入侵检测系统中BM模式匹配算法的改进

程玉青,梅登华
(华南理工大学 计算机科学与工程学院,广东 广州 510006)

摘 要:随着计算机网络的持续快速发展,网络安全问题日益突出,入侵检测技术也成为当前研究的热点。检测引擎作为入侵检测系统(IDS)的核心模块,基本上采用基于模式匹配的检测方法,模式匹配算法直接影响到系统的准确性和实时性能。文中介绍了目前最常用的BM模式匹配算法,以及其改进算法Boyer-Moore-Horspool(BMH)算法,在此基础上提出了另一种改进的BM算法。该算法减少了匹配次数,有效地加快了模式匹配的速度,提高了入侵检测的效率。

关键词:入侵检测;模式匹配;BM算法
中图分类号:TP393.08 **文献标识码**:A **文章编号**:1673-629X(2009)03-0172-03

Improvement of BM Algorithm for Pattern-Matching in Intrusion Detection System

CHENG Yu-qing, MEI Deng-hua
(School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China)

Abstract: With the rapid development of network, the online security especially the invading detection technique is becoming a hot topic. As the core module of intrusion detection system, detection engine generally uses the methods based on pattern matching. It directly influences the accuracy and real-time performance of the system. In this paper, the Boyer-Moore(BM) algorithm and its improvement algorithm, the Boyer-Moore-Horspool(BMH) algorithm, are described, then a new improvement algorithm is introduced. It can improve the match speed and the efficiency of the intrusion detection system.

Key words: intrusion detection; pattern matching; Boyer-Moore algorithm

0 引 言

随着网络的持续快速发展,信息安全和网络安全形势日趋严重和复杂化,入侵检测技术作为一种积极主动的安全防护技术,日益成为网络安全领域的研究热点。基于网络的入侵检测可以分为数据包的捕获、数据包的预处理以及对数据包进行攻击检测的过程。模式匹配是入侵检测系统所使用的基于攻击特征的网络数据包检测技术,也是IDS中一个最基本、最关键的技术。在实际的网络运行中,数据包的捕获速度与解释速度不能匹配,模式匹配速度的快慢直接影响到IDS的效率。

模式匹配问题可描述如下:对于给定的正文主串 $T = T_1,\cdots,T_n$(长度为 n)和模式串 $P = P_1,\cdots,P_m$(长度为 m),$(n >> m)$,要求在主串 T 中寻找等于模式串 P 的子串,如果在 T 中存在等于 P 的子串,则称匹配成功,函数值返回与 P 中第一个字符相等的字符在主串 T 中的序号,否则称为匹配失败,函数值返回为0。

收稿日期:2008-06-27
基金项目:国家自然科学基金与中国民用航空总局联合资助项目(66776816)
作者简介:程玉青(1978-),男,山东茌平人,硕士研究生,研究方向为计算机网络安全;梅登华,博士后,副教授,研究方向为计算机网络安全及可信计算。

1 现有模式匹配算法分析

1.1 BF算法和KMP算法

BF算法的思想是[1]:首先将 T_1 与 P_1 进行比较,若不同,就将 T_2 与 P_1 进行比较,…,直到 T 的某一个字符 T_i 和 P_1 相同,再将它们之后的字符进行比较,若也相同,则如此继续往下比较,当 T 的某一个字符 T_i 与 P 的字符 P_j 不同时,则 T 返回到本次开始字符的下一个字符,即 T_{i-j+2},P 返回到 P_1,继续开始下一趟的比较,重复上述过程。若 P 中的字符全部比较完,则说明匹配成功,返回本趟的起始位置,否则,匹配失败。

BF算法是较简单、直观的算法,但它可能产生不必要的回溯,因而模式匹配的效率、速率很低,其时间复杂度为 $O(m * n)$。KMP算法是由BF改进后不产生

回溯的一种算法[2]，其思想是：每当匹配过程中出现字符串比较不等时，不需回溯指针，而是利用已得到的“部分匹配”结果将模式向右“滑动”尽可能远的一段距离后，继续进行比较，从而提高匹配算法的效率。其时间复杂度为 $O(m+n)$，空间复杂度为 $O(m)$。

1.2 BM 算法

1977 年，Boyer 和 Moore 提出了一种新的字符串快速匹配算法—BM 算法[3]。BM 算法的关键是对给定的模式 $P=P_1,P_2,\cdots,P_m$，定义一个从字母到正整数的映射函数 dist：$c\{1,2,\cdots,m\}$，这里 $c\in\phi$（ϕ 为字符集），函数 dist 给出了文本中可能出现的字符在模式中的位置。

$$\mathrm{dist}(c)=\begin{cases} m;c \text{ 不在 } P \text{ 中出现或只出现在尾部} \\ m-j, j=\max\{j \mid P_j=c, 1\le j\le m-1\}; \\ \text{其余情况} \end{cases}$$

算法思想为：在匹配中模式串自左向右移动，而字符的比较却从右向左进行，当模式与正文在字符 T_i 处发生失配时，模式串向右移动 dist(T_i)，直至匹配失败或成功返回匹配位置。算法流程见图 1，移动过程见表 1。

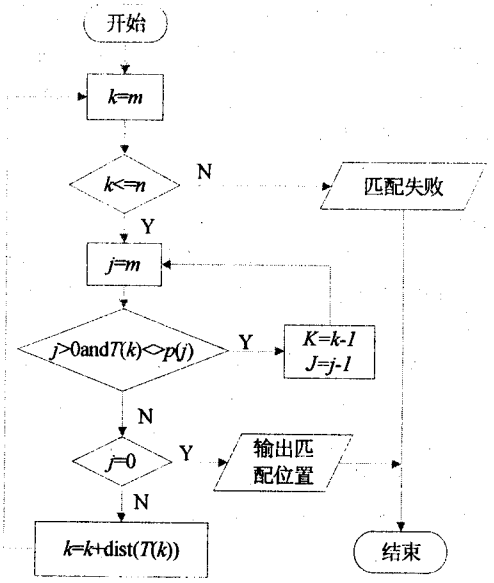

图 1　BM 算法流程图

表 1　BM 算法移动过程

substringjklgsmearchalgorithm
algorithm
algorithm
algorithm
algorithm
algorithm

1.3 BMH 算法

1980 年，Horspool 提出了对 BM 算法的一种改进算法——BMH 算法[4]。它首先比较文本指针所指字符和模式的最后一个字符，如相等再比较其余 $m-1$ 个字符。但无论文本中哪个字符造成了匹配失败，都将由文本中和模式最后一个字符对应位置的字符 T_i 来启发模式向右的滑动，滑动距离为 dist(T_i)。移动过程见表 2。

表 2　BMH 算法移动过程

substringjklgsmearchalgorithm
algorithm
algorithm
algorithm
algorithm

2 改进的 BM 算法

结合 BM 算法和 BMH 算法的优点，文献[5]提出在计算 dist 函数时，利用 T_i 的下一个字符 T_{i+1} 决定右移量。当下一个字符不在模式中出现时，它的右移量比 BMH 算法的右移量大，可使模式右移 $m+1$ 字符，而 BMH 算法即便是 T_i 不出现在模式串中，也最多移 m 距离。所以，通常情况下该算法比 BMH 算法快，但当 T_i 不在模式中出现，而 T_{i+1} 却出现在模式中时，该算法的效果就不如 BMH 算法。故当 BMH 算法移动距离大于该改进算法移动距离，且 T_{i+1} 在模式中唯一时，可以将模式直接跳至末端和 T_{i+1+m} 对齐，以实现最大右移量 $m+1$，基于上述思想，可以实现另一种改进的 BM 算法。算法流程见图 2，移动过程见表 3。

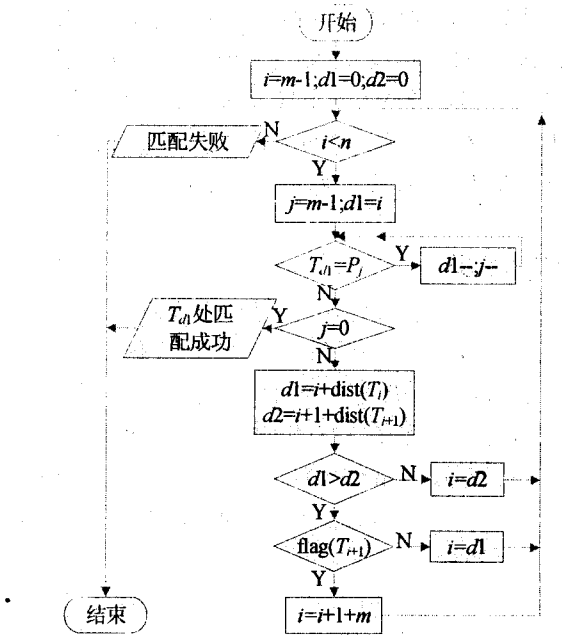

图 2　改进的 BM 算法流程图

运算过程是：首先计算出由字符 T_i 得到的右移距离 dist(T_i) 和模式串末端对应文本串字符 T_i 后一位置字符 T_{i+1} 得到的右移距离 dist(T_{i+1})，移动后模式末

端对应的字符下标分别是 $d_1 = i + \mathrm{dist}(T_i)$ 和 $d_2 = i + 1 + \mathrm{dist}(T_{i+1})$，再比较 d_1 和 d_2 的大小。如果 $d_1 < d_2$，将模式末端右移至 T_{d2} 字符处进行新一轮匹配；如果 $d_1 > d_2$，则须判断 T_{i+1} 在模式中出现的次数，若不出现或只出现一次，则直接右移 $m+1$ 个字符，若 T_{i+1} 在模式中出现多于一次，则将模式末端右移至 T_{d1} 字符处再进行下一轮匹配。用 flag 函数判断 T_{i+1} 在模式中出现的次数，定义如下：

$$\mathrm{flag}(c) = \begin{cases} 1; \text{字符 } c \text{ 在模式中只出现一次} \\ 0; \text{字符 } c \text{ 在模式中出现多于一次} \end{cases}$$

表 3 改进的 BM 算法移动过程

substringjklgsmearchalgorithm
algorithm
algorithm
algorithm

给定文本串：substringjklgmsearchalgorithm 和模式串：algorithm，分别用 BM 算法、BMH 算法和改进的 BM 算法进行匹配，匹配次数分别为 5 次、4 次、3 次，一次最大的移动量分别为 m、m、$m+1$ 个字符，可见该改进算法从匹配次数和最大移动量都取得了优势。匹配次数的多少和一次最大移动量以及最大移动量产生的概率有关，BM 算法产生最大位移量的情况是和模式串末端对齐的文本字符不在模式中，BMH 算法产生最大位移量的情况是和模式串末端对齐的文本字符不在模式中或该字符仅出现在模式末端，文中改进的 BM 算法产生最大位移量的情况是下一位字符不在模式串中或该字符在模式串中惟一且失配字符决定的移动量比该字符决定的移动量大两种情况，所以该算法产生最大位移量的概率比 BM 算法和 BMH 算法都要大。表 3 的移动过程也说明了这一点，在三次匹配中都实现了最大位移，极大提高了匹配效率。

3 实验结果

BM 算法的预处理阶段的空间复杂度是 $O(m+\sigma)$，σ 是与文本和模式相关的有限字符集的长度，查找阶段的最坏时间复杂度为 $O(mn)$，而其平均时间复杂度是亚线性的，最好情况下的性能是 $O(n/m)$[3]。BMH 算法的时间复杂度为 $O(n/m)$，改进算法的时间复杂度为 $O(n/m+1)$，可见比 BMH 算法的时间复杂度略优。选取 10M 的英文字母做文本，选取长度为 5、8、10、20、30 的英文字符串为模式串，分别用 BM、BMH 和改进算法进行匹配，结果见图 3，可见改进算法在匹配时间上优于原 BM 算法。

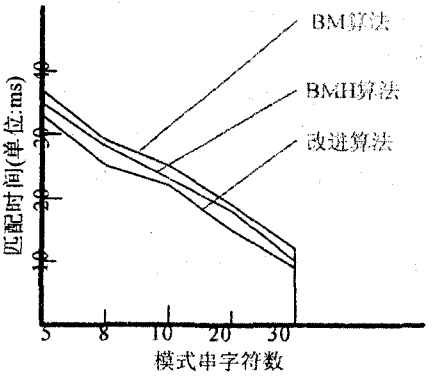

图 3 BM、BMH、改进算法比较

4 结束语

网络新应用的不断出现以及网络带宽的不断增加，使得目前的网络入侵检测系统的处理性能开始不能适应大流量网络环境的要求，这就迫切需要提高 IDS 的处理性能[6]。

文中对模式匹配算法 BH 和 BMH 作了简要的分析，并提出了一种改进算法，从理论分析和实验结果看，该算法减少了匹配次数，缩短了匹配时间，将其应用到入侵检测系统的检测引擎中，可以提高系统检测效率，改善系统性能。

参考文献：

[1] 彭 波. 数据结构[M]. 北京：清华大学出版社，2004.
[2] 傅清祥，黄晓东. 算法与数据结构[M]. 北京：电子工业出版社，2001.
[3] Boyer R S, Moore J S. A fast string searching algorithm[J]. Communications of the ACM, 1977, 20(10): 762－772.
[4] Navarro G, Raffinot M. 柔性字符串匹配[M]. 北京：电子工业出版社，2007: 19－23.
[5] Daniel M S. A very fast substring search algorithm[J]. Communications of the ACM, 1990, 33(8): 132－142.
[6] 伊 静，刘培玉. 入侵检测中模式匹配算法的研究[J]. 计算机应用与软件，2005, 22(1): 112－114.

(上接第 171 页)

[3] 何宝宏. IP 虚拟专用网技术[M]. 北京：人民邮电出版社，2002.
[4] 徐家臻，陈莘萌. 基于 IPSec 与基于 SSL 的 VPN 的比较与分析[J]. 计算机工程与设计，2004(4): 586－587.
[5] Bollapragada V, Khalid M, Wainner S. IPSec VPN 设计[M]. 袁国忠译. 北京：人民邮电出版社，2006.
[6] Metz C Y. IP 交换技术协议与体系结构[M]. 吴 靖等译. 北京：机械工业出版社，1999.