第30卷 第13期 计 算 机 工 程 2004年7月
Vol.30 №13 Computer Engineering July 2004
·安全技术· 文章编号：1000—3428(2004)13—0092—03 文献标识码：A 中图分类号：TP301.6

入侵检测系统中高效模式匹配算法的研究

杨 武[1]，方滨兴[2]，云晓春[1]，张宏莉[1]
（1. 哈尔滨工业大学计算机科学与技术学院，哈尔滨 150001；2. 国家计算机网络与信息安全管理中心，北京 100031）

摘 要：基于特征匹配的网络入侵检测系统的性能主要受模式匹配算法的影响。文章在对流行的网络入侵检测系统snort深入剖析的基础上，侧重研究如何利用各种高效模式匹配算法来优化snort的性能。重点介绍了一种基于BM思想的多模式匹配算法——SSPBM算法，结果表明采用SSPBM算法可以较大提高网络入侵检测系统的检测性能。

关键词：入侵检测；特征匹配；多模式匹配；网络安全

Study of Efficient Pattern Matching Algorithms in Intrusion Detection System

YANG Wu[1]，FANG Binxing[2]，YUN Xiaochun[1]，ZHANG Hongli[1]
(1. School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001；
2. National Computer Network and Information System Security Administration Center, Beijing 100031)

【Abstract】 The performance of a signature-matching-based network intrusion detection system (NIDS) is dominated by pattern matching algorithm. Based on the profile of the popular NIDS-snort, this paper studies how to utilize different efficient pattern matching algorithms to optimize the performance of snort. It mainly presents a multi-pattern matching algorithm based on the idea of BM algorithm-SSPBM algorithm. The result shows SSPBM algorithm can greatly improve the detecting performance of NIDS.

【Key words】 Intrusion detection；Signature matching；Multi-pattern matching；Network security

1 概述

随着Internet的飞速发展,网络规模日益扩大，网络应用趋向全球化，黑客入侵攻击事件也不断发生。传统的防火墙技术已经难以单独保障网络的安全，网络入侵检测系统（Network Intrusion Detection System，NIDS）作为一种积极主动的安全防护技术，已成为网络安全领域中研究的热点。

网络入侵检测系统通常采用被动监听的方式从关键网段捕获网络中传输的数据包，并利用各种检测分析方法从捕获的数据包中发现入侵证据。NIDS能够在不影响网络性能的情况下对网络进行监测，发现网络中的攻击事件。目前网络入侵检测系统中的检测分析方法主要分为两类：基于特征的检测（signature-based detection），又称为滥用检测（misuse detection），以及异常检测方法（anomaly detection）。基于异常的检测方法首先通过建立网络系统正常的行为或规范简档，然后与当前实际的网络行为进行比较，若出现较大的偏离则认为存在攻击行为，这种方法通常采用神经网络、概率统计和数据挖掘等技术；基于特征的检测方法通过将当前网络数据包与已知攻击及系统漏洞的特征库进行模式匹配来发现入侵。异常检测方法的最大缺点是学习需要时间，而且检测误报率高，很难适应大流量网络的实时检测要求。特征检测通常采用模式匹配的技术，易于实现而且检测精度高，因此目前大多数的网络入侵检测系统都采用基于模式匹配的特征检测方法。

网络流量的不断增大以及入侵特征库的不断更新，对基于特征匹配的NIDS的实时检测性能提出了挑战。当待分析网络数据流的产生速度超过了系统检测引擎的处理能力时，必然导致数据包来不及分析就被丢弃了。这些被丢弃的数据包很可能包含具有攻击特征的恶意数据，这样NIDS就会遗漏该攻击事件，产生漏报率（false negative）。一些攻击者往往会利用这一点，通过发送大量的无害数据包来使NIDS过载从而造成拒绝服务攻击（DoS攻击）或者逃避NIDS的检查（Evasion）。因此有必要提高基于特征匹配的NIDS在大流量网络环境下的检测性能。

基于特征匹配的网络入侵检测系统的性能主要受数据包和多个特征模式匹配过程的影响，即模式匹配算法是系统主要的性能瓶颈。本文在对传统的基于特征匹配的网络入侵检测系统snort进行了剖析，研究如何利用不同的模式匹配算法来最优化snort系统的检测性能。

2 传统NIDS——snort的性能评价

snort是一种流行的轻量级网络入侵检测系统[1]，它使用一种基于特征规则的检测引擎来对网络数据包进行内容模式匹配。snort在对网络数据包进行检查时，重复使用Boyer-Moore算法来将规则集中的每一条内容规则与数据包的有效载荷进行模式匹配。

为了量化分析snort系统在处理数据包时的性能瓶颈，我们对snort入侵检测系统进行了性能评测。系统实际的性能明显依赖于所使用的规则集以及所监测的流量特征。为此我们利用MIT Lincon Lab[2]提供的1998入侵检测训练数据集，选取1天的流量数据（容量大约160MB的Tcpdump文件）作测试。在该数据集上运行snort-1.8.3并选用完整的规则集，使用工具gprof来剖析snort系统的各个主要函数在运行过程中所占的时间百分比。运行代价排前5位的函数的剖析结果如表1所示。从表中可以看出，模式匹配过程是系统运行过程中代价最昂贵的部分，约占整个系统运行时间的30%。在其他的一些基

基金项目：国家“863”计划基金资助项目（2002AA142020）
作者简介：杨 武(1974—)，男，博士生，研究方向为网络安全、高性能计算；方滨兴、云晓春，教授、博导；张宏莉，副教授
收稿日期：2003-05-30 E-mail：yangwu@hit.edu.cn

于特征匹配的网络入侵检测系统中，也显示出模式匹配是系统主要的性能瓶颈。因此针对基于特征匹配的网络入侵检测系统的性能优化工作，主要是研究如何提高和改进模式匹配算法的效率。

表1 snort系统的性能剖析

函 数 名	功 能	运行时间百分比
mSearch	对内容选项进行模式匹配	30.32%
EvalOpts	对非内容规则选项进行检查	9.38%
EvalHeader	检查数据包头以分类	7.91%
CheckANDPatternMatch	检查模式匹配的边界	6.73%
CheckTcpFlags	检查TCP标志选项	6.43%

3 高效模式匹配算法分析

模式匹配过程可以抽象表示为这样一个问题：假定要在一个给定的文本串T中搜索一个模式串P的所有出现，其中P=$p_1p_2\cdots p_m$，$p_i\in\Sigma$，T=$t_1t_2\cdots t_n$，$t_i\in\Sigma$（Σ为字母表）。这是一个比较经典的单模式精确匹配问题，可以把这个问题扩展到基于模式集合的多模式匹配问题。

3.1 Boyer-Moore(BM)算法

BM算法是一种快速的单模式匹配算法。BM算法进行模式匹配时，沿着文本T从左到右移动模式P，却从右至左去比较字符以便在一个字符不匹配发生的时候可以将模式P移动更远的距离。BM算法主要依靠两张启发式移动表以便跳过文本T中无须匹配的字符： the good-suffix shift（优势跳转表）和the bad-character shift（劣势移动表）。

(1) 劣势移动表：假设模式P和文本T从左向右对齐，这时p_m对应着t_e并且p_m将要和t_e相比较。假设已经匹配上了j个字符，但是在模式p_{m-j}和文本t_{e-j}处发生了不匹配。如果字符t_{e-j}并不出现在P中，则可以沿着文本把模式安全地向右移动(m-j)个字符。更具体地说，BM算法计算一个数组 $\delta_1[1...|\Sigma|]$，$\delta_1[c]$计算了从P的最右端到文本T中字符c在P中最右出现的距离，即：$\delta_1[c]=\min\{q|P_{m-q}=c\}$。当发现一个不匹配时，可以把P向前移动$\delta_1[t_{e-j}]$-j个字符。

(2) 优势跳转表：通过在一个给定的不匹配发生之前检查已经匹配的字符串从而扩展了劣势移动表。假定P和T对齐并且p_m和t_e相比较，假设已经匹配上了j个字符但是在p_{m-j}和t_{e-j}发现了一个不匹配。如果知道子串$p_{m-j+1}p_m$在模式P中的最右出现到P的最右端的距离并且这个子串的前缀字符不是t_{e-j}，可以将P沿着文本T移动这个距离再进行比较。距离信息保存在数组元素$\delta_2[m-j]$中，计算数组$\delta_2[1\ldots m]$中每个数组元素值得到优势跳转表。

结合劣势移动表和优势跳转表，BM算法进行模式匹配的过程中，当匹配了j个字符而在文本的t_i处发生不匹配时，P向前移动max[$\delta_1[t_i]$-j, $\delta_2[m-j]$]个字符。因为δ_2的最小值是1，所以至少可以移动P的1个字符。

构造δ_1的时间复杂度是O(m+$|\Sigma|$)，构造δ_2的时间复杂度是O(m)，所以BM算法运行时总的时间复杂度通常被认为是O(n+m)。

3.2 Aho-Corasic(AC)算法

Aho和Corasick描述了一种应用有限状态自动机进行高效多模式匹配的算法。该算法利用多个模式串构建一个有限状态自动机并且通过使用该自动机来发现在对文本字符串的一次扫描过程中所有模式匹配的情况。多模式匹配问题可以抽象描述为：设P={$p_1,p_2,\cdots,p_k$}是一个模式集合，模式串p_i中的字母来自于一个固定的字母表Σ。多模匹配问题是发现P中的所有模式在文本T中的所有出现，T=$t_1t_2\cdots t_n$，$t_i\in\Sigma$。

AC算法的模式匹配的时间复杂度是O(n)，而且与模式的长度以及个数无关。无论模式p_i是否出现在T中，T中的每个字符都必须输入状态机中，所以无论是最好情况还是最坏情况，AC算法的运行时间总是O(n)。包括预处理时间在内，AC算法总共会花费O(M+n)的时间（其中M为所有模式的长度总和）。

对于多模式匹配来说，AC算法并行搜索模式集合要比应用BM算法逐一地搜索多个模式串要高效得多。然而基于有限自动机的算法必须逐一地查看文本的每个字符，这样会影响检测效率。BM算法能够利用跳转表跃过待搜索文本中的大段字符，从而提高搜索速度。将BM算法的启发式搜索技术应用到基于模式集合搜索的AC算法中，能够大大提高多模式匹配算法的效率。Commentz-Walter[3]首先提出了一种解决多模式匹配问题的算法，该算法结合了BM算法以及AC算法的特征。实践表明Commentz-Walter算法要比AC算法快很多。Baeza-Yates[4]也给出了一种组合BMH算法和AC算法的多模式匹配算法。AC_BM算法[5]根据一种前缀关键字树来计算劣势移动表和优势跳转表，从而可以跳跃式地并行搜索模式集合。

4 基于BM思想的多模匹配算法——SSPBM算法

基于BM算法跳跃式搜索的思想，本文介绍了一种新的高效多模式匹配算法——SSPBM算法。该算法主要分为两个阶段：预处理阶段和扫描阶段。

第一阶段预处理模式集合，在这个阶段建立3张表：移动表(SHIFT table)，后缀表(SUFFIX table)以及前缀表(PREFIX table)。移动表用来决定当文本被扫描的时候，文本中的多少个字符可以跳过检查。移动表和BM算法的移动表类似，但不完全相同。在构造移动表时，需要考虑的是一块大小为B的字符串的比较，而不是单个字符进行比较。设M为所有模式总的大小，M=k*m（m为模式集的最小长度，k为模式的个数），设c为字母表的大小。一个好的B的取值应该是$\log_c 2M$(本文B取值2)。假设现在移动表为每个可能的大小为B的字符串都包含一个入口，那么它的大小应为$|\Sigma|^B$（实际应用中常使用哈希技术将多个字符串映射到移动表中相同的入口，从而压缩了移动表的大小）。

当移动值为0时，要将文本后缀的B个字符子串和模式集中的一些模式后缀匹配。为了避免和每个模式都进行比较，使用哈希技术来最小化需要进行比较的模式数量。在构建移动表时，计算了B个字符的一个整数哈希值用作移动表的索引。使用相同的整数值作为另一表的索引，称为后缀表SUFFIX。后缀表的第i个入口SUFFIX[i]，包含一个索引指向最后B个字符的哈希值为i的模式列表，模式列表记为PATTERN_LIST。

当移动值为0时，文本需要遍历SUFFIX[i]值所指向的模式列表以进行模式后缀匹配。为了进一步降低模式搜索的范围，加速搜索进程，引入了另一张表，称为前缀表PREFIX。PREFIX[i]包含了所有模式头A个字符前缀的哈希值（本文A取值2）。

第二阶段进行扫描搜索，主要包括以下几步：

(1) 计算文本当前被扫描的B个字符的哈希值h(从t_{m-B+1}起)；

(2) 检查SHIFT[h]的值：如果>0，移动文本并转到步骤(1)；否则，转到步骤(3)；

(3) 计算文本前缀的哈希值（从当前的位置向左m个字符开始），称为text_prefix；

(4) 检查每一个满足SUFFIX [h]≤p<SUFFIX [h+1]的索引p，并且

当PREFIX[p]=text_prefix时，直接匹配文本与当前的模式（当前模式由PATTERN_LIST [p]给出）。

SSPBM算法整个扫描过程花费的时间是O(BN/m)（N为文本的大小，m为最小模式长度,p为模式的数量，M=m*p为模式的总长），当m>B时，SSPBM算法的平均时间复杂度呈亚线性。SSPBM总的时间复杂度是O(M)+ O(BN/m)=O(M+BN/m)。

5 试验结果与分析

利用MIT Lincon Lab[2]提供的1998入侵检测训练数据集，选取1天的流量数据（容量大约160MB的Tcpdump文件）作为测试数据集。Snort系统分别采用BM算法、AC算法以及SSPBM算法作为检测引擎，测试这3种情况下整个数据集的模式匹配时间与规则数的变化关系(规则集包括1 692条规则）。实验机配置：CPU PIII 1GB ，内存2GB，硬盘18GBSCSI，操作系统Linux 2.4.10。

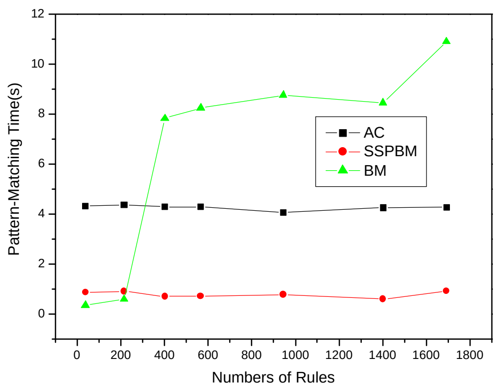

图1 BM、AC以及SSPBM算法在snort系统的模式匹配时间与规则数的关系

试验结果如图1所示，可以看出随着规则数量的增加，BM算法的匹配时间不断增大。SSPBM算法和AC算法的模式匹配时间基本不随规则数量而变化，这表明规则数量的不断增加不会影响这两种算法的匹配性能。在相同的规则数情况下（规则数超过300条），SSPBM算法和AC算法要比BM算法效率高很多，同时可以看出SSPBM算法要比AC算法的匹配效率高4~5倍。因此SSPBM算法是基于模式匹配的网络入侵检测系统优先选用的匹配算法。

6 结束语

网络应用的不断出现以及网络带宽的不断增加，需要提高网络入侵检测系统的处理性能以适应大流量网络环境的要求。本文对基于特征匹配的网络入侵检测系统snort的检测性能瓶颈进行了剖析，指出模式匹配过程是系统主要的性能瓶颈。在对传统的模式匹配算法分析的基础上提出一种新的基于BM思想的高效多模式匹配算法——SSPBM算法。将基于模式集并行搜索的AC算法和SSPBM算法分别应用到snort系统中，替换基于单模式重复执行的BM算法，可以极大提高系统的检测效率。

参考文献

1 Roesch M. Snort-lightweight Intrusion Detection for Network. Seattle, Washington, USA: Proceedings of LISA99: 13th System Administration Conference, 1999-11

2 Graf I, Lippmann R, Cunningham R, et al. Results of DARPA 1998 Offline Intrusion Detection Evaluation. http://ideval.ll.mit.edu/results-html-dir, 1998

3 Commentz-Walter B. A String Matching Algorithm Fast on the Average. Proc. 6th International Colloquium on Automata, Languages, and Programming, 1979

4 Baeza-Yates R A. Improved String Searching. Software-Practice and Experience 19, 1989

5 McAlerney J, Coit C, Staniford S. Toward Faster Pattern Matching for Intrusion Detection. DARPA Information Survivability Conference and Exposition, 2001

☆☆

（上接第40页）

4 模型的改进方向

为了进一步增强系统的功能，提高系统的安全性，可以对模型进行进一步改进：

(1) 用户管理设计：为降低安全风险，需要将用户分成几个层次，限制他们只能使用特定危险级别（轻度、中度和高度危险等级）的攻击工具；此外，还需要记录用户对服务器的登录和他们的行为。这些管理功能可以通过在服务器端数据库中添加用户管理库表和用户登录库来表实现。

(2) 通信协议的增强：采用用户权限设计后，相应需要对原有的通信协议作修改。此外，对于数据库的更新，可采用服务器端发送消息来升级客户端数据库的方式，这就需要添加新的ATP消息。

(3) 通信的加密设计：敏感的检测结果信息实施两层加密传输即内层是系统身份验证公钥加密算法以确保非用户无法窃取信息；外层是用户公钥加密算法以确保用户之间不会相互窃取信息，客户端保存私钥，服务器端的用户管理数据库表中保存用户的公钥。

(4) 与其他数据库的结合使用：加快完成分类描述体系的研究和部署工作，协调各个数据库的工作，提高工作效率。

5 结论

漏洞检测与主动防御系统模型是一种创新。它弥补了传统安全工具的缺陷，克服了类似产品的诸多缺点，比较好地实现了攻击工具的集成和控制，在新型安全防范平台中起着承上启下的重要作用。在实际测试中，这一系统模型体现出以下优点：攻击工具集成度高，智能化程度高；支持多用户操作，运行稳定快速安全；检测结果准确详尽，便于进行深入分析；攻击工具库组织合理，更新方便；能用于任何异构型网络，可以分布在网络中的任意位置；既可以单独工作，又可以与其他安全产品配合使用，达到漏洞扫描评估、服务检查、攻击性测试等目的；整个系统为将来的功能升级留有充足余地。今后，我们将在现有的基础上不断完善系统功能,并致力于在国内外市场的推广应用。

致谢

感谢我的同事冯涛、王磊、卞莹、牛永健为模型实现而付出的辛勤劳动。

参考文献

1 Disk J, Nomad S, Tremens I D. Phrack Magazine. http://www.phrack-dont-give-a-shit-about-dmca.org/show.php?p=56&a=6, 2000.5.1

2 Curry D, Debar H. Intrusion Detection Message Exchange Format Data Model and Extensible Markup Language (XML)Document Type Definition. draft-ietfidwg- idmef-xml-07.txt,January 30,2003, Expires: 2003: 07-31

3 Nessus Transfer Protocol White Paper. http://gd.tuwien.ac.at/infosys/security/nessuscvs/nessus/doc/ntp/

4 卞 莹, 张玉清, 郎 良等.主动防御攻击工具库设计.计算机工程与应用，(待发表)

5 Harrell J, Cannady J, Huggins D. A Model for an Attack Database System. http://www.scis.nova.edu/~cannady/afcea.pdf