分 类 号　TP393 TP393.08 TP368.1　　单 位 代 码　10183
密　　级　秘密　　研究生学号　2200810

吉　林　大　学
硕　士　学　位　论　文

嵌入式 Internet 防火墙模型及其部分组件实现研究

Research on Model of Embedded Internet Firewall
and Realization of Some Components

作者姓名：袁　兵
专　　业：计算机软件与理论
指导教师
及 职 称：付　宏　　教 授

论文起止年月：2001 年 9 月至 2003 年 3 月

谨以此文献给我的母亲

母亲将一生的心血倾注给了我。她不但赋予了我生命，抚育我成长，并教导我刻苦学习、努力生活，是我生命的摇篮，成长的基石。我感谢母亲，感谢她的伟大；我思念母亲，思念她的勤俭；我怀念母亲，怀念她的坚强。我将母亲的一切深深地刻在心中，永不磨灭。

二零零三年二月六日十八点五十分

吉林大学硕士学位论文原创性声明

本人郑重声明 :所呈交学位论文，是本人在指导教师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期：　　年　　月　　日

声　　明

未经本论文作者的书面授权，依法收存和保管本论文书面版本、电子版本的任何单位和个人，均不得对本论文的全部或部分内容进行任何形式的复制、修改、发行、出租、改编等有碍作者著作权的商业性使用（但纯学术性使用不在此限）。否则，应承担侵权的法律责任。

提　要

随着计算机技术和通信技术的飞速发展，越来越多的嵌入式系统设备融入Internet，形成了新的Internet形式：嵌入式Internet。嵌入式Internet的出现带给Internet更多的信息安全问题，以IPv6技术、“后PC”时代、嵌入式Internet技术和分布式防火墙技术等为基础，研究解决嵌入式Internet的网络安全问题已成为目前嵌入式Internet技术领域发展的重要方向。

本文主要研究嵌入式Internet的安全问题，提出了新的嵌入式Internet防火墙技术。首先，通过对嵌入式系统的分析，探讨了嵌入式Internet的相关技术，建立了嵌入式Internet和扩展Internet模型，并在此基础上研究了课题所采用的嵌入式Internet环境；其次，在分析传统Internet存在的网络安全威胁基础上，对嵌入式Internet和扩展Internet的特点及其存在的网络安全威胁进行了分析，针对传统防火墙技术所存在的缺点，引入分布式防火墙技术，在分布式防火墙技术的基础上进行修改更新，提出了嵌入式Internet防火墙技术模型；最后，研究了基于PC硬件模拟嵌入式系统的实现技术，对各防火墙组件进行了技术分析，并对嵌入式Internet防火墙中终端设备防火墙组件的实现技术进行了详细研究。

文章通过探讨Internet技术的发展，建立新的Internet模型，在分析新模型所存在的安全威胁基础上，提出一种能够解决其网络安全的新型防火墙技术，并研究了其具体结构和相应的实现技术。希望能为未来的防火墙技术的发展提供理论和技术依据。

目　录

第一章 绪 论

嵌入式系统已经渗透到生活的各个角落——电器、玩具、汽车、电视、录像机、医疗器械、飞机等，逐渐增多的嵌入式系统应用也成了推动 PC 技术发展的动力，而“后 PC 时代”[1]也正是基于嵌入式系统应用提出的新概念。

随着 Internet 技术的日益发展，嵌入式系统的 Internet 应用越来越多，技术也越来越成熟，嵌入式 Internet[38]作为 Internet 的一部分，不仅面临着复杂的技术问题，而且已经成为人们关注和研究的热点。

1.1 课题的研究意义

随着计算机技术和通信技术的飞速发展，越来越多的嵌入式系统设备融入 Internet，形成了嵌入式 Internet 网络结构，扩展了传统 Internet 的应用范围，使 Internet 朝着更广泛的方向发展，人类也进入了一个崭新的信息时代。

在信息化社会中，计算机通信网络在政治、军事、金融、商业、交通、电信、文教等方面的作用日益增大。各种各样完备的信息系统使得人类社会的一些机密和财富高度集中于计算机中，并依靠计算机网络接收和处理信息，实现其相互间的联系和对目标的管理、控制。如：电子商务(Electronic Commerce)、电子现金(Electronic Cash)、数字货币(Digital Cash)、网络银行(Network Bank)等，以及各种包含嵌入式系统专用网的建设，使得相关的安全问题显得越来越重要，成了这些应用的关键之所在。因此，网络安全已经成为数据通讯领域研究和发展的一个重要方向。随着嵌入式 Internet 的广泛应用，嵌入式 Internet 网络安全技术的研究会成为 Internet 安全发展的一个新热点，也将是信息科学一个新的重要研究领域，必然会受到人们的关注。

随着 Internet 的普及，网络犯罪和攻击活动愈演愈烈。据统计，仅西方八国集团各个成员国每年因网络犯罪造成的损失就高达 420 亿美元，仅去年全世界就损失了约 16000 亿美元，其中美国损失了 2660 亿美元，英国损失了 540 亿美元。据 Mi2g 报道，截至到 2002 年 8 月，全球黑客袭击成功的次数达 30839 次，与 2001 年的 31322 次、2000 年的 7821 次、1999 年的 4197 次和 1998 年的 269 次相比有较大的增加。很明显，全球的黑客攻击行为在逐年增加[9]。

Internet 的网络安全性问题已经成为 Internet 发展过程中必须面对的主要问题，作为未来 Internet 的一部分，嵌入式 Internet 的出现一方面扩展了 Internet 的应用领域，另一方面也在网络安全问题上给 Internet 带来新的难题。由于嵌入式 Internet 的开放性、灵活性，恰好给网络攻击者提供了更为广泛的空间和更灵活的手段。嵌入式 Internet 继承了 Internet 的大部分特点，由于它所包含的网络终端设备一部分是嵌入式设备，因此将 Internet 的网络安全技术照搬到嵌入式 Internet 中去应用是不合适的，甚至可能导致负作用。嵌入式 Internet 的网络安全性问题如果得不到解决，那么 Internet 的网络安全性就根本无从谈起。研究与开发更为有效的 Internet 网络安全技术，加快对嵌入式 Internet 网

络安全技术的研究与开发，成了摆在我们面前的时代重任。

1.2 国内外研究状况

嵌入式系统的兴起是在 1971 年由 Intel 公司推出有史以来第一颗微处理器 4004 开始，典型的嵌入式系统几乎让人感觉不到它的存在[4]。近十几年来，随着 Internet 技术和硬件技术的发展，嵌入式硬件设备的 Internet 应用也越来越多。但受到 IPv4 协议的 IP 地址资源和计算机硬件技术的限制，大量的嵌入式系统接入 Internet 不仅要考虑到硬件成本，还需要注意接入技术成本。例如，emWare 公司提出了 EMIT 技术[8](Embedded Micro Internetworking Technology，嵌入式微因特网互连技术)，通过给 8/16 位嵌入式系统增加相应的软、硬件，解决了 8/16 位的嵌入式系统接入 Internet 的问题。随着 IPv6 的应用和计算机硬件成本的降低，32/64 位嵌入式系统将逐渐成为嵌入式系统的应用主流，嵌入式系统直接与 Internet 互连将成为嵌入式 Internet 的主要接入技术方式。

嵌入式 Internet 技术的日渐成熟，相应就会产生其他一些新的课题，譬如说，嵌入式 Internet 的安全问题。对于嵌入式系统的 Internet 应用，涉及到的技术范围很广，本文主要侧重研究嵌入式 Internet 网络安全防火墙技术问题。因此，以下就嵌入式 Internet 安全相关技术的国内外研究状况进行简要描述。

1.2.1 IPv6 技术

在过去的 10 到 15 年间，连接到 Internet 的网络设备数量每隔不到一年的时间就会增加一倍。同时，Internet 规模的迅速增长也伴随着要求唯一 IP 地址接入 Internet 的嵌入式系统设备数量的激增，作为现有 Internet 支持协议的 TCP/IP，其中的 IPv4 只能提供 32 位的 IP 地址，IP 地址的数量不能满足实际应用需要，而且 IPv4 的安全性能还有待提高。因此，IP 协议就需要进行相应的升级，IPv6（IP 协议的下一版本）就是用来解决 IPv4 的地址紧缺以及安全性能缺乏的问题。1998 年夏末，新的 IPv6 RFC 获得了发表的批准，其中尤其值得注意的是：RFC 2373（IPv6 的寻址体系结构）替换了 RFC 1883；RFC 2374（一种 IPv6 可集聚全球单播地址格式）替换了 RFC 2073。其他新的 RFC 描述了 ICMPv6、IPv6 中的邻居发现和无状态自动配置[2]。

对 IP 地址日益增长的需要是 IPv6 发展的主要催化剂[1]。据估计，目前仅在无线领域，需要接入 Internet 的移动电话、PDA 和其它的无线设备就超过 10 亿个，每个设备都需要有自己唯一的 IP 地址。数十亿个家庭的 Internet 应用——从电视到冰箱再到电表，这些设备也将实现与的 Internet 连接，因此都需要自己的 IP 地址。

IPv6 带来的并不仅仅是地址的扩展，它的自动配置和自动发现特性将淘汰那些需要大量人力的昂贵方法，这种方法是管理动态主机配置协议(DHCP)服务器的，而多数大型机构都利用这种服务器来管理它们的 IPv4 地址。IPv6 将使用一种无国籍的自动配置方法，这种配置方法与一个接口 ID 相结合，例如机器现有的 MAC 地址以及来自本地路由器的网络前缀等，以便为其分配 IP 地址而不是依赖 DHCP 服务器来分配。

IPv6 也将提供一个对等网应用的基础设施，这个基础设施将允许使用端到端的全球寻址机制。这种设置将消除在一些大型企业网络边缘对网络地址转换(NAT)设备的需求，因为这些设备会减慢加密过程，而且会降低 VoIP、移动 IP 等的应用效率。

新版 IPv6 既保持了 IPv4 许多成功的特点，又对协议的细节做了许多修改，其修改涉及到更大的地址空间、灵活的报头格式、增强的选项、支持资源分配、支持协议扩展等五部分。IPv6 解决了地址空间耗尽和路由表爆炸等问题，而且为 IP 协议注入了新的内容，使得支持安全、主机移动以及多媒体成为 IP 协议的有机组成部分。协议的设计使路由器处理报文更加简便，协议的扩展性也更好。

IPv6 的实施方案在 IETF 的下一代转换工作组（NGtrans WG）中，Cisco 在 IPv6 基础设施的定义和配置中扮演了领导者的角色。Cisco 的 6Bone 路由器是 6Bone 基础设施的一个关键部分，这个设施是一个建立于现有 Internet 上的 IPv6 的实验网络，它为 IPv6 的开展和实施提供了一个真实的实验平台。另外，Cisco 还发布了支持新的 IPv6 特性的 Cisco IOS 12.2(2)T，这个版本允许在适应期期间 IPv4 和 IPv6 两种协议共存。目前，IPv6 的实验网 6Bone 已经遍布全球，因此，IP 协议从 IPv4 过渡到 IPv6 已经是历史必然[39]。人们花费数年甚至数十年时间构建起来的 IPv4 网络不可能一夜之间全部废弃而更新为 IPv6 的网络，因此，在过渡时期中显然需要 IPv4 和 IPv6 之间的互联互通[40]。转换的主要要求有三个：不中断 IPv4 服务、任何时候任何地点可用 IPv6 服务以及最低限度的运作成本。在 IPv6 的网络流行于全球之前，总有一些网络首先具有 IPv6 协议栈，利用隧道技术来连接这两种网络。IPv6 隧道的定义：当一个分组被封装并作为载荷在另一个 IPv6 分组中携带时，这个 IPv6 分组就称为 IPv6 分组（IPv6 Tunnel Packet）。在隧道分组的信源和信宿之间的转发路径就被称为一条 IPv6 隧道（IPv6 Tunnel），这种技术就被称为 IPv6 隧道技术。隧道技术一方面保护现有 IPv4 网络的投资，同时又使 IPv4 网络向 IPv6 的顺利迁移成为可能。相信在不久的将来，更多的基于隧道技术的 IPv6 网络会被建立并融入 Internet 世界中[40]。

由此可见，越来越多的嵌入式系统产品应用到人们的日常生活中，而这些嵌入式系统产品也需要和外界进行信息交互、资源共享以及远程控制等等，这些产品也就需要和 Internet 相连接以进行全球性的信息化。随着 IPv6 技术的广泛应用，如果再对于这些产品赋予 IPv4 的性能，那么必将会束缚其未来的应用，因此在考虑性价比的同时，优先考虑使用 IPv6 的支持。这些产品从数量上来讲，是现在 PC 数量的几何级数倍，同时，这么多的嵌入式系统产品在 Internet 上的应用也面临着更多网络安全方面的问题。IPv6 的出现，恰好从各方面都满足了这种需求。

1.2.2 “后 PC”时代[1]

IBM 总裁 Lou Gerstner 曾经说过：“PC 时代已经终结”。其表达的意思并

不是想说在未来的网络时代不会再使用 PC，而是在努力将大家的注意力从独立的、带有强烈技术韵味的 PC 产品转移到网络时代——电子商务时代。IBM 说[10]：“在 PC 走到了它一生中最关键的十字路口的时，IBM 推出全新的 PC 产品是一种必然，只有这样才能让大家真切的感受到‘后 PC’时代的 PC 产品是什么样的，IBM 的 NetVista A40 就是这样的新型计算机。拥有为最终用户进行优化、适用于电子商务、协调的电子生活方式（E-Life Style）、容易使用四大特色。”

英特尔公司副总裁欧德宁说[7]：“我觉得 PC 终结的说法完全错了。当今最令人激动的技术和产品，如 MP3 播放机、数码相机、数码摄像机，它们可以独立工作，但是在和 PC 相连时，它们的效果才是最好的。”

联想集团总裁杨元庆说[41]：“作为一种产品，PC 在欧美一些发达的国家，也许开始趋于饱和，但是在中国至少还有 5 年的高速发展的时期。如果从技术角度看，PC 会有更加广泛的应用，也许在未来我们家庭很多的电器里边都会有 PC 的影子。我们今天家里边的水表、电表要有人上门来抄数字，然后你得到这些水厂、电厂交费。未来可能这些都是数字化，通过家庭的电脑或者是网关，可以和互联网联在一起，自动抄数据，自动交费，使用网上银行。”

综上所述，无论哪种观点，有一点是相同的：一个以传统 PC 作为电子产品标志的时代已经过去，而一个以新型 PC 或者是“后 PC”作为电子产品标志的时代已经到来。无论是 IBM 的 NetVista A40，还是具有 PC 影子的信息家电，它们都不再是传统意义上的 PC 了。然而，从另外一个角度来看，所有的这些产品不正是嵌入式系统产品吗？

早在 20 世纪七八十年代，嵌入式系统已经开始应用于工业控制等领域，通过过去几十年的发展，嵌入式系统技术已经逐渐成熟。目前，嵌入式系统已经渗透到生活的各个角落——电器、玩具、汽车、电视、录像机、医疗器械、飞机等等，逐渐增多的嵌入式系统应用也成了推动 PC 技术时代发展的动力，现在，嵌入式系统带来的工业年产值已超过了 1 万亿美元[41]。美国著名未来学家尼葛洛庞帝 1999 年 1 月访华时预言：4~5 年后，嵌入式智能产品将是继 PC 和 Internet 之后的最伟大的发明。我国著名嵌入式系统专家沈绪榜院士 1998 年 11 月在武汉全国第 11 次微机学术交流会上发表的《计算机的发展与技术》一文中，对未来 10 年以嵌入式芯片为基础的计算机工业进行了科学的阐述和展望[41]。一个传统 PC 的时代已经逐渐开始宣告没落，新型 PC 的时代正在到来，因此可以用“‘后 PC’时代”来描述。

1.2.3 嵌入式 Internet 技术[38]

嵌入式系统是基于嵌入式设备或微处理器的控制系统，它将操作系统和功能软件集成于嵌入式硬件系统之中，专门用于执行特定的任务或任务组。嵌入式系统可以分为三类：军用嵌入式系统、商业嵌入式系统和家用嵌入式系统。军用嵌入式系统主要用于军事通信系统和控制系统；商业嵌入式系统用于楼宇自动化、过程控制及工厂自动化等；家用嵌入式系统主要用于智能家用电器及

其相关领域。

随着 Internet 技术的飞速发展，人们将不仅仅满足于信息资源的共享，也期望各种电子产品（如：激光打印机、传真机、电脑、VCD/DVD、电视机、安全监视器等）通过互联网络连接到一起，实现“硬资源”的共享。另一方面，基于现场总线的嵌入式系统近年来成为实施控制工业的主流，控制工业的趋势将是网络化，它必将连入其他网络而与各种网络实现网际互联。因此，将嵌入式系统与 Internet 结合起来已经成为目前嵌入式领域的一个研究热点[42]。

嵌入式系统的应用前景非常广泛，由于嵌入式系统减少了许多实际应用中的多元化问题，因此，越来越多的嵌入式系统需要通过互联网络进行相互互联。许多高容量的嵌入式系统应用是通过 UART、CAN、SPI 和 Ethernet 进行物理的连接，然而，支持这些不同的硬件标准是相同软件标准，许多软件标准的主动性、一致性是受到设计者的思想所支配的。正因为这样，即使两个基于 UART 的系统，尽管是基于同一个硬件标准，如果软件标准不统一，还是不能够彼此互联[16]。

嵌入式系统已经成熟起来，并得到了最为广泛的工业应用。它以 PC 不可比拟的结构灵活、稳定性和经济性成为计算机工业的高速增长点。8/16/32/64 位的嵌入式微处理器可以工作在许多空间狭窄、环境恶劣的环境或实体中，将成为人们感受自然和社会的扩展神经末梢和许多工业领域中专用的智能中央单元，或者是智能化的“专用螺钉”。下个世纪，嵌入式技术必将得到更广泛的应用[43]。当前，人们习惯了浏览 HTML 文档和收发电子邮件，只要嵌入式设备能在 Internet 上被远程控制和观测的时候，这就可以实现；用 PDA 在 Internet 上关掉在匆忙出门时忘记关掉的电饭锅，或者在办公室里通过 Internet 查找自己忘在家里笔记本上的电话，再者坐在家中观察远在千里之外的自己的蔬菜工厂中的蔬菜的长势.....有了嵌入式 Internet 技术，梦想从此不再遥远[1]。

目前嵌入式 Internet 的研究主要集中在：嵌入式系统的 Internet 接入技术、嵌入式 Web 服务器、支持网络的嵌入式操作系统以及嵌入式 Internet 开发工具四个方面。

在嵌入式系统的 Internet 接入应用中，利用 emWare 公司的 EMIT 技术，能将 Internet 延伸到 8、16 以及 32 位的嵌入式系统 [43]。嵌入式 Web 服务器[33]同传统 Web 服务器一样，是嵌入式网络信息服务的核心元素。嵌入式 Web 服务器的出现，有国外 NetMedia 的 SitePlayer[47]和国内东大新业的 Webit[1]。

emWare 的 EMIT 技术是在电子设备中添加远程管理能力，这样帮助公司增加其利润，提高其生产率和用户的满意度。并不是只有 emWare 公司认识到了将设备信息化的价值，那些同样看到这一点的公司和 emWare 一起建立了 ETI（Extend The Internet）联盟。ETI 联盟包括广泛的工业界佼佼者，每个都能为硬件设备网络化提供一个关键的部分。绝大多数的微处理器生产厂商都是属于 ETI 联盟的，在数据库、网络管理、ERP 和直接应用领域的领头羊也加入了 ETI 联盟，主要的开发工具提供商、网络服务提供商和产品的设计者也

同样是 ETI 的成员。ETI 联盟以每年增加一倍以上的速度发展，并且毫无疑问的会议这种速度快速发展。越来越多的软件公司、硬件设备制造商和服务提供商都看到了硬件设备网络化解决方案的潜力。这些公司都将提供补充的产品专门技能、服务技能、公司灵活性和技术技能用以连结数十亿种硬件设备[8]。

1.2.4 分布式防火墙

随着 Internet 的发展，其现存安全机制的脆弱性也暴露无疑，近几年对 Internet 的攻击屡见不鲜，这些攻击大都可以分为两类：一类是利用系统管理、配置不当、用户本身的误操作以及使用编写粗糙的软件等造成的漏洞对系统进行攻击。另一类更复杂的攻击是利用目前采用的 Internet 协议中的内在漏洞进行攻击[35]。这主要是因为目前 Internet 所使用的 TCP/IP 协议栈中没有任何协议提供通信双方的身份认证、对数据报内容的认证和加解密。显然，对于第一类攻击，防火墙无疑是非常好的工具[44]。

“防火墙”[3]这个词来自于建筑物中的同名机构，从字面意思上理解，它可以防止火灾从建筑物的一部分蔓延到其它部分。而对于 Internet 的防火墙来讲，指的是一种被动式防御的访问控制技术，通过在内部网和外部网的边界上建立相应的网络通信监控系统实现防御功能[45]。防火墙位于内、外部网络之间的某一个适当的扼制点（Choke Point）[46]上，目的是限制外部非法用户访问内部网络资源和内部非法向外传递信息。

防火墙技术主要有两种：包过滤技术和代理技术。后来又在此基础上出现了融合入侵检测功能的混合型防火墙、状态监测防火墙。但这些都不能满足越发开放、灵活的 Internet 安全需求，因此需要一种新型体系结构防火墙，为当今 Internet 网络安全提供更多的保障。分布式防火墙（Distributed Firewall）[13]便是这种新型的防火墙，从狭义来讲，分布式防火墙产品是指那些驻留在网络中主机如服务器或终端主机，并对主机系统自身提供安全防护的软件产品，用以保护网络中关键服务器及工作站免受非法入侵的破坏；从广义来讲，“分布式防火墙”是一种新的防火墙体系结构，它包含如下结构部分：网络防火墙、主机防火墙、中心管理[14]。

防火墙技术给 Internet 网络安全提供了保护，增强了抵御非法攻击的能力，而分布式防火墙则发挥了防火墙的优势，利用新的结构模型和技术手段，将 Internet 网络安全防御能力提高到一个新的台阶。

1.3 本文研究的主要内容

随着计算机技术和通信技术的发展，嵌入式系统已渗透到人们生活的各个角落，如：手持设备、信息家电、智能玩具、汽车、医疗器械、飞机等。如何有效的利用这些嵌入式系统，使其实现资源共享、信息通信和状态控制等功能，本文研究了嵌入式系统 Internet 应用的相关问题，并对嵌入式 Internet 的安全问题进行了研究，主要工作总结如下：

1、通过分析嵌入式技术和 Internet 技术的发展，研究嵌入式系统 Internet 应用的相关技术，提出嵌入式 Internet 和扩展 Internet 模型，并研究课题实现

所采用的嵌入式 Internet 环境及相关技术。

2、在传统 Internet 存在的网络安全威胁分析的基础上，研究嵌入式 Internet 和扩展 Internet 的网络安全性问题，并探讨嵌入式 Internet 和扩展 Internet 的特点及其存在的网络安全威胁。

3、探讨传统防火墙技术在应用中存在的缺点，研究分布式防火墙技术为嵌入式 Internet 提供安全保护的可行性，并对分布式防火墙进行改进，提出新的嵌入式 Internet 防火墙技术模型。

4、分析嵌入式 Internet 防火墙的体系结构，建立防火墙各个组件相应的结构模型，并进行深入地研究。

5、针对课题所使用的嵌入式 Internet 环境，探讨基于 PC 硬件的模拟嵌入式 Linux 系统的实现。详细分析嵌入式 Internet 防火墙各个组件的实现技术，并着重研究嵌入式 Internet 防火墙的终端设备防火墙组件的实现技术。

第二章　嵌入式 Internet 及扩展 Internet 模型

嵌入式系统的 Internet 应用给 Internet 带来了巨大的变化，推动了 Internet 技术的发展，扩展了 Internet 的应用领域，形成了嵌入式 Internet 模型(Embedded Internet Model)和扩展 Internet 模型(Extended Internet Model)。

2.1 嵌入式系统概述

在智能化设备与仪器应用场合，出于对产品体积、成本等因素的考虑，往往要求将计算机控制部分安装在设备内部，并且占用的空间尽可能小。在这种情况下，处理器一般没有多少可用内存，更没有可用的外存，而操作系统就装在有限的内存中（一般在 ROM 中），这样的系统称之为嵌入式系统。嵌入式系统是智能化设备与仪器的灵魂，而以嵌入式系统的形式隐藏在各种装置、产品和系统中的计算机称为嵌入式计算机。

与通用计算机相比，嵌入式计算机在应用数量上远远超过了通用计算机，一台通用计算机的外部设备中就包含了 5~10 个嵌入式微处理器，如：键盘、鼠标、软驱、硬盘。制造工业、过程控制、通讯、仪器仪表、汽车、船舶、航空航天、军事设备、信息家电等产品也是嵌入式系统的应用。

2.1.1 嵌入式系统的定义

对于嵌入式系统，业界内已经有了比较标准的定义。嵌入式系统是指以应用为中心，以计算机技术为基础，软件硬件可裁剪，适应应用系统对功能、可靠性、成本、体积、功耗严格要求的专用计算机系统。嵌入式系统是面向用户、面向产品、面向应用的。

2.1.2 嵌入式系统的组成

嵌入式系统通常包括底层的嵌入式硬件设备、嵌入式操作系统以及在操作系统上运行的嵌入式应用程序。用图 2.1 表示嵌入式系统的组成结构。

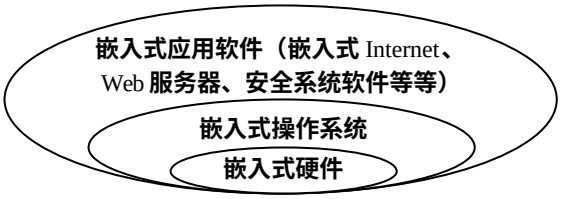


图 2.1　嵌入式系统结构图

根据图 2.1 所示，下面就嵌入式系统的各个组成分部分进行相关介绍。

1、嵌入式微处理器（MPU，Micro-Processor Unit）

嵌入式微处理器的基础是通用计算机中的 CPU。在嵌入式应用中，将 CPU 装配在专门设计的电路板上，只保留和嵌入式应用有关的母板功能，这样就可以大幅度减少系统体积和功耗。为了满足嵌入式应用的特殊要求，嵌入式 CPU 虽然在功能上和标准 CPU 基本上是一样的，但在工作温度、抗电磁干扰、可靠性等方面一般都作了各种增强。与工业控制计算机相比，嵌入式 CPU 还具

有体积小、重量轻、成本低、可靠性高的优点。目前，嵌入式 CPU 主要有 AMD186、PowerPC、ARM 等系列。

2、嵌入式操作系统

现有的嵌入式操作系统都是实时操作系统（RTOS，Real Time Operation System）。由于嵌入式系统硬件设备资源较少，处理器一般没有多少可用内存，更很少有可用的外存，而操作系统就装在这有限的内存中，所以嵌入式操作系统的体积就比较小。

目前嵌入式系统中，嵌入式微处理器的各项性能指标越来越高，一个微处理器往往要同时完成多项任务，所以为了提高系统的可靠性和开发效率，一般采用多任务的实时操作系统作为应用的系统平台。

国际上比较流行的嵌入式操作系统有很多，如：VxWork、Psos、QNX、Windows CE、JavaOS 等。据有关方面统计，世界上有 200 多家公司都在致力于嵌入式操作系统的研究开发，而国内也出现了以 Hopen 为代表的 RTOS。

对于这些嵌入式操作系统，在所采用的技术、体系结构和性能方面各有千秋，对应用环境和设备的支持各有侧重点。目前，对嵌入式实时操作系统（Embedded Real-Time Operating System）有以下几方面的要求：

·实时性。嵌入式系统一般带有实时性要求。

·系统可裁剪。由于嵌入式系统资源的可搭配性，因此，嵌入式操作系统应该有极强功能的结构性，操作系统功能要能够进行裁减配置，够用即可。

·网络支持。随着“后 PC”时代的到来，更多的嵌入式设备需要连接上网，成为嵌入式 Internet 的一部分，因此，需要提供必要的网络支持。

·功能可扩展。由于新型嵌入式设备的功能多样化，要求嵌入式操作系统除提供基本内核支持外，还需提供越来越多的扩展功能模块（含用户扩展），如：功耗控制、动态加载、嵌入式文件系统、嵌入式 GUI 系统和嵌入式数据库系统等。

3、嵌入式应用软件

RTOS 的引入，解决了嵌入式软件开发标准化的难题，随着嵌入式系统中软件比重不断上升、应用程序越来越大，开发人员的管理、应用程序接口标准化、程序档案的组织管理都成了嵌入式技术研究的大课题。引入 RTOS 相当于引入了一种新的管理模式，对于开发单位和开发人员都是一个提高。基于 RTOS 开发的程序，具有较高的可移植性。

2.2 嵌入式 Internet（Embedded Internet）

如果说在这个地球上人和人之间的通信方式变革能够带动一场 Internet 的革命的话，那么设备和设备之间、设备和人之间通信方式的变革，就必将导致另外一场以 Internet 为要素的嵌入式 Internet 革命。到那时，地球不再是 Internet 要素的地球村，而是一个具有电子化皮肤的“地球生物”。设想一下，在任何一个嵌入式设备中都具有一个嵌入式 Web 服务器，并将该 Web 服务器的客户端从“人”拓展到设备，那么我们就将“行为”作为浏览的对象搬到了嵌入式

Internet 上。

2.2.1 嵌入式 Internet 技术基础

从技术的角度讲，目前，嵌入式 Internet 技术领域涉及到以下四个方面：嵌入式系统的 Internet 接入技术、嵌入式 Web 服务器、嵌入式网络协议栈和嵌入式 Internet 开发工具。

2.2.1.1 嵌入式系统的 Internet 接入技术

嵌入式系统接入 Internet 主要的困难在于实现与 Internet 通信的协议对微处理器的要求比较高，这因为大量的嵌入式系统多采用 8 位和 16 位的低速微处理器。因此，目前对于嵌入式系统的 Internet 接入技术研究是嵌入式 Internet 领域研究的热点之一。通常具有如下两种技术模型：

1、间接接入技术模型

对于 8/16 位的嵌入式系统来讲，速度慢和内存小等系统性能缺陷是其实现 Internet 接入需要着重考虑的问题。如果在 8/16 位的嵌入式系统硬件设备中添加网络接口硬件，并在相应操作系统中实现 TCP/IP 协议栈，其系统性能仍达不到应用需要。因此，只能采用间接的接入方式，这种接入技术模型结构如图 2.2 所示。

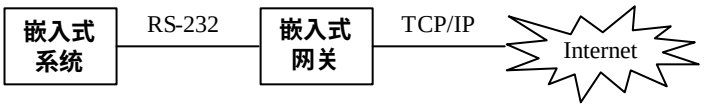


图 2.2　间接接入技术模型结构图

首先，嵌入式系统和嵌入式网关连接通信，连接方式采用传统的 RS-232 或 RS-485 等；再由嵌入式网关负责实现 TCP/IP 协议栈，并与 Internet 连接，完成嵌入式系统和 Internet 的信息交互。这种技术模型解决了以 8/16 位微处理器为核心的嵌入式系统的 Internet 接入。这种接入技术需要一个专门的嵌入式网关，而且网关和各个嵌入式系统之间的通信也会受到距离和速度的限制。对于过度分散的嵌入式系统而言，采用这种接入技术模型，成本将会增加。

基于这种接入技术模型，emWare 公司提出了 EMIT 技术，通过给 8/16 位嵌入式系统增加相应的软、硬件，充分解决了 8/16 位的嵌入式系统接入 Internet 的问题，得到了众多软硬件厂商的支持，是当前很具有前景的 8/16 位嵌入式系统的 Internet 接入技术。

2、直接接入技术模型

实现嵌入式系统直接与 Internet 相互连接主要是通过在嵌入式系统本身添加网络接口硬件，增加必要的软件支持，并采用相应的接入技术而实现的。直接接入技术主要是针对采用 32/64 位微处理器的嵌入式系统，这种接入技术模型结构如图 2.3 所示。

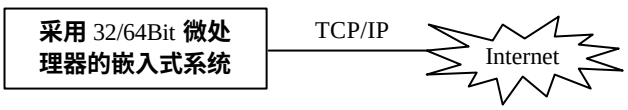


图 2.3　直接接入技术模型结构图

随着硬件技术的发展和制造成本的降低，32/64 位嵌入式系统微处理器的应用越来越广泛，将会逐步取代 8/16 位微处理器成为应用的主流。从 32/64 位微处理器本身性能来讲，能够集成网络接口部件，并且具有足够的速度和资源来实现网络协议栈。另外，现有的 32/64 位嵌入式操作系统也都实现了 TCP/IP 协议栈，并提供相应的网络软件。因此，32/64 位嵌入式系统已经具备了直接接入 Internet 的条件，这种接入技术模型可以使嵌入式系统直接与 Internet 相连，具有很大的灵活性。32/64 位嵌入式系统真正体现了“后 PC”时代的概念，是嵌入式系统的发展方向。

2.2.1.2 嵌入式 Web 服务器（EWS，Embedded Web Server）

嵌入式 Web 服务器同传统 Web 服务器一样，是网络信息服务的核心元素。传统的 Web 服务器是用户的中心，主要负责响应用户的服务请求。嵌入式 Web 服务器也就是“瘦服务器”，它在数目上远远超出了用户的数量，这样就能形成一种新型的服务模式：某个范围中数量较多的嵌入式 Web 服务器提供给数量相对较少的用户组进行信息交互。

一般来讲，Web 服务器指的是高性能的计算机，根据客户的需求，提供相应静态 Web 页的服务，同时提供给大量客户数据存储服务。嵌入式 Web 服务器的主要任务是使远程用户能够对嵌入式系统进行数据存取和状态控制，因此嵌入式 Web 页的内容需要动态生成。

HTTP 协议处于网络协议栈的上层，Web 服务器的核心就是网络协议栈。它主要负责应用程序数据在以太网或串行连接等网络通信中的数据传送。因此，嵌入式 Web 服务器的通用结构如下图所示：

图 2.4　嵌入式 Web 服务器结构图

由于人们对不同的嵌入式设备有不同的需求，而且嵌入式 Web 服务器的网络协议栈还有别于 PC 系统的网络协议栈，因此嵌入式系统对协议的支持是可选择的。目前，对于嵌入式 Web 服务器的研究已经进入实用化阶段，国外有 NetMedia 公司研制的世界上最小的、嵌入到网卡的嵌入式 Web Server，国内也有东大新业研制的 Webit。

2.2.1.3 嵌入式网络协议栈

目前，嵌入式系统所使用的网络协议栈的类型可分为两种：嵌入式 TCP/IP 协议栈和应用于无线移动通信的嵌入式 WAP 协议栈。作为应用于嵌入式系统的嵌入式网络协议栈，主要是在原有的 TCP/IP 协议栈和 WAP 协议栈基础上修改后应用于嵌入式系统的。在课题的研究过程中，对于无线移动通信的嵌入式 WAP 协议栈没有做太多考虑，因此，在这里就针对嵌入式 TCP/IP 协议栈

做相关的探讨。

TCP/IP（Transmission Control Protocol/Internet Protocol）是 70 年代中期美国国防部（DOD）为其 ARPNET 广域网开发的网络体系结构和协议标准，它代表了一个协议簇。以它为基础组建的 Internet 是目前国际上规模最大的计算机网络。

传统的 TCP/IP 协议在实时性方面做得不够好，而是把大量的经历花在保证数据传送的可靠性以及数据流量的控制上。对于一些实时要求比较高的场合，例如嵌入式系统领域，需要把传统 TCP/IP 协议在不违背协议标准的前提下加以改进，使其实时性得到提高，从而满足嵌入式系统应用的要求，这就是嵌入式 TCP/IP 协议栈。

当今国际上一些著名的嵌入式操作系统供应商，都在自己的嵌入式操作系统产品中应用了嵌入式 TCP/IP 网络组件，以便自己的实时系统具备网络通讯功能。由于是面向嵌入式应用的，因此这些产品在实时性、可移植性方面都做了各具特色的改进。

1、VRTXsa 中的 SNX

VRTXsa 是 Mentor Graphics Corp.公司开发的实时操作系统，SNX 是其中的 TCP/IP 网络组件。它是一个基于流的 TCP/IP，兼容 AT&T UNIX System V 的 Release 3 和 Release 4。向上提供两种编程接口，TLI（Transport Library Interface）接口库和 4.3BSD socket 接口库。SNX 不仅体现了一般网络的特性，而且也体现了其作为实时操作系统网络组件的特性。SNX 利用了 VRTXsa 超微内核的特性，在响应速度、吞吐量上体现了其实时性。

2、Nucleus 中的 NET

Nucleus 是 Accelerated Technology Inc.公司开发的实时操作系统，NET 是其中的 TCP/IP 网络组件。它最大的特点是引入了一种 ODH（Optimized Data Handling）机制，去掉了传统 TCP/IP 协议栈中各层之间不必要的数据拷贝，使得代码的效率更高。

3、pSOS 中的 pNA+

pSOS 是 Integrated Systems Inc.公司开发的实时操作系统，pNA+是其中的 TCP/IP 网络组件。它支持用于管理参与多目传送（multicast delicery）的计算机群的 IGMP（Internet Group Management Protocol），支持基于串口的 PPP，还支持 MIB-II。

综上所述，这些典型嵌入式网络协议栈产品具有以下共同点：

- 强移植性；
- 好的实时性；
- 可裁剪性；
- 具有各自实时操作系统平台。

2.2.1.4 嵌入式 Internet 开发工具

随着嵌入式系统接入 Internet 应用的增多，针对嵌入式 Internet 的开发工

具也越来越多，相应的开发技术也越来越成熟。目前，支持网络的嵌入式操作系统有 Microsoft 的 Windows CE、Wind River System 的 VxWorks 和科银京城公司的 DeltaOS 等等，相应也有很多成熟的嵌入式 Internet 开发工具产品，例如 Microsoft 公司的 Microsoft eMbedded Visual C++、Visual Basic，Wind River System 的 Tornado 以及科银京城公司的 LambdaTOOL 等。

另外，由于 Linux 操作系统本身的特点及其开放源码的特性，人们越来越多的关注嵌入式 Linux 的应用，同时嵌入式 Linux 网络开发也就成了嵌入式 Internet 开发的热点。

2.2.2 嵌入式 Internet 模型（Embedded Internet Model）

嵌入式系统的 Internet 接入存在两种不同的方法，通过这两种方法所构建的嵌入式 Internet 各不相同。因此，嵌入式 Internet 在局部来讲也相应的存在两种结构模型。

对于采用间接接入方法的嵌入式系统，为节约成本，由多个嵌入式系统与嵌入式网关相连，通过嵌入式网关组成嵌入式网络与 Internet 相连接。这种多个嵌入式系统和嵌入式网关所组成的嵌入式网络的结构如图 2.5 所示。

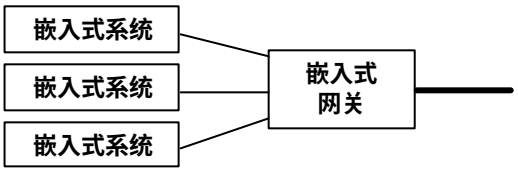


图 2.5　间接嵌入式网络结构图

对于采用直接接入方式的嵌入式系统，将其作为 Internet 的终端设备，采用现有网络拓扑结构直接接入 Internet，形成开放结构的嵌入式 Internet 的组成部分。

嵌入式 Internet 起源于把嵌入式系统与 Internet 结合起来的这种想法。嵌入式 Internet 指的是：在嵌入式系统应用领域中，以 Internet 技术为基础，使嵌入式系统与 Internet 相互连接，实现嵌入式系统与 Internet 之间的资源共享、信息通信和状态控制等功能，这种嵌入式系统与 Internet 之间的连接与应用就称为嵌入式 Internet（Embedded Internet）。

随着越来越多的嵌入式系统需要接入 Internet 以进行全球性的信息通信，使得嵌入式系统的 Internet 应用面临两方面的问题：网络协议的特殊化和用户与嵌入式系统网络之间的信息交互比较困难。嵌入式 Internet 在嵌入式系统应用领域的革命性举措是它有效地解决了嵌入式系统不统一的网络协议标准和人机接口的矛盾。嵌入式 Internet 硬件连接采用最常见的以太网；用全球性的 TCP/IP 协议来取代那些不统一的传输协议，也可以用蓝牙（Blue Tooth）或其他兼容 TCP/IP 协议的传输协议来替代。因此，具备了嵌入式系统与 Internet 连接的基础。另外，嵌入式 Internet 采用浏览器作为嵌入式系统网络与用户进行交互的接口，作为全球通用的浏览软件，浏览器实现了人机界面的统一。

通过嵌入式 Internet，嵌入式系统与 Internet 上的终端设备相互连接在一

起，可以提供给用户一个更为广泛的资源共享空间；并且通过嵌入式 Internet，用户可以获取嵌入式系统的状态信息并对其进行控制，实现用户与嵌入式设备的远程交互。

在嵌入式 Internet 解决了嵌入式系统 Internet 接入的两大技术问题过后，越来越多的嵌入式系统接入了 Internet，例如：信息家电、车载嵌入式设备、SOC（System on Chip，片上系统）、DSP（Digital Signal Processing，数字信号处理设备）等，扩展了嵌入式系统的应用范围。因此，可以用图 2.6 来描述嵌入式 Internet 模型（Embedded Internet Model）：

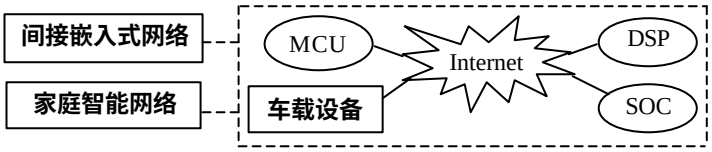


图 2.6　嵌入式 Internet 模型图

嵌入式系统通过各种方法接入 Internet，通过 Internet 实现资源共享、信息通信和状态控制功能。同时，嵌入式系统的 Internet 应用扩展了原有 Internet 的应用范畴。

嵌入式 Internet 的应用遍布到世界的各个角落，也扩展了 Internet 的应用范围，使 Internet 成为地球的“电子皮肤”。可以用图 2.7 来表示嵌入式 Internet 与 Internet 的关系：

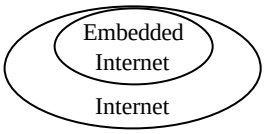


图 2.7　嵌入式 Internet 与 Internet 的关系图

2.3 扩展 Internet（Extended Internet）

2.3.1 Internet 应用模型概述

计算机网络是指多个计算机经由通信线路互联而组成的网络体系结构。Internet 实际上是由很多的局域网(LAN)、城域网(MAN)以及广域网(WAN)组成的，它是全球性网际网络。

计算机网络是由很多计算机或其他一些终端互联在一起组成的，因此，网络硬件分为主机（网络中的计算机以及其它类型的终端设备）和网络互联的辅助设备。具体一点来讲，计算机网络硬件设备基本上包括如下的系统设备：计算机、其他相关接入的终端设备、调制解调器(Modem)、通信控制器、多路复用器、分组组装/拆卸设备 PAD 和终端通信处理机等。

Internet 则是由各种计算机网络互相连接在一起而组成的，实际上组成 Internet 的最小因子是每个小的“通信子网”。每个“通信子网”的拓扑结构有星型、环型、树型、全连接型、交叉型和不规则型，很多具有以上这些拓扑结构的“通信子网”连接在一起，便组成了局域网、城域网、广域网乃至 Internet。

综上所述，Internet 是由很多局域网、城域网和广域网连接在一起而组成的，其组成的结构如图 2.8 所示：

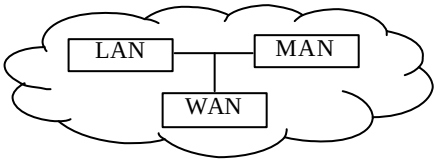


图 2.8　Internet 网络拓扑结构图

如图 2.8 所示，Internet 的结构模型是一个复杂的模型，其中包含有各种小的结构模型，因此，Internet 的应用会面对复杂的技术问题。

2.3.2 扩展 Internet 模型（Extended Internet Model）

当前，Internet 的 IP 协议还处于 IPv4 协议到 IPv6 协议的过渡时期，随着 IPv6 协议的逐渐应用，IP 地址资源的缺乏将会得到彻底的改变。由于 IPv6 中的地址域长度为 128 位，能为世界上任何潜在的网络用户提供地址，也可提供给每个嵌入式系统 IP 地址用以接入 Internet。

以往在谈论网络时所说的网络主机或终端，大部分指的都是计算机（Computer），所以组成的网络也称为：计算机网络（Computer Network），Internet 便是全球最大的计算机网络。现在，接入网络的设备不再全是计算机了，作为网络的主机或终端也不全是计算机，有可能是 SOC、DSP 等等，这些设备严格上来讲不是计算机系统，而是嵌入式系统。因此，嵌入式系统接入 Internet 扩展了 Internet 应用的局限性，从而形成了新的应用模型。我们把这种新的模型称为：扩展 Internet 模型（Extended Internet Model）。这种扩展 Internet 模型和原有 Internet 模型的主要区别在于：扩展 Internet 模型的网络主机或终端包含了现在和未来可以接入 Internet 的计算机和非计算机设备，同时，还包含诸如嵌入式网络等的特殊系统网络，其结构如图 2.9 所示。

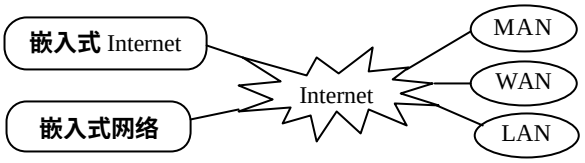


图 2.9　扩展 Internet 模型图

2.4 课题研究的嵌入式 Internet 环境

对于课题研究的嵌入式 Internet 环境，考虑到研究的前瞻性与连续性，必须对现有的网络环境做一些改进。另外，由于现有的一些技术应用还达不到嵌入式 Internet 应用的要求，只能做相应的变通。下面就课题研究所使用的嵌入式 Internet 环境做相关的论述。

2.4.1 基于 IPv6 的嵌入式 Internet

嵌入式系统设备接入 Internet 时，与 Internet 的连接需要实现相应的嵌入式网络协议栈，目前应用的较多的是：嵌入式 TCP/IP 协议栈和嵌入式 WAP

协议栈。普通的计算机系统，大部分都采用 TCP/IP 协议栈，TCP/IP 协议栈的应用有 TCP/IPv4 和 TCP/IPv6 两种版本，所以，课题研究的嵌入式 Internet 环境的网络协议栈需要面对是采用 IPv4 还是 IPv6 的问题。

目前来讲，绝大部分的嵌入式系统设备 Internet 应用所使用的 TCP/IP 协议栈都采用 TCP/IPv4 版本。但是 TCP/IPv4 的应用是不是适合现有的嵌入式 Internet 呢？可以想象：一个智能的家庭网络，其解决方案是通过利用类似 EMIT 技术形成智能家庭网关来实现，这个智能家庭网关具有网关的功能和路由的功能，并且是智能家庭网络的遏制点。在这种解决方案中，每个嵌入式系统设备采用的是 IPv4，拥有 32 位的 IP 地址。在这种结构模型中，家庭网络中的内部嵌入式系统设备只能通过网关与外界通信，而其本身不会直接暴露在 Internet 上，但要求家庭网络内部的设备都必须在一个局部范围内，不能够超出这个范围，从而导致了家庭智能网络的局限性。如果，将汽车车载系统这样的移动性很强的嵌入式系统与家庭智能网络互连，上面这种方法就不能够实现。

随着越来越多移动设备需要通过 Internet 与远程的其他嵌入式设备互联而形成嵌入式 Internet。在现有的 IPv4 的基础上来构建嵌入式 Internet 必将会带入 IPv4 所拥有的地址短缺、低安全性等缺点，同时，也会完全限制嵌入式 Internet 的本身的特点。如果采用 IPv4 来构建诸如智能家庭网络等相关的嵌入式 Internet，很明显，在建立好成熟的嵌入式 Internet 网络环境之后，随着技术的进步和需求的增长，嵌入式 Internet 的灵活性、开放性和远程性就需要将嵌入式 Internet 中的嵌入式设备直接接入 Internet，并且需要嵌入式 Internet 中的终端设备具有能够直接被网络其他设备访问的能力，IPv4 就会成为最大的技术瓶颈。因此，每个嵌入式 Internet 中的终端设备都应该被分配全球性唯一的 IP 地址资源，并且还可以通过 VPN 技术建立虚拟专有的嵌入式网络。这样，就可以提供给嵌入式 Internet 足够的发展空间，从而避免由于技术革新所带来的损失。

2.4.2 VPN 技术在嵌入式 Internet 中的应用

虚拟专有网 VPN（Virtual Private Network），简称虚拟网，是利用开放网络，比如 Internet 等，建立一个物理位置分布不同，逻辑上统一的网络。从实现技术上来讲，VPN 大多数采用信息加密技术和防火墙技术来实现。目前 SSL、SOCK v5、IPSec、PPTP、L2TP、SKIP 六种协议是用在 VPN 中的几种主要的安全协议。根据不同需要，虚拟网有三种基本的结构：

・VPN 防火墙和 VPN 防火墙结构，用于在公司总部和它的分支机构之间建立 VPN；

・PC 和 VPN 防火墙结构，用于在公司总部和远程用户之间建立 VPN；

・公司与商业伙伴、顾客、供应商、投资者之间建立 VPN，是外部网 VPN。

随着 Internet 网络通信的安全性日益受到人们的关注，网上的信息通过明文的形式传输，Internet 很难作为企业或者组织内部安全传输信息的载体，为

此，IETF 于 1998 年 11 月专门制定了用来增强 IP 站点间安全性的系列协议 IPSec。另一方面，VPN 技术可以使用户采用廉价的 Internet 公共网络组建自己的虚拟专用网，因此，在设计 VPN 网关时可以采用 IPSec 协议，使用 ESP（Encapsulating Security Payload）对传输的数据进行严格的保护，使用 AH（Authentication Header）进行用户与数据确认，提高 IP 节点间的安全性。

对于 VPN 的支持主要是利用 IPSec 协议，IPSec 协议的目的是对 IP 层传输提供各种安全服务，其安全协议主要由 ESP 协议与 AH 协议组成，ESP 协议设计净负荷封装与加密，AH 协议设计数据确认，根据需要可选择之一或皆选。IPSec 协议在 IP 端到端主机间或安全网关间执行，对 IP 传输进行保护。IPSec 可用来保护一条至多条通路。由于 IPSec 协议在 IP 层上实现，因此它实现的安全服务对于在它的较高层次上的协议如 TCP、UDP、ICMP 等都能使用。

在嵌入式 Internet 中，VPN 的应用主要是远程访问 VPN。远程访问 VPN 的客户端应尽量简单，因为一般的用户都缺乏专门训练。客户应该能够手工或自动的建立一条信道，即当客户每次想建立一个安全通信信道时，只需安装 VPN 软件；如果对于某些客户需要随时都建立 VPN 时，设备能够自动的进行配置。因为防火墙要监视大量用户，有时需要增加或删除用户，这样可能造成混乱，并带来安全风险，因此，防火墙应集中，并且管理也要容易些。当然，可以考虑对防火墙进行分布式应用和集中管理。

对于嵌入式 Internet 来讲，构成嵌入式 Internet 的终端设备除了包含传统 Internet 原有的终端之外，又增添了诸如信息家电、移动通信终端等嵌入式系统设备，通常这种设备具有不固定的物理位置、灵活的物理分布和远距离的终端通信等特点。对于这些设备终端，通过 Internet 及相关技术将它们连接起来，构成嵌入式 Internet。当然，在嵌入式 Internet 中，人们需要将这些设备中属于自己的构成安全独立的私有网络，来保护内部的信息安全，例如，将来需要建立的家庭智能网络。根据前面所描述的，可以利用 VPN 技术来实现这一点，其实现结构如图 2.10 所示。

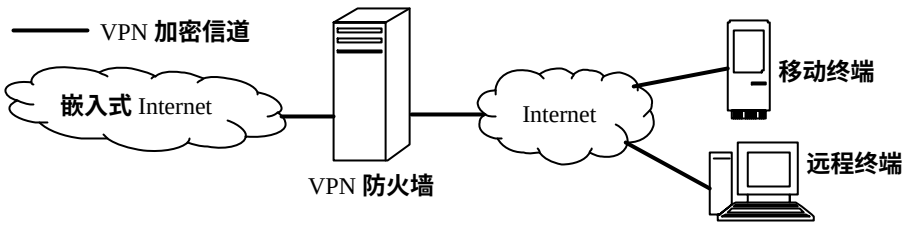


图 2.10　嵌入式 Internet 的 VPN 技术应用图

2.4.3 课题研究的嵌入式 Internet 结构模型

课题的研究将会持续很长一段时间，因此，从长远的角度来看，课题研究应该基于更具有长远发展的技术基础上来进行。课题研究的嵌入式 Internet 结构正是在基于 TCP/IPv6 的技术基础上建立起来的，这样，课题研究所选用的嵌入式 Internet 结构模型是一个以 TCP/IPv6 为基础，以 Internet 为传输媒介，

以移动技术、无线接入技术等为手段的虚拟专用网络。可以用图 2.11 来描述课题研究的嵌入式 Internet 结构。

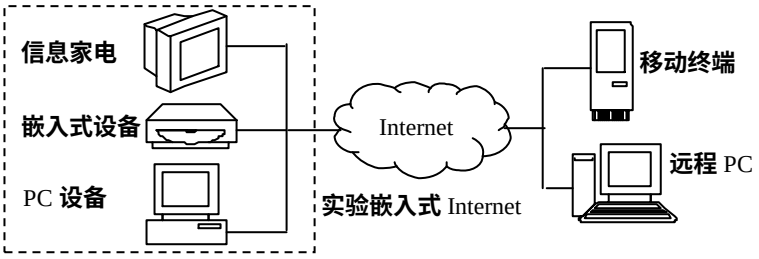


图 2.11　课题研究的嵌入式 Internet 结构图

第三章　嵌入式 Internet 网络安全分析

传统 Internet 面临着各种各样的安全隐患，要防范于未然，就必须深入分析其潜在的网络安全问题，寻找相应的对策。嵌入式 Internet 是在传统 Internet 的基础上发展而来，因此应该以分析传统 Internet 网络安全问题为基础，针对嵌入式 Internet 的网络安全进行分析。当然，对扩展 Internet 的网络安全性问题的分析也是必须的。

3.1 网络安全概述

网络安全是一个关系国家安全和主权、社会的稳定、民族文化的继承和发扬的重要问题，也是一个涉及计算机科学、网络技术、通信技术、密码技术、信息安全技术、信息论等多种学科的交叉学科。网络安全从其本质上来讲就是网络上的信息安全，它涉及的领域相当广泛。这是因为在目前的公用通信网络中存在着各种各样的安全漏洞和威胁。一般来说，凡是涉及到网络上信息的保密性、完整性、可用性、真实性和可控性的相关技术和理论都是网络安全所要研究的领域。下面给出网络安全的一个通用定义：网络安全是指网络系统的硬件、软件及其系统中的数据受到保护，不受偶然或恶意的原因而遭受到破坏、更改、泄露，系统能连续可靠正常地运行，网络服务不被中断。

从用户（个人、企业等）的角度来说，他们希望涉及个人隐私或商业利益的信息在网络上传输时受到机密性、完整性和真实性的保护，避免其他人或对手利用窃听、冒充、篡改、抵赖等手段对用户的利益和隐私造成损害和侵犯，同时也希望当用户的信息保存在某个计算机系统上时，不受其他非法用户的非授权访问和破坏。从网络运行和管理者的角度来说，他们希望对本地网络信息的访问、读写等操作受到保护和控制，避免出现“后门”、病毒、非法存取、拒绝服务以及网络资源非法占用和非法控制等威胁，制止和防御网络“黑客”的攻击。对安全保密部门来说，他们希望对非法的、有害的或涉及国家机密的信息进行过滤和防堵，避免其通过网络泄露，也避免由于这类信息的泄露对社会产生危害，对国家造成巨大的经济损失。从社会教育和意识形态角度来讲，网络上不健康的内容，会对社会的稳定和人类的发展造成阻碍，必须对其进行控制。

因此，网络安全在不同环境中的应用会得到不同的解释：

·运行系统安全。它侧重于保证系统正常的运行，避免因此系统的崩溃和损坏而对系统存储、处理和传输的信息造成破坏和损失，避免由于电磁泄漏产生信息泄露、干扰他人或受他人干扰；

·网上系统信息的安全。包括用户口令鉴别、用户存取权限控制、数据存取权限、方式控制、安全问题跟踪、计算机病毒防治和数据加密；

·网上信息传播安全。包括信息过滤，不良信息的过滤等，它侧重于防止和控制非法、有害的信息进行传播，避免公用通信网络上大量自由传输的信息

失控；

·网上信息内容的安全。它侧重于保护信息的保密性、真实性和完整性，避免攻击者利用系统的安全漏洞进行窃听、冒充、诈骗等有损于合法用户的行为。

显而易见，网络安全与其所保护的信息对象有关。本质是在信息的安全期内保证其在网络上流动时或者静态存放时不被非授权用户非法访问，但授权用户却可以访问。当然，网络安全、信息安全和系统安全的研究领域是相互交叉和紧密相连的。下面给出另一个网络安全的含义：网络安全的含义是通过利用各种计算机、网络、密码技术和信息安全技术，保护在公用通信网络中传输、交换和存储的信息的保密性、完整性和真实性，并对信息的传播及内容具有控制能力。网络安全的结构层次包括：物理安全、安全控制和安全服务。

本文所探讨和研究的嵌入式 Internet 防火墙系统，主要是从保护网络用户的角度来考虑的，是防御攻击和破译等人为性的安全威胁。所考虑的网络安全具有以下四个方面的特征：

·保密性。信息不泄露给非授权的用户、实体或过程，不会被其利用；

·完整性。数据未经授权不能进行改变的特征，即信息在存储或传输过程中保持不被修改、不被破坏和丢失的特性；

·可用性。可被授权实体访问并按需求使用的特性，即当需要时能够存取所需要的信息；

·可控性。对信息的传播及其内容具有控制能力。

计算机网络的发展，使信息共享应用日益广泛与深入。但信息在公共通信网络上存储、共享和传输，会被非法窃听、截取、篡改或毁坏而导致不可估量的损失。尤其是银行系统、商业系统、政府或军事领域对公共通信网络中的存储与传输的数据安全问题更为重要。如果因为安全因素使得信息不敢应用于 Internet，那么办公效率及资源的利用率都会受到影响，甚至使得人们丧失了对 Internet 及信息高速公路的信赖。

事物总是辩证的。一方面，网络提供了资源的共享性、用户使用的方便性，通过分布式处理提高了系统效率和可靠性，并且还具有可扩充性。另一方面，正是这些特点增加了网络攻击的可能性。对网络的威胁来自很多方面，并且随着时间的变化而变化。网络威胁是指对网络构成威胁的用户、事物、想法、软件等。网络威胁会利用系统暴露的要害或弱点，导致网络信息的保密性、完整性和可用性程度下降，造成不可估量的经济和政治上损失。威胁有两种：一种是无意的，一种是有意的。无意的威胁包括人为的操作错误、设备故障、自然灾害等很多不为人的意志转移的事件。有意的威胁包括窃听、计算机犯罪等人为的破坏。当前主要的网络威胁来自以下几个方面：

·自然灾害、意外事故；

·计算机犯罪；

·人为行为，比如使用不当，安全意识差等；

·“黑客”行为；由于黑客的入侵或侵扰，比如非法访问、拒绝服务、计算机病毒、非法连接等；

·内部泄密；

·外部泄露；

·信息丢失；

·电子谍报，比如信息流量分析、信息截取等；

·网络协议中的缺陷。

网络上的信息安全是网络安全的本质，但是在实际的应用中，信息安全与网络安全之间有着不同的侧重点。

3.2 传统 Internet 网络安全威胁分析

现在，计算机网络已经成为人们生活不可缺少的一部分，每天有成千上万的研究所、商家、政府部门和个人通过计算机网络交换信息，Internet 已经成为人们平时口中最为常见的话题。今天，一个普通的人可以很轻松的建立 Internet 服务器，和已有大多数的 Internet 服务器一样往往都存在一个共同的问题——网络安全问题。

网络安全问题是一个大范围的问题，其核心是指计算机网络上的信息安全问题。也就是说非法用户希望通过利用计算机网络系统的安全缺陷进行非法的网上信息窃取、伪造和破坏。

以下将对现有的 Internet 模型下存在的网络安全性问题进行分析，并利用特定的分类方法对现存的大多数网络攻击威胁进行分类。

1、对加密算法的攻击威胁

一般来说破译者可对密码进行惟密文攻击、已知明文攻击、选择密文攻击和选择明文攻击及穷举攻击，对特定算法还有特定攻击方法，如对 DES 这类迭代分组密码可选择差分密码分析法、能量攻击法；对公钥算法 RSA 可采用公用模攻击、低加密指数攻击、定时攻击等方法。每种加密算法都存在被有效攻击的威胁。

2、网络监听的攻击威胁

对于大部分的传输介质如 Ethernet、FDDI、Token-ring、模拟电话线、无线接入都可实施网络监听，其中尤以 Ethernet 与无线接入最为容易，因为这两者都是典型的广播型网络，所以很容易被人实施监听。

3、对网络协议弱点的攻击威胁

当初设计 Internet 各类协议时，几乎没有人考虑到网络安全问题，网络协议或缺乏认证机制，或缺少数据保密性。作为在 Internet 中应用最广泛的 TCP/IP 协议，从数据链路层到应用层协议的设计上都存在不同程度的安全弱点，攻击者可以利用这些弱点入侵网络。网络协议的攻击威胁如下：

1)、数据链路层。在 TCP/IP 分层模型的数据链路层，ARP 协议及与之有关的 Proxy ARP、Inverse ARP 数据包都可以被攻击者修改，从而改变数据包的流向，来达到攻击的目的。

2)、TCP/IP 层（传输层/网络层）。在 TCP/IP 的传输层和网络层中存在以下几种攻击威胁：

·利用 ICMP 的重定向消息破坏路由表，切断网络通信或窃听秘密通信；利用 ICMP 的不可达消息或 IGMP 的加入/离开组消息对单点或多点通信的主机实施拒绝服务（DoS，Denial of Service）攻击。

·通过伪造 IP 地址进行 IP Spoofing 攻击，或是结合 TCP 序列号攻击，实现 TCP/IP 劫持（TCP/IP Hijacking）及隐藏攻击者的地址，增加事后追踪的难度。这种网络攻击手段对进行网络反攻击存在一定的威胁。

·利用 Socket(套接字)进行 TCP 端口扫描,端口扫描可分为：TCP Connect 扫描、TCP SYN 扫描、TCP FIN 扫描和 Fragmentation 扫描等。端口扫描本身产生的攻击效应并不大，但这些方法是其他攻击的辅助手段，所以也存在一定的攻击威胁。

·借助海量数据包，使得目标主机耗尽 TCP 连接资源，这类攻击有“粘住”攻击、SYN-Flooding 攻击等。2000 年年初一系列针对国内外著名 Web 站点的分布式拒绝服务攻击就是大规模地采用了这种手段。

3)、应用层。在 TCP/IP 的应用层中包含了很多种协议，对于这些协议在设计实现时不可能都能够完全考虑到在安全性问题,因此可能存在如下的攻击威胁：

·伪造邮件冒充系统管理员，要求收信者发送口令或其他敏感信息；攻击者还可以对用户的邮箱或邮件服务器进行电子邮件轰炸。这种攻击威胁来自于邮件传输协议以及邮件服务器管理程序。

·利用 HTTP 允许相对 URL 的特性,攻击者改写某 Web 页上的所有 URL，扮演中间人（Men in the Middle）。IIS、Netscape Communicator Server for NT 等 HTTPD 都有 CGI-BIN 的安全问题，存在相应的攻击威胁。

·DNS 协议缺乏认证功能，攻击者可改变 IP 地址和域名的对应关系，从而绕过防火墙侵入系统。这种攻击威胁最大。

·利用 SNMPv1 无认证的弱点，攻击者只要给出有效的组名就能伪装成 SNMP 管理站点，因此，可以从各站点的 Agent 处取得大量的网络信息。

4、对软件设计弱点的攻击威胁

由于软件内部结构日益复杂，带来的一个直接后果就是安全隐患越来越多，而且当软件运行效率、可用性和安全性发生冲突时，大部分软件开发者会选择牺牲安全性。对软件设计弱点的攻击威胁如下：

1)、软件中使用弱的加密算法或是采用强加密算法但做了弱的实现。Bruce Schneier 就曾指出，GSM 加密算法，微软的 PPTF（Point to Point Tunneling Protocol）实现都存在该问题，一些软件中的随机数生成器会产生强度不足的随机数或忽视每次的随机初始化问题。这种有强算法弱实现的技术弱点会带来一定的攻击威胁。

2)、不同形式的缓冲区溢出会造成不同的危害，攻击者可能利用 UNIX 下

的 SUID 攻击取得 root 权限，输入极长的字符串作为命令参数或 URL 地址使 FTP、MAIL、Web 服务器运行错误而退出，Ping of Death 也可形成类似效果。

3)、利用 TCP/IP 协议处理程序中的错误实施使主机死机的攻击，如对 UNIX 和 Windows NT 用 TearDrop、Land 等工具攻击后，可使系统瘫痪。

4)、RPC（Remote Procedure Call）也会引起安全性问题，如 NFS 中从服务器端调出文件时缺乏控制，NFS 服务器容易错误地输出系统的数据库。

5)、攻击者利用移动代码 Java、ActiveX 和各种类 Script 程序中的漏洞，可以对服务器端与客户端发起攻击。

6)、其他攻击还有通过临时页面交换文件、未释放的窗口内存窥视用户密码和机密信息，借助不完善的系统恢复、备份程序等取得系统核心数据库，依靠远程 NT 登录的 SMB 窃取用户密码等。

5、对系统配置弱点的攻击威胁

这种攻击威胁是和前一种攻击相结合的，通常软件的某种特性只在特定系统环境下可能成为安全漏洞，而在其他场合表现正常。由于现代计算机系统的庞大复杂，很多系统管理员只使用系统默认配置或对更改配置后的安全后果不甚清楚，使攻击者的入侵更为容易，下面将从几个方面来描述相关的攻击威胁。

1)、使用一般不开放的 Telnet、finger 等服务器查询主机信息并试图登录，然后利用口令破译技术，破译不安全的弱口令，最后成功假冒合法用户。还可以利用配置不好的信任模型，靠 rlogin、rsh 等程序远程登录并执行命令。

2)、FTP、Web 服务器上文件、目录权限配置不当，容易混淆进程的拥有者和用户的执行权限，使得匿名用户透过 Internet 远程窃取、修改文件。

3)、对系统异常事件，如用户登录、打开敏感文件等的审计工作展开的不够完善，对系统日志文件未能定时查看清理，使得攻击者更容易隐蔽自己的攻击痕迹。

4)、为方便远程管理，管理员在防火墙中留下了隐蔽的 RAS 通道，从而使攻击者有可能利用该通道绕过防火墙的监控。

除了上述五种潜在的攻击威胁之外，其他的还有恶意程序（特洛伊木马、病毒）攻击，物理（偷窃加密机、盗取用户身份标识、抢劫数据中心或公钥存储中心）攻击，或因社会工程（Social Engineering）引起的密钥泄漏、权限让渡等。上述这些攻击手段都是传统 Internet 潜在的攻击威胁。

3.3 嵌入式 Internet 网络安全分析

3.3.1 嵌入式 Internet 的特点

嵌入式 Internet 是由嵌入式计算机系统、嵌入式瘦服务器、嵌入式网关和嵌入式因特网路由器等组成，这些已经成为嵌入式 Internet 时代的核心技术。嵌入式 Internet 的广泛应用将这个世界变得更加自动化、智能化和人性化。

嵌入式 Internet 广泛地应用到办公自动化、消费、通信、汽车、工业和军事领域，其典型应用包括：

·过程控制（Process Control），对生产过程中的各种动作、流程进行控制。

·网络通信（Telecommunication），如：程控交换机、路由器、BP 机、手机、桥接器、集线器和 Modem 等是网络通信的必备设备。

·智能仪器（Intelligent Instrument），如：示波器、医疗仪器。

·消费电子产品（Consumet Products），掌上电脑、数字电视、游戏机、洗衣机和微波炉等属于家庭和办公所用的消费电子产品。

·计算机外设（Computer Peripherals），包括打印机、扫描仪和磁盘驱动器等。

·军事电子（Military Electronics），如：雷达、电子对抗、坦克、战机和战舰等。

根据以上的这些应用，可以看出，嵌入式 Internet 的应用存在以下几个特点：

1、纯数据通信流量较小。

2、信息服务能力较弱。由于嵌入式系统的硬件资源有限，本身所包含的信息数据量较少，因此能够提供给其它系统的信息服务能力相应也比较弱。

3、较窄的服务带宽。由于嵌入式系统中硬件设备的处理能力有限，而且操作系统功能的裁剪，对于嵌入式 Internet 中信息服务请求的处理能力不足，因此，嵌入式 Internet 服务带宽较窄，容易产生阻塞。

4、远程控制。在嵌入式 Internet 中，嵌入式终端设备存在分散分布和远程化的特点，因此对其更新和维护大多需要通过远程控制来完成。

5、移动 IP。在嵌入式 Internet 中，大部分的嵌入式终端设备都是基于 WAP 协议（Wireless Application Protocol）的移动嵌入式终端设备。因此，在嵌入式 Internet 中，不能使用固定 IP 技术来进行身份鉴别，而应基于移动 IP 技术来完成。

6、操作系统的多样性。目前，在嵌入式 Internet 中，嵌入式终端系统使用着多种多样的嵌入式操作系统，这就造成嵌入式 Internet 中也存在操作系统的多样性。

7、VPN 和 IPSec 的应用。为了增加嵌入式 Internet 的应用范围及其可靠性，VPN 和 IPSec 技术得到了深入的应用。

3.3.2 嵌入式 Internet 网络安全威胁

由于嵌入式 Internet 的特点，嵌入式 Internet 的网络安全性问题与传统 Internet 的网络安全性问题有所不同，以下就嵌入式 Internet 的网络安全性问题进行分析。

嵌入式 Internet 网络安全问题是一个大范围的问题，其核心是指嵌入式 Internet 上的信息安全的问题。也就是说非法用户希望通过利用嵌入式 Internet 上系统的安全缺陷进行非法的网上信息窃取、伪造和破坏。以下主要是分析现有的嵌入式 Internet 模型下所存在的网络安全威胁。并用特定的方法对嵌入式 Internet 存在的网络安全威胁进行分类。

1、对加密算法的攻击威胁

在嵌入式 Internet 的应用领域，IPv6 技术将会得到广泛的应用，在 IPv6 技术的应用过程中，加密算法的应用也非常广泛，特别是在身份认证和用户权限划分方面。嵌入式 Internet 中的加密算法的攻击威胁相比传统 Internet 中存在的加密算法的攻击威胁具有更大的范围，它不仅包括了传统 Internet 中所具有的攻击威胁。特别是目前嵌入式 Internet 中 IPv6 和 IPv4 的共同使用，存在 IP 版本不同的攻击可能性。

2、网络监听的攻击威胁

根据第二章对嵌入式 Internet 结构模型的描述，可以了解到嵌入式 Internet 中终端设备物理分布的最大特性是不集中、分散和远程化。也就是说，在嵌入式 Internet 中，构成嵌入式 LAN、嵌入式 MAN、嵌入式 WAN 的设备终端在物理分布上和传统 Internet 存在很大的差别。同时，在嵌入式 Internet 中，移动设备终端也大量的存在，针对这一特点，构成嵌入式 Internet 网络的传输介质绝大部分都是 Ethernet 和无线通信传输。然而，对于通过 Ethernet 传输和无线通信传输技术来讲，由于这两种方法都是典型的广播型网络，所以很容易对其实施网络监听。并对其传输的信息进行截取，通过分析监听到的数据信息，就能够获取到相关信息，达到对嵌入式 Internet 的攻击目的。

3、对网络协议弱点的攻击威胁

在嵌入式 Internet 中，网络协议的应用和 Internet 中的网络协议应用一样，除了在部分嵌入式网络中会应用其特有的传输协议之外，基本上都会使用 Internet 网络协议。因此，嵌入式 Internet 针对其网络协议的弱点所存在的攻击威胁和 Internet 的网络协议弱点的攻击威胁相似。

嵌入式 Internet 也有 TCP/IP 协议的应用，和传统 Internet 一样，从数据链路层到应用层协议在设计上都存在不同程度的安全弱点，可能被攻击这加以利用而入侵网络。当然，针对 WAP 协议的攻击也和传统 Internet 中的 WAP 协议相似，所以在嵌入式 Internet 中，针对网络协议弱点的攻击威胁基本上和传统 Internet 存在的攻击威胁相同。但是也存在不同的地方，主要是嵌入式 Internet 局部范围内的通信协议种类相比传统 Internet 的多一些，同时在上层的应用上嵌入式 Internet 中的协议应用较少，因此嵌入式 Internet 的网络协议弱点攻击威胁相对少于传统 Internet。

当然，在嵌入式 Internet 中，会更多使用诸如 SSL、IKE、IPSec 等安全协议，针对这些安全协议也会存在一些相应潜在的攻击威胁。但这些加密协议在设计上对于安全的着重考虑，因此相比传统 Internet 来讲，安全性更加高一些。

4、对软件设计弱点的攻击威胁

嵌入式 Internet 中的终端设备，大部分都是嵌入式系统设备，这些嵌入式系统设备里面使用的都是嵌入式系统软件和应用软件。近年来，随着嵌入式技术的应用越来越广泛，嵌入式软件技术的日益复杂，也继承了计算机软件技术的应用发展方向。对于嵌入式系统设备的网络化来讲，这种趋势带来的一个直接后果就是嵌入式系统设备的安全隐患越来越多；而且当软件运行效率、可用

性和安全性发生冲突时，软件开发者也会选择牺牲安全性。因此，对于嵌入式软件自身的安全性也不容忽视。

嵌入式软件具有实时性、异步事件的并发处理、应用/操作系统一体化、应用可固化、鲁棒性和灵活性等特点。嵌入式软件的安全性，自然是嵌入式系统安全性中重要的一环，也是目前嵌入式软件技术的重要研究方向之一。

5、对系统配置弱点的攻击威胁

嵌入式 Internet 的系统服务相对来讲并不如传统 Internet 那么多，而且对于大多数嵌入式 Internet 终端设备和嵌入式 Web 服务器的系统配置，都可能是使用远程配置的方式，这就使得对于嵌入式 Internet 中的设备终端的系统配置面临更为严重的攻击威胁。

这种攻击往往和前一种针对嵌入式软件设计弱点攻击威胁相结合的，并和嵌入式 Internet 中的大部分终端设备的资源限制相关，通常因为嵌入式软件的特点以及嵌入式终端设备资源的限制会使得嵌入式 Internet 在系统配置管理方面趋于简单化、远程化，在正常的情况下这种管理方式没有问题，但是在特定的环境下可能成为安全漏洞。由于嵌入式 Internet 中的嵌入式系统设备数量巨大，同时这些终端设备和庞大复杂的现代计算机系统之间的相互融合，而且嵌入式 Internet 的物理分布不集中，使得很多系统管理员在系统配置时选择缺省配置或者即使更改也对更改后的安全后果不甚清楚，使系统配置更容易成为攻击突破口。

另外，除了上述五种潜在的攻击威胁之外，针对嵌入式 Internet 的攻击威胁还有很多。其中针对嵌入式 Internet 的物理分布不集中的攻击威胁是嵌入式 Internet 攻击威胁中最致命的，这种物理攻击涉及的范围非常广泛。其他的还有恶意程序攻击、密钥泄漏、权限让渡等。

3.4 扩展 Internet 网络安全分析

扩展 Internet 可以说是新一代 Internet，它是传统 Internet 在融合其他各种新型网络概念后而形成的。因此，在分析扩展 Internet 的网络安全性的时候，不但要考虑扩展 Internet 本身的特点，还要结合传统 Internet 和嵌入式 Internet 来进行分析。

3.4.1 扩展 Internet 的特点

扩展 Internet 是成熟的嵌入式 Internet 在传统 Internet 中的应用过程中扩展了传统 Internet 的应用领域而形成新的 Internet 模型。对于扩展 Internet 来说，它是容纳了传统 Internet 和嵌入式 Internet 后形成的新型应用模型，它是在“后 PC”时代发展起来的新一代 Internet。在扩展 Internet 中，包含了许多的异构、异域、异类的网络实体，因此扩展 Internet 的网络复杂度也就比传统 Internet 以及嵌入式 Internet 的网络复杂度要高得多。所以，在网络通信、信息传输以及控制过程中需要考虑到这些各种不同网络之间相互作用的问题，因此扩展 Internet 具有更多、更复杂的潜在网络安全威胁。

扩展 Internet 在继承传统 Internet 和嵌入式 Internet 特点的基础上，具有其

本身的应用特点：

1、网络结构复杂。扩展 Internet 是以传统 Internet 和嵌入式 Internet 为基础，根据前面的描述，可以知道其中存在相当多的不同的网络结构和网络应用。

2、网络协议众多。扩展 Internet 继承了传统 Internet 和嵌入式 Internet 中的所有通信协议。

3、网络内部通信不一致性。这种不一致性不但包括通信协议的不一致，而且还包括通信业务、通信信息流量、通信带宽等方面的不一致性。

3.4.2 扩展 Internet 网络安全威胁

对于扩展 Internet 的网络安全威胁，需要在传统 Internet 和嵌入式 Internet 的基础上共同来考虑。从应用的角度来看，扩展 Internet 的网络安全威胁既来自于传统 Internet 和嵌入式 Internet 的应用，同时也来自于传统 Internet 和嵌入式 Internet 之间的交互应用。

首先，扩展 Internet 的网络安全威胁继承了传统 Internet 和嵌入式 Internet 的网络安全威胁。

其次，传统 Internet 和嵌入式 Internet 的交互涉及到用户通过 Internet 对嵌入式 Internet 进行服务请求，嵌入式 Internet 需要满足用户类似 Internet 的服务请求，反之相同。由于扩展 Internet 存在网络协议众多的特点，对于这种交互服务，很容易进行探测、欺骗等攻击。

另外，由于扩展 Internet 存在结构复杂、通信不一致性等特点，对扩展 Internet 进行扫描、拒绝服务等这类针对终端系统的有限资源和不一致性的网络攻击很容易成功，并且会给扩展 Internet 带来巨大的损失。

3.5 网络安全关键技术分析

通过以上对网络安全性问题的分析，可以看出，网络安全包含了多方面的内容，不同的应用环境有不同的安全问题，不同的层次也有不同的安全要求。结合各种不同的应用环境，可以给网络安全定义一个结构层次，如图 3.1 所示：

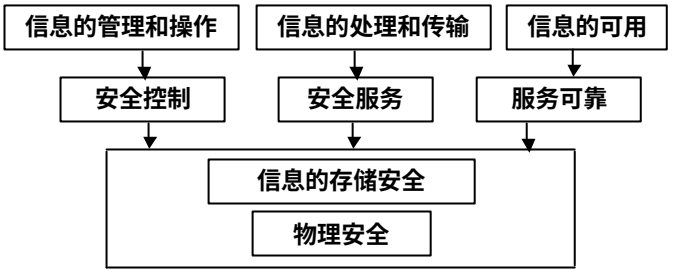


图 3.1　网络安全结构层次图

网络安全首先要保障网络上信息的物理安全。物理安全是指在物理介质层次上对存储和传输的信息的安全保护。物理安全是信息安全最基本的保障，是不可缺少和忽视的组成部分。一方面，研制生产计算机和通信系统的厂商应该在各种软件和硬件系统中充分考虑到系统所受到的安全威胁和相应的防护措施，提高系统的可靠性；另一方面，也应该通过安全意识的提高，安全制度的

完善，安全操作的提倡等方式使用户和管理维护人员在系统和物理层次上加强信息的保护。

安全控制是指在网络终端设备操作系统和网络通信设备上对存储和传输的信息的操作和进程进行控制和管理，主要是在信息处理层次上对信息进行初步的安全保护。安全控制主要是通过现有的操作系统或网管软件、路由器配置等实现。安全控制只提供了初步的安全功能和信息保护，仍然存在着很多漏洞和问题，但由于实际情况的限制，很难对此进行弥补和更改。

安全服务是指在应用层对信息的保密性、完整性和来源真实性进行保护和鉴别，满足用户的安全需求，防止和抵御各种安全威胁和攻击手段。这是对现有操作系统和通信网络的安全漏洞和问题的弥补和完善。安全服务主要包括：安全机制、安全连接、安全协议和安全策略。

以上从网络安全层次结构角度探讨了有关网络安全的注意环节。针对网络安全的这几个环节，现在已经发展出来了很多成熟的网络安全技术。其中网络安全的关键技术有：

- 主机安全技术；
- 身份认证技术；
- 访问控制技术；
- 密码技术；
- 防火墙技术；
- 安全审计技术；
- 安全管理技术。

第四章　嵌入式 Internet 防火墙技术研究

自从 1986 年来美国 Digital 公司在 Internet 上安装了全球第一个商用防火墙系统，提出防火墙概念后，防火墙技术得到了飞速的发展。特别是 1996 年以后，随着防火墙技术和密码技术的结合，防火墙市场得到了长足的发展。目前国外已经有 Check Point、symantec、Cyberguard、Netguard、Netscreen、Cisco 等公司，国内有天融信、清华比威、易尚、中软华泰等公司推出了功能各不相同的防火墙系列产品。

随着嵌入式 Internet 的发展，防火墙技术也需适应嵌入式 Internet 应用环境，成为新一代 Internet 安全保障的有效方式。

4.1 防火墙技术概述

Internet 的广泛应用也带来了极其敏感的副作用，这就是前面所说的 Internet 存在的网络安全攻击威胁。随着网络在商业和政府机关的普及，如何兼顾安全与方便，正日益引起人们的重视。

4.1.1 防火墙的概念

防火墙是一种常见的网络安全机制。“防火墙”这个词来自建筑物中的同名机构，从字面上的意思来理解，它可以防止火灾从建筑物的一部分蔓延到其它部分。Internet 防火墙的作用是：防止其上的不安全因素从外部蔓延到企业或组织的内部网络。

一般来说，Internet 防火墙是在一个特定的控制点（也称为遏制点，Check Point）上根据一定的规则进行对比，以判断特定类型数据包是通过与否。应用防火墙时，首先要明确防火墙的缺省策略，是接受还是拒绝。如果缺省策略是接受，那么没有明确应该是拒绝的数据包可以通过防火墙；如果缺省策略是拒绝，那么没有明确应该是拒绝的数据包不能通过防火墙。显然后者的安全性能更高。例如，假如已经设置防火墙禁止了对某系统提供的 FTP 服务（21 端口服务）的访问，而该系统有在另外的高端端口重开了 FTP 服务，如果缺省策略是接受，那么高端端口上的 FTP 服务是可以访问的；如果缺省策略是拒绝，则该 FTP 服务不可访问，使用时应该根据自己的实际情况选择这两种策略。

4.1.2 防火墙的分类

防火墙从技术角度可以分为两大类：包过滤防火墙（Packet Filtering）型和代理服务（Proxy Service）型；从应用角度也可以分为两大类：边界防火墙类型和桌面防火墙类型。

4.1.2.1 防火墙的技术分类

1、包过滤型防火墙

包过滤型防火墙根据数据包的包头中某些标志性的字段，对数据包进行过滤。当数据包到达防火墙时，防火墙根据包头中下列字段中的一些或全部进行

判断，决定接受还是丢弃该数据包：

- 源地址、目的地址；
- 协议类型（TCP、UDP、ICMP）；
- TCP/UDP 协议的源端口号、目的端口号；
- ICMP 消息类型等。

不同的防火墙产品还可能附加其它的过滤规则，如 TCP 协议标志（SYN、ACK、FIN 等）、进入或外出防火墙所经过的网络接口等。通过 TCP 或 UDP 端口进行过滤会带来较大的灵活性，特定的服务/协议是在特定的端口上提供的，阻塞了与特定端口相关的连接也就禁止了特定的服务/协议，所以用户可以根据自己的网络访问策略决定要阻塞那些协议。

通过设置包过滤规则，可以阻塞来自或去往特定地址的连接。例如，从收费方面考虑，校园网可能需要阻塞来自或去往国外站点的连接。从安全方面考虑，某组织的内部网可能需要阻塞所有来自外部站点的连接。

包过滤防火墙的弱点主要在于规则的复杂性。首先确定了基本策略（例如“拒绝”），然后设置一系列相反的（如“接受”）规则。但很多情况下，需要对已经设立的规则设定一些特例，这样的特例越多，规则就越不容易管理。例如，已经设立了一条规则允许对 TELNET 服务的访问，然后又要禁止某些系统对 TELNET 服务进行访问，那么只能为每个系统添加一条相应的规则。我们要防止添加的补丁规则会与整个防火墙策略产生冲突，过于复杂的规则将不易测试。

从概念上来讲，防火墙和网关（路由器）是不同的。但在具体实现中，包过滤防火墙通常还具有网关的功能，对数据包进行过滤后再转发到相应的网络，这样的包过滤防火墙或者网关称为“包过滤网关”。

2、代理服务型防火墙

代理服务防火墙可以解决包过滤防火墙的规则复杂性问题。所谓代理服务，是指在防火墙上运行某种软件（称为代理程序），如果内部网需要与外部网通信，首先要建立与防火墙上代理程序的连接，把请求发送到代理程序；代理程序接收该请求，建立与外部网相应主机的连接，然后把内部网的连接请求发送到外部网相应的主机。反过来也一样。内部网和外部网的主机之间不能建立直接的连接，而要通过代理服务进行转发。

一个代理程序一般只能为某几种协议提供代理服务，其他所有协议的数据包都不能通过代理程序（从而不可能在防火墙上开后门以提供授权服务），这就相当于进行了一次过滤；代理程序还有自己的配置文件，可对数据包的其他一些特性（如协议、目的地址、源地址等）进行过滤，有时这种过滤条件甚至比纯粹包过滤的功能还要强大。

包过滤和代理服务结合起来使用，可以有效的解决规则复杂性问题。包过滤防火墙只需要让那些来自或去往代理服务器的包通过，同时简单地丢弃其他包，其他进一步的过滤由代理服务程序进行。

使用代理服务可以根据协议/服务的某些细节进行过滤。代理服务禁止了源主机到目的主机的直接连接，可以由此实现一定程度的信息隐藏，使内部系统的名字不能通过 DNS 被外界知道，外界可以只知道代理服务器的名字，并通过它使用内部系统。代理服务器在转发数据包之前，还可以预先进行身份验证和日志纪录。

代理服务的转发与（包过滤）网关的转发层次不同。（包过滤）网关只是在 IP 层次上简单的转发数据包，代理服务器则在应用程序的层次上转发某种协议/服务的数据流。从这个意义上说，代理服务主机也经常被称为“应用程序网关（Application Gateway）”。

代理服务也有些缺陷。诸如在 TELNET 这样的客户/服务器型的协议中，需要两个步骤进行连接，有些代理服务程序要求对客户端程序进行一些修改，以适应代理服务的要求。

4.1.2.2 防火墙的应用分类

边界防火墙是指设置在网络边界，在内部企业网和外部互联网之间构成一个屏障，通过包过滤和代理等其他技术进行网络存取控制，以保护内部网的安全。

桌面防火墙是指在本机上运行，通过包过滤技术和其他相关技术保护本机访问 Internet 的安全。

4.1.3 防火墙的体系结构

防火墙的体系结构有多种，下面就其中主要几种传统体系结构进行介绍。

4.1.3.1 双穴网关（Dual-homed Gateway）

双穴网关是包过滤网关的一种替代，与包过滤网关一样，双穴网关也位于外界 Internet 与内部网络之间，并且通过两个网络接口分别与之相连。但是，IP 转发功能被禁用，网关功能是通过提供代理服务而不是通过 IP 转发来实现的，显然，只有特定类型的协议请求才能被代理服务处理。双穴网关实现了“缺省拒绝”策略，可以得到较高的安全性。

另一种双穴网关的使用方法是，要求用户先登录到双穴网关，再从上面访问外界。这种方式一般不提倡，因为在防火墙上是最好保留尽可能少的账户。

4.1.3.2 屏蔽主机型防火墙（Screened Host Firewall）

这种防火墙其实是包过滤和代理功能的结合，其中代理服务器位于包过滤网关靠近内部网的一侧。代理服务器只安装一个网络接口，它通过代理功能把某些服务传送到内部某些主机。而包过滤网关把那些危险的协议过滤掉，不让他们到达代理服务器那里。

从安全性考虑，内部网络与外界 Internet 之间的通信都通过代理服务器进行，包过滤网关通过过滤对代理服务器进行保护。这样，包过滤网关应该只让那些来自或去往代理服务器的数据包通过，而丢弃其他所有数据包。

但是有的协议没有相应的代理服务。如果使用这些协议的风险经过考虑认为可以接受（则相应数据包称为“可信任的”），可让该协议的数据包通过包过

滤网关。这些包不经过代理服务器，直接去往相应的服务器。这种结构可以用图 4.1 进行描述。

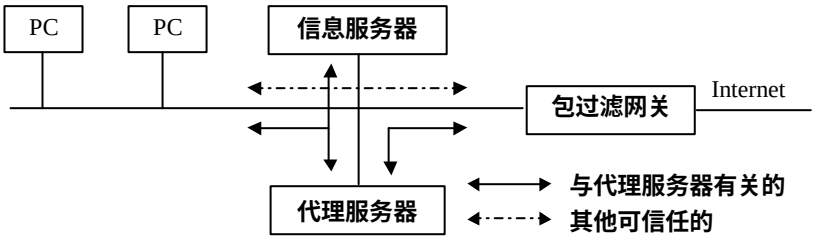


图 4.1　屏蔽主机型防火墙

这种类型的防火墙主要的安全问题也就在于“可信任”上。前面提到了包过滤方式的缺点在于规则配置比较复杂，规则数目增多容易互相冲突。如果只放行与代理服务器相关的数据包，包过滤网关的规则配置将非常简单；由于对于每一种“可信任的”数据包，都需要为他们分别配置包过滤规则，这样又把复杂性问题遗留了下来。而且这些“可信任的”服务仍然有可能成为潜在的安全漏洞。解决这个问题的最好方法就是寻找“可信任的”服务的代理软件，好在各种服务的代理软件越来越多了。

4.1.3.3 屏蔽子网型防火墙（Screened Subnet Firewall）

这种防火墙是双穴网关和屏蔽主机防火墙的变形。如图 4.2 所示，该系统中使用了两个包过滤网关在内部网络和外界 Internet 之间隔离出一个受屏蔽的子网。有些文献称这个子网为“非军事区（DMZ）”。代理服务器、E-mail 服务器、各种信息服务器（包括 Web 服务器、FTP 服务器等）、Modem 池及其它需要进行访问控制的系统都放置在 DMZ 中。

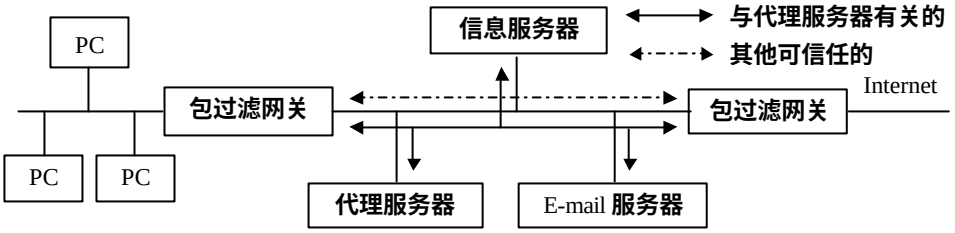


图 4.2　屏蔽子网型防火墙

与外界 Internet 相连的那个网关称为“外部路由器”，它只让与 DMZ 中的代理服务器、E-mail 服务器、信息服务器有关的数据包通过，其他所有类型的数据包都被丢弃，从而把外界 Internet 对 DMZ 的访问限制在特定的服务器范围内。内部路由器的情况也如此。

这样，内部网与外界 Internet 之间没有直接连接，它们之间的连接要通过 DMZ 来中转，这与双穴网关的情况是一样的。不同的是屏蔽子网防火墙使用了包过滤网关把数据包转发到特定系统，代理服务器只需要安装一个网络接口。

通过配置，可以做到只有 DMZ 的代理服务器、E-mail 服务器、信息服务

器是外界不可见的，外界无法知道其它系统的存在，无法通过他们的名字直接访问它们。与屏蔽主机防火墙一样，那些没有代理又需要使用的服务，需要在外部路由器处网开一面,这就造成了一定的安全威胁。但是在屏蔽子网结构中，可以把这些需要服务的系统直接放置在 DMZ，这些系统通过代理服务器与内部子网联系。虽然，这种方法并不完美，但对于需要高度安全性的系统来说，可以算作一个不错选择。

4.2 分布式防火墙

防火墙的发展从技术角度来讲已经历了几代,第一代防火墙为包过滤路由器（Packet filtering Router）或筛选路由器（Screening Router），是基于过滤技术实现的；第二代防火墙为代理服务器（Proxy Server），属于应用层的过滤；第三代防火墙是具有状态监控功能（ Stateful Inspection）,在网络层对数据包的内容进行检查。这些传统防火墙不能完全解决网络安全问题，随着网络攻击手段和信息安全技术的发展，产生了一种新的防火墙技术——分布式防火墙。

4.2.1 传统防火墙存在的问题

不管你喜欢与否，Internet 已经成为世界上最活跃的领域。一方面，人们试图扩大他们信息交换的连接性和有效性，充分利用现代 Internet 提供的所有好处。另一方面，人们又担心他们的隐私被侵犯，不得不一起合作来保护他们的数据或名誉不被一些恶意的用户所破坏，甚至有可能是公司或者政府的间谍。防火墙的出现解决了这一问题，实现了在两个网络间根据一些安全策略进行通信过滤，从而保护网络的安全。

从结构方面来讲，传统防火墙依赖以下两个概念：

·严格的拓扑结构。这指的是传统防火墙将整个网络分成两个部分：一边被防火墙保护，因此可以信任；另一边通常是意味着外部的 Internet，简单的来说是不被信任的。

·进入点，有时叫做扼制点（chokepoint）。所有在信任和不信任网络之间的通信都是通过这个点，并完成不同的存取控制。

随着 Internet 的扩张，Internet 的可用性、有效性、连接性、分散性和移动性需求已经有了很大的增长，这种需求决定了传统防火墙很难满足当今的需求。当传统防火墙的方案在小型和中型网络中运行得很好的时候，在大型网络中的应用就会遭受到一些限制，而且 Internet 的移动性和分散性需求是传统防火墙技术不能满足的。体现在：

1、拓扑结构的依赖

依靠拓扑结构就意味着网络被防火墙人工分成信任的和不可信任的两部分，一个计算机是否能够被信任，完全依靠它是否在物理上和可信任网络相连接。在这种情况下，假设一个职员为了参加一个会议将膝上电脑带出了城，那么它就不能够被公司的防火墙所保护,因为这个膝上电脑并没有和公司的网络相联接。另一方面，一个参观者的膝上电脑就可以被公司的防火墙所保护，因为它在物理上和公司的网络相联接。

在后一个例子中，那个参观者的膝上电脑受到保护或许并不被希望，甚至严重一点来讲，那个膝上电脑将被看成为一个内部主机，虽然它没有被信任的条件，结果这个膝上电脑就有机会来存取那些公司所保护的信息。对于前者，现在的解决方案是利用安全隧道 SSH 或 VPN 来组建公司私有的网络连接，然后再返回到 Internet。这个解决方案增加了额外的资金耗费和多余的连接。对于后者，通常利用多个防火墙来将网络分成更小的部分，而每一部分拥有不同的安全策略。但是这样的解决方案又增加了网络结构和网络管理的复杂性，这将直接影响网络的连接和配置。

2、单个进入点

对于传统防火墙技术，单个进入点是执行防火墙定义的安全规则的地方。所有进入和出去的数据包都应该通过这个点，并且通过安全规则的检查。一些站点在他们的单个进入点处网关上使用各种不同的病毒扫描程序，同时这增加了防火墙的负荷，而且，还会因为单个进入点的失效而导致整个内部网络的失效。

3、主机认证

在传统防火墙技术里，主机是通过 IP 地址来区别他们是“内部的”还是“外部的”主机，这就使得 IP 地址的哄骗技术成为可能，例如，黑客可以通过诸如 rlogin 等协议发起基于劫持内部主机到其他可信任系统通信的攻击。

传统防火墙另外的缺点是不同主机的不同连接请求很难得到承认，特别那些基于 DHCP 或者 BOOTP 协议的网络。DHCP 使用的目的是解决那些 IP 地址不够或者有许多移动用户的网络（或子网）。提到这个问题是因为当主机从网络上启动后，IP 地址是被动态分配的，由于保留主机的 IP 地址是可能的，因此主机总是在启动后获得相同的 IP 地址。但是这种配置失去了 DHCP 所提供的灵活性。另一个解决方案是将多个 IP 地址范围用不同的存取规则分配给 DHCP 服务器。但是在现在应用的主机中仅仅是通过其 MAC 地址进行授权认证，但是 MAC 地址却能够通过软件镜像。

4、包过滤技术

在传统防火墙中，包过滤经常用来检查数据包的包头中一些通用的字段，然后将这些字段信息与安全规则相比较，同时一些没有定义的特殊字段则被忽略了。这些被忽略的字段就能够被黑客应用来构建一些特殊的 IP 包，这些 IP 包在那些没有被定义的字段中设置特殊的值，用来鉴别目标操作系统的信息。例如 ICMP Echo 请求包中的 TOS（Type of Service）字段，一个 ICMP Echo 请求回答应该用相同的 TOS 的值。下面的例子表示了一个对于 Linux（内核是 2.2.14）利用 sing 命令的 ICMP Echo 请求的回应，sing 命令能够将一个 ICMP Echo 请求中指定的特殊字段的值都显示出来，如下所示：

```
[root@shizi bin]# ./sing –echo –TOS 8 host_address
SINGING to host_address (IP_Address): 16 data bytes
16 bytes from IP_address:  seq=1 tt1=234 TOS=8 time 6.070 ms
16 bytes from IP_address:  seq=2 tt1=234 TOS=8 time 5.731 ms
16 bytes from IP_address:  seq=3 tt1=234 TOS=8 time 6.082 ms
```

所有的回应指定的 TOS 字段的值都是一样的。可以用 tcpdump（UNIX 的一个工具）工具进行更进一步的分析。这种行为已经被 Windows2000、Novell Netware 和 UNIX 排除掉了，它们不再允许 ICMP Echo 回应包中使用 TOS 字段。

所有的协议中都存在许多没有被定义的字段，例如 TCP 和 UDP，对于包过滤来讲，很难将包的每个字段进行检查。当包过滤技术不能足够了解数据包连接信息的时候，就很难判断一个 TCP 包是不是正在进行的对话中的包。因此，攻击者可以构造一个欺骗的数据包来进行“隐秘扫描”攻击。

一个好的例子便是 FTP 协议。一般来讲，FTP 客户端利用端口命令来指定端口号接收从其他方面来的入站数据包，但是这并没有被防火墙所了解，防火墙只看到入站的数据包试图达到不被允许的端口，因此，整个会话就被阻塞了。现在的解决方案是建立应用级的代理来处理这些协议。UDP 包也存在同样的问题。

4.2.2 分布式防火墙技术

Steven M. Bellovin 在 1999 年提出来一种新的防火墙技术——分布式防火墙。其基本的想法就是用一些规则语言将安全规则进行集中定义，然后利用分布式系统管理工具将安全规则分布到每个网络终端主机上，这些主机对进出其中所有的数据包进行过滤，这些数据包被接收或者被拒绝是根据相应的安全规则和身份认证来决定的。

4.2.2.1 分布式防火墙概念

传统防火墙依靠拓扑结构和控制进入点（Entry Point）来实现网络安全防御功能。也可以说是依靠一种假设：每个在进入点（也就是防火墙）一边的人都应该被信任，并且在进入点（Entry Point）另一边的任何人都至少是潜在的敌人。近年来，随着大量的 Internet 设备连接，这种假设也成为了一种疑问，外部网络的用户是否都是不被信任的，如果是被信任的，则需要允许外部用户访问防火墙的内部网络；另一方面，在加密隧道没有被应用的时候，用于 Internet 连接的电信交换机也需要被保护。

防火墙是一个规则控制工具，它允许管理者来设置对通过的网络数据包进行过滤的规则，防火墙则按照这些规则对通过的网络数据包进行过滤。在分布式防火墙中，规则仍然集中定义，但过滤在每一个网络终端（End Point）处执行。分布式防火墙依赖以下三个概念：

- 一种可以描述某种连接应该被允许或禁止的分布式规则语言，如：KeyNote。
- 一种分布式系统管理工具，例如 Microsoft 的 SMS 或 ASD。
- 网络安全协议，如 IPSec。

分布式防火墙的基本思想是：利用一种编译器将规则语言翻译成一种内部格式，再通过分布式系统管理工具将这些规则文件分布到所有的终端主机上，而这些主机上的防火墙按照相应规则对进出主机的网络数据包进行过滤，同时

还需对每个数据包的发送者进行身份鉴别，从而决定如何处理数据包，达到保护主机的目的。

目前，有很多规则语言可以使用，包括类似 Firmato 的面向文件配置语言，和现有绝大多数商业防火墙上所使用的 GUI 通用规则语言，例如 KeyNote。

如何鉴别可信任主机是分布式防火墙中最重要的。可以用 IPSec 加密的名字作为主机身份的标识，这样的身份标识是比较安全、可靠的身份标识。这种技术抛开了 IP 地址，主机身份标识方法不容易被欺骗，因此可以避免身份“哄骗”的网络攻击。如果一个主机已经利用加密名字标识被授权，那么无论主机物理位置处在那个地方，这个授权都能够被安全的使用。

规则能够被分布式管理工具动态的分布到终端系统中。例如，一个协议许可服务器或者安全服务器能够询问某种通信是否应该被允许。传统防火墙也能够做到这一点，但重要的是传统防火墙缺少了对网络请求上下文的了解。分布式防火墙系统的终端主机可以了解到相关信息，这些信息能够在网络协议上传输，但是会增加其复杂性。

在一个典型的分布式防火墙系统中，终端主机用户并不一定需要是了解系统的管理员。为了简化系统管理，允许一定级别的集中控制，可以利用分布式系统管理工具对终端主机进行系统管理，也可以利用分布安装规则补丁软件的方法。大多数的分布式防火墙系统中，都是利用分布式系统管理工具来实现。

由于终端主机的系统管理员不再必须是分布式防火墙系统的系统管理员，分布式防火墙还不受拓扑技术的限制，同时终端主机的身份还可以用主机的名字来标识，因此，在定义分布式防火墙规则的时候就显得更加灵活多样。

在利用终端主机名字进行身份认证时，如果某个不可信任的终端主机通过了身份认证，而分布式防火墙系统中的其它终端主机上的防火墙组件通过规则过滤还能够进行进一步的判断，仍然可以鉴别出这个已经通过的不信任终端主机，因此分布式防火墙系统具有比传统防火墙更为有效的身份认证机制。在分布式防火墙系统中，所有的终端主机在某种意义上来讲具有相同的安全性能。在传统的防火墙应用中，如果任何一个内部计算机被控制，那么从这个主机上发动的网络攻击则非常危险，然而，在分布式防火墙系统中，这种即使控制分布式防火墙中的一个终端主机也很难发动有效的网络攻击。例如，网络电子邮件，大多数拥有防火墙的站点都留下有限的几个指定的主机来接收外界的电子邮件，它们依次将电子邮件传递给内部的邮件服务器。传统的防火墙通过允许内部邮件网关的 SMTP（Port25）连接来实现这个功能，而连接到内部其它的主机就会被阻塞，在传统的防火墙内部，Port25 的连接是没有限制的。如果使用分布式防火墙，所有的计算机都有一些连接到 Port25 的安全规则。邮件网关允许外部任何主机对这个端口 25 的连接，但对于其它内部的终端主机，需要根据内部终端主机的身份认证来判断，分布式防火墙仅仅允许内部主机与邮件网关的连接，即使一个被控制的内部终端主机也不可能在受分布式防火墙保护的网络中通过邮件的 Bug 进行攻击。

当然，分布式防火墙还有其它的一些优势。首先是整个网络不再只有一个扼制点（chokepoint），它使得网络不会再因为防火墙中的一个点的失效而导致整个网络的失效，同时分布式防火墙所保护网络的吞吐量（Throughput）不再受到防火墙速度的限制。

分布式防火墙的另一个优势在于，其终端主机本身知道何时监听特殊的数据连接，能够在探测到数据连接后随时响应，而传统防火墙并不具有认识终端主机意图的能力，只能通过对各种协议外部可见的信息进行分析，来判断终端主机的意图。例如，一个进入的 TCP 数据包的 ACK 位被设置了，那么有时就会假定这个数据包为合法的数据包，原因是如果这个数据包为一个正在进行对话的一部分，那么这个数据包只能是合法的，当然这个对话初始是被防火墙允许的。因此，冒充 ACK 包能够被用来作为“隐秘扫描”的一部分。同样的，正确对待 UDP 数据包对于防火墙来说也是很困难的，因为它们不能说出它们是否是回答外部的询问，如果是就是合法的，否则它们就是来自外部的攻击。如果是诸如 FTP 协议的数据包，这个优势更为明显。

分布式防火墙最重要的优势在于能够保护那些不在拓扑结构范围里的主机，这些主机一般都是利用电信交换机(Telecommuter)来实现网络的切换，进行安全通信，当其在切换时，利用传统的方法就能够保护这种计算机。但是这需要利用切换通道的技术将 Internet 通道切换到相应的网络，同时负责通道的回收，然而当这种通道没有被建立时，它不能保护任何计算机。相比而言，分布式防火墙能够在所有的时间都保护计算机，并不用考虑是否建立了通道，它可以利用 IPSec 对所有数据包进行加密处理，从而能够对 Internet 上任何一个主机发送来的数据包进行鉴别。

4.2.2.2 分布式防火墙体系结构

从狭义上来讲，与传统防火墙产品相比，分布式防火墙产品是指那些驻留在网络中的计算机内部，如服务器和终端主机，并对整个网络系统自身提供安全防护的防火墙技术。它能够保护内部网络，防止来自外部网络的攻击，同时还能够保护内部网络自身不受来自内部网自身的攻击，提供给内部网络最强大的保护能力。

从广义来讲，分布式防火墙是一种新的防火墙体系结构，它融合了各种类型的传统防火墙技术，是新一代的防火墙技术，其结构图 4.3 所示。

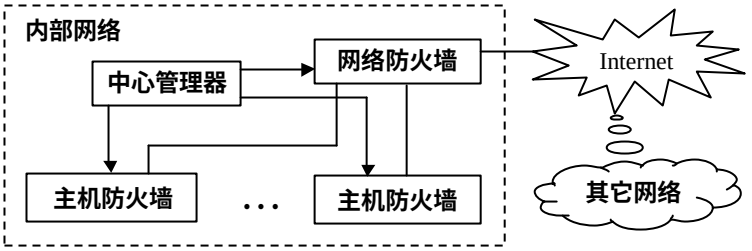


图 4.3　分布式防火墙结构图

在图 4.3 中，分布式防火墙由多种传统防火墙所组成，它们通过中心管理

部件来进行管理，实现分布式控制。分布式防火墙包含以下几部分：

·网络边界防火墙。用于内部网与外部网之间和内部子网之间的网络安全控制系统，也就是传统的网络边界防火墙。

·主机防火墙。主要是用来保护网络中终端主机的网络安全，这些主机的物理位置可能在内部网中，也可能在内部网以外，例如：托管服务器和移动办公的便携机。

·中心管理器。在分布式防火墙中，每个防火墙部件是根据防火墙本身安全性要求的不同而布置在网络中需要的位置上，但整个防火墙的安全规则又是统一制定和集中管理的，安全规则的分发及日志的集中管理都是中心管理器应具备的功能。中心管理器是分布式防火墙系统的核心和重要特征之一。

目前，还存在一些先进的防火墙技术，这些防火墙技术都是把传统防火墙技术与其他网络防护技术相结合而形成的。例如，将防火墙技术与网络入侵检测技术相结合，形成集成入侵检测的防火墙技术模型。

4.3 防火墙技术方案改进分析

现有分布式防火墙的技术方案在实际应用中是以传统 Internet 技术为基础的。随着 Internet 技术的发展和嵌入式 Internet 的成熟，分布式防火墙技术也需要进行相应的更新以适应 Internet 技术的发展。

4.3.1 远程节点

对于传统 Internet 来讲，使用分布式防火墙是一种最为安全灵活的安全防范手段。但在嵌入式 Internet 中，由于有很多终端主机的物理位置是分散分布的，因此网络中存在很多的远程节点。利用分布式防火墙技术实现对这些远程节点的保护不太现实，我们需要在分布式防火墙技术的基础上进行改进。通常是利用传统防火墙技术和分布式防火墙技术相结合，实现对远程主机进行保护，对远程主机进行身份合法性的验证。在这种混合应用中，一些主机在防火墙后面，然而其他一些远程主机则处于防火墙的外面。

4.3.2 虚拟网络技术（VPN）

虽然分布式防火墙技术不依赖于拓扑结构，但在分布式防火墙系统应用中，也需要利用拓扑结构技术来保护内部网络。一般来说，现有网络的拓扑结构存在较多的遏制点，利用分布式防火墙技术可以消除因为单个扼制点而引起的失效。但是，即使在分布式防火墙中利用拓扑结构技术也不能保护远程主机，因此需要将这些远程节点组合起来，形成一个安全的虚拟内部网络，这就需要使用虚拟网络（VPN）技术。当然，在应用虚拟网络技术的同时，IPSec 协议的应用也是必要的。

在 VPN 中，基于主机 IP 地址的身份认证的安全性较弱，很容易导致外部的地址哄骗攻击，因此，需要在 VPN 中采用基于主机名字的身份认证。另外，IPSec 协议正是用来保证内部主机到远程主机的通信安全，IPSec 协议提供反欺骗的保护，保护 VPN 中所有的主机，无论是本地的还是远程的。

4.3.3 特殊应用的通道

大多分布式防火墙技术在被应用时，由于拥有种类合适的路由，则三重路只能用于少数的一些协议中。例如，应用级代理的防火墙需要一些相应的协议，经由这些协议的数据包能通过由 IPSec 保护的通信信道路由到防火墙的内部，因此，只有在通过特殊应用的通道并经过防火墙的相应处理过后，数据包才能够传送到 Internet。

4.3.4 防火墙的硬件化

分布式防火墙中，终端主机中的防火墙有两种实现方法：基于软件的和基于硬件的。对于基于硬件的防火墙，例如 NIC，它拥有自己的处理器和内存等硬件资源，也有自己的操作系统和防火墙功能模块，不被终端主机用户所存取，能够满足应用的抗抵抗能力需求。在现实应用中，大多数终端主机的防火墙都是基于软件实现的防火墙，他们不能满足应用的抗抵抗能力（Tamper Resistance）需求，来阻挡网络内外有敌意的、无意识的用户运行恶意代码。但是，也可以将防火墙从主机操作系统上层拿开，将防火墙的实现放置于终端主机的底层，独立于终端主机的操作系统来实现。

当然，还可以通过不将防火墙在应用层实现来达到和防火墙硬件化同样的效果。在这种方式下，即便是拥有终端主机的 root/administrator 权限的用户，都不能更改防火墙在终端主机中的特定安全规则，同时也不能关闭防火墙，但终端主机用户在与中心管理规则不冲突的情况下，可以制定额外的安全规则。

4.4 嵌入式 Internet 防火墙技术模型

显然，在嵌入式 Internet 中有很多嵌入式终端系统和远程的终端设备，分布式防火墙技术已经不足以满足嵌入式 Internet 网络安全的需求。由于分布式防火墙技术具有其优越性，我们可以在此基础上进行改进，形成新的防火墙技术模型来保护嵌入式 Internet 的网络安全，我们称这种技术为：嵌入式 Internet 防火墙技术。

嵌入式 Internet 具有移动 IP、远程管理、终端主机物理位置分布不集中的特点。要对分布式防火墙进行修改，首先需要了解实现分布式防火墙所需要的三个组件：

·一个解决安全需求的规则描述语言。简单的来讲，分布式防火墙的安全规则等同于传统防火墙中的包过滤规则。另外，为了实现委托授权和认证的目的，规则描述语言及其解决机制也应该有可信任度的支持。

·一个分布式系统管理工具。如果可能，这个工具可以是基于 IPSec 的协议管理工具，或者是其他的一些协议管理工具。不管是通过协议管理工具，还是作为规则对象描述的一部分，都必须保证规则传输的完整性。

· 一个实现对进出的网络数据包和网络连接进行按规则过滤的工具。

根据前面的描述，在嵌入式 Internet 中，需要考虑原有防火墙所考虑的 PC 终端设备，还需要考虑嵌入式设备、移动嵌入式设备以及远程的嵌入式终端设备或 PC 终端设备，同时还要增加对 VPN 的支持。另外，对于不同终端设备

的保护还需要使用与终端设备相应的防火墙组件。因此，在对分布式防火墙技术基础上进行改进，可以得到图 4.4 所描述的嵌入式 Internet 防火墙结构模型：

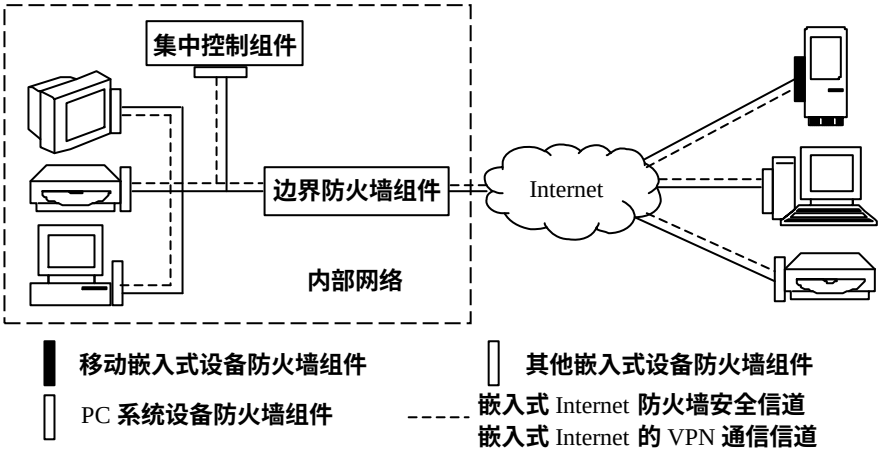


图 4.4　嵌入式 Internet 防火墙结构模型图

根据图 4.4 所示，从物理位置分布角度来看，嵌入式 Internet 防火墙可以分为两大部分：一部分是物理位置相对集中的内部网络，这些内部网络在网络的出口点处放置边界防火墙组件，通过边界防火墙来增强内部网的安全性；在嵌入式 Internet 防火墙系统结构中，这种内部网络可能会有很多个，因此，需要分布多个边界防火墙组件。另一部分是物理位置分散的外部系统设备网络，这些终端系统通过嵌入式 Internet 连接在一起，然后通过 Internet 与内部网络通信，并利用 VPN 技术构成嵌入式 Internet 的 VPN，当然，必需在这些终端设备上安装相应的防火墙组件。

嵌入式 Internet 防火墙的结构比较复杂，其主要由以下三部分组成：

·集中控制组件。集中控制组件主要是用来对整个嵌入式 Internet 防火墙进行管理控制。集中控制主要包括用户管理、安全规则控制、防火墙系统配置等功能。

·边界防火墙（Perimeter Firewall）组件。边界防火墙组件是用在网络拓扑结构的关键遏制点上，实现对内部网络的保护，增强内部网络的安全性。同时，还需要结合分布式系统结构，实现防火墙组件对 VPN 的支持。

·终端设备防火墙组件。根据网络终端设备的不同类型，终端设备防火墙组件可以分为三种：移动嵌入式设备防火墙组件，主要适用于移动通信设备，如：手机、PDA 等，这些移动通信设备都是嵌入式系统设备；其他嵌入式设备防火墙组件，主要是适用于类似信息家电、通信中继设备等嵌入式系统中，这些系统设备也是嵌入式系统；PC 系统设备防火墙组件，指的是在嵌入式 Internet 中，PC 系统作为嵌入式 Internet 的终端设备时，所必须使用的防火墙组件。

在嵌入式 Internet 中，很多终端设备都是处在远程，很容易被攻击者从物理位置上或通过其他方式截取、破坏甚至替换，以达到攻击内部网络的目的。

在嵌入式 Internet 防火墙中，如果使用上面的方法进行攻击，数据包没有通过终端设备防火墙组件的处理，就不能通过嵌入式 Internet 防火墙的集中认证，因此会被认为是不被信任的终端而被封锁。

4.5 嵌入式 Internet 防火墙技术模型结构分析

这部分主要就嵌入式 Internet 防火墙结构模型进行分析。嵌入式 Internet 防火墙由三种组件组成：集中管理组件、边界防火墙组件和终端设备防火墙组件，以下分别就各部分的结构进行分析。

对于终端设备防火墙组件，首先，需要实现对终端设备的网络安全保护功能，主要实现按规则过滤功能；其次，防火墙组件不但能够进行本机的管理，还需要实现防火墙组件的集中管理机制；另外，防火墙组件需要对用户进行认证，确定本机用户以及网络用户的身份；最后，嵌入式 Internet 防火墙要适应嵌入式 Internet 对 IPv6 的支持，以及 VPN 在嵌入式 Internet 中的应用。可以用图 4.5 来描述终端设备防火墙组件的结构。

<table>
<tr><td colspan="2">应用层管理模块</td></tr>
<tr><td>集中管理接口模块</td><td>内核过滤模块</td></tr>
<tr><td>VPN 支持模块</td><td>认证模块</td></tr>
<tr><td>IPv6 支持模块</td><td>与操作系统相关内核</td></tr>
<tr><td colspan="2">嵌入式 Internet 终端设备操作系统</td></tr>
</table>

图 4.5　终端设备防火墙组件结构图

对于边界防火墙组件，首先，需要实现对内部网络安全的保护功能，主要是进行按照控制规则进行过滤，以及对内部网的通信进行代理；其次，边界防火墙组件需要既能够进行自主的控制管理，还要实现防火墙组件的集中管理机制；另外，边界防火墙组件也同样需要对内部网以及外部网的用户进行身份认证；最后，边界防火墙组件还需要提供对 IPv6 和 VPN 的支持。因此，可以用图 4.6 来描述边界防火墙组件的结构。

<table>
<tr><td colspan="2">应用层管理模块</td></tr>
<tr><td colspan="2">应用层代理与过滤模块</td></tr>
<tr><td>集中管理接口模块</td><td>内核过滤模块</td></tr>
<tr><td>VPN 支持模块</td><td>认证模块</td></tr>
<tr><td>IPv6 支持模块</td><td>与操作系统相关内核</td></tr>
<tr><td colspan="2">边界防火墙硬件设备操作系统</td></tr>
</table>

图 4.6　边界防火墙组件的结构图

对于集中控制组件，首先，需要实现对嵌入式 Internet 各个组件的控制管理功能；其次，集中控制组件也同样需要对所有的嵌入式 Internet 防火墙用户进行身份认证；最后，集中控制组件也需要对 IPv6 和 VPN 的支持。因此，可以用图 4.7 来描述集中控制组件的结构。

<table>
<tr><td colspan="2">应用层控制功能、管理功能模块</td></tr>
<tr><td>集中管理接口模块</td><td rowspan="2">认证模块</td></tr>
<tr><td>VPN 支持模块</td></tr>
<tr><td>IPv6 支持模块</td><td>与操作系统相关内核</td></tr>
<tr><td colspan="2">集中管理硬件设备操作系统</td></tr>
</table>

图 4.7　集中控制组件的结构图

综上所述，可以看出嵌入式 Internet 防火墙中的每种组件之间都拥有较为相似的结构模型。因此，可以将这些共同点归纳出来，从结构上给嵌入式 Internet 防火墙中的所有组件划分层次，如图 4.8 所示。

<table>
<tr><td>集中管理层</td></tr>
<tr><td>身份认证层</td></tr>
<tr><td>加密层</td></tr>
<tr><td>安全传输层</td></tr>
</table>

图 4.8　防火墙组件的层次结构图

第五章　嵌入式 Internet 防火墙实现研究

在课题研究中，嵌入式 Internet 防火墙实现分为两部分：模拟嵌入式系统环境的实现研究和嵌入式 Internet 防火墙终端设备防火墙组件实现研究。

5.1 基于 PC 硬件的嵌入式系统设计

对于嵌入式 Internet，其硬件环境相对复杂，比较难搭建，研究使用的嵌入式设备仅利用普通的 PC 硬件设备来模拟构造；操作系统选择是利用对开放源码的 Linux 进行裁减修改，设计实现嵌入式 Linux 操作系统。从而实现了具有嵌入式系统性能的计算机系统，以此来模拟嵌入式系统，用于课题研究的嵌入式 Internet 环境的搭建。

5.1.1 硬件系统设计

嵌入式系统通常使用其特有的接口和设备，如 PCMCIA 接口、LCD 显示屏、触摸屏、DOC（Disk on Chip）、IBM 的 MICRODRIVE 等体积很小存储容量不太大的存储器。大多数嵌入式系统在系统启动后，内核和所有的应用程序全部都在内存或者 RAM 中。而 PC 上通常配有软盘、硬盘、CDROM 等各种存储设备和键盘、鼠标、显卡、显示器等输入输出设备，还有 ISA 、IDE、AGP、USB 等接口，这些设备在嵌入式系统中通常是无法使用，嵌入式系统也根本用不上这些设备。

嵌入式操作系统的存储体系都是由 Flash memory（或 ROM）和 RAM 两部分组成的，但由于没有足够的设备来烧录程序，课题设计的这个嵌入式系统里并不包含 ROM 这部分，而 Flash memory 的价格也比较昂贵，因此这部分的功能由其它部件来实现。ROM 主要是用来存储经常用到而不常变动的数据或程序，如初始化程序、引导程序、内核等。通常放在一张软盘上，每次需要启动操作系统的时候，可以直接用软盘来启动并装入系统。选用应用最广泛的 3.5 英寸软驱和相应的双面高密度软盘即可。一个大小为几百 K 的 Linux 系统内核正常运行时一般的需要 4M 左右的内存，嵌入式 Linux 系统所处理的应用程序都是小型的，功能也比较单一，因此用 8M 或 16M 的 DRAM 足够了。

我们设计的嵌入式系统主要是用来构建课题研究所使用的嵌入式 Internet 环境，网卡是不能缺少的，选用的是 NE2000 兼容系列 10M RJ-45 接口的 PCI 以太网卡。选用该网卡主要出于两方面考虑，一方面是该网卡比较兼容，Linux 系统本身就带有该种网卡的驱动程序，可以直接使用而不用自己编写，能节省大量时间；另一方面，这种网卡比较普遍，价格也较低，容易购买，并且性能颇佳，是一种性价比较高的配件。

嵌入式系统需要有比较好的人机交互界面，系统中的这部分当然更不能缺少。所以应选的输出设备是显示器，由于显示器本身不能直接接到主板上，不是即插即用的设备，则需要中间部件连接，该部件就是显卡。小影霸系列的显卡，性能是相当的优良，并且 Linux 也自带该类设备的驱动程序。显卡的显存

是一个比较敏感的问题，随着越来越多的嵌入式应用涉及到图像、多媒体等数据，因此，可以选用显卡时可考虑较大一点的显存。

主板的选用就没有什么特别的要求了，主板主要功能就是协调各个硬部件之间的工作，使各个部件的性能发挥到最佳。最重要的一点是该主板必须有良好的兼容性，能支持所选用的网卡、显卡和处理器。因此，要求主板至少有两个 PCI 插槽，赛扬 233 底座支架。这种主板是有二级总线的主板，对处理器的性能有一定的影响。通过各方面的权衡考虑，得出如下设计方案：

- CPU：塞扬 233MHZ
- 内存：8M 或 16M DRAM（PC100 或 PC133）
- 软驱：1.44M 三寸软区
- 主板：有串行接口（RS232）的 PCI 主板
- 显示器：14 寸的普通显示器
- 网卡：1 0M RJ-45 接口的 PCI 以太网卡（NE2000 兼容系列）
- 显卡：小影霸 TNT2 系列 M64（显存 8M）

这些硬件通过整合之后，就构成了一个模拟的嵌入式系统硬件系统。从表面上看，它仍然是一个 PC 硬件结构，但实际上它的结构功能已经和现有 PC 机相差甚远，比较接近嵌入式设备结构简单、功能单一、体积小和资源少等特点。该系统上电后，通过 BIOS 对硬件进行自检，初始化硬件设备，然后由软盘引导并启动系统，把整个系统内核都调入到 RAM，随后 root 文件系统也都统统进入 RAM。这样系统就启动成功，可以进行一些操作和开发。

虽然这个计算机系统和真正的嵌入式系统不太一样，但结构功能基本上都是一样的，基本上能够模拟嵌入式系统的性能，可以在此基础上进行实验研究。

5.1.2 嵌入式 Linux 操作系统的设计与实现

目前，市面上众多商业性嵌入式操作系统厂商都在努力地为自己争取嵌入式市场的份额。但它们价格昂贵、源码不公开，使得嵌入式应用软件的兼容性差，这种差的兼容性导致了商业嵌入式操作系统在对设备的支持方面存在较大的问题，也使得软件的移植变得比较困难。为了解决这个问题，许多人在考虑使用嵌入式 Linux 操作系统。这是由于 Linux 具有开放的源代码、精巧高效的内核、完整的网络功能和良好的可裁剪性，比较适合信息家电等嵌入式系统的应用，因此吸引了许多开发商的目光，成为嵌入式操作系统的一个发展方向。但也有另外的观点：因为嵌入式 Linux 在实时性、可靠性和可服务性方面有许多缺陷，只适用于实时性不强且系统资源较丰富的应用，或者作为研究对象。

Linux 本身不是一个实时系统，Linux 的内核并不提供对事件优先级的调度和抢占支持，但是可以利用 Linux 的特性给 Linux 增加实时调度的能力。实现方法是双内核系统，即利用 Linux 内核，同时增加一个实时内核，两个内核共同工作，获得别的实时系统所不能达到的优势。其实，双内核的解决方案在很早以前就已经提出。大概在 20 年前，贝尔实验室的开发人员就准备开发一种名为 MERT 的实时操作系统。这种操作系统拥有两个内核，实时内核和分时通用内核，实时内核用来运行实时任务，通用内核用来运行普通任务，这种

设计方法的优势就在于实时内核可以利用非实时 OS 内核的一些优势来开发。双内核的体系结构可以用下面的图来描述：

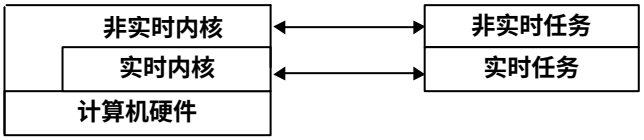


图 5.1　双内核结构图

Linux 内核对网络协议栈的设计是从简洁实用的角度出发，实现出一整套的网络协议模块。Linux 不仅可以支持一般用户需求的 Ftp、Telnet 和 rLogin 协议，还能提供对网络上其他机器内文件的访问（如 NFS，网络文件系统）。Linux 还可以支持 SLIP（Serial Line Interface Protocol）和 PLIP（Parallel Line Interface Protocol)协议,使得通过串口和并口线进行连接成为可能。通过 AX.25 协议，Linux 可以提供无线连接方式；通过在 Linux 上开发 Novell 标准的 IPX 协议，Linux 可以访问 Netware 网络；通过 Apache 公司开发的免费网络服务器，可以利用 Linux 系统作为强大的网络服务器，提供 Internet 上电子商务和数据提供服务。

下面将主要介绍 Linux 中的网络支持，还将对 Linux 中的网络设备驱动程序进行分析，以及对 Linux 网络设备支持进行解释。并提出如何利用 Linux 的网络协议栈来设计应用于嵌入式系统的网络协议栈以及相关设计思想。

5.1.2.1 Linux 网络概述

网络和 Linux 是密切相关的，Linux 所支持的 TCP/IP 协议栈是目前 Web 上最流行和通用的网络协议。Linux 的网络实现是以 4.3 BSD 为模型的，它支持 BSD sockets（及一些扩展）以及所有的 TCP/IP 网络。Linux 选用这个编程接口是因为它很流行，并且有助于应用程序从 Linux 平台移植到其它 Unix 平台。Linux 下的 TCP/IP 网络协议栈的各层之间是通过一系列互相连接层的软件来实现 Internet 地址族的，结构层次如下图所示：

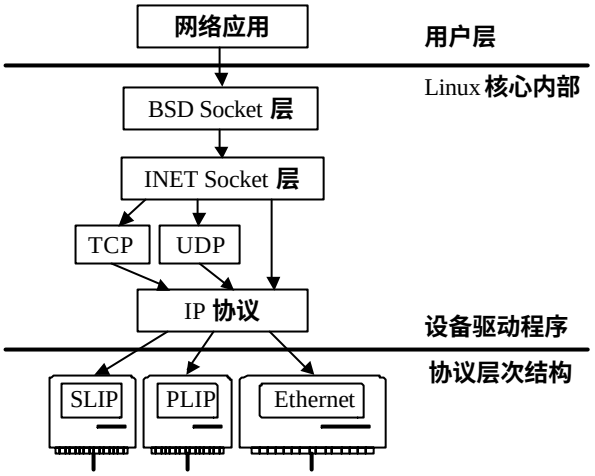


图 5.2　Linux 下的 TCP/IP 网络协议层次图

5.1.2.2 Linux 网络设备驱动程序的分析

为了使嵌入式 Linux 适应嵌入式 Internet 的需求，网络设备的处理是设计的重点之一，因此需要分析 Linux 网络设备的相关问题，即针对网卡设备驱动程序的分析，集中于对 Linux 与网卡相关部分的分析。

1、Linux 中网络驱动程序介绍

在 Linux 中，将所有的外围设备都归结为三类：字符设备、块设备和网络设备。Linux 对所有的网络设备进行抽象，并且定义了一个统一的概念——接口（Interface），所有的网络硬件的访问都是通过接口进行的。每一个已经驱动了的网络设备，都用一个 struct device 的数据结构表示。在内核启动或者驱动模块插入时，通过网络驱动程序，向系统注册检测到的网络设备。在进行网络数据传输的时候，网络驱动程序需要负责通过标准的接口将数据发送到相应的网络层，或者向网络发送数据包。

需要注意的是，网络接口并不是在 Linux 的文件系统中，而是在内核中的一个 struct device 数据结构中被定义。每一个 device 的数据结构都是在驱动的时候被创建的，而不像字符设备或者块设备那样，即使不存在物理设备的设备文件也还是存在的。

对于系统中的所有网络设备，都是通过一张网络接口管理表 dev_base 统一进行管理的。dev_base 是指向 struct device 的链表，在系统初始化完成之后，系统检测到的所有网络设备都将自动的保存在这张链表中，链表中的每一个字节都代表着一个系统检测到的网络设备。当系统需要发送数据的时候，网络子系统根据系统路由表选择相应的网络接口进行数据传输；而当接收到数据包时，通过驱动程序登记的中断服务程序进行数据的接收（纯软件实现的网络设备除外）。

2、重要的驱动过程

网络设备驱动的方法有两种：通过模块驱动和内核启动时自动检测。通过模块驱动的方法是 Linux 中使用模块设计的一种方案。Linux 的内核是将需要的功能编译在一起的，如果 Linux 内核增加一项功能，就将它的实现直接放在内核的代码中。为了让 Linux 的内核体积不至于庞大，采用编译成模块的方式。当需要用到这个模块时，用 shell 命令的 insmod 将该模块插入到内核运行空间；如果不需要了，用 rmmod 命令将该模块卸载。使用内核启动检测的方法有所不同。在系统启动的时候，内核把所有编译在内核支持的网卡设备都初始化在 dev_base 链表里，然后对每个节点都调用自己的 init 函数指针。如果该函数返回成功，那么该节点对应的设备保留；否则，该节点对应的设备不存在，将该节点删除。这样，在系统初始化的最后，在 dev_base 里面剩下的所有节点就全是系统检测到的网络设备了。

5.1.2.3 嵌入式 Linux 系统网络协议栈的实现

嵌入式系统网络协议栈的实现有两种途径，一种是复用 Linux 现成的网络代码，以获得对 TCP/IP 以及其它网络协议栈的支持；另一种途径是通过对

Linux 网络协议代码的分析，精简代码，将网络协议的代码作为操作系统服务的一部分。

1、重用 Linux 网络协议栈代码

在嵌入式 Linux 操作系统的设计中，重用 Linux 的网络协议栈源代码，很大程度上减少了工作量和开发成本。由于 Linux 协议栈已经相当成熟，在代码效率、兼容性和网络安全方面都很完善，如果嵌入式系统对内核的体积要求不是太高，那么使用 Linux 的裁减内核就成为可能，并且可以将网络部分代码内嵌重用。

2、重写网络协议栈

在微内核的基础上编写所需要的网络协议栈，可以根据需求，在实现需要功能的基础上尽量减少代码量，减少占用的空间，提高执行效率。这种方法应用于大部分嵌入式系统的定制开发过程中,其存在开发难度大、周期长的缺点。

5.1.3 嵌入式 Linux 操作系统的裁剪实现

嵌入式 Linux 系统需要下面三个基本元素：引导程序、Linux 微内核（由内存管理、进程管理和事务处理构成）和初始化进程。如果要让它有更多的功能且保持小型化，还可以加上文件系统、TCP/IP 网络支持、GUI（图形用户界面）和设计精简的应用程序，并将其放在 ROM、RAM 或 FLASH 中启动。由于嵌入式 Linux 操作系统的高度灵活性，开发者可以很容易地对它进行定制或作适当开发，来满足实际应用需要。

要裁减修改 Linux 为嵌入式 Linux，需要了解 Linux 系统启动的基本过程。所有的 x86 系统都是通过执行 ROM 中的代码加载启动盘的 0 柱面 0 扇区中的代码来启动整个系统的。在 Linux 系统中启动盘的 0 柱面 0 扇区中含有的是启动装载器 LILO，它定位、装载、最后执行内核。一旦内核装载后，接着试图加载并登陆磁盘中的 root 文件系统，装载完毕并登陆成功后，就会看到一行信息：

VFS: Mounted root (ext2 filesystem) readonly.

然后，系统发现 init 程序并执行它，init 程序寻找它的配置文件/etc/inittab，并开始执行其中的脚本，这些脚本是一些 Shell 命令的组合，用来执行加载所需模块、装载 SWAP、初始化网络、装载 fstab 中列出的所有驱动器等命令。最后启动一个叫 getty 的程序，它负责 console 和 ttys 之间的通信，它在显示器上打印 login 提示符并激活 login 程序，login 处理登陆的有效性并建立与用户的对话，至此启动过程完毕。也可以通过将 init 直接指向/bin/bash，使系统直接启动 Shell 解释程序。

5.1.3.1 选择内核

裁减修改制作嵌入式 Linux 启动盘，首先必须创建系统内核，常采用对内核进行手工配置，去掉不必要的模块的方法来裁剪内核。要想裁剪内核体积，创建它时就得把不必要的功能去掉，如去掉对不必要设备的支持。但是一定要记住保留内核对 RAMDISK 和 Ext2 的支持，否则启动盘将不能工作。其过程

如下：

以 root 超级用户登录，进入目录/usr/src/linux，执行#make menuconfig 来对内核模块进行配置。依次执行#make dep 和#make bzImage（内核不大则执行#make zImage），执行该命令后，将在/usr/src/linux/arch/i386/boot 目录下生成内核文件 bzImage（或 zImage）。若配置中加入了模块的支持，还需要执行#make modules 和#make modules_install。拷贝新内核到/boot 目录，修改文件/etc/lilo.conf，其中加入如下代码：

```
image = /boot/bzImage
label = new
root = /dev/hdxx    （自己的启动硬盘分区）
read-only
```

执行命令#lilo 载入新内核，用#reboot 重启系统，在出现 lilo:时键入 new。插入软盘执行#dd bs=1k if=/usr/src/linux/arch/i386/boot/bzImage of=/dev/fd0，这样将内核拷贝到软盘上。用命令#ls –s 将在/usr/src/linux/arch/i386/boot/bzImage 得到内核的大小，并将这个大小记录下来以被后面使用。

5.1.3.2 root 文件系统的制作

制作能应用于搭建嵌入式 Internet 环境的嵌入式 Linux 系统只需要下面几个基本元素：引导工具、Linux 微内核、硬件驱动程序、嵌入式 Web Server 和 Browser、文件系统和 TCP/IP 网络堆栈。

1、内核的裁剪

经过对 Linux 内核源代码的阅读，以及对实际嵌入式系统功能的分析，许多 PC 中的功能模块对于嵌入式系统来说都是多余的，但需要如下的功能模块：

- TCP/IP 支持模块。
- RAM DISK 支持模块。在内存中模拟硬盘，存放 Linux 运行所须的文件系统。
- 网卡的支持模块。我们选择 NE2000 PCI 网卡，把它直接编译到内核中，虽然会导致内核尺寸变大，但是能简化系统。

具体来说有两种方法：内核编辑和内核配置。内核编辑就是通过直接修改、增添或者删除内核源文件以及头文件来达到要求。内核配置是配置文件——Makefile 文件，可以用：make config、make xconfig、make menuconfig、make oldconfig 等。

首先以 root 的身份登录到系统中，然后执行以下命令：

1)、#cd /usr/src/linux　进入内核源文件所在目录，即编译内核的工作目录。

2)、#make mrproper　确保源代码目录下没有不正确的.o 文件（目标文件）以及文件的互相关联。

3)、#make menuconfig　对编译内核的选项进行详细配置。

4)、#make dep　正确设置所有文件依赖关系。

5)、#make clean　清除所有已存在的目标文件。

6)、#make bzImage　生成一个压缩比率最大的内核。

至此，内核制作完成。

2、Root 文件系统的制作

Linux 的运行除了内核，还需要一个基本的 Root 文件系统。Root 文件系统是 Linux 启动必须的部分，里面放置要用到的配置文件、应用程序、驱动程序和脚本文件等。具体的制作过程如下：

1)、#dd if=/dev/zerp of=rootfs bs=1k count =2500 生成一大小为 2500k 临时文件来存放根文件系统。If=后面的参数是要读入的文件，这里是系统中的零设备。Of=后面的参数是要写入的文件。bs 描述每次处理的数据块的大小，count 描述读写的次数。

2)、#losetup /dev/loop0 rootfs losetup 将该文件与一循环设备连接起来。循环设备使得我们可以用一个文件来模拟一个文件系统。

3)、#mke2fs －m 0 /dev/loop0 在此循环设备上制作 ext2 文件系统，同时用 －m 0 来保证为 root 用户不留任何的空间。

4)、#mount －t ext2 /dev/loop0 /mnt 将循环设备安装到/mnt 目录下。

5)、按照以下所述内容在/mnt 目录下建立相应的目录和文件，这里假设/mnt 为以后作成的文件系统的根目录，以下用“/”来表达。/bin，通常是放置系统的一些基本命令文件的目录；/etc，是放置系统配置信息的文件目录，同时也放置了一些系统初始化文件；/sbin，这个目录下放置一些系统扩展命令；/usr，目录下放置了一些用户经常需要使用的程序和一些配置数据；/lib，目录下放置了系统的一些库文件，很多命令或者程序运行是都要用到这些库文件；/dev，系统设备文件目录，按照标准设备定制；/var，主要放置系统参数和配置。

6)、#umount /mnt 卸载循环设备。

7)、#gzip －9 rootfs 用最大比例压缩制作完的文件系统。

3、整合过程

接下来把内核、根文件系统和引导程序集成到一张软盘上去。在整合之前要先检查 bzImage 和 rootfs.gz 两个文件的大小，如果超出了 1.44M，则需要对 rootfs.gz 作进一步的裁剪。接着要设置内核镜象文件中的根设备的位置，可以通过命令 rdev 来设置，rdev 是专门用来对内核的各个参数进行设置的工具。设置的命令如下：

・#rdev bzImage /dev/fd0 设置内核的根设备为软盘。

然后设置内核镜像文件 RAMDISK 的偏移量用来指出如何确定定位 root 文件系统，计算好以后用 rdev －r 来设置。假设对应于内核的偏移量是：xxxxx，运行命令如下：

・#rdev –r bzImage xxxxx

还要设置内核加载根文件系统的方式（只读还是读写）。在所设计的嵌入式 Linux 系统中，只有一次内核加载的过程，因为 RAMDISK 中的文件系统即为系统在之上运行的 root 文件系统，所以需要修改这个参数，用 rootflags 命令来实现：

・#rootflags bzImage 0

最后一步是放置内核和文件系统：

・#dd if=bzImage of=/dev/fd0 bs=1k 将内核写到软盘从第一扇区开始的连续空间里。

・#dd if=rootfs.gz of=/dev/fd0 bs=1k seek=内核数据块数（这里是 325） 将压缩后的 root 文件系统传输到软盘上紧接着内核的空间上，这里 dd 的参数 seek 表明了写指针的位置，也就是紧挨着内核的数据块的位置。

至此，整个系统制作完毕，用它就可以直接启动搭建的硬件平台，系统可以自动检测到网卡，登录系统以后，可以通过 ipconfig 命令来配置网络。比如 IP 地址、子网掩码等参数，然后就可以访问网络了。在文件系统中，放入了

telnet、ftp 和 ping 等网络工具，当配置好网络以后，就可以用这些工具来测试网络，并能够访问网络上的资源了。

5.1.4 基于嵌入式 Linux 的 WEB Server 和 Browser

要搭建课题研究的嵌入式 Internet 环境，也就需要基于裁减后得到的嵌入式 Linux 操作系统编写自己的嵌入式 Web Server 和 Browser。

5.1.4.1 嵌入式 Linux 中 WEB Server 的实现

目前在 Internet 上广泛应用的是 WWW 系统，这种系统用 HTML 文件格式传播信息，用统一资源定位符（URL）连接 Internet 的信息资源，按照 HTTP 协议在浏览器和 WWW 服务器之间通信，WWW 服务器又称为 Web 服务器。Internet 的底层通信协议是 TCP/IP 协议，而 TCP 协议则为通信的双方建立一条虚电路，保证所有的数据包都能按正确的次序到达目的地。TCP/IP 只是实现计算机之间的二进制数据传输，对这些数据如何解释，这是上层协议的事情。开发 Web 应用程序，必须了解的是其上层的 HTTP 协议。

1、HTTP 协议

HTTP 是一个应用层协议，目前广泛应用于 Web 浏览器和 Web 服务器之间的通信。HTTP 用字符串进行通信，所传送的信息称为 HTTP 消息（HTTP-Message）。HTTP 消息有两种类型：浏览器传送给服务器的请求消息和服务器传送给浏览器的响应消息。

2、HTTP 协议的作用原理

HTTP 协议工作的原理包括四个步骤：

1）、连接：Web 浏览器与 Web 服务器建立连接；

2）、请求：Web 浏览器通过 socket 向 Web 服务器提交请求；

3）、应答：Web 浏览器提交请求后，通过 HTTP 传送给 Web 服务器；

4）、关闭连接：当应答结束后，Web 浏览器与 Web 服务器必须断开。

3、实现 Web 服务器的程序设计

根据 HTTP 协议的作用原理，实现最基本的 GET 请求的 Web 服务器程序的方法如下：

1）、创建 Socket 对象，监听端口 80；

2）、等待、接受客户机连接到端口 80，得到与客户机连接的 Socket；

3）、从 Socket 中读取浏览器发过来的请求信息；

4）、从请求信息中获取请求类型；

5）、如果 HTML 文件存在，则打开 HTML 文件，把 HTTP 头信息和 HTML 文件内容通过 Socket 传回给 Web 服务器，然后关闭文件，否则发送错误信息给 Web 浏览器；

6）、关闭与相应 Web 浏览器连接的 Socket 字。

5.1.4.2 嵌入式 Linux 浏览器的实现

通过裁减修改实现嵌入式 Linux 系统的主要目的是为了搭建嵌入式 Internet 环境，另外，还需要一个基于嵌入式 Linux 的浏览器（Browser），来浏览 Internet 的资源。要设计实现一个浏览器，首先需要了解浏览器的主要功能：响应 Web 请求。Web 请求是由客户端发出的，当浏览器接到用户的命令后，根据用户给出的所要浏览的地址，再由域名解析系统把服务器名字解析为

具体 IP 地址，然后送给服务器一个连接请求。当服务器接收了这个请求时，连接也就建立了，然后开始数据传输。

一般来说，一个浏览器的服务请求包含如下几步操作：

1)、浏览其中的某个 URL（URL 由三部分组成：协议名字、存放页面的服务器名和服务器上存放的绝对路径）被使用者选中；

2)、浏览器使用 DNS 查询 URL 中域名所代表的 IP；

3)、DNS 发回一个回应，通知浏览器查询域名所代表的 IP；

4)、浏览器通过 TCP 协议连接到 Web 服务器的 80 端口；

5)、浏览器向 Web 服务器发送一个 GET HTTP：//localhost /index.html 请求；

6)、Web 服务器将 Index.html 发送回来；

7)、浏览器将 index.html 这个页面显示出来；

8)、浏览器结束这次 TCP 连接。

TCP 提供的是一种可靠的、复杂的、面向连接的服务，它通过三段式握手过程建立连接，用四个分组交换序列终止连接。

使用 C 语言编制一个小型的简单的文本浏览器，它能够把服务器上的文本文件传到客户端，并显示在屏幕上。该程序的实现步骤分为三步。第一步负责收集信息并使用这些信息填充 sockaddr_in 结构，它首先初始化该结构为全空，并将协议组设为 Internet 协议，随后进行服务器名称解析，然后查询 DNS 主机名，并把它的结果用函数 memcpy()复制到该结构，最后查询协议入口。第二步使用 socket()函数建立套接字。最后一步使用 connect()函数将套接字实际连接到远程端。此时，TCP 开始握手，两台计算机开始对话。程序通过编译完成后，得到浏览器 my_browser，使用方法如下：

./my_browser 服务器名 服务器绝对地之下的文本文件

例如：./my_browser localhost /index.html

5.2 嵌入式 Internet 防火墙技术研究

前面已经给出了嵌入式 Internet 防火墙的技术模型，并对模型进行了相应的分析。这一部分将对嵌入式 Internet 防火墙的实现技术进行分析，并讨论其实现方案。

5.2.1 嵌入式 Internet 防火墙组成结构分析

在嵌入式 Internet 防火墙中，主要由三部分组成：集中控制组件、边界防火墙组件和终端设备防火墙组件。以下分别就这三种防火墙组件作相关论述。

5.2.1.1 集中控制组件组成结构分析

集中控制组件主要是用来对整个嵌入式 Internet 防火墙进行管理控制的，是防火墙唯一的管理控制组件。集中控制主要包括用户管理、安全规则管理和防火墙系统配置等功能。因此，集中控制组件可以说是嵌入式 Internet 的核心，必须将集中控制组件放置在嵌入式 Internet 防火墙体系结构保护的内部网内，同时还需要在集中控制组件自身放置防火墙组件，以增加其安全性。

在嵌入式 Internet 防火墙中，集中控制组件本身是通过 PC 系统来实现的，因此可以从硬件和软件的角度对集中控制系统进行结构分析。首先，集中控制

组件所需的硬件设备是一个完整的 PC 硬件系统，配置相应的网络硬件设备。其次，对于集中控制组件的支撑操作系统，考虑到系统的安全性，一般来说大多使用 Linux，并选择对 IPv6 的支持。另外，为了提高集中控制组件的安全性，还需给整个集中控制系统配置 PC 终端系统防火墙。最后，为了使集中控制组件能够实现嵌入式 Internet 防火墙的集中管理，还需要在集中控制系统中添加安全规则描述语言和分布式系统管理工具。

5.2.1.2 边界防火墙组件组成结构分析

边界防火墙组件是作为加强嵌入式 Internet 内部网安全的一个组件，同时也是在嵌入式 Internet 中实现 VPN 的重要组成部分。为了增强嵌入式 Internet 内部网的安全，根据内部网络的拓扑结构，利用传统 Internet 防火墙技术，在内部网络中搭建边界防火墙，使内部网络得到类似于传统防火墙提供的保护功能，达到保护内部网络的效果。对于 VPN 技术在嵌入式 Internet 的使用，必须在嵌入式 Internet 中使用 VPN 网关，目前解决 VPN 网关需求的方案是在边界防火墙组件中集成 VPN 网关的功能，因此在嵌入式 Internet 防火墙系统结构中，边界防火墙组件同时被作为实现 VPN 技术的支持组件。

基于以上两个作用，边界防火墙组件一般都被放置在嵌入式 Internet 内部网的遏制点（Check Point）上。对于嵌入式 Internet 内部网来说，与 Internet 或其他网络的连接使其可能具有很多个遏制点，因此在嵌入式 Internet 防火墙体系结构中，边界防火墙组件的数量和嵌入式 Internet 的内部网具有的遏制点的数量一样。

边界防火墙组件和市场上现有具有 VPN 支持的边界防火墙相似。在嵌入式 Internet 防火墙系统结构中，边界防火墙组件还强调对 IPv6 的支持。边界防火墙组件一般采用的是硬件防火墙的技术方式，利用单独的 BOX 结构，自行设计硬件环境，使用相应的 Linux 操作系统，使防火墙达到最高的性能。

5.2.1.3 终端设备防火墙组件组成结构分析

终端设备防火墙组件是本论文研究的重点。在嵌入式 Internet 防火墙系统结构中，终端设备防火墙组件不但作为内部网的终端设备的网络安全保护工具，还是嵌入式 Internet 中远程终端设备网络安全的保护工具，同时也是这些远程终端设备通过嵌入式 Internet 安全、有效地访问内部网络资源的必备工具。

在嵌入式 Internet 中，终端设备分为两大类，一类是 PC 系统终端，另一类是嵌入式系统终端。从这一点来讲，嵌入式 Internet 与传统 Internet 之间的差别就是需要考虑嵌入式设备终端的问题；嵌入式 Internet 防火墙与分布式防火墙之间的差别就在于分布式防火墙只是考虑如何增强基于 PC 系统的内部网安全性能，而忽略了嵌入式系统终端及其它一些系统终端的网络安全，同时也很少考虑远程终端设备的网络安全问题，嵌入式 Internet 防火墙把这些因素都考虑进去，形成整体的网络安全体系结构和解决方案。

嵌入式系统终端也分为两大类：移动嵌入式系统终端和其他嵌入式系统终端。前者主要包括手机、PDA 等利用 WAP 等无线传输协议传输的嵌入式终端

系统，后者主要包括那些除了利用无线传输协议的移动嵌入式终端系统之外的嵌入式系统，如信息家电等等。

移动嵌入式系统终端中的防火墙组件，其硬件环境是移动嵌入式硬件系统，操作系统是嵌入式操作系统，对网络的支持是 WAP 协议栈。从这些终端设备物理位置的分布来看，有在嵌入式 Internet 内部网的，也有在远程的。对于内部网的移动终端设备，因为受到了边界防火墙组件的保护，这些终端设备的防火墙组件的安全规则考虑的问题范围集中于内部网络；而对于远程的系统终端，因为没有边界防火墙的安全保护，防火墙组件对自身的安全要求很严格，同时为了能和内部网络进行安全通信、资源共享以及交互控制，防火墙组件必须提供安全的身份认证机制。另外，由于现有支持移动嵌入式系统终端的嵌入式操作系统种类很多，在防火墙组件的实现上就相对比较复杂。但是，这些防火墙组件很容易实现层次和接口的统一。

其它嵌入式系统终端的防火墙组件，硬件环境是普通的嵌入式硬件系统，操作系统同样是嵌入式操作系统，对网络的支持是 TCP/IP 协议栈。从物理位置分布来讲，有在内部网的，也有远程的。内部网中的嵌入式终端设备，安全规则考虑的范围也就限于内部网络；而远程的嵌入式系统终端，安全要求很严格，同样需要防火墙组件拥有安全的相互认证机制。这种嵌入式系统终端的防火墙组件在不同的嵌入式操作系统上也能容易的实现层次和接口的统一。其它嵌入式系统终端的防火墙组件是本文研究的重点所在。

5.2.2 嵌入式 Internet 防火墙技术分析

下面就嵌入式 Internet 防火墙系统防火墙组件的实现技术进行详细分析。

5.2.2.1 集中控制组件技术分析

集中控制组件在嵌入式 Internet 防火墙技术模型中具有重要的作用，是用来实现防火墙系统规则配置和系统配置功能，主要包含以下几个部分：

1、规则（Policy）

嵌入式 Internet 防火墙的实现思想是：利用一种编译器将规则语言编译成防火墙内部的格式，集中管理组件通过防火墙中的分布式系统管理工具将这些规则文件分布到所有的防火墙组件上，这些防火墙组件对所有进出防火墙组件的网络数据包按照相应的规则进行对比判断和数据包发送者身份鉴别，从而决定对这些数据包如何处理。

目前，由于有很多种规则语言可以使用。其中包括类似 Firmato 配置语言和绝大多数商业防火墙上所使用的 GUI 通用规则语言，如：KeyNote 规则语言。而在实现嵌入式 Internet 防火墙系统的过程中，考虑到网络中的嵌入式系统本身的特点，我们选择使用 KeyNote 规则描述语言。

规则需要能够动态的被放入嵌入式 Internet 防火墙中的每个防火墙组件中，这些规则便是防火墙组件对网络数据包进行安全性判断的根据。传统防火墙也能够做到这一点，但是传统防火墙缺少了对规则相关请求上下文之间的了解，而嵌入式 Internet 防火墙终端系统可以通过相应的规则语言来了解规则相

关请求的上下文。

在嵌入式 Internet 中，规则分为两大类：一种是系统配置规则，另外一种是安全过滤规则。系统配置规则是对嵌入式 Internet 中的集中控制组件、边界防火墙组件和终端防火墙组件在嵌入式 Internet 防火墙系统中的系统权限、协同的描述，主要是包括嵌入式 Internet 防火墙各个组件的系统配置。管理者可以在集中控制组件中通过接口定义各种防火墙组件的系统配置规则，然后将这些系统配置规则分布到相应的防火墙组件中去，实现对防火墙组件的系统管理功能。

安全过滤规则是嵌入式 Internet 中的边界防火墙组件和终端防火墙组件在各自组件的硬件系统平台上需要实现功能的一种描述。这些防火墙组件按照集中控制组件提供的安全过滤规则进行网络数据包的安全过滤，用来实现防火墙组件的安全控制，完成嵌入式 Internet 防火墙系统的功能。在嵌入式 Internet 中，安全过滤规则的制定必需和嵌入式 Internet 中各种防火墙组件相对应。不同的防火墙组件需要有不同的过滤规则，因此，嵌入式 Internet 防火墙安全规则可以细分为三类：边界防火墙组件安全规则、PC 系统防火墙组件安全规则和嵌入式系统防火墙组件安全规则。

1)、边界防火墙组件安全规则。边界防火墙组件安全规则主要用于实现嵌入式 Internet 边界防火墙的过滤。

2)、PC 系统防火墙组件安全规则。从嵌入式 Internet 防火墙的体系结构来划分，PC 系统防火墙组件安全规则分为两种类型：一种是在嵌入式 Internet 内部网络中的 PC 系统防火墙组件的安全规则；另一种是分散在内部网络之外、通过 Internet 与内部网络通信的远程 PC 系统防火墙组件的安全规则。这两种规则在构成上存在不同的侧重点，对于内部网的 PC 系统防火墙组件，着重要求内部系统之间的融合，同时因为有了边界防火墙的保护，在安全级别上相对可以降低一些，而身份认证手段也比较丰富；对于远程 PC 系统防火墙组件，着重要求与嵌入式 Internet 内部网安全的通信、资源共享以及交互控制，这样就对远程 PC 系统防火墙组件提出相应的要求。另外，从 PC 系统的类型来看，目前存在的 Windows 系列和 Unix 系列两大类，对于这两大类的 PC 系统防火墙组件，虽然在组件的实现方面存在很大的差别，但从安全规则方面来讲，还是能够做到接口的统一。

3)、嵌入式系统防火墙组件安全规则。和 PC 系统防火墙组件一样，从嵌入式 Internet 防火墙的体系结构来划分，嵌入式系统防火墙组件安全规则也分为两种类型：一种是在嵌入式 Internet 防火墙系统内部网络中的嵌入式防火墙组件的安全规则；另一种是分散在内部网络之外、通过 Internet 与内部网络通信的远程嵌入式防火墙组件的安全规则。这两种规则也有其不同的侧重点，对于内部网的嵌入式防火墙组件，也同样着重要求与内部系统之间的融合，在安全级别上可以相对降低，身份认证手段也非常丰富；对于远程嵌入式防火墙组件，也着重要求与远程嵌入式 Internet 内部网安全的通信、资源共享以及交互

控制，这样就对远程嵌入式防火墙组件提出相应的要求。由于嵌入式设备系统种类很多，因此，从嵌入式设备系统类型角度来看，嵌入式防火墙组件安全规则又可以划分成：移动嵌入式防火墙安全规则和其它类型嵌入式防火墙安全规则。在嵌入式 Internet 中有使用 WAP 协议的移动嵌入式系统，移动嵌入式防火墙安全规则主要针对这类嵌入式系统设备的；而嵌入式 Internet 中的其它类型嵌入式系统设备则按照其它类型嵌入式防火墙安全规则来过滤。目前，存在很多种的嵌入式系统，每种嵌入式系统的操作系统也不尽相同，不同的嵌入式系统其防火墙组件的实现也不尽相同，但在对外的接口方面也是统一的。

2、规则管理（Policy Management）

嵌入式 Internet 防火墙系统中防火墙组件使用的安全规则的获得方式有两种。一种方式是：先由集中控制组件来制定，然后其他每个防火墙组件获得相应的安全规则来实现安全保护，称这种方式为集中管理（CM，Concentrative Management）；另一种方式是：每个防火墙组件本身提供一个安全规则管理接口，用户可以通过这个管理接口在自身的权限下对安全规则做相应的管理，称这种方式为自主管理（IM，Independent Management）。

在嵌入式 Internet 防火墙系统结构中，安全规则的管理非常重要，这决定了整个嵌入式 Internet 防火墙的安全性能。嵌入式 Internet 防火墙模型中，将各个防火墙组件的安全规则管理纳入集中管理机制，以确保整个防火墙的安全性能，因此，CM 机制的优先级是高于 IM 机制。

使用 CM 机制对嵌入式 Internet 防火墙组件进行管理，这是一个互动的管理过程。也就是说，在 CM 过程中，集中管理组件既是主动者，同时也是被动者。由此，从集中管理组件角度来看，嵌入式 Internet 防火墙中的防火墙组件安全规则的 CM 实现方式有两种：一种是集中管理组件主动的方式，称之为主动型（Initiative Type）；另一种是集中管理被动的方式，称之为被动型（Passive Type）。

1）、主动型（Initiative Type）实现方法

主动型实现方法的实现思想是：主动的通过相应的工具，将规则分布到所有的防火墙组件中去。

在嵌入式 Internet 防火墙中，集中控制组件主动的将规则分布到所有的防火墙组件的方法有两种：定时轮询更新（Timing Circuit Update）和修改时更新（Changing Update）。

定时轮询更新的基本思想是：在嵌入式 Internet 防火墙中，集中控制组件作为防火墙的中心组件，定时的对防火墙中在线的每个防火墙组件进行轮询，判断每个防火墙组件的类型，以及这些防火墙组件的版本是否是最新版本，如果不是最新版本，则主动要求防火墙组件进行规则更新。

修改时更新的基本思想是：首先在初始的时候，集中控制组件记录整个嵌入式 Internet 防火墙系统中的所有防火墙组件，并将它们归类。管理员在集中控制组件上对某类防火墙组件或某特定的防火墙组件的规则进行修改更新，当

修改更新操作完成后，集中控制组件可以和相应防火墙组件类的集合或者指定的防火墙组件进行通信交互，要求防火墙组件进行相应的规则更新操作。

我们可以通过图 5.3 和图 5.4 来描述这两种实现方法：

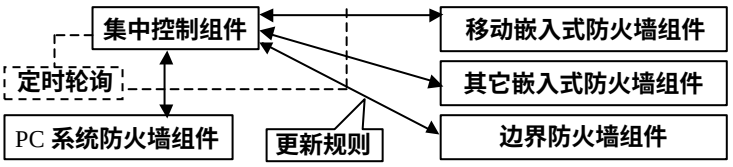


图 5.3　定时轮询规则更新法

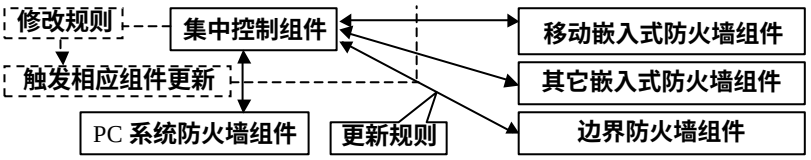


图 5.4　修改时规则更新法

2）、被动型（Passive Type）实现方法

被动型实现方法的实现思想是依靠嵌入式 Internet 中所有防火墙组件定时地通过网络向集中管理组件进行询问，集中控制组件判断相应防火墙组件中的规则版本，决定是否需要防火墙组件进行相应的规则更新操作。

在嵌入式 Internet 防火墙中，每个防火墙组件向集中控制组件进行规则更新查询需要达到一定的要求，即通过防火墙组件自身主动而集中控制组件被动的实现方法一定要使防火墙组件在规则更新方面尽可能和集中控制组件中规则更新保持一致，因此需要在规则更新的时间和方法方面做到最优。有关防火墙组件的规则更新机制将在后面探讨。图 5.5 描述了其实现方法：

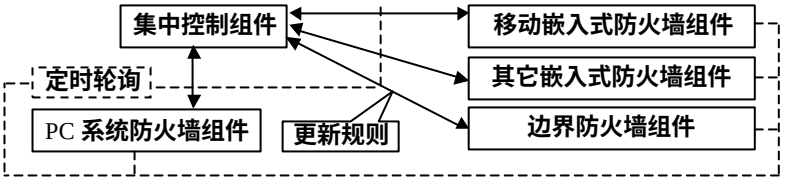


图 5.5　被动型实现方法图

3、用户管理

在嵌入式 Internet 防火墙功能结构中，用户管理包括：如何进行合法用户的权限划分、如何对用户进行身份的合法性验证以及如何授予合法用户相应的权限等。可以用图 5.6 来表示：

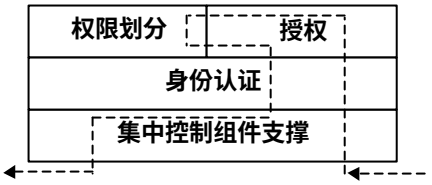


图 5.6　用户管理的过程图

当然，集中控制组件还需要一个安全加密层，当集中控制组件与其他防火墙组件进行通信和信息交互时，所传输的数据都需要通过安全加密层的加密处理。这样不但能够保证数据在网络中的非明文传输的安全性，同时还能实现一定数据的身份认证作用，大大增强了防火墙所保护网络的安全性。

另外，在含有集中控制组件的计算机系统中同样需要嵌入式 Internet 防火墙组件来进行保护，因此，集中控制组件需要和防火墙组件相互结合。

5.2.2.2 边界防火墙组件技术分析

边界防火墙组件是在继承传统的边界防火墙的功能上发展而来的。顾名思义，从边界防火墙所处的位置来看，是在 Internet 和 Intranet 之间，它的功能是保护内部网络的安全。实现嵌入式 Internet 防火墙中的边界防火墙组件，所涉及到的技术有多种：

1、包过滤（Packet Filter）

包过滤技术是边界防火墙组件按照规则对网络数据包传输进行控制的基本技术。网络上的数据被分割成为一定大小的数据包，每一个数据包中都会包含发送方的 IP 地址和接收方的 IP 地址、TCP/UDP 源端口和目标端口、TCP 链路状态等，同时，数据包还被防火墙组件中的安全加密层进行了加密，并添加了相应的补充信息。边界防火墙组件通过对数据包进行相应的处理，读取出数据包中相关的信息和规则进行比较，判断这些数据包是否是可信任数据包。一旦发现是不可信任的数据包，防火墙便会将这些数据包拒之门外。通过这种方法，边界防火墙组件在网络的关键遏制点处保证了内部网络的安全，当然，系统管理员也可以根据实际情况灵活制订边界防火墙组件的过滤规则。

2、代理服务（Proxy Service）

边界防火墙组件应用于网络结构关键的遏制点处，它起着一定的隔绝内部网络和外部网络通信的作用。为了确保内外网络之间的通信，边界防火墙组件中需要添加一定的代理功能，实现与传统代理服务器防火墙相似的处理功能。使边界防火墙组件具有较高的安全性，并能“阻隔”内外网络之间的数据包，实现监视和控制应用层网络数据包的作用。

3、状态检测（Stateful Inspection）

当然，传统边界防火墙的状态检测技术也能够应用到边界防火墙组件中，使边界防火墙组件可以对网络各个层次的数据包进行主动的、实时的监测，并对数据包进行分析、处理，能够有效地判断出在网络各层中的非法入侵。

4、混合防火墙技术

传统边界防火墙将包过滤和应用代理、状态检测的方法结合起来，便形成了混合防火墙技术。利用这种技术，传统边界防火墙具有了更为广泛的、有效的安全保护功能，因此，在嵌入式 Internet 防火墙系统中，也可以将这种技术应用到边界防火墙组件中去，增强网络的保护能力。

随着 VPN 技术在嵌入式 Internet 中的应用，构建嵌入式 Internet 的 VPN 的需求越来越多。通过使用嵌入式 Internet 防火墙，在嵌入式 Internet 防火墙

中添加 VPN 技术的支持，来满足这种应用需求。因此，需要在边界防火墙组件提供相应 VPN 功能支持，也就是说，边界防火墙组件担着实现 VPN 安全网关的功能。

嵌入式 Internet 防火墙系统中的数据都经过了相应安全加密层的处理，因此，边界防火墙组件也需要实现安全加密层，用来处理这些加密的数据。

另外，边界防火墙组件的规则管理，可以参考终端设备防火墙组件的规则管理，具体内容将在终端设备防火墙组件部分进行详细分析，这里就不再介绍了。

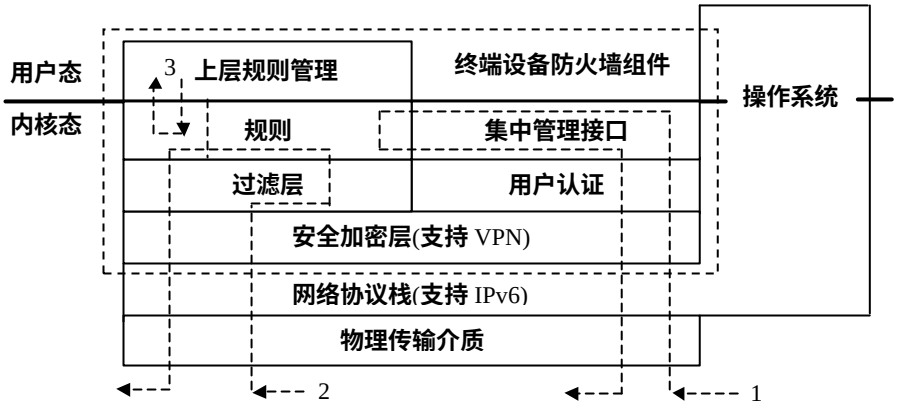
对于边界防火墙组件的硬件环境，为了提高边界防火墙组件的性能和效率，可以针对边界防火墙组件自身的特点，设计定制相应的硬件环境。同时，通过对开放源码的 Linux 进行配置修改，使其适应相应的硬件环境，作为边界防火墙组件运行的操作系统。

5.2.2.3 终端设备防火墙组件技术分析

在嵌入式 Internet 防火墙系统中，终端设备防火墙组件分为三部分：移动嵌入式设备防火墙组件、其他嵌入式设备防火墙组件和 PC 系统设备防火墙组件。本文将着重讨论终端设备防火墙组件的实现技术，终端设备防火墙组件的实现需要注意以下几点：

1、统一的层次模型

为了能够和集中控制组件进行交互以及能够通过 VPN 技术构成嵌入式 Internet 的 VPN，因此，所有的终端设备防火墙组件需要有统一的层次模型。嵌入式 Internet 防火墙组件的层次结构模型是以功能为基础的，主要包含：按规则的安全过滤、规则的获取、规则权限管理、身份认证、信息加密、VPN 的支持以及 IPv6 的支持。



1　防火墙组件进行集中管理的操作过程
2　防火墙组件进行按安全规则过滤操作过程
3　防火墙组件进行上层规则管理的操作过程

图 5.7　终端防火墙组件层次功能技术模型图

对于嵌入式 Internet 防火墙的终端设备防火墙组件来讲，存在不同的硬件

系统平台，如：PC 硬件系统和嵌入式硬件系统，同时也需要不同操作系统的支持，如：PC 操作系统和嵌入式操作系统。另外，由于终端设备系统在 Internet 中接入方式的不同，相应硬件设备支持和操作系统中所包含的网络协议栈就有所不同，例如，一般的 Internet 接入技术使用的是 TCP/IP 协议栈，而通过无线通信手段接入 Internet 使用的则是 WAP 协议栈。根据终端设备防火墙组件的统一的层次功能，可以归纳出如图 5.7 所示的技术模型。

2、安全加密层与 VPN 支持

安全加密层对嵌入式 Internet 防火墙系统中所有的信息进行安全加密，确保嵌入式 Internet 中所有的数据安全地在开放的 Internet 中传输，保证传输过程的安全。一般来讲，安全加密层可以采用 IPSec 技术来实现。而对于那些使用无线通信协议的移动嵌入式终端设备，由于其网络协议栈不是 TCP/IP，因此，安全加密层的实现就需要作相应的改进。为了使防火墙组件具有统一的层次模型和接口，并保持信息在嵌入式 Internet 中的一致性，安全加密层需要有跨协议栈的能力，做到协议栈之间的统一。安全加密层的实现需要考虑对 VPN 的支持。

3、身份认证

在嵌入式 Internet 防火墙体系结构中，防火墙组件分散分布在 Internet 各处，必须避免在嵌入式 Internet 中出现身份欺骗以及未授权用户的非法访问，需要对用户的身份进行认证；另外，数据的传输、集中控制组件和防火墙组件间的规则更新都需要对通信双方进行身份认证；同时，还需要对进出嵌入式 Internet 网络数据包的所有者进行身份认证，以确保整个网络通信、资源共享、交互控制的合法性和安全性。有关防火墙身份认证的相关技术，本文不做更多的研究。

4、规则权限管理及其管理流程

对于终端防火墙组件来说，要完成对嵌入式 Internet 终端设备的保护就需要终端设备防火墙组件能很好完成按规则的网络数据包过滤，因此防火墙组件需要对其规则实行非常严格的管理。规则分为两类，一类是安全过滤规则（Secure Policy），另一类是系统配置规则（System Policy）。另外，由于防火墙组件需要进行集中管理，所以，规则又可以分为集中规则（Centralize Policy，即 CPolicy）和自主规则（Independent Policy，即 IPolicy）。对于规则本身而言，由于存在多个管理者，因此，规则管理的协调和规则的一致性问题都是非常重要。

必须将属于集中控制组件管理的集中规则作为规则的首要部分，而防火墙组件自身能够管理的自主规则作为规则的次要部分。也就是说，在防火墙组件中 CPolicy 的优先级高于 IPolicy，用户不能够通过防火墙组件修改 CPolicy，而集中控制组件可以通过 CPolicy 来限制 IPolicy。

嵌入式 Internet 防火墙中，规则管理流程分为两部分：CPolicy 的集中管理流程和 IPolicy 的自主管理流程。IPolicy 的自主管理一般在防火墙组件内部

进行，其操作流程和一般操作一样，这里就不再详细分析。CPolicy 的集中管理流程是防火墙组件管理的重点，它是结合集中控制组件规则的主动型和被动型的实现方式来实现的。

嵌入式 Internet 防火墙系统中的防火墙组件规则的集中管理流程分为两部分：集中控制组件主动流程和集中控制组件被动流程。

集中控制组件主动流程过程是：集中控制组件对每个防火墙组件进行轮询，判断每个防火墙组件的类型以及这些防火墙组件的规则版本是否是最新版本，如果不是最新版本，则主动要求防火墙组件进行规则更新。同时，管理员在集中控制组件上对某类防火墙组件或某特定的防火墙组件的规则进行修改更新，当修改更新操作完成后，集中控制组件便和相应的防火墙组件类的集合或特定的防火墙组件进行通信交互，要求相应的防火墙组件进行规则更新，并及时执行防火墙组件规则更新操作。

集中控制组件被动流程过程是：防火墙组件在每次启动接入嵌入式 Internet 的时候，首先，需要和集中控制组件进行通信，判断自己现有规则是否是最新版本的规则，如果不是，则需要防火墙组件立即向集中控制组件进行规则更新操作。同时，为了确保规则更新的及时性和确定性，当防火墙组件在嵌入式 Internet 上时，所有的防火墙组件需要记录集中控制组件的轮询记录，当等待的时间和集中控制组件轮询间隔时间相等时，如果没有等到集中控制组件的轮询，则防火墙组件就主动向集中控制组件进行交互通信，查询其规则的版本，判断是否需要规则更新，依次重复以上操作，称此为超时申请更新。当然，为了增加规则的一致性，可以通过缩短集中控制组件的轮询间隔时间来实现。

综上所述，嵌入式 Internet 防火墙组件的规则管理流程如图 5.8 所示。

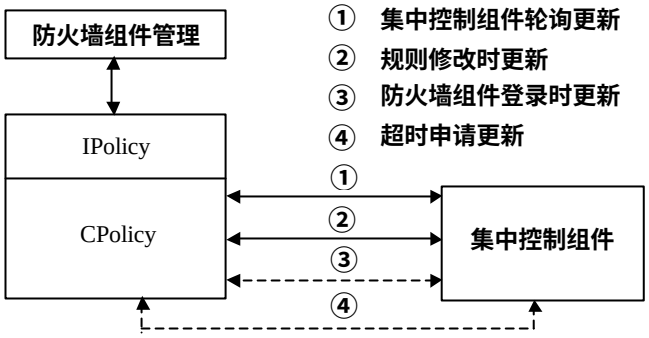


图 5.8　防火墙组件的规则管理流程图

5、按安全规则（Secure Policy）的过滤

在嵌入式 Internet 防火墙系统中，所有终端设备防火墙组件都必须具有保护终端设备系统的功能，终端设备防火墙组件主要是利用对通过终端设备网络数据包进行按规则过滤来实现的。因此，可将终端设备防火墙组件看成是一个利用包过滤技术的“主机”（包含 PC、移动终端和其它嵌入式终端设备）防火墙。

1)、数据包的构建

在嵌入式 Internet 中,一部分设备终端是通过 TCP/IP 与 Internet 建立连接,一部分是通过WAP。本文只针对基于TCP/IP 的终端设备防火墙组件进行讨论,而基于 WAP 的实现不做详细的分析,但其原理基本上和基于 TCP/IP 的一样。

TCP/IP 网络传输的数据包的构建过程是:每一个协议层都用特殊的连接对数据包进行“打包”,打包的过程是这样的,每一层把从上一层得到的信息作为它的数据来处理,并且在这个数据上加上自己的报头,报头包括与那层有关的协议信息,主要信息是:IP 源地址、IP 目标地址、协议类型、TCP 或 UDP 源端口、TCP 或 UDP 目标端口和 ICMP 消息类型等。然而,当数据从低层传输到高层时,这个过程正好相反,数据包的报头信息在每一个协议层被相应地剥去。

2)、包过滤(Packet Filter)的原理

网络数据包过滤是一种控制访问技术,用于控制进出网络的数据包。它被放置在网络的适当位置,对数据包进行判断,决定如何处理,判断的依据就是相应的安全规则。数据包过滤在这里是通过嵌入式 Internet 防火墙的终端设备防火墙组件来实现的。防火墙组件详细检查通过终端设备的网络数据包,根据防火墙组件的安全规则来决定它是否应该发送给终端设备上层。

终端设备防火墙组件主要负责对网络数据包进行处理,它对数据包的处理主要有两种选择:让数据包通过和丢弃这个数据包。

3)、数据包过滤技术的实现方式

终端设备防火墙组件的数据包过滤正是分析数据包中所含的重要报头信息和没有被反映在数据包报头中关于数据包的其他信息,如:嵌入式 Internet 防火墙的各个组件加密层的加密信息等,然后通过这些信息来进行判断处理。由此可见,数据包过滤主要有以下几种实现方式:

·基于地址的过滤。按地址过滤是一种最简单的数据包过滤技术。这种方法是基于数据包源和目标地址的数据包的限制技术,能允许指定的终端设备和本终端设备进行对话,反之则可以禁止指定的终端设备和本终端设备的不安全连接。由于嵌入式 Internet 中包含了较多的移动终端设备,这些移动的终端设备在嵌入式 Internet 中不可能一直保持相同的 IP 地址,因此,基于地址过滤的方法不适合于移动终端设备的应用。

·基于协议的过滤。基于协议的过滤是根据系统设计的原则来禁止或允许某种协议类型的数据包。

· 基于加密信息的过滤。在嵌入式 Internet 防火墙系统中,每个防火墙的组件都具有相应的安全加密层。当终端设备进行网络通信的时候,所有的信息都必须通过防火墙组件的安全加密层进行加密,因此,在安全加密层加密时会附带一些嵌入式 Internet 防火墙相关的附加信息,用来增强防火墙内设备之间通信的安全性。过滤时,可以根据这些附加信息来判断网络数据包的合法性。

· 基于 ICMP 消息类型的过滤。ICMP 用于 IP 状态和消息控制,ICMP 数

据包被包装在 IP 数据包的包体中，它没有源或目标端口，而是一套已定义好的消息类型代码。许多数据包过滤系统中都实现了基于 ICMP 消息类型的数据包过滤，而基于 IPv6 的 ICMP 协议能更好地做到这一点。ICMP 协议只是 TCP/IP 协议簇中的一个，但不能应用于 WAP 协议，因此，基于 ICMP 协议的过滤不能应用于移动终端设备防火墙组件。

4)、数据包过滤的操作

通常，在终端设备防火墙组件中数据包过滤的操作包含如下的一些过程：

- 设置防火墙组件中数据包过滤安全规则；
- 分析通过终端设备的网络数据报头；
- 如果一个规则不允许该数据包通过，此数据包被丢弃；
- 如果一个规则允许该数据包通过，此数据包被放行；
- 如果一个数据包不满足任何规则，此数据包被阻塞。

5.3 嵌入式 Internet 防火墙实现研究

在嵌入式 Internet 防火墙的实现过程中，集中控制组件里包含了规则描述语言、分布式工具等，实现起来相对复杂；边界防火墙组件实现技术和传统 Internet 边界防火墙组件的实现技术相似；终端设备防火墙组件的实现基本上分为两大类：基于 TCP/IP 的防火墙组件和基于 WAP 的防火墙组件，这两类防火墙组件的实现技术也基本一致。根据现有的技术基础和实验环境，本文主要集中基于 TCP/IP 的终端设备防火墙组件的实现研究，而关于基于 WAP 的防火墙组件的实现可以参考基于 TCP/IP 的防火墙组件的实现。

5.3.1 基于 Linux 和嵌入式 Linux 终端设备防火墙组件的实现

在 Linux 操作系统中，有免费的 Linux 防火墙。Linux 防火墙虽然只是操作系统的一小部分，但它已经拥有了商业防火墙的大部分功能和性能，并且经历了大量实际应用的检验，具有很大的实用价值。它可以实现如下的功能：网络地址翻译、负载均衡、包过滤、日志、流量统计和 VPN 等。其中，在 IP 层的包过滤功能是 Linux 防火墙的核心功能。 目前，ipchains 是 Linux 中成熟的防火墙软件，也是构筑 Linux 防火墙系统的关键部分。

5.3.1.1 ipchains 概述

为了便于在 Linux 的基础上实现终端设备防火墙组件，需要先从 Linux 网络层通信入手分析。Linux 网络层采用统一的缓冲区结构 skbuff，在底层从网络设备接收到数据包后，由系统分配一块内存，然后将数据整理成 skbuff 的结构存放在此内存中。实际上，在网络协议处理的时候，数据均是以 skbuff 的形式在各个层之间传递、处理。每一个单独的 skbuff 都被组织成双向链表的形式。其主要结构如下：

```
struct sk_buff
{
    struct  sk_buff   *next;    /*列表中的下一个缓冲区*/
    struct  sk_buff   *prev;    /*列表中的前一个缓冲区*/
```

```
struct sk_buff_head   *list;     /*当前工作列表*/
#if CONFIG_SKB_CHECK   /*调试时检测缓冲区*/
int magic_debug_cookies;
#endif
struct sk_buff   *link3;         /*用于 IP 协议层缓冲链的链接*/
struct sock   *sk;               /*使用的套接字*/
unsigned long   when;            /*用于计算 rtt 值*/
struct timeval   stamp;          /*分组到达时间*/
struct device   *dev;            /*接收分组的网络设备*/
……
}skbuff
```

Skbuff 的强大功能在于它提供了众多指针，可以快速定位协议头的位置，同时，也保留了许多数据包信息（如使用的网络设备等），以便协议层根据需要灵活应用。

网络硬件设备接受和发送比特流，IP 协议层调用一系列的底层函数对分组进行过滤，其中有三个关键函数：ip_rcv()、ip_forward()、ip_output()，用来分别处理 IP 层的接收、转发和发送工作。ipchains 防火墙的功能函数将在此三个函数中调用。用户通过 write()和 read()两个接口函数完成与上层的通信。

5.3.1.2 ipchains 防火墙的包过滤功能原理分析

从理论上来讲，ipchains 的包过滤是在 IP 层工作，但为了可以实现更多的功能，还必须利用其它层的信息。在 ipchains 中，使用命令对规则链进行维护，可以有效地定义包过滤规则。ipchains 本质上是包过滤器，它检查到达网络接口的 IP 包，根据事先定义好的安全过滤规则进行比较，然后再转发给其它接口。在 Linux 内核中有三条内置的规则链（input chain、forward chain、output chain），分别对应接收检测、转发检测和发送检测，数据在接收、转发或发送之前都需要通过相应规则链的检查。同时还有一条用户规则链，允许用户自己定义规则，起到对三条规则链进行扩展的作用。

运行 ipchains 的主机可以拥有许多网络接口，每个接口都连结在不同的网络上。当然，安装在 PC 及其上的防火墙只有一个接口，但对于过滤内外网信息的防火墙则至少应有两个独立的接口，一个连接到内部网络，另一个连接到外部网络。数据包从一个接口进入，经由过滤链（chains）传递给另一个接口。

每一条 chain 包含一系列过滤规则和链的缺省策略，它们主要包含如下处理操作：

ACCEPT（接受）：允许数据包通过防火墙。

REJECT（驳回）：丢弃数据包，并发送一个 ICMP 错误消息给该数据包的发送者。

DENY（拒绝）：直接丢弃数据包，不向发送者提供任何错误信息。

REDIRECT（转发）：不管数据包的目的地是哪里，转发到本地系统的制定端口。

RETURN（返回）：该策略支队用户定义规则有效，它直接返回调用链。

ipchains 防火墙的包过滤实现过程如下：

1）、当一个分组进来时，内核首先检查它是不是被篡改。主要是检查校验码，如果校验码不对，则此分组被拒绝。

2）、通过规范型检测来检验那些不规范的分组，通常写在链的前面，特别是 input 链。

3）、分组进入 input 链，由 input 链根据接收规则对其进行处理。

4）、对接收的分组进行判断，如果此分组时有原来的分组经过伪装后的回应，则对其进行伪装，然后送到 output 链处理。

5）、如果分组不是以前伪装过的分组则通过路由机制判断分组的目的地址是本机器还是需要转发给其他远程机器的。

6）、如果分组目的地是本机器，则直接送到 output 链。

7）、如果是需要转发的分组则进入 forward 链。Forward 链对那些需要转发的分组进行详细检查，通过了在进行转发。

8）、检查 output 链和分组的详细信息，符合则让它通过。

5.3.1.3 ipchains 防火墙的包过滤功能代码分析

ipchains 防火墙的包过滤部分的源文件有：ipfw.c、ipfw.h（IP 包过滤功能）、firewall.c、firewall.h（包过滤功能接口）。在 ip_fw.c 中，使用 ip_chain 数据结构来描述防火墙的规则链，它包含了指向链中的一条规则的指针和链的缺省策略。每个规则由 ip_fwkernel 数据结构（ip_fw.c）描述，其中很重要一项就是 ip_fw 数据结构（ip_fw.h）：

```
struct ip_fw
{
  struct in_addr   fw_src, fw_dst;        /*源地址和目的地址*/
  struct in_addr   fw_smsk, fw_dmsk;      /*源地址和目的地址掩码*/
  __u32    fw_mark;                       /*分组标记*/
  __u16    fw_proto;                      /*协议号*/
  __u16    fw_flg;                        /*flag 标志*/
  __u16    fw_invflg;                     /*反转标志*/
  __u16    fw_spts[2];                    /*源端口范围*/
  __u16    fw_dpts[2];                    /*目的端口范围*/
  __u16    fw_redirpt;                    /*重定向端口*/
  __u16    fw_outputsize;                 /*输出最大分组*/
  char    fw_vianame[IFNAMSIZ];           /*接口名*/
  __u8    fw_tosand, fw_tosxor;           /*分组优先级*/
};
```

其中包含了要匹配的所有信息（源、目的地址；源、目的端口；协议；flag 标志；接口名等），尽管时间信息并没有加进去，但可以通过不同时间更换不同规则的策略来处理与时间有关的规则。

下面结合 ipchains 防火墙的源代码来分析包过滤的全过程：

首先，由 ip_fw_init()完成包过滤部分的初始化工作。然后调用 IP 层的接收函数 ip_rcv()，通过检查长度、版本号来校验并判断接收的信息是否正确。如果没有错误就调用包过滤检测函数 call_in_firewall 进行规则匹配，返回值为匹配出来的规则。若规则先是同意接受该数据包，就查找路由 ip_route_input（不同意则将包丢弃）。此时路由信息已包含在了 skbuff 数据结构的 dst 项中，紧接着调用 skb->dst->input(skb)继续处理。对发往本地高层协议的包，则调用 ip_local_deliver()，对转往其它机器的数据包，则调用 ip_forward()。Io_forward()中包含了 call_fw_firewall()的调用。在发往底层的时候调用 ip_output，其中包含了 call_out_firewall()调用。在经伪装后的数据包回来时，其目的 IP 是防火墙组件的 IP，经路由后，也送入 ip_local_deliver()处理。在 ip_local_deliver()内部先解伪装，然后再查一次路由，发往本地的直接送往高层，否则依然调用 ip_forward()。

实际上，在 call_in_firewall、call_fw_firewall、call_out_firewall 的定义中可以发现，核心过程都集中在 ip_fw_check()这个函数中，它完成了数据包与规则的实际匹配，把每一个 IP 包与规则链中的每一条规则按照链表的组织顺序一一比较，通过返回规则的行动项，来决定对该数据包如何处理。

5.3.1.4 基于 Linux 和嵌入式 Linux 终端设备防火墙组件的实现研究

为了保证规则的安全性，规则采取本地存储，核心加密的形式以防用户对其进行修改，如果防火墙组件发现规则损坏，则触发定时规则更新的事件，向集中控制组件重新获取规则。如果获取规则失败，则暂时不对数据包进行过滤和加密，就不能添加合法身份信息到数据包，这样取消用户在嵌入式 Internet 防火墙系统中的合法身份。

Linux 内核中内置了包过滤功能模块，防火墙软件一般只是通过命令行对内核中的包过滤功能模块进行操作。由于 Linux 的不同内核对包过滤的支持方式不完全相同，因此，每推出一个新的内核版本，相应的包过滤防火墙软件也要进行升级。

Linux 下的包过滤软件很多。平常用的最广的是 ipchains、iptables 系列。这一系列最早是在 1994 年年底，从 BSD 的 ipfw 移植到 Linux 下的，它在不同内核版本下有不同的版本，分别是：ipfwadm（2.0 内核）、ipchains（2.2 内核）、iptables（2.4 内核），它们依次是前一个的直接升级版本。到目前为止，使用最广泛的是 ipchains（这主要是因为目前 Linux 的稳定内核版本是 2.2），一些重要的 Linux 发行版，如 RedHat、Turbo Linux 等都缺省安装了 iptables 本身应该是随 Linux2.4 内核一起工作的，由于目前 Linux2.4 内核还在紧张开发中，所以使用的人还不多，但实际上 iptables 在内核 2.3.15 之后就可以使用了。课题研究所使用的是基于 Linux2.2 内核的 ipchains。我们在基于 2.2 内核的 ipchains 的基础上进行开发，实现在 Linux 系统和嵌入式 Linux 系统平台上的终端设备防火墙组件。

另外，必须在 Linux 上建立安全加密层，安全加密层处于防火墙组件里的最下层，因此需要通过添加设备驱动程序的方式添加。

5.3.2 基于 Windows 和 WinCE 终端设备防火墙组件的实现

在 Windows 和 WinCE 系统中，都是使用 TCP/IP 协议栈，而且操作系统中的网络协议栈的实现也基本上是一样。其中，数据链路层是网卡驱动程序，负责和 Windows 进行沟通。Windows 使用 NDIS（Network Driver Interface Standard）和网卡的驱动程序进行通信。NDIS 是 Windows 的网络驱动程序接口标准，这就意味着从网卡驱动程序到协议驱动程序的数据都要利用 NDIS 这个规范来进行操作。

随着硬件设备的迅速发展，设计与开发设备驱动程序的需求也越来越大。设备驱动程序是硬件设备连接到计算机系统的软件接口，使应用程序以标准的方式访问硬件设备。设备驱动程序的优劣不仅影响到硬件设备的兼容性和硬件设备性能的发挥，还关系到系统工作的稳定性。驱动程序设计得不好会使硬件设备的性能大打折扣，使用户不能完全享受设备应有的优越性能。所以关于设备驱动程序的设计研究就显得尤其重要了。而基于 Windows 和 Win CE 终端设备防火墙组件的实现就是利用了 Windows 驱动程序技术。

5.3.2.1 Windows 平台设备驱动程序技术

WDM（Windows Driver Model）是微软提出的一种全新的设备驱动程序模型，它是在 Windows NT 内核驱动程序模型（Kernel-mode Device Driver Model）的基础上发展而来的，增加了对即插即用（PnP），电源管理（Power management），Windows 管理接口（WMI）等新的硬件标准的支持。然而 WDM 最重要的改进是提供了一系列驱动程序类（class drivers）。一个驱动程序类完成某一功能领域的工作，这样的驱动程序类包括 USB 类，1394 类，流设备类和 HID 类等等。

从内在组成结构来看，WDM 设备驱动程序是一种栈式结构，完成不同功能的驱动程序位于栈中不同层。从驱动程序设计者的角度看，WDM 设备驱动程序是按一定规则组织起来的一组函数集。下面具体阐述这两种结构。

1、WDM 设备驱动程序的栈式结构

WDM 设备驱动程序通常由一个驱动程序栈构成，栈中一系列驱动程序通过依次调用下层驱动程序来完成各自的功能。例如，USB 端口驱动程序就是由以下各层驱动程序组成，在栈底是 USB hub 驱动程序用以驱动 USB 控制芯片，其上是 USB 小驱动程序（Mini-Driver）和 USB 类驱动程序提供对特定的 USB 设备类（如：USB 接口的数字相机）的控制，上层则是厂家开发的针对厂家某种具体型号 USB 设备的驱动程序。

针对栈中不同层的设备驱动程序，WDM 中定义了不同的设备对象（Device Object）。设备对象就是一些用以控制实际物理设备的数据结构。栈中最底层的总线驱动程序对应的设备对象叫作物理设备对象（PDO）。栈中间完成设备功能的驱动程序（Function Driver）对应的设备对象叫作功能设备对象（FDO）。

其他各层驱动程序对应的设备对象叫作过滤器（Filters）。

当安装硬件设备时，操作系统中即插即用管理部件按照设备驱动程序的要求构建设备对象栈，首先是最底层的总线驱动程序检测到实际的物理设备，创建设备 PDO。创建了 PDO 后，即插即用管理部件就查询注册表，根据注册表信息依次加载过滤器（Filter）和功能驱动程序（Function Driver），生成相应的设备对象并建立设备对象与驱动程序的对应关系。这样一个完整的设备驱动程序栈就建成了。

设备驱动程序栈中的各层驱动程序完成各自不同的功能，功能驱动程序处理设备的 I/O 请求包（IRP），如读写请求包和配置请求包；总线驱动程序管理设备与系统的交互，负责具体的信号传输；而过滤器则监督或修改 IRP 流，做一些额外的处理动作。

2、WDM 设备驱动程序的函数集结构

WDM 设备驱动程序可以看成是按一定规则组织起来的一个函数集，其中的函数被操作系统所调用，以响应应用程序的 I/O 请求。针对硬件设备功能要求的不同，驱动程序实现的函数集也有所不同。一些函数，如 DriverEntry 和 AddDevice 函数以及基本的 I/O 请求处理函数是每个 WDM 设备驱动程序都应实现的，而其他函数则由设计者根据需要选择实现。譬如，如果驱动程序要将 IRP 排队，那么必须实现函数 StarIO；而要支持 DMA 传输驱动程序就必须包含函数 AdapterControl。

5.3.2.2 基于 Windows 和 WinCE 终端设备防火墙组件的实现研究

基于 Windows 和 Win CE 终端设备防火墙组件的实现是利用了 Windows 平台的驱动程序技术。Windows 平台的驱动程序实现的主要思想是：首先实现拦截通过网络的数据包，然后对拦截到的数据包按照定义的安全规则进行对比，并且检查该数据包发送者的身份是否合法，通过安全规则的对比和身份的鉴别，用以决定是否允许被拦截的数据包通过，并且决定是否需要日志记录和报警。另外，该驱动程序还需负责于集中控制组件之间的交互，实现防火墙组件的规则更新。因此，驱动程序的实现是此类终端设备防火墙组件实现的关键。本文针对终端设备防火墙组件的实现研究正是以 Windows 平台 WDM 驱动程序技术为基础的。

同许多 Win32 应用程序一样，WDM 驱动程序是 PE 格式的，但是它没有 WinMain 或 Main 这样的入口。一个 WDM 驱动程序的标准入口是函数 DeviceEntry。该函数主要完成初始化，设置 IRP 处理入口和生成相关设备对象的工作。在 DriverEntry 中登记的 IRP 处理函数中，处理基本 I/O 请求的函数和处理设备打开与关闭的函数在实现上与 NT 下基本相同，在这里就不再赘述。因此，DriverEntry 函数的实现是终端设备防火墙组件实现中驱动程序的主要函数。

利用 Windows 平台的 WDM 驱动程序技术，笔者实现了基于 Windows 和 WinCE 终端设备防火墙组件，其中，Windows 平台的 WDM 驱动程序的结构

框架程序部分源代码参考附录。最后，终端设备防火墙组件应用程序的主界面如下图所示：

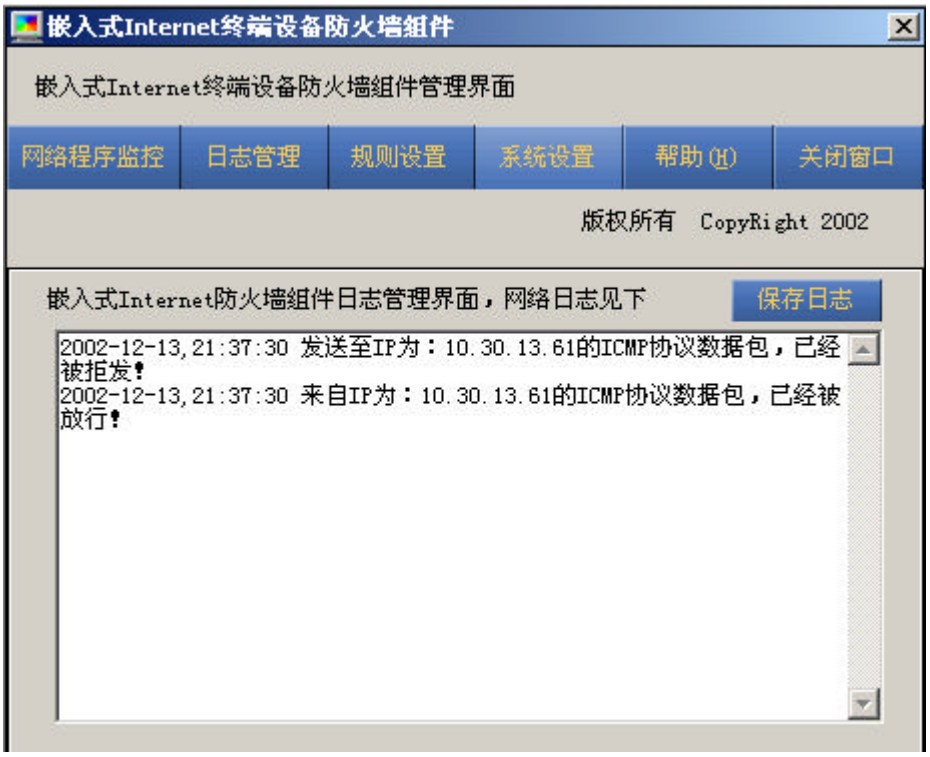


图 5.9　**终端防火墙组件应用程序的主界面**

第六章 结 论

Internet 技术的飞速发展，使得 Internet 的安全性也面临着更多、更大的威胁。随着嵌入式 Internet 技术广泛应用，接入 Internet 的嵌入式系统设备也越来越多，这为 Internet 带来了更多的安全隐患。与此同时，网络攻击技术与攻击力度的提高，对 Internet 的网络安全技术提出了更高的要求，嵌入式 Internet 防火墙技术的出现恰好在一定程度上解决了这些问题。

嵌入式 Internet 防火墙技术解决了嵌入式系统设备接入 Internet 所带来的安全问题，它能适应嵌入式系统设备资源较少的限制，还能够应用于保护开放的远程终端设备的安全，可以实现对嵌入式 Internet 乃至扩展 Internet 的安全保护。嵌入式 Internet 防火墙技术是一种新的技术，它是在分布式防火墙技术基础上进行修改更新，以适应嵌入式 Internet 需要而得来的。目前，分布式防火墙技术已经有了比较成熟的应用，但对于嵌入式 Internet 防火墙技术，国内外还基本上没有相关的研究见报。

作者详细分析了嵌入式 Internet 技术的研究现状，建立了嵌入式 Internet 以及扩展 Internet 模型；对嵌入式 Internet 的网络安全问题进行了研究，并在分布式防火墙技术上进行改进更新,形成了嵌入式 Internet 防火墙的体系结构；针对嵌入式 Internet 防火墙的结构模型，分析了相应的实现技术，并研究了其终端设备防火墙组件的实现技术。

本文的主要研究工作和结论概述如下：

1、通过分析嵌入式技术和 Internet 技术的发展，研究了嵌入式系统的 Internet 接入技术，提出了嵌入式 Internet 和扩展 Internet 模型，并在此基础上研究了课题实现所采用的嵌入式 Internet 环境及相关技术。

2、在分析研究传统 Internet 存在的网络安全威胁的基础上，分析了嵌入式 Internet 和扩展 Internet 的网络安全性问题，并对嵌入式 Internet 和扩展 Internet 的特点及其存在的网络安全威胁进行了研究。

3、分析了传统防火墙技术在现有 Internet 应用中存在的缺点，讨论了分布式防火墙技术为嵌入式 Internet 提供安全保护的可行性，通过对分布式防火墙进行改进，提出了嵌入式 Internet 防火墙技术模型。

4、讨论了嵌入式 Internet 防火墙的体系结构，建立了防火墙各个组件相应的结构模型，并对此进行了深入地研究。

5、针对课题所使用的嵌入式 Internet 环境，研究了基于 PC 硬件的模拟嵌入式 Linux 系统的实现。详细分析了嵌入式 Internet 防火墙各个组件的实现技术，着重研究了嵌入式 Internet 防火墙的终端设备防火墙组件的实现技术，并介绍了部分研究成果。

在论文结束之际，总结过去研究工作的同时，作者对嵌入式 Internet 防火墙技术的相关研究提出以下几点建议和设想：

1、嵌入式 Internet 研究的基础是基于 IPv6 技术的，但由于课题研究环境的限制，IPv6 技术的应用考虑得比较少。因此，在以后课题的研究过程中，需要更多的考虑 IPv6 技术，以 IPv6 技术作为防火墙组件实现的基础。

2、在嵌入式 Internet 防火墙体系结构中，安全加密层和身份认证是其重要的组成部分，它们决定了在开放的嵌入式 Internet 中谁才是可信任的。随着网络技术的发展，需要更为安全的信息加密和身份认证系统，因此，在未来的研究中，这两部分应该作为考虑的重点之一。

3、规则是嵌入式 Internet 防火墙实现安全功能的重要依据，所有的防火墙组件功能都依赖于规则，规则的表达、传递和保护也是嵌入式 Internet 防火墙中需要着重考虑的。因此，规则语言和分布式系统管理工具也就成了防火墙实现的重要部分，可以选择自主开发或使用成熟的规则描述语言和分布式系统管理工具也是防火墙实现的重要课题。

4、随着无线移动通信的广泛使用，在嵌入式 Internet 中，基于无线传输协议的移动终端设备也越来越多。由于移动终端设备种类繁多，所使用的操作系统也各不相同，因此，研究开发统一的移动终端设备防火墙组件也是课题后续研究需要关注的问题。

致　谢

本文的全部研究工作是在付宏教授的亲切关怀和悉心指导下完成的。导师渊博的知识、敏捷的思维、严谨扎实的科研作风以及求实的治学态度给我留下了深刻的印象。她严格要求自己的学生，督促我们认真地对待学习。从她那里，我不仅学到了学术上的知识，而且提高了我独立工作及发现问题、解决问题的能力。在结稿之际，向她表示最诚挚的敬意和深深的谢意。

在论文的研究过程中，得到了刘衍珩教授、李雄飞教授、赵永哲副教授、高强副教授的帮助和指点，在此向他们表示深深的谢意。我的师弟吴鸿韬在我的课题研究过程中都给予我积极的帮助与关怀，98 级本科生张兆勇、杨向广同学参与了课题的实验环境搭建，马天刚、郭威、肖传伟、吴相豪、程耀东、刘伟、陈鹏同学在我的日常生活和课题研究中也给与我很大的帮助，在此一并谢过。在作者的成长过程中还得到了苑森淼教授的教诲和鼓励，老师、朋友们的热心帮助使得本文得以顺利进行。

这里我要特别感谢我的母亲，母亲将她一生的心血都给了我，我无法用任何言语表达对她的谢意。我的父亲、大舅母以及其他的亲朋好友在我生活和学习中的大力支持和信任表示衷心的感谢。

在这几年学习生活中，我的女朋友钮志华在我最困难的时候给予我大力地支持、鼓励和帮助，使得课题研究得以顺利的进行，在此表示深深的谢意。

最后向所有曾经给予我鼓励和帮助在此未能提及的前辈、同学和参考文献的作者表示感谢。

参考文献

[1] 赵海. 嵌入式 Internet. 北京：清华大学出版社，2001

[2] 沙斐等. IPv6 详解. 北京：机械工业出版社，2000

[3] 宋书民等. 防火墙技术指南. 北京：机械工业出版社，2000

[4] 探矽工作室，嵌入式系统开发圣经，北京：中国青年出版社，2002

[5] 林晓东. 网络安全关键技术研究. 北京邮电大学博士研究生毕业论文，1999

[6] 邹思轶. 嵌入式 Linux 设计与应用. 北京：清华大学出版社，2002

[7] http://www.computer.org/internet/. IEEE Computer Society

[8] http://www.emware.com/. emWare, Inc.

[9] http://www.mi2g.com/. Mi2g limited

[10] http://www-900.ibm.com/cn/. IBM, Inc.

[11] Robert E. Filman，Embedded Internet Systems Come Home, IEEE Internet Computing, 2001.5(1): 52-53

[12] Janne Riihijärvi, Petri Mä hö nen, and Mika J. Saaranen, Providing Network Connectivity for Small Appliances: A Functionally Minimized Embedded Web Server, IEEE Communications Magazine. 2001.39(10): 74-79

[13] Steven M. Bellovin. Distributed Firewalls. login, 1999, November: 39-47

[14] Wei Li. Distributed Firewall. December 5th, 2000. URL: http://www.cs. helsinki.fi/u/ asokan/distsec/documents/li.ps.gz

[15] Steven M. Bellovin. Computer Security - An End State? Communications of the ACM, 2001, March, 44(3): 131-132.

[16] Bill Gilvino. ZiLOG and the Embedded Internet – a white paper. http://Microcontroller. com, September 2000

[17] Rick Drohan. Embedded Internet Provides a New Business Opportunity – eServices. Embedded Internet Conference 2000

[18] David Kline. The Embedded Internet. http://www.wired.com/wired/archive /4.10/es_ emb-edded_pr.html

[19] Sotiris Ioannidis, Steven M. Bellovin. Building a Secure Web Browser. Usenix Conference, June 2001.

[20] Peter M. Gleitz, Steven M. Bellovin. Transient Addressing for Related Processes: Improved Firewalling by Using IPV6 and Multiple Addresses per Host. Proceedings of the Eleventh Usenix Security Symposium, August 2001.

[21] Sotiris Ioannidis, Angelos D. Keromytis, Steve Bellovin, and Jonathan M. Smith, Implementing a Distributed Firewall. in Proceedings, ACM Conference on Computer and Communications Security, Athens, GREECE (November 2000):190-199

[22] John Ioannidis, Steven M. Bellovin. Implementing Pushback: Router-Based Defense Against DDoS Attacks. NDSS, February 2002.

[23] NEWPORT Electronics, Inc. Embedded Internet: iSeries Meter and Controllers - Direct Connection to Ethernet. http://www.newportus.com /i/iEmbed.htm
[24] Thomas Herbert. Embedding TCP/IP. http://www.embedded.com/internet/
[25] Jonathan M. Smith, Kenneth L. Calvert, Sandra L. Murphy, Hilarie K. Orman, Larry L. Peterson. Activating Networks: A Progress Report. IEEE Computer, 1999, April 32(4): 32-41
[26] Angelos D. Keromytis, Sotiris Ioannidis, Michael B. Greenwald, Jonathan M. Smith. Scalable Security Mechanisms for the Internet. 2001, January. MS-CIS-01-05, CIS Department, University of Pennsylvania
[27] D. Scott Alexander, Paul B. Menage, Angelos D. Keromytis, William A. Arbaugh, Kostas G. Anagnostakis, Jonathan M. Smith. The Price of Safety in an Active Network. Journal of Communications and Networks, 2001, March 3(1): 5-18.
[28] K. G. Anagnostakis, S. Ioannidis, S. Miltchev, J. Ioannidis, M. Greenwald, J. M. Smith. Efficient Packet Monitoring for Network Management. in Proceedings, IEEE NOMS, Florence, IT, 2002, April:423-436
[29] Matt Blaze, John Ioannidis, Angelos D. Keromytis. Trust Management for Ipsec. In Proceedings of the Internet Society Symposium on Network and Distributed Systems Security (SNDSS) 2001:139-151
[30] Matt Blaze, Joan Feigenbaum, John Ioannidis, and Angelos D. Keromytis. The KeyNote Trust-Management System, Version 2. Request For Comments (RFC) 2704, September 1999
[31] Steven M. Bellovin. A Technique for Counting NATted Hosts. Proc. Second Internet Measurement Workshop, November 2002
[32] ZiLOG Inc. Embedded Internet Solutions with Extreme Connectivity from ZiLOG. http://www.microcontroller.com/
[33] McCombie, B.,Journal. Embedded Web servers now and in the future. Real-Time Magazine, Real-Time, 1998, Jan.-March (1): 82-83
[34] Sotiris Ioannidis, Steven M. Bellovin, and Jonathan M. Smith. Sub-Operating Systems: A New Approach to Application Security. in 10th SIGOPS European Workshop, September, 2002.
[35] Bellvin S M. Security problems in the TCP/IP protocol suite. Computer Communication Review, 1998, 9(2):32-48
[36] Deborah Radcliff. Feature: Firewalls Reach Out. March 26th, 2001. URL: http://www. nwfusion.com/net.worker/news/2001/0326firewalls.html
[37] 钟卓新. “9.11”之后的美国信息安全措施. 计算机安全，2002, 9
[38] 袁兵，付宏，吴鸿韬. 嵌入式 Internet 与扩展 Internet 研究. 2002 通信技术新展望——第八届全国青年通信学术会议论文集:503-507
[39] 陈其松，谢晓尧. IPv4 到 IPv6 的过渡策略及其测试. 贵州工业大学学报（自然科学版），2001, 2:68-70
[40] 李春霞，孙魁明. IPv4 向 IPv6 迁移的过渡策略. 北京师范大学学报（自

然科学版），2001, 8, Vol.37(4):465-470
[41] 吕京建，肖海桥. 面向 21 世纪的嵌入式系统. 半导体技术，2001，26(1):1-3
[42] 庄言春,何熙文. 基于 Internet 的嵌入式模糊控制技术. 微处理机,2001，5(2):18-20
[43] 安继芳. 嵌入式微处理器因特网组网技术（EMIT）概览. 半导体技术，2000，9，Vol.25(4):7-9
[44] 王常杰，秦浩，王育民. 基于 IPv6 的防火墙设计. 计算机学报，2001，2，24(2):219-223
[45] 刘克龙，蒙杨，卿斯汉. 一种新型的防火墙系统. 计算机学报，2000，3，23(3):231-236
[46] 朱树人，李伟琴. 防火墙 HTTP 代理用户认证的实现技术. 计算机工程与应用，2001，6:29-31
[47] http://www.netmedia.com. NetMedia Inc.
[48] 张根源. 嵌入式系统与 Internet 技术. 微计算机信息，2000，3:19-20
[49] 胡欣杰. 嵌入式系统应用研究. 微计算机信息，1999，6:4-5
[50] 栗大超等. 嵌入式系统的 Internet 互连技术. 微计算机信息，2000，6:4-5,11
[51] 鲍可进. 基于工业 PC 的嵌入式测试系统. 微计算机信息，1999，4:71-72
[52] 吕华峰等. Internet 防火墙及其 Linux 实现. 计算机工程与应用，2001，8:41-44,103
[53] 罗宇等. 一种嵌入式操作系统设计. 计算机工程与应用，1999，2:51-53
[54] 陈定君等. 嵌入式软件仿真开发系统的设计与实现. 计算机工程与应用，1999，4:59-62
[55] 李晔. 嵌入式系统监测服务器的实现. 微计算机应用，2001，4:209-212
[56] 李刚等. 关于 Java 技术在嵌入式系统中的应用研究. 小型微型计算机系统，2001，9:1138-1140
[57] 李志飞等. WAP 在 TCP/IP 网络上的协议模拟方法. 小型微型计算机系统，2001，6:679-682
[58] 李宗伯等. 在嵌入式 Java 芯片中使用即时编译技术. 计算机研究与发展，2001，3:375-379
[59] 郭晓东等. 嵌入式系统虚拟开发环境的设计与实现. 计算机研究与发展，2000，4:413-417
[60] 陈科等. 网络入侵检测系统和防火墙集成的框架模型. 计算机工程与科学，2001，2:26-28,36
[61] 陈栃立等. VPN 系统体系结构及基于安全网关的实现. 通信技术，2001，2:42-44,60
[62] 谢庆杰等. IPSec 策略管理机制. 计算机工程，2001，8:109-111
[63] 杨向荣等. 入侵检测技术研究与系统设计. 计算机工程与应用，2001，

16:1-4
[64] 刘克龙. 一种新型的防火墙系统. 计算机学报，2000，3:231-236
[65] 刘玉莎等. 嵌入式防火墙系统的实现. 计算机工程与应用，2000，7:135-138
[66] 曾重等. Linux 环境下动态防火墙技术的研究及实现. 计算机应用研究，2001，5:49-51
[67] 赵欣等. 网上 IP 劫持攻击的研究，软件学报，2000，11:515-519
[68] 张少波等. 分布式防火墙中 FTP 透明代理的研究与实现. 计算机应用研究，2001，7:68-70
[69] 袁曙等. 使用 Linux 构筑复合型防火墙. 计算机应用研究，2001，7:71-72

作者在攻读硕士学位期间发表的相关论文

1. 袁兵，付宏，吴鸿韬. 嵌入式 Internet 和扩展 Internet 研究. 第八届全国青年通信学术论文集，2002.11.1-4. 被评为会议优秀论文，并收录入北京邮电大学学报 2003 年增刊.
2. 付宏，袁兵. 嵌入式 Internet 应用研究. 计算机工程与应用，已录用，待发表.

附　录

下面将列举实现 Windows 和 Win CE 终端设备防火墙组件中 WDM 设备驱动程序的程序框架实现的部分源代码。如下：

```
#pragma hdrstop
#pragma NDIS_INIT_FUNCTION(DriverEntry)
//
//入口函数，这里用来完成缓冲区初始化、HOOK 和设备的创建
//
NTSTATUS DriverEntry(
    IN  PDRIVER_OBJECT      DriverObject,
    IN  PUNICODE_STRING     RegistryPath
    )
{
    NTSTATUS status=STATUS_SUCCESS;
    if(!InitBuffer())
        return NDIS_STATUS_FAILURE;   //  申请、初始化网络数据包缓冲区

    //HOOK
    if(GetNdisModuleAddress() && m_NdisBaseAddress != NULL)
    {
        …….      //实现截取网络数据包，代码略。这部分为驱动程序功能实现的关键。
    } //HOOK

    DriverObject->MajorFunction[IRP_MJ_CREATE]                  = YBPacketOpen;
    DriverObject->MajorFunction[IRP_MJ_CLOSE]                   = YBPacketClose;
    DriverObject->MajorFunction[IRP_MJ_READ]                    = YBPacketRead;
    DriverObject->MajorFunction[IRP_MJ_WRITE]                   = YBPacketWrite;
    DriverObject->MajorFunction[IRP_MJ_CLEANUP]                 = YBPacketCleanup;
    DriverObject->MajorFunction[IRP_MJ_DEVICE_CONTROL]   =  YBPacketIoControl;
    DriverObject->DriverUnload = YBPacketUnload;

    YBPacketCreate(DriverObject, RegistryPath);   //  创建实现 DeviceIoControl 的设备

    return status;
}

VOID YBPacketUnload(
    IN PDRIVER_OBJECT           DriverObject
)
{
    PDEVICE_OBJECT      DeviceObject;
    PDEVICE_OBJECT      OldDeviceObject;

    if(m_pNdisSend != NULL)
    {
        ……      //释放对网络数据包的截取，代码略。
    }  // UnHook  Ndis Function

    DeviceObject = DriverObject->DeviceObject;
    while (DeviceObject != NULL)
    {
        OldDeviceObject=DeviceObject;
```

```
            DeviceObject=DeviceObject->NextDevice;
            YBPacketDelete(OldDeviceObject);
        }
}

NTSTATUS YBPacketOpen(          //对应驱动程序 IRP_MJ_CREATE 消息的实现函数
    IN PDEVICE_OBJECT DeviceObject,
    IN PIRP Irp
)
{
     return 0;
}

NTSTATUS YBPacketClose(         //对应驱动程序 IRP_MJ_CLOSE 消息的实现函数
    IN PDEVICE_OBJECT DeviceObject,
    IN PIRP Irp
)
{
     return 0;
}

NTSTATUS YBPacketCleanup(       //对应驱动程序 IRP_MJ_CLEANUP 消息的实现函数
    IN PDEVICE_OBJECT DeviceObject,
    IN PIRP Irp
)
{
     return 0;
}

NTSTATUS YBPacketRead(          //对应驱动程序 IRP_MJ_READ 消息的实现函数
    IN PDEVICE_OBJECT DeviceObject,
    IN PIRP Irp
)
{
     return 0;
}

NTSTATUS YBPacketWrite(         //对应驱动程序 IRP_MJ_WRITE 消息的实现函数
    IN PDEVICE_OBJECT DeviceObject,
    IN PIRP Irp
)
{
     return 0;
}

NTSTATUS YBPacketCreate(
     IN   PDRIVER_OBJECT        DriverObject,
     IN   PUNICODE_STRING       RegistryPath
)
{

     NTSTATUS                       status = STATUS_SUCCESS;
     PXPACKET_DEVICE_EXTENSION      pDeviceExtension      = NULL;
     PDEVICE_OBJECT                 pFilterDeviceObject   =NULL;
     UNICODE_STRING                 usTempName;
     UNICODE_STRING                 SymbolicLinkName;
```

```
    RtlInitUnicodeString(&usTempName, XPACKET _DEVICE_NAME);
    status = IoCreateDevice(
                IN  DriverObject,
                IN  sizeof(XPACKET_DEVICE_EXTENSION),
                IN  &usTempName,
                IN  FILE_DEVICE_XPACKET,
                IN  0,
                IN  FALSE,
                OUT     &pFilterDeviceObject
                );

    RtlInitUnicodeString(&SymbolicLinkName, XPACKET _DOS_DEVICE_NAME);

    status = IoCreateSymbolicLink(
                    &SymbolicLinkName,
                    &usTempName);

    if (!NT_SUCCESS(status))
    {
        IoDeleteDevice(pFilterDeviceObject);
        return( status );
    }
    return status;
}

VOID YBPacketDelete(        // 删除一个设备
    IN  PDEVICE_OBJECT          pDeviceObject
)
{
    PXPACKET_DEVICE_EXTENSION pDeviceExtension;
    UNICODE_STRING SymbolicLinkName;

    pDeviceExtension = (PXPACKET_DEVICE_EXTENSION)pDeviceObject->DeviceExtension;

    RtlInitUnicodeString(&SymbolicLinkName, XPACKET_XFILTER_DOS_DEVICE_NAME);
    IoDeleteSymbolicLink( &SymbolicLinkName );

    pDeviceExtension->ulNodeType = 0;
    pDeviceExtension->ulNodeSize = 0;

    IoDeleteDevice( pDeviceObject );

    DebugPrint("Device was deleted!");

}

NTSTATUS
YBPacketIoControl(
    IN  PDEVICE_OBJECT DeviceObject,
    IN  PIRP  Irp
)
{
    PIO_STACK_LOCATION IrpStack;
    IOCTLPARAMS IoControl;

    DebugPrint("<==ybPacketIoControl...\n");

    IrpStack = IoGetCurrentIrpStackLocation(Irp);
```

```
    IoMarkIrpPending(Irp);
    Irp->IoStatus.Status = STATUS_PENDING;

     IoControl.dioc_IOCtlCode = IrpStack->Parameters.DeviceIoControl.IoControlCode;
     IoControl.dioc_cbInBuf = IrpStack->Parameters.DeviceIoControl.InputBufferLength;
     IoControl.dioc_cbOutBuf = IrpStack->Parameters.DeviceIoControl.OutputBufferLength;
     IoControl.dioc_InBuf = Irp->AssociatedIrp.SystemBuffer;
     IoControl.dioc_OutBuf = Irp->AssociatedIrp.SystemBuffer;

     ManageIoControl(&IoControl);

     Irp->IoStatus.Information = IoControl.dioc_cbOutBuf;
    Irp->IoStatus.Status = STATUS_SUCCESS;
     IoCompleteRequest(Irp, IO_NO_INCREMENT);

     return STATUS_SUCCESS;
}

// DeviceIoControl 处理函数
DWORD ManageIoControl(PIOCTLPARAMS pVtoolsD)
{

     PVOID pVoid;
     DWORD* pOutBuffer = (DWORD*)pVtoolsD->dioc_OutBuf;
     DebugPrint("IoControl Process!");

     switch(pVtoolsD->dioc_IOCtlCode)
     {
          case IOCTL_YBPACKET_CREATE_RULE_BUFFER:
               {
                    // 申请安全规则的内存空间
                    pVoid = CreateMemory(*(DWORD*)pVtoolsD->dioc_InBuf);
                    pVtoolsD->dioc_cbOutBuf = 0;
               }break;
         case ……
          default:
               DebugPrint("Other Command!");
               break;
     }

     return NDIS_STATUS_SUCCESS;
}
…….        //程序的其他源代码略
```

摘　要

随着硬件技术和 Internet 技术的发展，越来越多的嵌入式系统设备融入 Internet，形成了新的 Internet 模型：嵌入式 Internet 和扩展 Internet 模型，它们的出现也给 Internet 带了更多的安全问题。本文主要研究解决这些安全问题的防火墙技术。

嵌入式系统的 Internet 应用主要涉及到：嵌入式系统的 Internet 接入技术、嵌入式 Web 服务器、嵌入式网络协议栈和嵌入式 Internet 开发工具。本文在分析嵌入式 Internet 相关技术的基础上，提出了嵌入式 Internet 和扩展 Internet 模型。嵌入式 Internet 起源于把嵌入式系统与 Internet 结合起来的这种想法，在嵌入式系统应用领域中，以 Internet 技术为基础，使嵌入式系统与 Internet 相互连接，实现嵌入式系统与 Internet 之间的资源共享、信息通信和状态控制等功能，这种嵌入式系统与 Internet 之间的连接与应用就称为嵌入式 Internet，其模型如图 1 所示。扩展 Internet 模型主要特点是：它的网络主机和终端包含了现在和未来可以接入 Internet 的计算机和非计算机设备，另外，还包含诸如嵌入式网络的通信子网，其模型如图 2 所示。

图 1　嵌入式 Internet 模型图

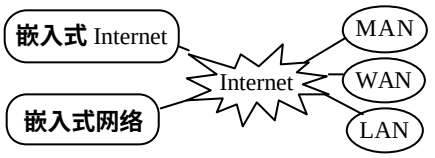


图 2　扩展 Internet 模型图

在分析传统 Internet 所存在的网络安全威胁的基础上，针对嵌入式 Internet 和扩展 Internet 的特点，分析了二者潜在的网络安全隐患，探讨了分布式防火墙在解决嵌入式 Internet 安全方面的可能性，并在分布式防火墙技术基础上进行修改更新，提出了新的嵌入式 Internet 防火墙技术模型，其结构模型如图 3 所示：

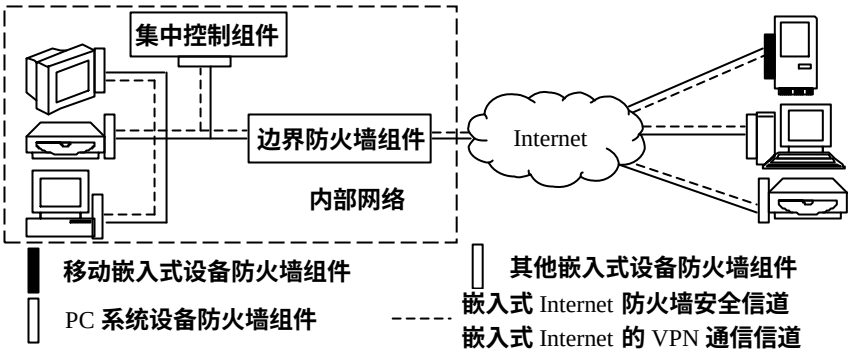


图 3　嵌入式 Internet 防火墙结构模型图

从结构上来看，嵌入式 Internet 防火墙可以分为两大部分：一部分是物理位置相对集中的内部网络，这在网络的出口点处放置边界防火墙组件，用来增

强内部网的安全性。另一部分是物理位置分散的外部设备系统网络，这些终端系统通过 Internet 连接在一起，然后通过 Internet 与内部网络通信。

嵌入式 Internet 防火墙主要由：终端设备防火墙组件、边界防火墙组件和集中控制组件组成。终端设备防火墙组件首先需要实现对终端设备的安全保护功能，主要实现按规则的过滤；其次，还要能够进行本机的管理和集中管理机制；另外，需要对用户进行认证，确定本机和其它用户的身份；最后，需要提供对 IPv6 和 VPN 的支持。对于边界防火墙组件，首先需要实现对内部网络的安全保护功能，主要是按规则的过滤和应用代理；其次，需要实现自主的控制管理和集中管理机制；另外，需要对内、外网络的用户进行身份认证；最后，也需要提供对 IPv6 和 VPN 的支持。而集中控制组件首先需要实现对嵌入式 Internet 各个组件的控制管理功能；其次，需要对所有嵌入式 Internet 防火墙的用户进行身份认证；最后，也需要提供对 IPv6 和 VPN 的支持。

在嵌入式 Internet 防火墙中，每种组件都拥有相似的结构模型，可将其分为四层：集中管理层、身份认证层、安全加密层和安全传输层。

本文利用普通的、便宜的 PC 硬件设备来构造模拟嵌入式硬件系统，并通过对开放源码的 Linux 进行裁减修改，构建嵌入式 Linux 操作系统。两者相结合，得到比较接近嵌入式系统性能特点的模拟嵌入式计算机系统，来搭建嵌入式 Internet 环境，用于课题的研究。

通过对嵌入式 Internet 防火墙各组件的组成结构和实现技术分析可知，集中控制组件在嵌入式 Internet 防火墙中具有重要的作用，是用来实现防火墙安全规则和系统配置管理功能，主要包括规则制订、规则管理和用户管理。边界防火墙组件的技术实现分为包过滤、代理服务和复合型防火墙技术三种。终端设备防火墙组件分为移动嵌入式设备防火墙组件、其他嵌入式设备防火墙组件和 PC 系统设备防火墙组件三部分；移动嵌入式设备防火墙组件主要适用于移动通信设备，如手机、PDA 等；其他嵌入式设备防火墙组件主要是适用于类似信息家电、通信中继设备等嵌入式系统中；PC 系统设备防火墙组件是指在嵌入式 Internet 中，PC 终端系统所使用的防火墙组件。

集中控制组件的实现相对复杂，需要规则描述语言、分布式工具等；边界防火墙组件实现技术和传统 Internet 边界防火墙组件的实现技术相似；终端设备防火墙组件的实现分为两类：基于 TCP/IP 和基于 WAP 的防火墙组件，这两类防火墙组件的实现技术原理基本一致。根据现有的技术基础和实验环境，本文着重讨论基于 TCP/IP 的终端设备防火墙组件的实现技术，而基于 WAP 的实现可以参考基于 TCP/IP 的实现技术。

终端设备防火墙组件的实现包括：统一的层次模型、安全加密层与 VPN 的支持、身份认证、规则权限管理及其管理流程和和按规则的过滤。本文的实现分为两部分：基于 Linux 和嵌入式 Linux、基于 Windows 和 WinCE。基于 Linux 和嵌入式 Linux 防火墙组件的实现主要是通过修改 Linux 操作系统中的 ipchains 防火墙来完成的。而基于 Windows 和 WinCE 防火墙组件的实现则采

用了 Windows 平台 WDM 设备驱动程序技术。

本文通过探讨 Internet 技术的发展，建立新型的 Internet 模型，分析新的 Internet 模型所存在的安全威胁，提出一种能够解决这种网络模型安全的新型防火墙技术，并研究了其具体结构和相应的实现技术。希望能为未来的防火墙技术的发展提供理论和技术依据。

关键词：嵌入式系统　嵌入式 Internet　分布式防火墙　嵌入式 Internet 防火墙

ABSTRACT

Embedded System has already been used wherever in our lives, and it has promoted the development of PC technology too. Proposed the "After PC" times based on application of Embedded System. With the development of computer and communication technology, more and more Embedded System equipments incorporate Internet. It makes a new Internet forms: Embedded Internet. The appearance of Embedded Internet has brought more security questions. So, the paper mainly research on the network security technologies of Embedded Internet.

Embedded Internet technologies include: Embedded Internet access technologies, Embedded Web Server, Embedded Internet protocol stack and Embedded Internet developing instruments. Using Embedded System brings the enormous change to Internet, forms Embedded Internet model, and produces Extend Internet Model too. Embedded Internet originated from this kind of idea of combining Embedded System with Internet. In Embedded System applying fields, based on Internet technologies, makes Embedded System and Internet joint each other, and realize resource-sharing, information communication and status control between Embedded System and Internet. This kind of connection between Embedded System and Internet is called Embedded Internet. Its model is as Figure 1 shows. The characteristic of Extend Internet Model is that host computers and terminal stations include these equipments that can set in Internet now and future, in addition, include also such as embedded network. It's model as Figure 2 shows.

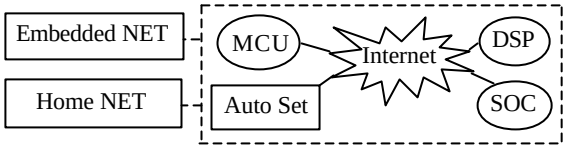


Figure.1 Embedded Internet Model

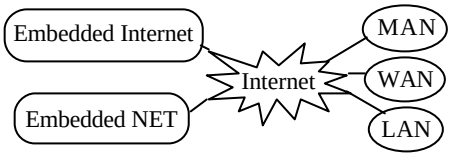


Figure.2 Extend Internet Model

Traditional Internet exists several kinds security threatened as follows: Encrypt algorithms, Network monitoring, Protocols weakness, Software design weakness and System disposes weakness. Embedded Internet application has its own characteristics, so, the network security question of Embedded Internet is different from traditional Internet. It based traditional Internet security question, considering Embedded Internet itself potential network security question. Extend Internet is shaped by Traditional Internet being merged other kinds of networks. It inherits the threatening of Traditional Internet and Embedded Internet, and it still has it's own characteristics. Embedded Internet has the characteristics like below: mobile IP, long-range management, and the physics position distributed and concentrated terminal host computer. Through improving Distributed Firewall technology, we can add some technologies like below: remote computer, VPN, special application tunnel, and hardware-firewall. New firewall technology comes and protects Embedded Internet security, we can call it: Embedded Internet Firewall. It's structure model as Figure 3 shows.

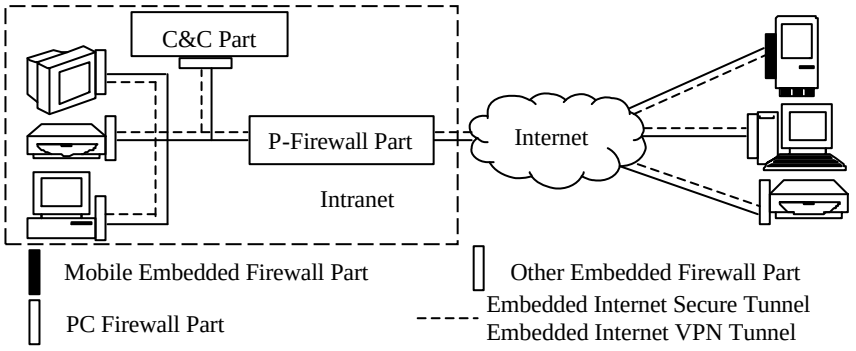


Figure.3 Embedded Internet Firewall Model

Embedded Internet Firewall can be divided into two major parts from this structure. One part is inside network, and it's physics position partially relative concentrated, these inside networks put the packet-filter firewall of border in some places of export of the network. Through it, firewall can strengthen security of inside network. Another part is the physical position scattered outside system equipments network, these terminal systems are linked together through Internet.

Embedded Internet Firewall's constitution mainly is like below: Centralized and Control Part, Perimeter Firewall Part, and Terminal Device Firewall Part. Terminal Device Firewall Part needs to realize the security protection function of the terminal device, and mainly realizes packet-filter function by security policy, it can also carry on the mechanism of concentrates management and machine's management, it needs to carry on users' authentication and confirm identities of this machine and other users and it needs to offer suppuration of IPv6 and VPN. Perimeter Firewall Part needs to realize the inside network's security protection function at first, mainly packet-filter according to policy and proxy, it needs to realize independent management and concentrates the mechanism of management, it needs users identity authentication of the internal and outside network and it needs to offer suppuration of IPv6 and VPN. Centralized and Control Part needs to realize Embedded Internet each part's management at first, it needs to carry on identity authentication to all users in Embedded Internet firewall and it also needs to offer suppuration IPv6 and VPN.

Each kind of part in Embedded Internet Firewall has similar structure model. To sum up, we can give out the unified hierarchical structure. It is divided into four layers as follows: Concentrate Administration Layer, Identity Authentication Layer, Encrypt layer and Security Transmit Layer.

Hardware environment of Embedded Internet environment is relatively complicated, and is more difficult to put up. So, the subject can only make use of ordinary, cheap PC hardware equipments to construct the embedded hardware equipments to simulate while studying. Besides, through utilizing and cutting down revising to open source code Linux to produce Embedded Linux operating system. Combining the two parts, can get simulate embedded computer system to put up embedded Internet environment. We can use it in this subject. Through combining the two parts, can form a simulating embedded system. This system is still a PC

system, but has already relatively been close to the performance characteristic of the embedded system in fact.

Through analyzing the realization of Embedded Internet Firewall's compositions structure technology can know, Centralized and Control Part has important function in Embedded Internet Firewall. It uses to realize that the security policy and firewall system management functions. It mainly includes: policy made, policy management and users management. The realization technologies of Perimeter Firewall Part can be divided into three kinds: packet-filter, proxy service and complex firewall technologies. Terminal Device Firewall Part can be divided into three parts: Mobile Embedded Equipment Firewall Part, Other Embedded Equipment Firewall Part and PC Firewall Part. Mobile Embedded Equipment Firewall Part is suitable for the mobile communication equipment, for instance: Cell-phone, PDA, etc. Other Embedded Equipment Firewall Part is mainly suitable for Information Product, relaying in equipments of communication, etc. PC Firewall Part means that in Embedded Internet, PC system needs it.

The realization of Centralized and Control Part is relatively complicated. It needs a kind of languages to describe policy, and a distributed tool, etc. The realization technologies of Perimeter Firewall Part are similar with Traditional Firewall's implementation technique. The realization technologies of Terminal Device Firewall Part can be divided into two kinds: based TCP/IP and WAP firewall part. The implementation technique principles of these two kinds of firewall parts are unanimous basically. According to existing technological foundation and experiment environment, this paper emphatically research the firewall part implementation technique based on TCP/IP. And the implementation technique firewall part based on WAP can consult the part based on TCP/IP.

The realization of Terminal Device Firewall Part includes: Unified Layer Model, Security Encrypt Layer and VPN support, Identity Authentication, Policy Management and Packet-filter by Policy. The realization in this paper is divided into two parts: Based on Linux and Embedded Linux, based on Windows and WinCE. The realization of firewall parts based on Linux and Embedded Linux was mainly finished through revising ipchains firewall in Linux operating system. And the realization of firewall parts based on Windows and WinCE has adopted Windows platform WDM equipment driver technologies.

This paper has set up Embedded Internet and Extend Internet model, analyzing its characteristic and potential network security threaten. Through improving the Distribute Firewall technology, we have put forward Embedded Internet Firewall. Analyzed every part of Embedded Internet Firewall in detail, and studied the corresponding implementation technique. Finally, researched the implementation technique of Terminal Device Firewall Part, and offered theoretical foundation and technology for development of the Embedded Internet Firewall Technology.

Keywords: Embedded System, Embedded Internet, Distribute Firewall, Embedded Internet Firewall