改进BM算法及其在网络入侵检测中的应用

孙文静[1,2] 钱 华[3]
(南京理工大学 南京 210094)[1] (南京审计学院 南京 210029)[2]
(南京联迪信息系统有限公司 南京 210019)[3]

摘 要 传统BM算法存在一些无用的比较,影响了字符串的匹配速度,降低了入侵检测效率。为此,提出一种改进BM算法,并将其用于网络入侵检测系统的检测引擎中。实验结果表明,较采用BM算法的Snort检测器,改进BM算法构建的网络入侵检测系统可有效降低误报率和漏报率,提高入侵检测率与时间利用率。显然,这对提升网络入侵检测系统的整体能力非常有用。

关键词 入侵检测,改进BM算法,检测效率,误报率与漏报率
中图法分类号 TP311 **文献标识码** A

Improved BM Algorithm and Its Application in Network Intrusion Detection

SUN Wen-jing[1,2] QIAN Hua[3]
(Nanjing University of Science & Technology, Nanjing 210094, China)[1]
(Nanjing Audit University, Nanjing 210029, China)[2]
(Nanjing Liandi Information Systems Co. Ltd., Nanjing 210019, China)[3]

Abstract The traditional BM algorithm has some useless comparison, affecting the string matching speed and reducing the efficiency of intrusion detection. Therefore, this paper proposed an improved BM algorithm, applied it to the engine of network intrusion detection system. Experimental results show that, compared with BM algorithm which employs Snort detection, a network intrusion detection system constructed by improved BM algorithm can effectively reduce the false positive rate and false negative rate, and improve intrusion detection rate and time utilization. Obviously, this network intrusion detection system is very useful for enhancing the overall capacity.

Keywords Intrusion detection, Improved BM algorithm, Detection efficiency, False positive rate and false negative rate

模式匹配是入侵检测系统中最重要,也是最常用的一种信号分析方法。其基本任务是把存放在入侵检测系统规则集中的已知入侵规则(模式)与系统获取的网络包或者重构的TCP流中的文本进行匹配。如果匹配成功,则判断发生了网络入侵。

随着网络攻击技术的发展和攻击手段的多样化,描述攻击行为的特征数目呈指数上升,检测算法的效率已成为误用检测技术的瓶颈,直接影响系统的实时性能。因而,如何改进字符串匹配的搜索算法、提高检测速度,是目前IDS研究的重点之一。

BM(Boyer-Moore)算法由Boyer和Moore提出[1],是一种有效的字符串匹配算法,较多地应用于分布式网络入侵检测系统中。

应用中人们发现,BM算法本身有许多意义极小甚至无用的比较,影响了字符串匹配效率。为此,本文提出一种改进BM算法。实验表明,将其用于基于Snort的分布式入侵协同检测系统的检测引擎匹配算法,可有效减少匹配时间,提高模式匹配的速度。

1 BM算法

BM算法的具体匹配是从P的右端向左进行,进行中考察比较并考虑文本中可能出现的字符在模式中的位置。该算法的基本思想是:

(1)匹配自右向左进行;

(2)若匹配失败发生在$P_j \neq T_i$,且T_i不出现在模式T中,则将模式右移直到P_1位于匹配失败位T_i的右边第一位(即T_{i+1}位),若T_1在P中有若干地方出现,则应选择$j=\max\{K|P_k=T_i\}$;

(3)若模式P后面K位和文本T中一致的部分,有一部分在T中其他地方出现,则可以将T向右移动,直接使这部分对齐,且要求一致部分尽可能的大。

以T="This is a test of the Boyer Moore algorithm.",P="algorithm"为例。匹配从'*'处开始。

algorithm←pattern

876543210←字符的移动值

首先比较的字符是模式串中最右边的字符'm'。

到稿日期:2013-03-11 返修日期:2013-06-09 本文受国家发改委发改办[2012]3179号下一代互联网络扫描与补丁管理系统产业化项目基金资助。

孙文静 女,博士生,讲师,主要研究方向为计算机网络;钱 华(1983—),女,硕士,主要研究方向为信息技术与安全。

*

Text　This is a test of the Boyer Moore algorithm.

Pattern　algorithm

*

在文本中正在比较的是与'm'相对应的字符'a',它在模式中也出现过,其移动值是8,所以将模式直接向右移动8格,进行下一次比较。这种移动通常称为"GOOD SUFFIX"移动。

*

Text　This is a test of the Boyer Moore algorithm.

Pattern　algorithm

*

这次在文本中进行比较的字符'f'没有在模式中出现过,这就需要另一种策略来处理这种移动。BM算法的设计是如果出现模式中的字符与文本中没有在模式中出现过的字符相比较的情况,模式可以完全移过文本中出现的不匹配的字符。这种移动被称为"BAD CHARACTER"移动。各字符的移动值如下:

algorithm←pattern

123456789←字符的移动值

*

Text　This is a test of the Boyer Moore algorithm.

Pattern　algorithm

*

因为是'm'与'f'失配,所以这次移动使用'm'的移动值9。移动后,下一个比较的字符是'e',它同样不在模式中。所以这次移动还是"BAD CHARACTER"移动,其移动值还是9。

*

Text　This is a test of the Boyer Moore algorithm.

Pattern　algorithm

*

这次比较的'a'仍是"GOOD SUFFIX"移动,移动值是8。

*

Text　This is a test of the Boyer Moore algorithm.

Pattern　algorithm

*

最后,反相比较循环通过比较模式中的所有字符与对应位置文本中的字符,在文本的末端发现一个成功的匹配。

2　BM算法的改进

这里提出BM算法的改进宗旨是减少不必要的比较次数。具体改进规则为:将模式串与文本串对齐进行比较,在发生字符比较不匹配后根据文本串所对应的模式串最后位置的下一个字符信息来确定偏移量,即不匹配时滑动位数的最大值增为$m+1$。而传统BM算法中此时最大的偏移量为m。

算法实现采用坏字符后缀规则[2,3],即只计算skip数组的值。在计算skip数组时考虑下一个字符的情况,即利用下一个字符决定右移量。算法伪代码如下:

```
void N_BM(char * p,char * t,int ship[])
{
for(i=0;i<maxchar;i++)
skip[i]=m+1;
for(i=0;i<m;i++)
skip[p[i]]=m-i;
k=m-1;
while(k<n)
{
j=m-1;
while((j>==0)&&(p[j]==t[k+j-(m-1)]))
{
j--;
}
if(j==0)
return t[k];
else k=k+ship;
}
}
```

根据以上算法,以在"perfected searching example"搜索"search"字符串为例。

文本串:perfected searching example

模式串:search

第1步　对齐文本串和模式串,从右向左比较。第一次匹配从文本串第6个字符处开始比较,结果发现不匹配,于是需要把模式串往后移动。改进后的算法是根据紧跟在当前子串之后的那个字符't'获得位移量,而't'没有出现在模式串"search"中,则按照skip[i]=m+1可以直接跳过6+1的距离,从'c'之后的那个字符开始作下步的比较。

文本串:perfected searching example

模式串:search

第2步　从'a'开始比较,比较的结果,第一个字符又不匹配,再看其后的字符'c',它在子串中出现在第4位,按照skip[p[i]]=m-i,把子串向右移动6-3位,使得两个'r'对齐,因此,整个匹配过程移动了两次模式串就找到了匹配位置。

BM算法在查找阶段的时间复杂度为$O(m*n)$,而改进后的算法与BM算法的结构基本相同,但在查找阶段的时间复杂度为$O(m(n-m))$,它在原有模式串移动的基础上加大了在匹配失败后向后跳跃的幅度,从而减少了比较次数,提高了运算效率。

图1为本文实验结果。实验选用开放源代码的Snort 2.0进行评测[4]。选择的规则集是Snort 2.0标准规则集的子集,选取了769条要求对数据包内容进行检查的规则备用,测试采用的数据源为10MB大小的文本,模式数量分别为50,100,150,200,250。

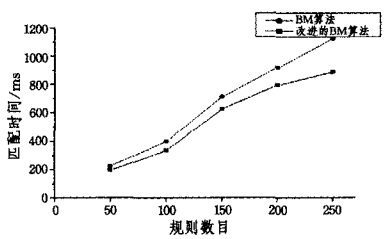

图1　匹配时间比较

由图1可以看出,改进后的BM算法有效地减少了匹配的时间,提高了模式匹配的速度,且提高速度与规则集的容积大小成正比。

3 改进 BM 算法在网络入侵检测中的应用

图 2 为本文实验搭建的网络入侵检测系统。各模块主要功能如下：

(1)数据包捕获模块：主要负责监听网络接口卡，抓取流过的数据包，并过滤出系统需要的数据。

(2)预处理模块：主要实现 IP 包分片重组、TCP 流重组等功能。

(3)协议分析模块：对预处理模块采集的数据进行协议分析，将其还原为各种不同协议(IMCP、TCP、UDP、HTTP 等)的数据包，再根据不同协议的具体内容对其进行分析，检测入侵行为，并将这些数据包交给规则匹配模块进行处理。

(4)规则匹配模块：对协议分析模块处理后的数据根据不同的协议调用不同的规则库进行模式匹配，判断是否有入侵行为发生。

(5)响应模块：对入侵行为作出响应，如向控制台输入报警信息。

(6)规则库：存储入侵特征规则，每条规则表示一种入侵行为，在进行协议分析和模式匹配时要根据该库中的规则进行入侵检测。

(7)规则解析模块：对规则库中的规则进行解析并读入内存。

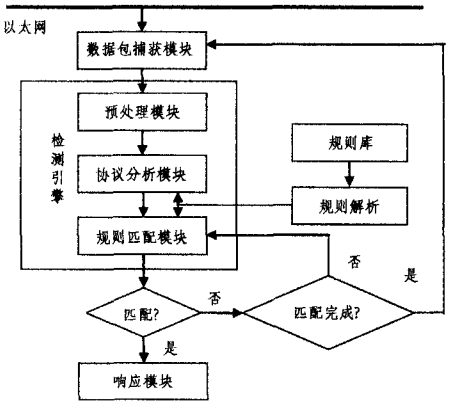

图 2 网络入侵检测模块

本文的入侵检测系统模型是基于 Snort 来设计与实现的，提出的改进 BM 算法用于检测引擎中。

测试比较以 Snort 做参照，让 IDS 对进入到受保护系统的数据进行检测，以确定检测系统能否发现其中的入侵。实验中利用 MIT Lincoln Lab[5] 提供的 1999DRAPA 入侵检测测试数据集，该数据集是目前最常用的测试 IDS 的数据包之一。

NIDS 在一台以 Windows XP 作为 OS 的服务器上，网卡为混杂模式，把用于存储和检测结果的数据库和日志也放在该主机上，采用 MySQL 实现，数据库管理界面使用 VC 实现，管理员可以通过其查看和处理检测结果和日志记录。

实验分别对 Snort 系统和 NIDS 进行测试。Snort 算法本身就是采用 BM 算法。

测试结果如表 1－表 3 所列。其中：

误报率＝(检测前是正常包而检测后认为是异常包的个数/检测前正常包的总数)×100%

漏报率＝(检测前是异常包而检测后认为是正常包的个数/检测前异常包的总数)×100%

检测率＝((检测前正常检测后也正常的数据包＋检测前异常检测后也异常的数据包)/包总数)×100%

误报率降低率＝((采用 BM 算法的误报率－采用改进 BM 算法的误报率)/采用 BM 算法的误报率)×100%

检测率提高率＝((采用改进 BM 算法的检测率－采用 BM 算法的检测率)/采用 BM 算法的检测率)×100%

漏报率降低率＝((采用 BM 算法的漏报率－采用改进 BM 算法的漏报率)/采用 BM 算法的漏报率)×100%

检测时间减少率＝((采用 BM 算法的检测时间－采用改进 BM 算法的检测时间)/采用 BM 算法的检测时间)×100%

表 1 采用 BM 算法的 Snort 检测结果

包总数	2808	3468	4005
误报率	5.70%	7.52%	8.68%
漏报率	12.56%	11.39%	8.16%
检测率	87.84%	84.25%	83.13%
检测时间(s)	43.56	53.89	71.34

表 2 采用改进 BM 算法的 NIDS 检测结果

包总数	2808	3468	4005
误报率	4.27%	6.45%	7.96%
漏报率	9.49%	10.12%	7.08%
检测率	89.78%	87.58%	88.34%
检测时间(s)	37.75	47.92	57.42

表 3 检测结果比较表

包总数	误报率降低率(%)	漏报率降低率(%)	检测率提高率(%)	检测时间减少率(%)
2808	25.09	24.44	2.21	13.34
3468	14.23	11.15	3.95	63.58
4005	8.29	12.54	5.90	19.51
平均改进率	15.87	16.04	4.02	32.14

由表 3 可见，相对于采用 BM 算法的 Snort 检测器，采用改进 BM 算法的 NIDS 系统 3 组实验数据所对应的误报率降低率、漏报率降低率、检测率提高率、检测时间减少率分别在 8.29%～25.09%、11.15%～24.44%、2.21%～5.90%、13.34%～63.58 之间，检测效率与检测效果均优于传统 BM 算法。显然，这对提升网络入侵检测能力具有积极意义。

结束语 传统 BM 算法存在一些无用的比较，影响了字符串匹配速度，从而降低了入侵检测效率。为此，本文提出一种改进 BM 算法，其具体改进规则为：将模式串与文本串对齐进行比较，在发生字符比较不匹配后根据文本串所对应的模式串最后位置的下一个字符位置信息来确定偏移量，即不匹配时滑动位数的最大值增为 $m+1$。而传统 BM 算法中此时最大的偏移量为 m。实验表明，采用改进 BM 算法的 NIDS 系统较采用 BM 算法的 Snort 检测器，可有效降低误报率和漏报率，提高入侵检测率与时间利用率，这对提升相应的网络入侵检测系统的整体能力非常有用。

参 考 文 献

[1] Boyer R S, Moore J S. A fast string searching algorithm [J]. Communications of the ACM, 1977, 20(10): 762-772

[2] 李洋，王康，谢萍. BM 模式匹配改进算法[J]. 计算机应用研究，2004, 21(4): 58-59

[3] 杨薇薇，廖翔. 一种改进的 BM 模式匹配算法[J]. 计算机应用，2006, 26(2): 318-319

[4] Roesch M. Snort: Lightweight Intrusion Detection for Networks [C]//LISA' 99 Proceedings of the 13th USENIX Conference on System Administration. 1999: 229-238

[5] http://www.ll.mit.edu/IST/

[6] 魏旻，王一帆，李玉，等. 基于 WIA－PA 网络的周界入侵检测系统设计与实现[J]. 重庆邮电大学学报：自然科学版，2013, 25(2): 148-153