防火墙入侵阻止功能的设计与实现

乔向东，杨　仝，张景伟
(空军工程大学 电讯工程学院，陕西 西安 710077)

摘　要：针对防火墙内嵌入侵阻止功能模块的研究，提出了2种不同的设计方案，方案1利用基于libipq库的netfilter的排队(QUEUE)动作以及snort_inline技术实现对攻击数据包的丢弃，方案2则是综合利用syncookies技术、netfilter的新模糊包速率匹配、PSD检测、U32检测技术并结合对防火墙内核改造来实现对主流拒绝服务攻击(DOS)攻击包的阻止；在综合比较的基础上，依据方案2完成了实验模块的设计与实现；攻击测试结果表明，该模块具备了较好地防御主流DOS攻击的能力。

关键词：防火墙；入侵阻止；设计
中图分类号：TP391　　**文献标识码**：A　　**文章编号**：1009-3516(2007)04-0065-04

防火墙入侵阻止(IPS)功能是近两年出现的一个新概念，它与传统意义下的入侵检测系统[1](IDS)有着较大的区别。与入侵检测系统不同，防火墙入侵阻止功能则是以防火墙接收、转发的数据包为数据源信息，对其进行分析检测，检测到攻击数据包后立即实施阻断。而入侵检测系统本身则不能实现阻止攻击的功能[2]。

1　设计方案

1.2　模块设计功能

防火墙入侵阻止模块将要实现的主要功能包括：

1)在保证访问控制引擎功能正常的前提下，提供对内网重点网段或主机的攻击防护，入侵阻止模块可拦截对防火墙自身的多种主流攻击数据包。

2)向用户提供设置入侵阻止参数的CLI和GUI接口，便于用户根据需求自行设置。

3)鉴于网络攻击种类繁多，目前仅考虑对主流常见的网络攻击行为实施阻止，包括DOS攻击中的各类flood攻击(Syn，Udp，Icmp等)、端口扫描攻击以及其他诸如tcp spoofing，winnuke，icmp fragment，abnormal icmp，ping of death，teardrop，Chargen_DoS，snork等网络攻击。

基于上述设计思想，提出了以下2种防火墙入侵阻止模块的设计方案。

1.2　方案1

方案1的核心技术路线是在netfilter的filter表的转发检查点(NF_IP_FORWARD)(也还可以包括进入检查点NF_IP_LOCAL_IN)上利用基于libipq库的netfilter排队(QUEUE)动作通过netlink接口将待转发数据包发送到用户空间，并由Snort_inline实现对用户空间接收到的数据包进行检测，若检测到攻击就采取相应的处理动作，而正常数据包则再次返回内核由netfilter的其余规则完成正常的访问控制动作。Snort_inline是Snort[3-5]的一个变体，其规则语法、处理流程、检测引擎等均与Snort完全一致，主要区别在于：①它不是利用libpcap获取数据包而是从netfilter的检查点上获取数据；②Snort_inline支持3种新的规则行为，即drop(丢弃并日志)、reject(拒绝)和sdrop(丢弃不日志)。方案1的最主要优点就是原理相对简单，且对现有防

收稿日期：2006-12-08
基金项目：空军工程大学电讯工程学院科研基金资助项目(120059 DG040825)
作者简介：乔向东(1970-)，男，陕西佳县人，副教授，博士，主要从事信息融合与信息安全研究.

火墙内核改动不大,实现难度较小。但是随着研究的不断深入,发现该方案仍然存在以下 3 点问题。第 1 就是 Snort_inline 的攻击检测类型尽管大而全,但仔细研究其检测规则集(我们采用的是 Snort_inline - 2.2.0a)后发现其中的许多规则过于简单、机械。第 2 是 Snort_inline 会对接收到的数据包进行关于所有攻击类型的检测匹配,不仅对防火墙性能影响较大而且难以实现用户对所阻止攻击的类型进行细粒度控制。第 3 是对于专门的入侵检测系统而言,大量的检测数据由内核发往用户空间以独立进程存在的入侵检测程序是可以接受的;但对于作为访问控制设备的防火墙而言,一定量的数据进入用户空间后再返回内核,势必影响转发效率,况且此前防火墙的所有的访问控制动作(代理除外)全部在内核完成。由于方案 1 的上述缺陷,重新讨论并设计新的方案 2。

1.3 方案 2

在考虑方案 2 时,希望所有的入侵阻止动作均在内核空间完成,无用户空间程序参与(用户配置命令除外)。要实现这一目的,显然也只能再次利用 netfilter 的包过滤机制,这样构造入侵检测规则就成为问题的关键。然而对于 Synflood 和端口扫描这样的攻击行为,用基本的 netfilter 匹配(match)是难以实现这个目的的。例如对各类 flood 攻击,通常的检测方法是以接收特定类型数据包的速率(包/秒 pps)作为依据的,一旦速率大于设定阀值则判定攻击发生;而对于端口扫描攻击则是以相同地址的主机连接目标主机不同端口的频率作为依据,若频率大于设定阀值就判定攻击发生。而这种包速率以及端口连接频率的检测是要以 netfilter 支持的匹配为前提的,这就需要寻找可资利用的 netfilter patch 或自行设计实现。

为此,跟踪、下载了最新的 netfilter 补丁 patch - o - matic - ng - 20060330[6] 找到了 3 个对于后续研究非常重要的补丁:模糊包速率匹配(fuzzy)、端口扫描检测匹配(PSD)以及数据包任意字节匹配(U32)。其后,对这 3 项匹配做了代码分析和并编译进内核进行功能测试,测试结果符合预期效果。有了这 3 项 netfilter 匹配项的支持,并结合 syncookies 技术(一种应对 TCP 半连接的技术),通过对防火墙内核的改造来实现一个轻量级的入侵阻止模块。该方案的优点首先在于所有的入侵检测规则与访问控制规则均在 netfilter 整体框架内;其次是由于不再依赖第 3 方的规则库因而可以灵活地构造更具针对性的规则;第 3 就是用户可以灵活控制诸如攻击类别、监测网段、速率阀值等参数。

2 入侵阻止模块实现

2.1 规则的组织结构

为了便于管理考虑在内核中注册一个专门针对入侵阻止规则的表 packet_ips,以下简称 IPS 表,该表的 hook 检查点仅只设在 NF_IP_FORWARD 和 NF_IP_LOCAL_IN 上,相应的检查机制与 filter 表类似。以下为主要数据结构。

1)注册的 packet_ips 表

```
static struct ipt_table packet_ips
 = { { NULL, NULL }, "ips", &initial_table.repl, IPS_VALID_HOOKS, RW_LOCK_UNLOCKED,
NULL, THIS_MODULE };
```

2)在 packet_ips 表中注册的 nf_hook_ops 结构体 ipt_ops

```
static struct nf_hook_ops ipt_ops[ ]
 = {{ { NULL, NULL }, ipt_ips_hook, PF_INET, NF_IP_LOCAL_IN, NF_IP_PRI_FILTER }
 { { NULL, NULL }, ipt_ips_hook, PF_INET, NF_IP_FORWARD, NF_IP_PRI_FILTER }};
```

3)ipt_ops 中的钩子函数 ipt_ips_hook

```
static unsigned int ipt_ips_hook(unsigned int hook, struct sk_buff * *pskb, const struct net_device *in,
const struct net_device *out, int ( *okfn)(struct sk_buff *))
{
return ipt_do_table(pskb, hook, in, out, &packet_ips, NULL);
};
```

用户的配置命令可在 Forward 链上添加类似如图 1 所示规则。注意,packet_ips 表中 FORWARD 和 INPUT 链的缺省策略为 ACCEPT,而自建的 tcp - packet,synflood,icmp - packet,udp - packet 以及 portscan 链中

规则的 target 应为 DROP,这样才能确保不匹配任何入侵特征的数据包最终实现 FORWARD 或 INPUT 链的缺省策略。

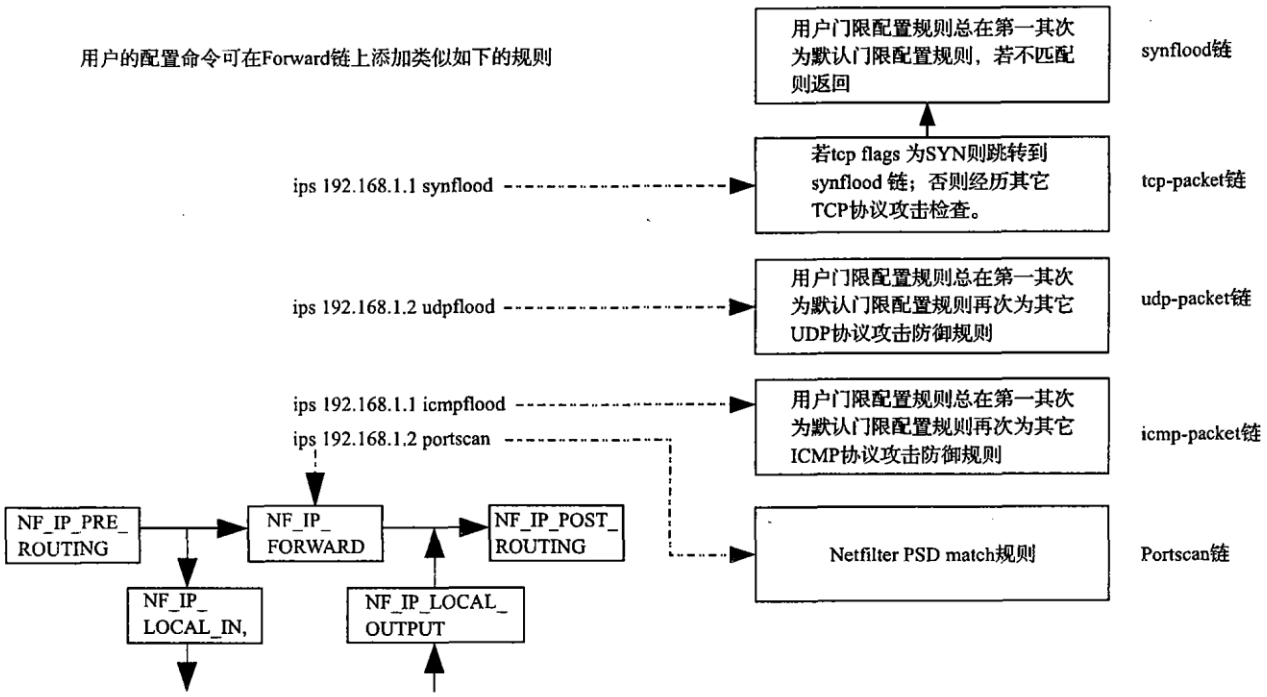

图 1　模块规则组织

2.2　规则的添加

完成建立自建规则链后,就可以向规则链中添加相应的默认的入侵阻止规则。下面以向 tcp - packet 链添加阻止 Localhost_Source_Spoof 攻击为例说明规则添加过程。

1)定义数据结构

```
struct ipt_entry * e; //规则
struct ipt_entry_target * target; //规则目标/动作
struct ipt_entry_match * match; //规则匹配项
struct ipt_u32 * u32; //U32 匹配的数据结构
```

2)定义规则头,包括规则匹配后的动作或跳转的目标链名称

3)计算规则匹配项的长度和规则长度,本例中的规则匹配项长度 msize 为

```
msize = IPT_ALIGN (sizeof (struct ipt_entry_match)) + IPT_ALIGN (sizeof (struct ipt_u32));
size += IPT_ALIGN (sizeof (struct ipt_entry)) + msize;
```

4)为规则和规则匹配项分配空间,利用 U32 作为规则匹配来匹配 Localhost_Source_Spoof 攻击特征

```
u32 = (struct ipt_u32 *) match - > data;
u32 - > ntests = 1;
u32 - > tests[0]. nnums = 2;
u32 - > tests[0]. location[0]. number = 12;
u32 - > tests[0]. location[1]. nextop = IPT_U32_AND;
u32 - > tests[0]. location[1]. number = 0xFF000000;
u32 - > tests[0]. nvalues = 1;
u32 - > tests[0]. value[0]. min = 0x7F000000;
u32 - > tests[0]. value[0]. max = u32 - > tests[0]. value[0]. min;
```

5)调用 iptc_insert_entry 函数完成规则加载

2.3　用户命令设计

入侵检测模块的命令主要用于完成用户对入侵阻止模块的参数配置,具体包括攻击检测阀值命令和网段(主机)攻击监视命令以及辅助三大类,共 19 条命令。前者所包含 6 条命令具体如下,:

(no) icmp rate threshold LL - HH //配置(取消配置)icmp flood 攻击的包速率上(HH)、下(LL)门限值
(no) syn rate threshold LL - HH //配置(取消配置)syn flood 攻击的包速率上(HH)、下(LL)门限值
(no) udp rate threshold LL - HH //配置(取消配置)udp flood 攻击的包速率上(HH)、下(LL)门限值
上述配置中包速率单位为 pps。网段攻击监视命令(8 条)包括:
(no) ips icmp A. B. C. D/M //配置(取消配置)阻止对 A. B. C. D/M 网段的 ICMP 协议类攻击
(no) ips tcp A. B. C. D/M //配置(取消配置)阻止对 A. B. C. D/M 网段的 TCP 协议类攻击
(no) ips udp A. B. C. D/M //配置(取消配置)阻止对 A. B. C. D/M 网段的 UDP 协议类攻击
(no) ips psd A. B. C. D/M //配置(取消配置)阻止对 A. B. C. D/M 网段的端口扫描攻击
其它命令(5 条)包括:
start syncookies //针对 syn flood 攻击,启动 syncookies
stop syncookies //停止 syncookies
(no) log ips drop //配置(取消配置)对丢弃的攻击数据包进行日志
show ips //显示当前防火墙入侵阻止模块配置信息
完成模块实现后,对该模块的网络攻击阻止能力进行了大量攻击测试,测试日志表明该模块实现了预期的设计目的,对前述主流攻击行为起到了较好的防御目的。

3 小结

本文探讨了防火墙入侵检测模块的设计原则以及 2 种设计方案,并依据方案 2 完成了实验模块开发。攻击测试结果表明,该模块具备了较好的防御主流 DOS 攻击的能力。

参考文献:

[1] 陈铁柱. Snort 规则集的优化[J]. 海军航空工程学院学报,2005,20(6):664 - 666.
[2] 薛静锋,宁宇鹏,阎 慧. 入侵检测技术 [M]. 北京:机械工业出版社, 2004.
[3] 刘文涛. Linux 网络入侵检测系统 [M]. 北京:电子工业出版社, 2004.
[4] Linux - 2.4.20 - 8 内核源码 . [EB/OL]. [2005 - 10 - 30]http://www. kernel. org
[5] Snort - inline - 2.2.0a 源码 . [EB/OL]. [2005 - 10 - 30]http://www. snort. org
[6] Netfilter - Extension - Howto 文档 . [EB/OL]. [2006 - 03 - 30]http://www. netfilter. org

(编辑:田新华,徐楠楠)

Design and Implementation of a Firewall IPS Module

QIAO Xiang - dong, YANG Tong, ZHANG Jing - wei
(The Telecommunication Engineering Institute, Air Force Engineering University, Xi'an 710077, China)

Abstract: Two different design schemes about firewall IPS module are brought up. In the first scheme snot - inline technique and the QUEUE action of netfilter are used to achieve dropping of attack data packet. In the second one, IPS about Denial of Service attack is achieved with the benefit of combination of synccokies, the new netfilter fuzzy match, PSD match, U32 match with an improvement on the firewall kernel. After comprehensive comparison, the module is developed according to the second scheme. The experiment shows that the designed module works well in defending main DOS attack.

Key words: firewall; intrusion prevention; design