一种基于 Linux 的应用层网关的设计与实现

Design and Implementation of Application Gateway System Based on Linux

彭维平 （河南理工大学 计算机学院 河南 焦作 454003）
杨爱梅 （河南工业大学 河南 郑州 450052）
李子臣 刘 攀 （河南理工大学 计算机学院 河南 焦作 454003）

摘要：要保证网络资源用于正常的学习和工作而不是被滥用，除了从制度方面进行限制之外，一个应用层网络管理系统是不可缺少的。设计和实现了一个基于 GNU/Linux 和自由软件的应用层网关系统，通过一套基于 B/S 结构的网关管理软件对网关过滤规则进行管理和维护，为应用层的网络管理提供一种灵活、有效、可靠且低成本的解决方案，测试结果完全符合预期目标。

关键词：应用层网关 GNU/Linux iptables netfilter

1 引言

一直以来，计算机网络中的防火墙（firewall）或者网关（gateway）对于网络信息的过滤和控制主要体现在对来自或者发往特定端口或者特定 IP 地址的信息进行匹配。这种基于网络层的控制在过去很长一段时间是有效的，因为在这个时期出现的大多数的网络应用程序是使用固定端口的，并且服务器的 IP 地址常常是固定而且明显的。随着网络的发展，很多新的应用程序不再使用固定的端口，服务器和客户机的角色逐渐混淆，服务器的 IP 地址常常有多个而不是固定在一个地址上，原来的基于网络层的控制就显得不够灵活，甚至完全无法工作。要解决网络资源用于正常的学习和工作而不是被滥用这个问题，除了从制度方面进行限制之外，对于一些无力购买专业级防火墙的中小型企业或初、高级中学来说，通过一套基于应用层的网关对各种网络服务进行限制和管理，是非常可行和有效的。

本文设计和实现了一个基于 GNU/Linux 和自由软件的应用层网关系统，通过一套基于 B/S 结构的网关管理软件对网关过滤规则进行管理和维护，为应用层的网络管理提供一种灵活、有效、可靠且低成本的解决方案。

2 系统目标

可以对客户端能否使用各种网络软件（例如 QQ，BT 等）进行控制。可以对客户端访问的网址（URL）进行基于字符匹配的过滤。控制的方法是简单的，规则是容易维护的，而且具有很好的灵活性。无论使用者是否有专业知识都能进行基本的维护。系统应具有良好的可扩展性，可以通过添加新模块的方式来进行功能扩展。提供身份鉴别功能，进行规则维护需要提供适当的身份信息。整个系统有效、安全、易管理。

3 系统结构及原理

通过对系统目标的分析，我们获得一个如下的原型系统（基本结构如图 1）。

（1）使用 GNU/Linux 作为操作系统平台；

（2）使用 NetFilter 作为核心处理模块；

（3）使用 iptables 作为用户态软件对 NetFilter 进行管理；

（4）使用一种 CGI 提供用户管理界面；

（5）使用一种数据库存储规则和系统日志；

本系统的基础是 Linux2.4.* 内核中推出的一套全新的 NetFilter/iptables 包过滤机制。netfilter 是 Linux 核心中的一个通用架构，比以前任何一版 Linux 内核的防火墙系统都要完善。它提供了一个抽象、通用化的

框架，该框架定义的一个主要功能就是包过滤系统。其主要实现功能如下：

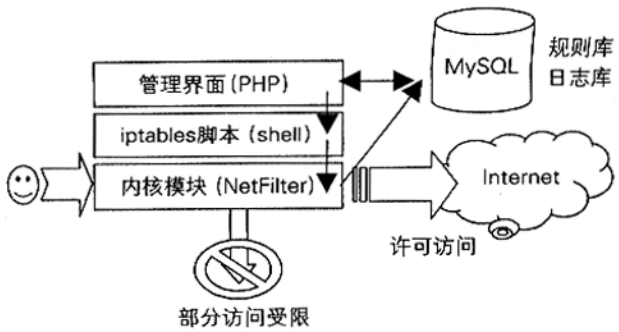

图 1 系统的基本结构

首先，为每种网络协议（如 IPv4，IPv6 等）定义一套钩子函数（其中 IPv4 定义了 5 个钩子函数），这些钩子函数在数据包流过协议栈的几个关键点被调用。在这几个点中，协议栈将把数据包及钩子函数标号作为参数调用 Netfilter 框架。

其次，内核的任何模块可以对每种协议的一个或多个钩子进行注册，实现挂接，这样当某个数据包被传递给 Netfilter 框架时，内核能检测是否有模块对该协议和钩子函数进行了注册。若注册了，则调用该模块注册时使用的回调函数，这些模块就有机会检查、修改、甚至丢弃该数据包，也能指示 Netfilter 将该数据包传入用户空间的队列。

最后，传入用户空间的队列数据包，将被用户进行异步处理。一个用户进程能检查数据包、修改数据包、还可以重新将数据包注入到内核中。但 IP 层对真正要抛弃 IP 数据包之前都要进行检查。

当有用户请求抵达内核模块时，NetFilter 会根据预先的策略定义对用户请求进行分拣，阻止非法请求，并且允许合法请求。而 NetFilter 的策略是依靠内核包过滤管理工具 iptables 具体实施的。一个 iptables 命令包含如下五个基本部分：

(1) 工作表：filter、nat、mangle，缺省表为 filter。

(2) 使用链：指定表下面的某个链，也可以是用户建立的新链，还可对链进行管理：-N 建立新链 -X 删除空链 -P 改变内建链的原则 -L 列出链中的规则 -F 清除链中的所有规则。

(3) 规则操作：-A 添加规则 -I 插入规则 -D 删除规则 -R 替代规则 -L 列出规则。

(4) 目标动作：ACCEPT 保持和原来的一致继续传递数据包 DROP 丢弃数据包，不再继续传递 QUEUE 队列化数据包，为用户空间处理准备 RETURN 返回到前面调用链。

(5) 匹配数据包条件：包括基本匹配条件、匹配条件扩展、目标动作扩展。其中基本匹配条件：-P 指定协议：tcp udp icmp；-s 源地址和 -d 目的地址：单一地址指定 192.168.1.1 或 www.domain.com，多个地址指定 192.168.1.0/255.255.255.0；-I 输入接口和 -o 输出接口：INPUT 链只可能有 -i 网络接口，OUTPUT 链只可能有 -o 网络接口，FORWARD 链既可以有 -I 网络接口，也可以有 -o 网络接口；-f 包碎片：用来指定超大数据包划分后的第二个及其以后的数据包碎片。还有几个常用的命令：-j 跳转命令、-m 匹配命令、-! 相反指定命令。匹配条件扩展：TCP：匹配源端口 -sport，目的端口 -dport，及 tcp 标记组合、选项等；UDP：匹配源端口 -sport 和目的端口 -dport；ICMP：匹配 icmp 类型，后跟数字类型或编码，可求帮助 -p icmp -help；MAC：匹配接收到的数据的 mac 地址，只用于 PRER - OUTINGG 链和 INPUT 链；MARK：匹配 nfmark；OWNE：匹配用户 ID，组 ID，进程 ID 及会话 ID（仅用于本地产生的数据包）；LIMIT：匹配特定时间段内的数据包限制；STATE：匹配特定状态下的数据包；TOS：匹配 IP 头的 TOS 字段的值。目标动作扩展：LOG：将匹配的数据包传递给 syslog（）进行记录；ULOG：将匹配的数据适用用户空间的 log 进程进行记录；REGECT：不仅丢弃数据包，同时返回给发送者一个可配置的错误信息；MIRROR：互换源和目的地址以后重新传输该数据包。

在本系统中，我们使用一个 shell 脚本将所有的 iptables 命令集合起来进行执行。而该 shell 脚本是通过系统提供的一个基于 PHP + MySQL 的规则维护模块来产生的。值得注意的是，这里的规则不同于 NetFilter 的策略，这里的规则非常简单而且容易理解，不需要任何专业知识也可以对规则进行维护。

例如："对 ip 地址范围 192.168.0.1～255 在上午 8:00 到下午 6:00 之间禁止使用 QQ，允许使用 MSN，并且过滤新闻类网站。"对应的 iptables 语句：

```
iptables -I FORWARD -s 192.168.0.0/24 -m string --string "news" -j DROP
```

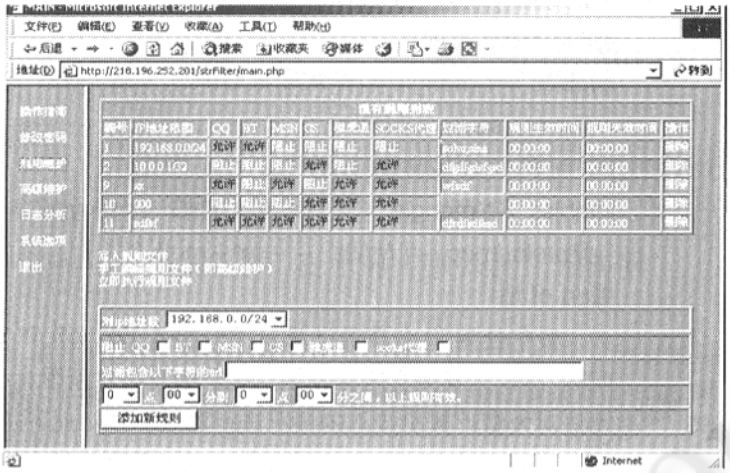

图 2　规则维护界面

iptables -t mangle -A POSTROUTING -s 192.168.0.0/24 -m layer7 --l7proto qq -m time --timestart 08:00:00 --timestop 18:00:00 -j DROP

iptables -t mangle -A POSTROUTING -s 192.168.0.0/24 -m layer7 --l7proto msnmessenger -m time --timestart 08:00:00 --timestop 18:00:00 -j ACCEPT

4　系统实现

4.1　环境配置与测试

首先需要安装一个 Linux 操作系统，由于我们需要使用的字符匹配补丁（ipt_STRING）只能工作在 2.4.xx 下，所以我们需要使用 2.4 这个分支的内核。另外我们还需要手工编译安装 apache + mysql + php 和 iptables 及其补丁，并测试能否正确解析 php 代码以及 php 代码和 Mysql 数据库的连接是否正确。

4.2　PHP 核心程序

按功能分，主要提供了五个部分的 PHP 脚本程序。第一部分为身份验证的脚本程序；第二部分为系统参数设置的脚本程序；第三部分为“规则维护”的脚本程序；第四部分为“执行规则”的脚本程序；第五部分为日志分析的脚本程序。

身份验证的脚本程序主要提供了用户名、密码的修改和验证功能。由于进行防火墙及规则的配置仅允许在内网中进行，因此，程序只是简单地将用户名和密码存于文件中，并对该文件内容进行了适当的加密。

系统正常运行设定一些前期参数主要包括以下几项：IP（可管理的 IP 地址列表，用分号隔开，斜杠后面的数据表示掩码）、Filename（规则文件名，默认保存在程序所在目录下）、NatIP（规则文件将对此处所列出的 IP 做源地址转换）、Crond（规则文件每隔此时间执行一次，单位为秒）。

规则维护的操作界面如图（二）所示，这是系统的核心部分。页面上部的表显示了当前数据库中所存放的规则的内容，操作员可以通过页面下端的选项添加新规则。提供选项内容包括四部分：（1）选定待实施规则的 IP 地址段；（2）选定此规则应拒绝的服务（如：QQ、BT、MSN、雅虎通、socks 代理等）；（3）设定“过滤包含以下字符的 URL”；（4）设定此规则有效的时间段。

针对不同的用户，提供了两种不同的规则维护设置的方法：

普通维护 在 PHP 脚本中，设置和查看规则主要通过执行系统的 iptables 命令来实现。通过对 iptables 命令格式进行研究，对其常用命令格式中的参数进行分解，提供了一个针对 iptables 命令中的参数进行简单设置和选择的脚本页面。用户只需对页面中的内容进行简单填写和选择就能生成所需的规则。

高级维护 作为补充，PHP 脚本也提供了高级设置功能，用户可以手工编辑规则文件，即直接在页面上修改和删除已有的规则和输入新的规则。

日志分析的脚本程序根据访问记录，以正常访问量、试图登陆 QQ 量、试图使用 BT 量、试图使用 MSN 量、试图访问被过滤的网址量以及其他访问量所占比例，生成简单的统计图表。

5　系统测试

添加如下规则，对内网地址 192.168.0.1～254 做 NAT，并且对 QQ/BT/MSN 等做一定限制。

```
#加载基本模块
/sbin/modprobe ip_tables
#加载 ftp 相关模块
/sbin/modprobe ip_nat_ftp
/sbin/modprobe ip_conntrack_ftp
#打开内核的转发功能
```

```
echo 1 > /proc/sys/net/ipv4/ip_forward
#清除输入规则
iptables -F INPUT
#清除转发规则
iptables -F FORWARD
#清除 nat 规则
iptables -F POSTROUTING -t nat
#允许转发
iptables -P FORWARD ACCEPT
#对来自数据库中 natIP 网段的地址进行转发
iptables -t nat -A POSTROUTING -o eth0 -s 192.168.0.0/24 -j MASQUERADE
#以下是数据库中的规则,可根据需要进行调整和优化。
#禁止关键字"sina",禁止访问所有包含 sina 字符的网站。
iptables -I FORWARD -s 192.168.2.0/24 -m string --string "sina" -j DROP
#限制某一个 IP 访问端口 23 的并发数在 2 以内
iptables -A FORWARD -p tcp --syn --dport 23 -m connlimit --connlimit-above 2 -j DROP
#设置 IP 192.168.2.3 在周一到周五早上 8 点到下午 6 点能访问互联网
iptables -I FORWARD -s 192.168.2.3 -m time --timestart 8:00 --timestop 18:00 --days Mon,Tue,Wed,Thu,Fri -j ACCEPT
#阻止 bittorretn 协议:
iptables -t mangle -A POSTROUTING -m layer7 --l7proto bittorrent -j DROP
#阻止 MSN:
iptables -t mangle -A POSTROUTING -m layer7 --l7proto msnmessenger -j DROP
#阻止 QQ 协议:
iptables -t mangle -A POSTROUTING -m layer7 --l7proto qq -j DROP
#阻止 CS:
iptables -t mangle -A POSTROUTING -m layer7 --l7proto counterstrike -j DROP
#阻止雅虎通:
iptables -t mangle -A POSTROUTING -m layer7 --l7proto yahoo -j DROP
#阻止 socks 代理:
iptables -t mangle -A POSTROUTING -m layer7 --l7proto socks -j DROP
#限制 BT 协议的带宽:
iptables -t mangle -A POSTROUTING -m layer7 --l7proto imap -j MARK --set-mark 3
#End of file
```

我们使用 12 台普通计算机同时接入 192.168.0.0 网段,测试结果完全符合预期目标。由于条件所限,无法对大量计算机接入的情况进行测试。预计此环境至少可以支持 50 客户端同时接入,可以满足一般小型公司对应用层网络管理的需求。

6 总结

本文利用 Linux 内核中推出的一套全新的 NetFilter/iptables 包过滤机制,通过 PHP 编程工具开发的网关管理软件对网关过滤规则进行管理和维护,为应用层的网络管理提供了一种灵活、有效、可靠且低成本的解决方案,测试结果完全符合预期目标。

本文作者创新点在于使用一个 shell 脚本将所有的 Iptables 命令集合起来进行执行。而该 shell 脚本是通过系统提供的一个基于 PHP + MySQL 的规则维护模块来产生的,这里的规则不同于 NetFilter 的策略,这里的规则非常简单而且容易理解,不需要任何专业知识也可以对规则进行维护。

参考文献

1 李晓峰、张玉清、李星,《Linux2.4 内核防火墙底层结构分析》[J],计算机工程与应用,2002,(14) p138.

2 博嘉科技《Linux 防火墙技术探秘》[M]. 北京:国防工业出版社,2002.199-202.

3 VICTOR Castro. Roll Your Own Firewall with Netfilter [J/OL]. Linux Journal, Oct, 2003.

4 RUSTY Russell. Linux 2.4 Packet Filtering HOWTO [EB/OL]. http://www.netfilter.org/documentation/HOWTO//packetfiltering-HOWTO.html, Jan, 2002.

5 乔晓丹、张鹏,一个基于 Linux 操作系统的嵌入式网关的实现[J],微计算机信息,2005 年 10 期,P26.