各信宿的路由, 根据获取的路由信息, 可建立以 MLManager 为根的路由图。图 14 是图 13 中网管系统的路由图。

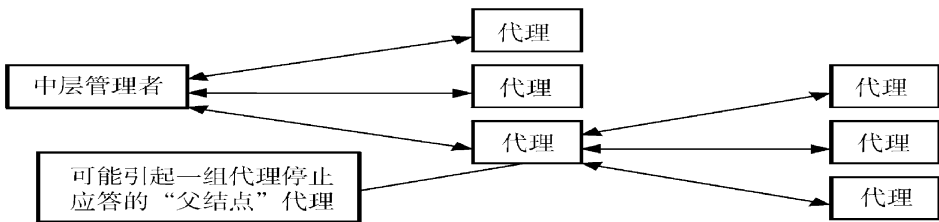


图 14 MLManager 到各代理的路由图

Fig. 14 Schemetic of route from MLManager to agents

当有一组代理同时停止对 MLManager 的轮询应答时, MLManager 的事件处理报警模块按深度优先遍历已建立的路由图, 获取这组代理间的“互联关系”, 并依据它建立更合理的报警方法: 若这组停止应答的代理中有一个是其余代理的“父结点”(如图 14), 则报告该代理停止应答, 而将其后的代理报告为状态未知。如不存在这样的“父结点”, 则分别向管理者报告每个停止应答的代理。

事件处理报警模块的功能: 1) 向上层管理者有条件地转发代理的 trap 报文; 2) 向上层管理者无条件地转发代理的鉴别失败 report 报文; 3) 及时发现无应答或有应答错误的代理; 4) 及时发现被监控参数值超过阈值的代理; 5) 依据 MLManager 和 MLManager 需轮询代理的 IP 地址建立并存储 MLManager 到各代理的路由图; 6) 有条件地向上层管理者报警。

事件处理报警模块每隔 maxtime (s) 被总控模块激活一次, 其任务如下: 1) 发现近期内超过阈值的所有代理—按 threshold-table 的设置依次检查 history-table 中各代理在近期内 的轮询结果值; 2) 发现近期内无应答或有应答错误或鉴别失败的所有代理—查看 response-state-table 中各代理在近期内 的应答情况; 3) 处理报警。

报警条件为: 1) 一个代理在近期内连续两次无应答, 且从 MLManager 到该代理的路径上没有停止应答的代理(遍历已建立的路由图); 2) 一个代理在近期内连续两次有应答错误(可能是代理软件故障); 3) 监控结果值低于下限阈值; 4) 监控结果值高于上限阈值; 5) 下属代理发来 trap 报文; 6) 下属代理发来 report 报文, 且报告的原因为鉴别报文失败; 7) 在 event-log-table 内近期无相同的事件记录(超过上限阈值、低于下限阈值被认为是不同的事件)。

处理报警为: 1) 若一个备监控的代理满足 a 和 g 或 b 和 g 或 c 和 g 或 d 和 g 或 e 和 g 或 f, 则在 event-log-table 中记录事件类型和事件时间, 并立即按 alarm-table 的设置向指定的管理者发送 SNMPv2trap 或 report 报文。否则, 只在 event-log-table 中记录事件类型、事件时间, 但不报警; 2) 若一个代理停止应答且沿 MLManager 到该代理的路径上有停止应答的代理, 则也在 event-log-table 中记录该代理停止应答的时间, 事件类型为“状态未知”, 并按 alarm-table 的设置向指定的管理者报告该代理为“状态未知”; 3) 若一个代理发来 report 鉴别失败报文或超时报文或越权访问管理信息报文, 则在 response-state-table 相应于该代理的轮询记录域 valid 中分别填入 3 或 4 或 5; 4) 若某代理超过阈值, 则根据该代理的 agentID、requestID 在 response-state-table 表域 valid 中置 2。

图 15 是事件处理报警模块与相关表(history-table、treshold-table、response-state-table、alarm-table、event-log-table、partytable) 的关系。

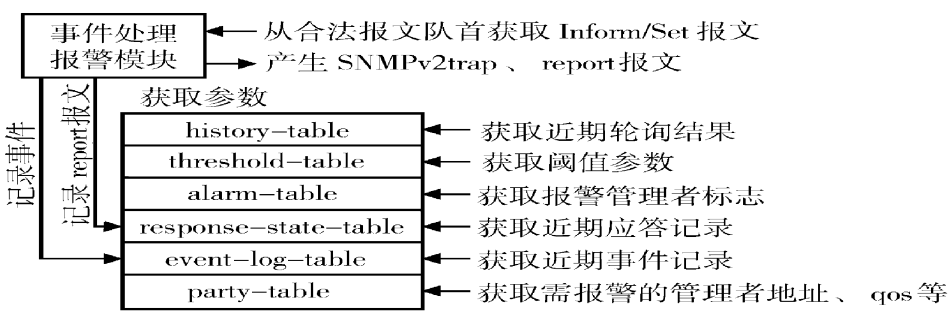


图 15 事件处理报警模块与相关表的关系

Fig. 15 The relation between the alarm module and its associated tables

2 评 述

随着网络规模和复杂性的增加, 网络管理正朝分布式方向发展。分布式网管环境中, 中层管理者的作用越来越受到重视, 管理企业(级)网络不可或缺。本文根据当前分布式网管系统的需求, 提出了一种基于 SNMPv2 的中层管理者设计方法, 它的管理信息库设计弥补了 M2M-MIB(SNMPv2 中层管理者管理信息库)的不足, 并具有现有同类产品自动轮询、数据处理和存储、数据处理和报警功能, 可作为它们的替代产品。本文的 MLManager 设计具有下列特点。1) ML-MIB 的设计遵循 SNMPv2-SMI 标准, 它收集的管理信息可供任何支持 SNMPv2 的网管系统抽取。2) 选用 IETF 最近推荐的安全协议作通信协议, 具有更好的安全性和互操作性。3) 采用动态的轮询机制, 与文献[18]中的同类产品的固定轮询机制相比, 能更及时地捕捉到异常情况, 从而大大提高带宽利用率。4) 使用 RMON 的滞后机制控制报警数量, 在报警时考虑代理间的互联关系, 使 MLManager 的报警质量优于现有同类产品。5) 对运行环境要求不高, 设计易于实现。6) 能从任何支持 SNMPv1、SNMPv2 的代理处收集管理信息。7) 使用标准的 SNMP 命令与管理者和代理通信, 因而易于集成到任何支持 SNMPv2 的网管系统中。

本文旨在就分布式网管技术做一探讨, 仅提出了一种设计方法, 具体实现尚有许多方面需进一步研究探索。

参 考 文 献

- [1] 徐 波, 周明天. TCP/IP 网络管理技术研究[J]. 微型计算机, 1996, 16(1): 20 ~ 23
- [2] MANFRED R G. Hierarchical network management: a concept and its prototype in SNMPv2[J]. Computer Network and ISDN System, 1996, 28: 441 ~ 452
- [3] 杨 锐, 白英彩. 网络管理专家系统研究[J]. 小型微型计算机系统, 1997, 18(2): 15 ~ 20
- [4] JANDER M. Midlevel managers ease SNMP information overload[Z]. Data Communication, 1993
- [5] 冯 明. 实用网络管理技术[M]. 北京: 人民邮电出版社, 1995
- [6] CASE J D. Protocol operations for version 2 of the simple network management protocol (SNMPv2)[R]. RFC1448, 1993
- [7] CASE J, Rose M, Waldbusser S. Manager-to-manager management information base[R]. RFC 1451, 1993
- [8] WATERS G. User-based security model for SNMPv2[R]. RFC1910, 1996
- [9] WALDBUSSER S. Remote network monitoring management information base[R]. RFC 1513, 1993
- [10] 朱三元编著. 网络通信软件设计指南[M]. 北京: 清华大学出版社, 1995
- [11] 周明天, 汪文勇编著. TCP/IP 网络原理与技术[M]. 北京: 清华大学出版社, 1994
- [12] 李伟琴, 张 辉, 王学民. 基于 SNMP 协议的网络管理系统[J]. 微型计算机, 1994, 14(2): 1 ~ 5
- [13] 高瑞振, 白英彩. 网络拓扑图的搜索实现[J]. 上海微型计算机, 1997

A NEW SCHEME OF NETWORK MIDDLELEVEL MANAGER

ZHANG Jian-wei¹, ZHOU Hua-sheng²

(1. Department of Mathematics, NIM, Nanjing 210044; 2. China Economic Institute of Coal, Yantai 264005)

Abstract: This paper presents a new scheme of SNMPv2-based midlevel manager. It is hoped that the new scheme can remedy the defects of M2M MIB (Management Information Base of SNMPv2 midlevel manager). Its polling way and alarming mechanism is superior to and its security and interoperability are better than the existing products of the same type.

Keywords: the Simple Network Management Protocol (SNMP); Management Information Base (MIB); midlevel manager