网络应用软件监控系统中管理代理的设计与实现①

Design and implement of the Management Agent in the Application Software Network Monitor System

费洪晓 裘方敏 康松林 谢文彪
(中南大学 信息科学与工程学院 湖南 长沙 410075)

摘要:文中实现的管理代理是基于SNMP的应用软件监控系统的关键模块之一,运行在网络分布环境中的受控站点上,是系统的通信中心,同时担负监控信息收集、组织和发送的任务。它利用SNMP的扩展协议分发和响应管理站点的控制命令,通过文件映射技术与监控模块共享监控信息,利用消息映射机制与监控模块实时交互,并能自主管理和独立处理紧急情况。文中还详细设计了其中的RRP协议和管理代理的安全性,有效的提高了系统的安全性和稳定性。

关键词:管理代理 SNMP 文件映射 消息映射

1 引言

目前SNMP(Simple Network Management Protocol,简单网络管理协议)的管理范围仅限于对网络硬件设备的管理,如路由器,集线器,交换机等,与网络硬件设备管理相比,网络应用软件的管理涉及较少[1,2]。随着网络技术的发展,出现了各种运行在网络环境中的应用程序,包括一些敏感的系统,如银行系统、证券系统等。为了保证这些应用程序安全高效地运行,必须对其进行有效的监控。通过分析研究现有的网络安全和网络管理的现状,提出了ASNMS(Application Software Network Monitor System,网络应用程序监控系统)系统。ASNMS是一个面向网络环境中应用软件的网络监控系统,实现了在管理站点对网络中的应用程序的实时监控,能及时发现被监控应用程序的异常状态,并及时采取有效的措施。ASNMS系统主要包含三个模块:监控模块、管理代理和管理站点[3]。文中设计实现的管理代理模块是ASNMS系统的通信中心,运行在受控站点上,它的功能的设计和实现对整个系统的实时和安全等性能具有十分关键的作用。

2 系统初始实现

ASNMS系统运行在网络中,因此系统的初始十分重要。初始化系统时,管理代理将网络中的受控站点情况发送给管理站点,使监控模块和管理站点建立联系,即根据当前受控站点的数目、连接情况等形成ASNMS系统。在初始化阶段,管理代理在系统的安全方面的作用致关重要。

管理代理随各受控站点自启动后的功能为:

(1) 监听管理站点发送来的要求注册的广播报文或者各个管理代理自身发送注册申请来与管理站点建立连接;

(2) 接收各个应用程序的监控模块的注册申请信息或发送要求应用程序监控模块注册的广播信息来与各监控模块建立连接。

2.1 网络连接建立

网络连接的建立,即管理代理和管理站点建立连接,包括申请和注册两个阶段,通过UDP协议实现,内容具体格式如图1。

字节	8	11	5	>=0
	UDP头	MCSP头	RRP头	RRP数据

图 1 RRP协议在UDP报文格式中的位置

其中MCSP是Monitor & Control System Protocol

① 基金项目:国家自然科学基金资助(60173041),湖南省自然科学基金资助(05JJ30119)

的简写，是在 SNMP 协议的扩充[3]。管理代理和管理站点通信连接的协议为 MCSP 中的 RRP（Register Request Protocol，注册请求协议）协议，设计的具体的格式如图 2 所示。其中，注册类型为 1 时为注册，为 0 时为申请。注册状态根据注册类型字段，为 1 时表示注册申请成功或注册成功。注册申请报文不包含 RRP 数据部分。

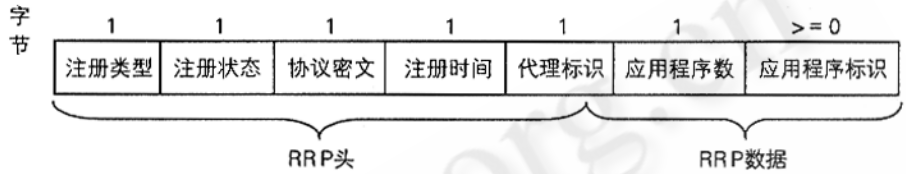

图 2　RRP 协议结构

在申请阶段，管理代理根据管理站点的要求注册的广播信息，发送注册申请报文，或者管理代理根据默认的管理站点信息发送注册申请报文。为安全起见，管理代理在注册申请报文中加入了一个长度为 1 个字节的协议密文，以标识身份。管理站点在收到管理代理的申请注册的报文后，读取注册类型、注册状态和协议密文，在确认代理的身份后，发送注册申请成功的报文，该报文中填充了相应的协议状态字段及协议密文字段，密文字段是经过特定的处理方式处理后的密文，以标识管理站点的正确身份。

在注册阶段，管理代理根据站点的相应报文及受控站点的监控的应用程序情况，填充注册类型、应用程序数和各应用程序标识字段（每个应用程序的标识为 1 个字节）等字段，然后发送该注册的报文给管理站点。管理站点在收到管理代理正式注册的报文后，确认管理代理的身份后，才发送注册成功的响应报文，其中站点填充的字段有注册状态字段和密文字段及注册时间字段。

2.2　站内连接建立

站内连接建立，即受控站点中管理代理和各个监控模块连接的建立，也包括申请和注册两个阶段，通过 windows 消息来实现。监控模块以线程的形式存在，管理代理利用 postThreadMessage[5] 函数与监控线程通信。向线程发送消息的函数原型为：

BOOL PostThreadMessage (DWORD idThread, UINT Msg, WPARAM wParam, LPARAM lParam)；

其中：idThread 为线程 ID 号；Msg 为消息字符串；wParam、lParam 为附加参数。

管理站点与管理代理间协议的注册消息参数设置如表 1。根据消息的参数的设置，实现相应功能的 Windows 消息，对应的消息的功能见表中的操作说明。

3　主要监控功能的实现

ASNMS 系统的监控功能包括受控站点内监控信息的收集发送和监控命令转发。监控信息的收集发送指在受控站点将监控模块监控到的信息收集到一起，组织成与管理站点协议的格式，并发给管理站点。监控命令的转发指管理站点发送监控命令到受控站点中特定的应用程序的监控模块，监控模块根据命令处理后，由管理代理发送响应。

表 1　协议的注册消息

消息字符串	WPARAM	LPARAM	操作说明
a2m_Reg_Respond	代理进程 ID	监控线程 ID 密文	代理同意注册的响应
a2m_Reged_Respond	协议密文	未用	代理注册完成的响应
m2a_Reg_Request	监控线程 ID	未用	监控模块申请
m2a_Reg	监控线程 ID	协议密文	监控模块注册

3.1　监控信息的收集发送

监控信息的收集发送包括监控信息的收集和组织发送。受控站点中，各应用程序的监控信息存储在各自的内存映射文件中。管理代理对监控信息的收集，其实是对内存映射文件的遍历。通过遍历内存映射文件可获得实时的被监控应用程序的状态，监控变量的值等。对监控信息的组织发送是为了将监控信息组织成与管理站点协议的结构，以便发送给管理站点。

监控信息采用内存文件映射[4]的方式来实现共享，是出于安全性和效率的考虑。因为内存映射文件相当于对内存的读写，同时内存映射文件还可以实现操作系统的内存机制自动回收功能，普通的文件无法

实现这样的性能。监控模块将监控信息以内存映射文件的形式存储在内存中，管理代理利用内存文件读写操作函数对监控文件进行读写，并运用有效的同步机制来防止管理代理与监控模块同时访问同一个监控信息文件而出错。

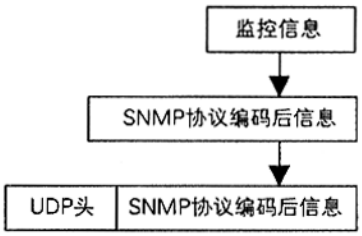

图 3　监控信息的组织

送给管理站点前，还要对 MIB 子树信息线性化处理。为了使监控信息通过网络发送给管理站点具有一定的安全性，首先对监控信息进行编码，使监控信息符合简单网络管理的标准，编码采用 SNMP。然后将编码后的数据用 UDP 包的形式发送给管理站点，监控信息的组织如图 3。同时，管理代理还提供对过大 UDP 包的分组发送机制，分组后填充相应 MCSP 协议的字段（MCSP 协议结构如图 4），保证管理站点在收到 UDP 包后能正确有效的重组。在管理站点收到代理发送的报文后，重组 UDP 包，然后按照 SNMP 解码，并将线性存储的 MIB 子树信息映射到管理站点的 MIB 数据库中供用户界面浏览。

3.2　监控命令的转发

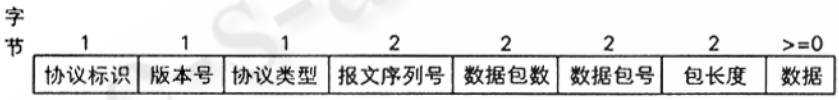

图 4　MCSP 协议格式

监控信息是以 MIB 树的形式存储，在逻辑上以二叉树形式反映应用程序中监控变量的结构（如所在的类、父类等信息）。设计中采用了三种节点，即应用程序节点、类节点和变量节点，分别存储相应的信息。管理代理采用中序遍历方式搜寻整 MIB 棵树，读取监控变量的数据值及所对应的监控模块、应用程序进行标识等信息。各个监控模块的内存映射文件的唯一标识为其注册时发送的注册消息的监控模块的线程 ID 再加上协议的字符串组成。

由管理代理定时遍历 MIB 树，将监控模块所监控的应用程序的监控变量的类名、变量名、变量类型、变量值取出，并在管理代理内暂存。对存储有监控信息的 MIB 子树的刷新有两种方式：

（1）由监控模块激发，即当监控模块检测到应用程序中监控到的变量发生改变时，发送消息给管理代理，管理代理再将变量改变的情况用 UDP 报文形式发送给管理站点，由管理站点根据具体的情况发送控制命令；

（2）有管理代理激发，即管理代理多次遍历 MIB 中均未发现有任何改变时，主动要求监控模块刷新数据监控文件，防止由于某些突发事件造成数据丢失。

管理代理通过中序遍历方式读取监控信息，在发

管理代理负责转发管理站点的各种控制给监控模块。管理站点的命令采用 UDP 包的形式发送到管理站点。管理代理根据接收的 UDP 包内容，识别各个命令的目标监控模块对象（通过应用程序的标识识别），然后通过 Windows 消息的形式发送给指定的监控模块，监控模块完成相应的操作后，再由管理代理给管理站点发送一个响应报文。

管理站点发送的命令有查询被监控的应用程序的状态、刷新被监控变量、锁定监控变量、锁定应用程序窗口等命令。命令类型的识别，在 MCSP 协议的协议类型字段中读取，相应 MCSP 协议的数据字段中填充特定的协议，如前面的 RRP 协议报文，此外还有 GDP（Get Data Protocol，获取 MIB 对象值协议），SDP（Set Data Protocol，设置 MIB 对象值协议）等报文[3]。通过协议的报文和 Windows 消息，管理代理、管理站点和监控模块三者协调工作。发送命令的 Windows 消息在受控站点注册有全局唯一的消息句柄，这样代理和应用程序监控模块就能通过该消息句柄识别消息类别。同样，出于安全的考虑，协议的报文和相应的消息函数参数中，携带有密文，以达身份识别验证的作用。消息映射技术能够满足受控站点内的通信的实时性及安全性要求。

4 管理代理安全性设计

管理代理除了以上几个主要功能的设计，为了达到系统的安全性和稳定性等要求，对代理进行下列功能的设计。

4.1 自动启动、隐藏和互斥功能

为了实现管理代理工作的隐藏性，提高 ASNMS 系统的安全性，管理代理应实现在受控站点上对用户的透明性，即管理代理能够随着受控站点的启动而自动启动，并且用户不能察觉其在工作。同时，管理代理应能够保证在受控站点上只有一个管理代理在运行，防止冲突的产生。

代理的自启动通过修改注册表中 Software \Microsoft Windows \Current Version \Run 下的键值或者通过修改 win. ini 文件中"[windows]" 数据段的值来实现代理的自启动。

代理的隐藏通过将管理代理设计 Windows 服务的形式来实现，即调用 RegisterSerivceProcess 函数在 Window 下隐藏应用程序，不让它出现在任务列表中。RegisterSerivceProcess 函数的主要功能是将程序注册成为一个服务模式程序，就是说使用此函数注册的应用程序，可以防止当注销用户时被其他程序关闭。同时可以实现在任务管理器的进程列表中被用户结束。还可以通过是去掉应用程序的标题来达到隐藏的目的[6]。

互斥即保证在同一时刻只有一个实例运行。可以使用一个 Win32 API 同步对象 mutex 来实现。当有新的代理启动时，查看 mutex，若发现还有其他的管理代理在运行就马上退出，这样就可以避免在受控站点中存在多个管理代理的混乱局面。

4.2 退出及异常处理

退出包括两个方面，一个是监控模块的注销，一个是管理代理自身的退出。对于监控模块的注销，管理代理通知管理站点该模块的退出，经同意断开与该监控模块的通信。管理站点收到信息后，释放对该模块监控的应用程序的监控以节省资源。管理代理本身则随受控站点的退出而退出，在退出前也同监控模块注销时的处理类似，要通知管理站点，避免通信异常的发生。当管理代理监控到监控模块工作异常或异常退出了，管理代理也具有定期清理的功能，通过查看内存映射文件中存储的各监控模块的状态变量实现。如果被监控的应用程序出现异常，管理代理也能采取相应的措施，如终止应用程序。

4.3 与监控模块封装

将监控模块与管理代理结合，使 ASNMS 系统最终提供给用户的是一个统一的模块，只要有监控模块的应用程序就能实现管理站点对其的监控。用户无需考虑管理代理的执行部分，只需设定所需监控的变量即可，使操作尽量简单易行。这样可以防止由于用户的疏忽或操作错误而造成监控模块与管理站点之间通信的中断，使监控数据无法及时上传，造成严重的后果。封装实现的一种方法是将管理代理作为监控模块资源的一个部分，当管理代理应故中断时，监控模块能够利用资源自启动一个管理代理，从而使监控能顺利的继续，使系统更具安全性。

5 小结

管理代理实现了管理站点和应用程序之间有效的交互通信，有效分担了管理站点的负担。ASNMS 系统在引入了管理代理之后，实现了分布式管理，适用当前网络高度分布的需要。本文介绍的 ASNMS 网络监控系统管理代理通过实验测试，基本实现预期的功能，满足系统的实时性和安全性要求。

参考文献

1 费洪晓、康松林，一种实现 MIB 信息传输的方法[J]，长沙铁道学院学报，2002，20(2)：43～46.

2 宋渊明、杨明，信息与计算机通信网络安全技术研究[J]，信息技术，2003，27(4)：44～46.

3 费洪晓、康松林、施荣华，基于 SNMP 的网络应用软件监控系统设计与实现，计算机工程与应用，2004，40(15)：122～125.

4 康松林、费洪晓、施荣华、彭凯，网络应用软件监控系统中监控模块的设计与实现，中南大学学报(自然科学版)，2004，35(6)：993～997.

5 陈坚主编，Visual C + +. NET 深入编程与实例剖析[M]，西安：西安科技大学出版社，2002. 116～129.

6 美国微软公司编著，Microsoft Windows CE Programmer's Guide[M]，北京：北京希望电子出版社，1999.