一种策略分流的入侵防御及恢复系统架构①

杨忠明[1], 秦 勇[2], 蔡昭权[3]

[1](广东科学技术职业学院 计算机工程技术学院, 珠海 519090)
[2](东莞理工大学 计算机学院, 东莞 523808)
[3](惠州学院 教育技术中心, 惠州 516007)

摘 要: 通过充分利用入侵检测各类产品的安全防护特点, 本文设计了一种策略分流的入侵防御及恢复系统架构. 采用双 NIDS 系统作为前端检测模块, 通过策略分流, 使得双 NIDS 系统全面覆盖入侵检测的各个协议层, 充分发挥两种 NIDS 系统的检测优势, 实现高效的入侵检测. 并结合 HIDS 的主机日志防护机制及关键内容恢复机制, 在即便出现入侵破坏数据的情况下, 仍可保证系统的关键部位安全.

关键词: 入侵检测; 策略分流; 入侵恢复; NIDS; HIDS

Intrusion Prevention and Recovery System Architecture Based on Policy Shunt

YANG Zhong-Ming[1], QIN Yong[2], CAI Zhao-Quan[3]

[1](College of Computer Engineering Technical, Guangdong Institute of Science and Technology, Zhuhai 519090, China)
[2](College of Computer Science, Dongguan University of Technology, Dongguan 523808, China)
[3](Research Department, Huizhou University, Huizhou 516007, China)

Abstract: By making full use of the security features of intrusion detection products, this paper designs an intrusion prevention and recovery system architecture based on policy shunt. The system uses the double NIDS system as the front-end detection module, making the double NIDS system completely cover the various protocol layers of intrusion detection by policy shunt and gives full play to the advantages of both NIDS detection system to achieve the high-efficiency intrusion detection. In addition, the system combines with the host logs protection mechanisms and the key recovery mechanisms. Therefore, even in the case of intrusion and data destruction, the system can ensure the security of the critical parts in the system.

Key words: intrusion detection; policy shunt; intrusion recovery; NIDS; HIDS

1 引言

对于网络中潜在的入侵攻击, 传统的防火墙只能提供静态的被动的防护, 无法对实时攻击或异常行为及时作出反应, 因而能够自动调整策略设置的以防御入侵行为的入侵检测系统(IDS, Intrusion Detection System), 以其动态的安全策略备受青睐.

在入侵检测系统架构以及安全模型的研究方面, 专家学者们贡献了许多宝贵的研究成果.

1994 年, 美国的 S.Forrest 教授提出了从主机角度区分“自我”和“非我”的思想[1], 并在两年后建立了 UNIX 特权进程的“自我”定义, 将其引申到基于主机的入侵检测系统 HIDS(Host-based Intrusion Detection System)中, 运用“自我”和“非我”区分“抗原”与“抗体”. 然而这种“自我”思想在面对高速网络海量数据中的异常行为分析具有较大的局限性. 在基于统计分析的入侵检测系统方面, Denning 归纳出了均值与标准偏差、时间序列分析等 5 种模型[2], 通过对用户历史行为进行建模, 运用数学方式的量化分析检测异常行为, 但这种分析方法很难选取合适的阈值来作为异常模式的判断条件, 使得基于统计分析的入侵检测模型的误报率一直高居不下. Kumar 提出了模式匹配系统[3], 将入侵信号分为存在、序列、规则、其他模式四种模式, 通

① 基金项目: 国家自然科学基金项目(61170193);广东省工业高新技术领域科技计划项目(2013B010401036);广东省高等学校优秀青年教师培养计划项目(YQ2014187);广东省自然科学基金项目(s2013010013432);广东省教育厅科技创新项目(2013KJCX0178)

收稿时间:2016-06-03;收到修改稿时间:2016-07-14 [doi:10.15888/j.cnki.csa.005620]

© 中国科学院软件研究所 http://www.c-s-a.org.cn

过收集的信息与已知的模式数据库进行对比, 从而确认异常行为, 但模式匹配系统仅针对已知入侵行为, 对于未知的入侵行为及其变异难以提取攻击特征.

目前主流的入侵检测系统大多基于开源的 Snort 系统, 在其之上进行优化改良形成产品. Snort 通过捕捉和分析网络数据包, 运用模式匹配对网络信息跟已知攻击特征库进行对比, 及时阻断异常行为, 若发现未知攻击行为即将其补充进攻击特征库[4]. Snort 系统所开源的大部分检测规则针对传输层及以下的层的入侵检测, 对于应用层协议及内容的分析比较薄弱,无法对基于应用层协议攻击行为进行有效检测. 随着新的应用层协议不断出现, 应用层协议的形式与种类更为繁杂, 基于Snort的架构进行入侵检测从架构上限制了检测的准确性和效率. Bro 是一种在 Snort 之后提出的新开源入侵检测系统, 其基础架构上设计了强大的策略脚本语言及应用层协议分析框架等, 更有利于针对应用层协议的入侵检测进行扩展[5].

由于任何单一的入侵检测模型都存在着明显的局限性, 而在如今网络大数据时代, 需要更为完善更为全面的入侵检测系统. 本文旨在提出一种策略分流的入侵防御及恢复系统架构, 融合各种 IDS 的优势理念, 实现高效的入侵检测, 并恢复系统原状态.

2　入侵检测技术

2.1 NIDS

由于 NIDS 是目前应用最为广泛的入侵检测系统类型, 常见的 NIDS 有 Snort 和 Bro.

Snort 以网络数据作为分析数据源, 通过收集网络层实时流量中的数据包, 剥落数据包中的包头, 利用 TCP-IP 栈解码技术分离出 IP、协议、端口及敏感信息等内容, 预处理器将其格式化并以日志的形式记录下来. 检测引擎从日志中提取出格式化数据, 利用约定的语法写成的检测规则对其进行内容搜索和匹配. 检测规则可以根据实际情况新增不同检测粒度的规则以适应不同需要的入侵检测, 能够发现出来自外界高危害的攻击行为, 如缓冲区溢出攻击、端口扫描、SMB 探针等. Snort 体系结构如图 1 所示.

Snort 的实现较为简单, 结构松散, 本质上是一个搭载匹配分析数据包的嗅探器, 而规则集中的预设规则都较为简单, 大部分规则针对网络层和数据链路层, 面向应用层的检测规则较少, 且不对应用层协议解码. 粗线条的检测方式依靠较为简单的判断条件而非缜密的规则关联来甄别数据包的特性, 很容易产生漏报或错报的情况, 报警准确性不高; 而 Snort 通过 libpcap 全面捕获数据包的做法不仅消耗大量的系统资源, 同时还使得其他模块无法接纳海量数据而令执行效率处于较低水平[6].

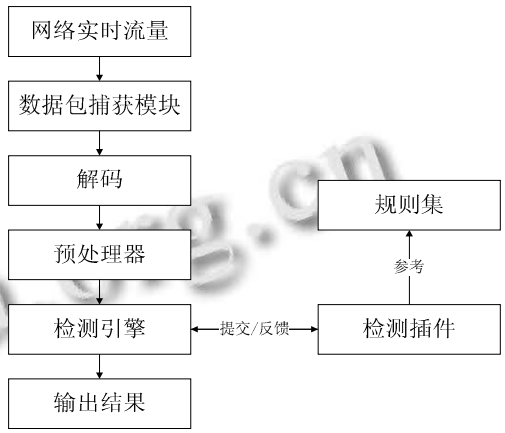

图 1　Snort 体系结构图

Bro 是一个基于 Unix 的开源网络入侵检测系统(NIDS), 从内部结构来看, 跟 Snort 较为相似. Bro 也采用 libpcap 捕获网络数据包, 注重细致的流量分析和协议状态分析, 不同的是, 其将所有链接均看作处在虚拟链接状态, 将 IP 地址、端口等信息封装成数组, 越底层需要处理的数据量越大, 越往上层, 则逐步递减, 有利于每个数据项的层次性处理, 简化了一部分数据分析的操作[7]. 事件引擎在低级别分析的时候利用正则表达式进行模式匹配, 当检测到入侵行为的时候, 通过上下文标识机制来标识该攻击行为. 策略组件对事件引擎产生的事件队列归类, 据此形成应对策略, 解释器检索出相应的事件处理脚本并执行. Bro 运作流程如图 2 所示.

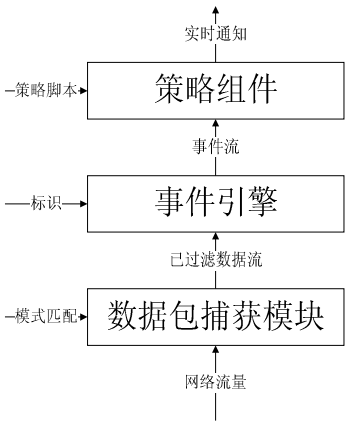

图 2　Bro 运作流程图

© 中国科学院软件研究所 http://www.c-s-a.org.cn

然而 Bro 存在不少缺陷, 例如不支持拦截 UDP 包, 没有优质的数据分析工具等. 数据方面, Snort 支持检查多种数据库, 并可以将过滤结果写入数据库, 而 Bro 并不支持, 且不具备检测数据攻击. 但 Bro 的优势也相当明显, 面向应用层的规则集数量更多, 检测范围更大, 检测粒度更细.

2.2 HIDS

基于主机的入侵检测系统(HIDS)以主机中的日志文件作为数据源. HIDS 启动后, 备份主机中关键的系统文件和可执行文件, 监视其完整性, 并定期检查校验和. HIDS 还监视主机端口, 设置特定端口访问报警机制, 并限制端口传输文件的类型和大小. 而针对日志文件, 监视日志中的系统事件, 当存储的文件发生变化的时候, HIDS 将对比新的条目和攻击标识, 若匹配成功则报警, 并管控所涉及的登录用户, 启动防御机制, 封存目标文件或恢复关键文件的备份, 以保证主机正常运行[8]. HIDS 运作流程图如图 3 所示.

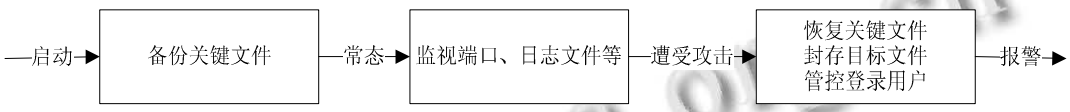

图 3 HIDS 运作流程图

由于 HIDS 是基于主机的入侵检测系统, 与操作系统的契合度要求较高, 主要面向主机的本地监控, 无法提供面向网络的入侵检测手段, 若主机处于网络中, 很难抵御来自网络的攻击行为. 如当主机遭受针对网络协议栈漏洞[9]的攻击时, 极可能由于系统协议栈溢出死机.

3 分布式反馈控制入侵检测防御系统

基于上述入侵检测技术的研究, 考虑到 NIDS 和 HIDS 的针对性不同, 无法单一地完成安全防护, 本文提出一种结合开源的 NIDS 与 HIDS 系统结合的入侵检测防御系统, 其基本思想是利用 Snort 和 Bro 两个 NIDS 系统作为防范攻击的过滤模块, 由于 Snort 和 Bro 各有优势, 利用策略分流将捕获到的数据包分类分发到不同 NIDS, 使得 NIDS 可以有效的发挥自身长处, 通过规则集进行模式匹配, 对已知攻击类型的数据包直接丢弃处理并且联动防火墙阻挡攻击 IP 的访问. 同时利用分布式 HIDS 系统补充和完善体系框架, 使得主机中关键区域得到有效监控. 当 NIDS 发现新的入侵行为时, HIDS 主动诊断受保护的服务器群健康状态(如检查日志、账户状态等), 若发现系统出现异常或关键部位数据被访问, 即触发数据恢复模块对关键数据进行恢复并且将其锁定, 暂时屏蔽来自所有用户的危险指令(包括远程访问等), 同时通过邮件、短信等方式向管理员报警. 只有经过管理员确认了主机状态安全, 手动设置 HIDS 撤销锁定操作后, HIDS 将自动复原攻击行为, 方便管理员根据其特性编写新的过滤规则, 同时将新规则反馈到 NIDS 中以避免类似攻击行为的二次危害, 至此完成服务器群的入侵检测及防御. 系统架构如图 4 所示.

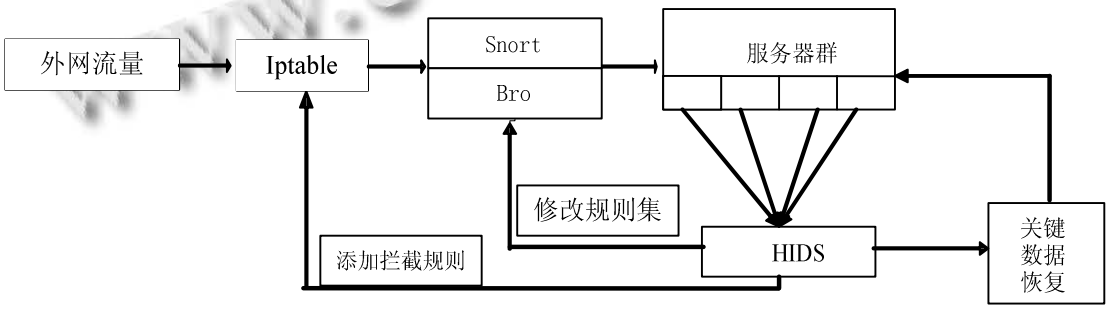

图 4 系统架构图

整套系统并不是简单的安装和集成开源入侵检测软件, 而是采用分布式和负载均衡方式处理通过防火墙的流量数据, 对于两个 NIDS, Snort 和 Bro 从源代码上进行改动与优化, Snort 侧重于网络底层协议的分析,

© 中国科学院软件研究所 http://www.c-s-a.org.cn

Bro 提供了一些关键的高级特性, 有相当多的新特征和强大的策略脚本语言, 侧重于应用层协议的分析, 本系统通过策略分流, 利用 Snort 和 Bro 各自已知的优势, 将数据包分发至不同的 NIDS, 使得 Snort&Bro 充分发挥各自优势, 节省系统资源. 系统具有分布式、负载均衡以及策略分流的特点, 保证了两者的效率和更好地避免多余动作.

此外引入的 HIDS 系统, 能够对可疑攻击行为做进一步分析, 并通过审查系统日志及关键数据完整性获知系统状态, 获取攻击证据, 有助于网络审计, 并且利用关键数据恢复模块对受损的关键数据进行恢复, 对整个系统有补漏作用, 最后将处理结果反馈至防火墙和 NIDS.

4 入侵检测模拟

如图 3 所示, 当来自外网的流量通过防火墙, 由于防火墙属于静态防御结构, 新型的攻击行为不容易被识别出来, 这时部署于防火墙后的 Snort&Bro 复合 NIDS 就起到了对看上去正常的流量做分析, 试图从中发现不被防火墙发现的异常行为数据包.

以针对 25 端口的检测为例, 首先在/etc/snort/snort.conf 中配置规则头: alert tcp $EXTERNAL_NET any -> $HOME_NET 25

针对 25 端口的扫描规则为: preprocessor portscan: $HOME_NET 25 7 /var/log/portscan.log, 可为后续的检测和甄别提供证据. 当检测到可疑数据包的时候, 就可以判断 25 端口处于危险状态, 需要对 25 端口进行严密监控, 这时候马上启动一些针对性规则并设为高优先级别.

以标签远程栈溢出漏洞(CNNVD-200712-216)(CVE-2007-6435)为例, 检测规则如下:

alert tcp $EXTERNAL_NET any -> $HOME_NET 25 (msg:"SMTP Novell GroupWise client IMG SRC buffer overflow"; flow:to_server,established; content:"<IMG"; nocase; pcre:"/<img\s*src\s*\x3D[^>]{244}/i"; metadata:policy security-ips drop, service smtp; reference:bugtraq,26875; reference:cve,2007-6435; classtype:attempted-user; sid:13364; rev:2;)

经过 Snort&Bro 过滤和审查的流量才能进入服务器群, 而被规则集经过模式匹配拦截下来的数据包则被分离出其中的数据特征(如源 IP、数据包类型、敏感信息等). Snort&Bro 组件会将以上特征自动记录下来作为证据, 在向规则集添加相关新规则以防范后续的攻击行为的同时向管理员报警. Snort&Bro 工作流程如图 5 所示.

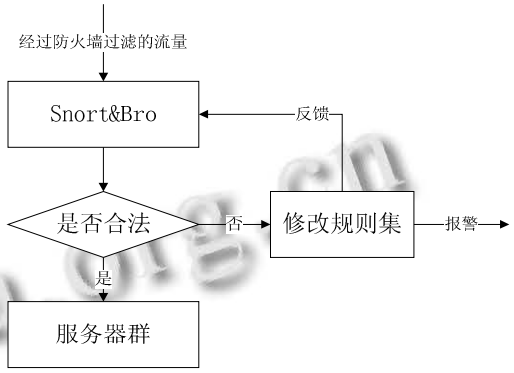

图 5 Snort&Bro 工作流程图

如果 Snort&Bro 依然无法没发现流量中异常行为的数据包, 当伪装的攻击行为试图对主机已被重点监控的关键区域(如权限文件、配置文件等)进行篡改的时候, HIDS 将是保护系统不被破坏的最后防线. 以非法软件提权为例, 由于一般上传的文件均只具有低权限, 而最终得到的 shell 是高权限的, 也即是说当某个高权限进程的的父进程是低权限的, 那么即可以认为是异常行为. 针对这种情况, 可设置如下规则:

```
{
"dsc":"Local Privilege Escalation",
"cache":{
    uid>0
},
"rule":{
    ip=cache.ip,
    ppid=cache.ip.pid
    uid=0
}
}
```

若监控区域已经被修改, HIDS 中的关键数据恢复模块将会被触发, 在修复的同时根据 Snort&Bro 日志及数据包备份确认攻击行为的来源及其他未被发现的危险操作, 将分析结果反馈至防火墙和 Snort&Bro, 同时锁定主机所有危险指令, 并向管理员报警. HIDS 联动流程如图 6 所示.

© 中国科学院软件研究所 http://www.c-s-a.org.cn

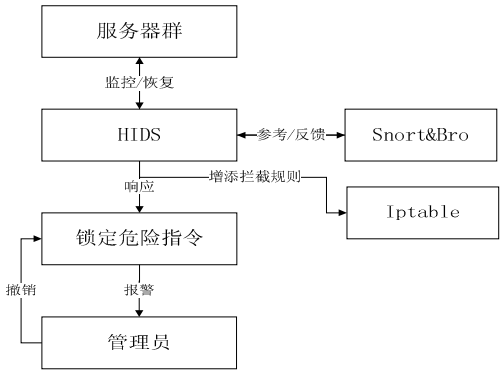

图 6 HIDS 联动流程图

5 小结

本文提出了一种策略分流的入侵防御及恢复系统架构设计, 双 NIDS 架构具有 Snort/Bro 双处理机制, 两套针对不同协议层次的规则库, 在 Snort 或 Bro 分析数据时, 各自有侧重分析的协议层次, 利用策略分流, 使得数据在 Snort&Bro 中最终会被全面分析, 有效降低漏报率和错报率. 而它们各自的负载量比单个规则库的 NIDS 分析数据的负载量要低, 占用 CPU 资源较少, 本系统比一般 NIDS 系统在资源(CPU/内存/磁盘/网络带宽)的占用比例上有所降低.

引入了 NIDS 与 HIDS 混合结构后, 可在现行成熟的入侵检测规则无法检测新型攻击手段的情况下, 利用 HIDS 的日志分析与关键数据完整性检测等手段, 及时发现入侵行为的发生, 快速恢复关键数据, 减低系统被破坏的风险. 然而由于 HIDS 基于操作系统的特性, 使得 HIDS 需要针对不同类型不同版本的操作系统研发不同的安全引擎, 大大增加了工作量和实现的难度.

参考文献

1 Yang H, Li T, Hu X, et al. A survey of artificial immune system based intrusion detection. Scientific World Journal, 2014, (2): 713–730.

2 Wang W, Zhang X, Pitsilis G. Abstracting audit data for lightweight intrusion detection. International Conference on Information Systems Security. Springer Berlin Heidelberg. 2010. 201–215.

3 Gupta S, Kumar P, Abraham A. A profile based network intrusion detection and prevention system for securing cloud environment. International Journal of Distributed Sensor Networks, 2013, (1): 8–10.

4 Albin E, Rowe NC. A realistic experimental comparison of the Suricata and Snort intrusion-detection systems. 2012 26th International Conference on Advanced Information Networking and Applications Workshops (WAINA). 2012. 122–127.

5 Morais A, Cavalli A. A distributed and collaborative intrusion detection architecture for wireless mesh networks. Mobile Networks & Applications, 2014, 19(1): 101–120.

6 Wahid MNA, Zulkarnain ZA. Applying packet generator for secure network environment. Journal of Computer Science, 2011, 7(5): 790–799.

7 Ghorbani AA, Wei L, Tavallaee M. Network Attacks. Advances in Information Security, 2010, 47:1–25.

8 Wang X, Liu H, Er D. HIDS: A multifunctional generator of hierarchical data streams. Acm Sigmis Database, 2009, 40(2): 29–36.

9 Soghoian C. Insecure flight: Broken boarding passes and ineffective terrorist watch lists. Social Science Electronic Publishing, 2007, 261: 5–21.

© 中国科学院软件研究所 http://www.c-s-a.org.cn