基于 Winsock SPI 技术的包过滤研究*)

甘利杰　丁明勇　杨永斌
(重庆工商大学计算机科学与信息工程学院　重庆 400067)

摘　要　本文介绍的 SPFireWall 防火墙实现了控管规则的制定、控管规则的修改与删除、底层分析与拦截程序、界面监视控制程序。通过进程间共享变量以及 WINDOWS 消息响应机制完成底层与界面的数据交流。运用 Winsock SPI 技术，编写服务处理函数替换 WINDOWS 的默认网络服务处理函数。本技术实现方便，在很大程度上也提高了效率。

关键词　包过滤，防火墙，网络安全，Winsock SPI

Research on Packet-Filter Based on Winsock SPI Technology

GAN Li-Jie　DING Ming-Yong　YANG Yong-Bin
(Computer Science & Information Engineering College, Chongqing Technology & Business University, Chongqing 400067)

Abstract　Make control rule, edit and delete control rule, analysis and dissuade program, GUI monitor program are SPFireWall's main tasks in this article. It can shift the Windows network service function to the author' service function by using Winsock SPI technology. Procedure-shared variable and Windows message mechanism complete data exchange between the DLL and the GUI. This firewall is a packet-filtering type firewall. It can be easily implemented, and can improve the efficiency greatly by using this technology.

Keywords　Packet-filter, Firewall, Network secure, Winsock SPI

1　引言

Internet 互联网是一种高信息量和即时性信息网络，它这样的特点就要求这个网络有极高的安全性，保证网络信息的可靠性。当然安全问题就成了 Internet 诸多技术中的一个至关重要的环节[1]。网络中最大的问题是怕受到攻击，攻击一般分为 4 种：中断(Interruption)、拦截(Interception)、修改(Modification)、伪装(Fabrication)。中断一般针对可用性上的破坏，特别是对硬件的破坏，如切断网线等；拦截是指非法的用户获得了网络中传输数据的操作，如窃听网络中的数据；修改是指非法的用户擅自窜改在网络中传输的数据；伪装是指非法的用户伪装成网络中某一个合法的用户。攻击又可分为主动攻击和被动攻击，主动攻击一般破坏性较大，但同时也比较好预防；被动攻击一般不好预防，因为我们有时很难检测到它的存在。网络中的系统的安全问题一是病毒的破坏，二是恶意攻击。同时它也可能利用病毒作为破坏工具，一般利用系统本身的漏洞或网络软件的漏洞，再就是应用软件的"后门"，一旦一个系统"后门"洞开，后果将不堪设想[2]。

近年来，网络安全已经成为人们关注的一个焦点，且有愈演愈烈的趋势。为了抵御来自网络的各种攻击，保证网络的安全，大多数人和企业都采用防火墙来抵御来自网络的攻击。一般来说，防火墙可以分为包过滤型的防火墙和应用网关型防火墙两大类。包过滤防火墙[1~3]的优点是成本较低，性能开销小，处理速度快。缺点是定义复杂，容易出现因配置不当带来问题，允许数据包直接通过，容易造成数据驱动式攻击的潜在危险。而代理防火墙的最突出的优点就是安全。由于每一个内外网络之间的连接都要通过 Proxy 的介入和转换，通过专门为特定的服务如 Http 编写的安全化的应用程序进行处理，然后由防火墙本身提交请求和应答，没有给内外网络的计算机以任何直接会话的机会，从而避免了入侵者使用数据驱动类型的攻击方式入侵内部网。但是相对来说，代理型防火墙速度较慢，不太适合用于高速网络。

包过滤防火墙中的关键技术包括网络封包的截获和解析。数据包的截获是捕获流进和流出的数据包，以用来对它进行解析。数据包的截获又有不同的类型，一是对全部的数据包进行捕获，二是随机的方式，再是根据一定的算法对特定性质的包进行捕获。数据包的解析是对截获的数据包的性质进行分析，分析数据包头文件，大小，使用的协议，近端 IP 及端口号，远端 IP 及端口号等。目前网络封包截获技术主要有三种：用传输层过滤驱动程序截获网络封包，用 NDIS 中间驱动程序截获网络封包，用 Winsock 2 SPI 截获网络封包[4]。

为了研制适合高速网络的防火墙，并且实现相对容易，本文采用 Winsock 2 SPI 技术，因为 Winsock 2 SPI 上手比较容易，而开发要求的其他知识也相对比较少，且其处于应用层，比较灵活，CPU 资源占有少，效率较好。

2　防火墙的结构设计

设计防火墙需要做两个大的方面的工作，一是底层的封包拦截与过滤程序(根据不同的防火墙还可以加入其他许多功能)；二是用于人机交互的界面程序。

要编写底层的封包拦截程序，首先需要了解 Windows 的工作方式、网络的协议结构以及这些协议在 Windows 操作系统中是如何实现的，如图 1 所表示。

甘利杰　主要研究方向：软件工程、操作系统等；丁明勇　主要研究方向：软件工程、操作系统等；杨永斌　主要研究方向：软件工程、操作系统等。

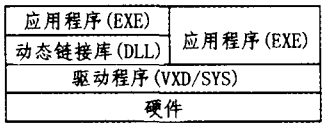

图 1 Windows 网络协议的实现

Winsock 引入了 SPI 的编程接口，这种技术可以在 Socket 中插入一层以完成诸如传输质量控制(QoS)、扩展 TCP/IP 协议栈、URL 过滤以及网络安全监控等功能，如图 2 所示。SPI 程序以动态链接库的形式存在，它工作于应用层，为上层 API 调用提供接口函数。在自己编写的 SPI 程序安装以后，所有的 Winsock 请求都会自动的发送到这个程序并由这个程序完成对网络的调用。不过，由于系统原来提供的 SPI 已经能够完成对网络的调用，因此一般来说，研究者都不会去重新编写其对网络的调用程序，而往往会直接使用系统提供的网络调用函数，既简单又安全，而这种工作模式也就是人们通常所说的“钩子”程序了。

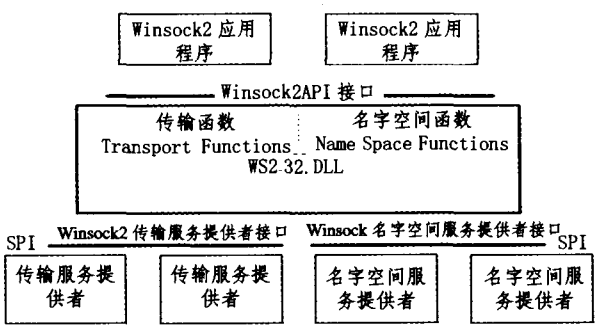

图 2 Winsock 提供者结构图

Winsock 2 SPI 除了完成网络传输的传输服务提供者(Transport Service Provider)，还提供了友好名称的名字空间服务提供者(Name Space Service Provider)，传输服务提供者传输服务提供者能够提供建立通信，传输数据，流量控制和错误控制等服务。名字空间服务提供者把一个网络协议的地址属性和一个或多个用户友好名称关联起来，这样就可以启用与协议无关的名字解析方案。

3 软件结构

数据包过滤软件主要由底层网络封包分析与过滤程序和界面应用程序组成。

3.1 底层动态链接库

3.1.1 封包分析模块

封包分析模块在底层动态链接库中，其主要功能是将由网络传过来的封包进行分析，判断发送封包的主机 IP 地址，判断接收封包的端口，得到该封包发送所使用到的网络协议类型等等。这部分内容可以说是整个防火墙软件工作的基础，分析得出的内容将成为后面防火墙软件决定是否放行或者拦截该封包的根本资料。同时，这部分分析得出的网络封包的信息也会实时地显示到界面应用程序的封包监视画面上以利于用户的监视与检查。除了检查收到的封包以外，如果有必要的话，此部分还能对本机发出的封包也进行相关的检查，并同时将检查得到的信息存放到系统的共用变量中，使得封包过滤程序能够用之进行处理。

3.1.2 封包过滤模块

封包过滤模块的主要作用在于对封包作出相关的动作，例如：拦截、放行等。此模块将与另外的两个模块交流。首先是封包分析模块。在此模块进行相关动作之前，必须先通过封包分析模块得到所以操作的封包的相关信息，它得到这些信息的手段则是通过共享的变量。在进行封包过滤时，此模块将会取出共享变量的值，同时，再通过与界面应用程序的控管规则所共享的相关变量值得到目前系统设置中的控管规则，然后将封包分析得出得相关信息与变量中保存得控管规则进行逐条对比。如果发现有完全匹配控管规则得封包，可以很容易得按照该条控管规则规定得动作来操作。如果没有发现完全匹配的控管规则，而只有一些关键字段匹配(应用程序名、网络 IP、工作时间等)，就会考虑根据人们习惯，从而对得到的最后一条匹配规则的管制动作去反进行操作。这部分内容是防火墙进行工作的主要依据，是规则的执行者。在执行规则的同时，由于与界面应用程序共享的进程间变量，也会同时根据用户的设置而得到实时的更新。

3.2 界面应用程序

3.2.1 封包监视模块

封包监视模块的主要工作便是将封包解析模块分析得到的封包信息实时的显示在页面上。封包分析模块在得到封包以后，首先分析封包的结构与相关信息，与此同时，也会以 Windows 消息的方式通知封包监视模块，并将分析得到的封包信息保存到相关的共享变量中。在封包监视模块得到解析模块发出的消息以后，将会从该共享变量中去取出分析后的信息，并将这些信息显示到界面上。这部分是用户借以了解底层动态链接库工作的窗口，因此也具有特别的意义。在此界面上，用户还可以自行控制是否要该界面对封包的接收与发送情况进行显示，也可以决定该界面的显示方式。

3.2.2 封包规则制定模块

封包规则制定模块的主要作用便是控制封包的过滤。用户通过该模块进行控管规则的设置、删除、修改等动作。首先，用户可以通过界面上的规则添加按钮向系统中添加新的控管规则，这些控管规则在添加并保存后会存储到磁盘上的规则文件中，用户添加规则以后，这些新的规则也会实时的加载到系统内存中，供底层封包过滤模块用以匹配。除此之外，用户还可以选择已经存在的控管规则进行编辑和修改。然而，在用户点击确定或者应用按钮之前，这些新添加或修改的控管规则只会存在于内存之中，只有当用户点击了确定或应用以后，这些规则才会被保存到磁盘上的控管规则文件中供下次调用。在这之前，用户随时可以点击取消按钮来取消掉所添加或修改的内容。而系统也会调用文件读写类来从原来的控管规则文件中读出原有的控管规则到内存。

4 软件实现

4.1 服务提供者函数

(1)动态链接库的入口函数

每一个 DLL 程序都需要有一个程序的入口点。此函数在系统调用 DLL 时起作用，这也是系统调用时的标准接口。

(2)服务提供者入口函数

WSPStartup 是服务提供者的入口函数即 Windows Sockets 应用程序调用 SPI 的初始化函数，如图 3 所示。在操作系统进行网络接收或发送时将会从这里出发来做相应的处理。在调用者(如 SPI 客户机)调用 WSPStartup 时，便通过一个被当作参数投送的函数派遣表打开另外的 30 个 SPI 函数，

(下转第 122 页)

从许多方面进行深入的分析和研究。尤其 Web 世界的信任问题，更是 Web 计算亟需面对和解决的重要问题。

参 考 文 献

1 TIM BERNERS-LEE. The world wide web: past, present and future[DB/OL]. http://www.w3.org/2002/04/Japan/Lecture.html, 1996.8
2 Khare R, Rifkin A. Weaving a web of trust[DB/OL]. http://www.4k-associates.com/trust.html, 1997
3 Kunii T L. Cyberworld modeling integrating cyberworlds, the real world and conceptual worlds[A]. In: 2005 International Conference on Cyberworlds (CW'05), 2005. 3～11
4 Berners-Lee T. WS-SW: integrating applications[DB/OL]. http://www.w3.org/2003/Talks/05-gartner-tbl/slide1-0.html, 2003.5
5 Wilson M, Matthews B. The future of the world wide web? [J]. BNCOD 2004, LNCS 3112, 2004.4～15
6 Golbeck J, Parsia B, Hendler J. Trust networks on the semantic web[A]. In: Proceedings of Cooperative Intelligent Agents. 2003
7 OASIS. XRI 2.0 FAQ[DB/OL]. http://www.oasis-open.org/committees/download.php/15695, 2005.12
8 Reed D, Strongin G. The dataweb: an introduction to XDI[DB/OL]. http://www.oasis-open.org/committees/download.php/5115/wd-xdi-intro-white-paper-2004-01-20.pdf, 2004.1
9 Mui L, Mohtashemi M, Halberstadt A. A Computational Model of Trust and Reputation[A]. In: Proc. of the 35th Hawaii International Conference on System Sciences, Big Island, Hawaii, IEEE (2002)
10 Heymans S, Nieuwenborgh D V, Vermeir D. Preferential reasoning on a web of trust[A]. ISWC 2005, LNCS 3729, 2005.368～382
11 Golbeck J, Grau B C, Halaschek-Wiener C. Semantic web research trends and directions[A]. PReMI 2005, LNCS 3776, 2005.160～169
12 Gelernter D. Mirror worlds: the day software puts the universe in a shoebox… how it will happen and what it will mean[M], Oxford University Press, 1991

(上接第 113 页)

传输服务提供者便由这 30 个函数组成。

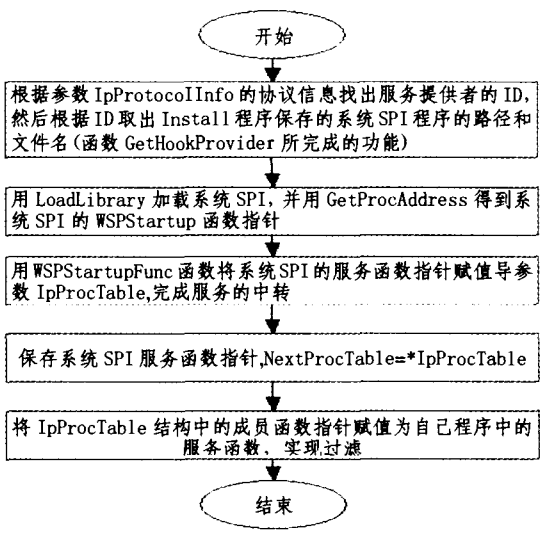

图 3 WSPStartUp 程序流程

WSPStartup 完成了对 SPI 服务函数的过滤，所有的 Windows Sockets 调用都会首先经自己的服务函数，然后自己的服务函数大多调用已经保存在 NextProcTable 中的系统服务提供者函数。

(3)得到服务提供者路径的相关函数

一个是用于得到此防火墙自己的服务提供者路径的函数，另外一个是用来得到已经被防火墙自己的服务提供者函数所替换掉的系统自己的服务提供者函数。这两个函数的实现都是通过读取注册表的相关信息来实现的。而注册表的相关内容则是在安装防火墙系统时，由防火墙安装程序在安装过程中写入的。

(4)用于提供服务的相关函数

Windows 用于提供服务的函数共有 30 个。可以视需要来重载这些函数。

4.2 管制服务提供者函数部分

这些函数通过调用与控管规则直接打交道的控管规则匹配管制函数来得到结果。

4.3 控管规则管制函数

这一部分代码主要用于匹配内存中的 Session 与控管规则，通过匹配得到的内容来决定是否放行封包。所有关于控管规则与 Session 函数的决策都在此一部分完成。

4.4 封包处理函数部分

封包处理函数部分的主要任务是在收到或发送封包的时候会在内存中创建 Session 来保存封包的相关信息，以供底层动态链接库的其他函数调用。这一部分的封包处理函数正是用来处理这些 Session 所用。在这一部分的函数中，主要包括了 Session 的初始化、Session 的创建、Session 的删除、查找、修改等。另外还有几个函数负责与界面应用程序的通信等。

4.5 界面程序设计与编码

界面程序设计部分，首先需要考虑的便是程序的人机界面。由于要获得用户输入的控管规则，因此控管规则添加部分是必不可少的，同时也需要让用户在添加完控管规则以后有个实时的反馈，所以需要有一个窗口单独显示控管规则的内容，选择使用 MFC 提供的 List 控件来完成这项工作。除了考虑控管规则的添加以外，还需要考虑控管规则的编辑与删除等。因此，在添加按钮后，又新加了编辑与删除按钮。同时，在此页面上也包含了设置系统工作模式的几个控件。用户只需通过点击这几个控件就可以选择系统的工作模式：全部放行、全部拒绝、询问。而底层动态链接库也会根据这里的工作模式的设置来决定其工作方式。在用户点击添加规则的按钮以后，还需要有一个页面来显示添加规则的界面，用户可以自行定制规则。另外，为了使该防火墙在用户最小化界面的情况下仍然能够使用，因此，构建了一个隐藏的主窗体作为在界面最小化时与底层动态链接库进行消息通信之用。

结论 Winsock SPI 编程，所编写的文件以 DLL 的方式存在。编程与调试都比较方便，同时其可移植性也较佳。只有告之开发者 DLL 所提供的接口便可以借用此 DLL 对表现层进行开发，而不需要另写底层代码。此防火墙工作对于系统资源的占有很少，同时，由于其工作在应用层，因此能够很灵活的对使用网络的应用程序进行管理。通过在过滤规则中选择相关的应用程序名称，用户就可以决定对这些应用程序的网络封包的操作动作，十分方便。除却必须到驱动程序层进行控制的协议类型以外，该防火墙对于其他的网络服务都可以有效的进行监控。

参 考 文 献

1 张琳，李璇华. 网络组建、管理与安全[M]. 人民邮电出版社，2000
2 Ziegler R. Linux 防火墙. 余青霞，周钢译. 人民邮电出版社，2000.10
3 Vacca J. Intranet 的安全性[M]. 电子工业出版社，2000
4 Aggarwal C, Wolf J L, Yu P S. Caching on the World Wide Web[J]. IEEE Transactions on Knowledge and Data Engineering, 1999, 11(1)