基于 Snort 的入侵检测系统性能优化

韩忠秋
（四川大学 计算机学院，四川 成都 610065）

摘 要：通过对 Snort 的规则匹配方式和模式匹配算法进行分析，为了提高基于 Snort 的入侵检测系统检测效率，提出了在规则匹配过程中充分利用处理函数的参数之间的关系，从而动态减少无效匹配次数，在模式匹配阶段采用改进的模式匹配算法提高匹配速度，从根本上优化了入侵检测系统的检测性能。

关键词：入侵检测系统；规则匹配；模式匹配；性能优化

The Performance Optimization of Snort-based Intrusion Detection System

HAN Zhong-qiu
(Computer Department, Sichuan University, Chengdu,Sichuan 610065,China)

Abstract: Snort is a mature open source code network invasion detection system. The rule matching mode and pattern matching arithmetic have been analyzed in this paper. For improve the speed of the snort-based intrusion detection, we utilized relationship between parameters, significantly reduced invalid rules in the running time. By using the improved pattern matching arithmetic to increase the matching speed in the pattern matching phase, the detection performance was optimized ultimately.

Key words: intrusion detection system; rule matching; pattern matching; performance optimization

随着计算机网络和信息化技术的发展，越来越多的电子商务系统、企业信息化平台建筑在互联网上，人们在享受网络带来的资源共享及信息交流的同时，也不得不面对黑客和网络入侵者给网络安全带来的威胁。所以，网络中计算机系统的安全问题日益成为人们关注的热点[1]。入侵检测技术作为一种积极主动的防御手段，已成为当今网络安全体系的重要组成部分。而 Snort 作为一种开放源代码的入侵检测系统已经得到广泛的应用。它采用的是基于规则的网络信息搜索机制，通过对数据包内容进行模式匹配来检测多种不同的入侵行为和探测活动。但是，由于 Snort 只按照规则归类的方法进行规则匹配，而且对数据包内容的模式匹配采用的是 Boyer-Moore（简称 BM）算法，使得其对规则匹配的重复次数过多，模式匹配效率不高，影响了检测系统的整体性能。为此，本文试图将规则匹配策略并行化，对于各条件既按照规则本身进行分类，也按照条件分类；同时引入横向淘汰方法，如果一个规则中的一个条件为假，那么就不匹配此规则中的其他条件，在模式匹配过程中采用一种改进的匹配算法，从而能有效地减少数据包的匹配时间，提高了基于 Snort 的入侵检测系统的检测效率。

1 基于 Snort 的入侵检测系统

基于 Snort 的入侵检测系统主要包括数据包捕获、预处理、数据检测和报警日志四个部分。系统体系结构如图 1 所示。

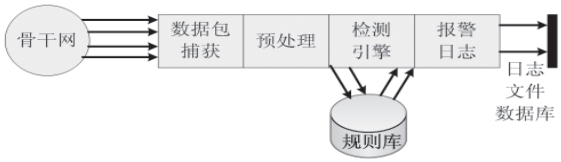

图 1 基于 Snort 的入侵检测系统体系结构

系统首先从网卡上捕获数据包并送交给数据包解码器，由解码器对各种协议栈上的数据包进行解析、预处理，然后把处理结果提交给检测引擎进行规则匹配。如果有匹配的数据包则报警，与日志子系统进行响应并记录日志。其中，数据包嗅探和解码的全部工作都是围绕着网络协议栈中的各层协议定义展开的，包括了数据链路层协议和 TCP/IP 等层协议的定义。系统中的每个子例程使用事先定义好的数据结构，从原始网络流量数据中解析出协议信息，这些子例程按照网络协议栈自下而上的顺序调用，从数据链路层到传输层，直到应用层协议结束。在协议解析的过程中，强调执行的速度，所完成的主要功能是形成协议处理链表，解析出 IP 地址、端口号和数据包负载（其中包含了与攻击相关的内容）等信息，以便检测引擎子系统执行下一步的分析工作。

检测引擎是入侵检测部分的核心模块。它主要负责按照启动时加载的规则，对每个数据包进行分析。它的作用是探测数据包中是否包含着入侵行为。基于 Snort 的入侵检测系统将所有已知的攻击以规则的形式存放在规则库中，每一条规则由规则头和规则选项两部分组成。规则头对应

于规则树结点 RTN(Rule Tree Node)，包含动作、协议、源（目的）地址和源（目的）端口及数据流向这样一些公共信息，Snort 把这些具有相同条件的规则链接到一个集合中，用 RTN 结构来描述；规则选项对应于规则选项结点 OTN(Optional Tree Node)，包含一些特定的检测标志、报警信息、匹配内容等条件，每个选项的匹配子函数（插件）放到 FUNC 链表中[2]。其结构如图 2 所示。

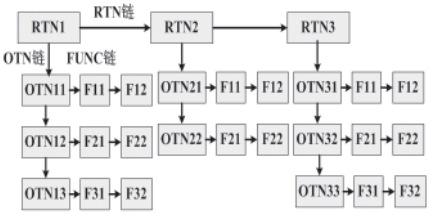

图 2 基于 Snort 的入侵检测系统规则结构图

规则头和规则选项都有自己的匹配子函数，在匹配方式上可以分为规则头匹配和规则选项匹配。在进行规则匹配时首先根据该数据报的 IP 协议决定与哪个规则树进行匹配；然后与 RTN 结点依次进行匹配，当与某个规则头相匹配时，接着向下与 OTN 结点进行匹配。每个 OTN 结点都包含了一条规则的全部选项，它包含的一组函数指针就是用来实现对这些条件的匹配操作。当检测得知数据报与某个 OTN 结点的所有条件相符合时，即判断此数据报为攻击报文。

日志／警报子系统在运行时用户可以通过控制台进行选择。系统采用不同的方式记录工作日志，用户可以通过查询和审计日志信息来了解当前系统的工作情况，同时用户还可以选择系统工作在报警模式下，从而当系统检测到攻击时会及时向用户报警并输出报警信息，使用户能及时地做出响应。

2 系统性能优化

2.1 规则匹配效率的改进

为提高规则匹配的速度，对于规则头和规则选项中的条件参数，打破只按照规则归类的方法，而是既按照规则归类也按照匹配子函数归类的方法。这样在每个匹配子函数中都附有一个不同规则相同条件对应的参数集合。在进行规则匹配时先按照匹配子函数的顺序，然后在每个函数内只是针对不同规则中的条件参数进行处理，而不是一遍又一遍地调用相同的函数，这样可以节省重复调用函数的系统时间。

首先，在原版基于 Snort 的入侵检测系统中规则头链表的基础上，把各个匹配函数的条件参数组成一个新的十字链表，参数链的内容包括：地址参数链和端口参数链，并对地址和端口进行分类，处理函数只处理某类中的第 1 个参数，其他的参数都直接使用这个参数的处理结果，改进的规则头链表如图 3 所示。

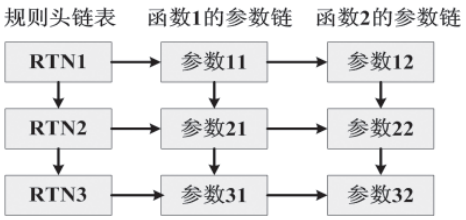

图 3 改进的规则头链表结构

对规则头进行匹配时，先找到规则树，然后按照规则头条件的次序顺序处理，而在具体的处理函数中对参数列表中的每个参数进行处理，对于具有相同分类的参数，只引用上次的处理结果就可以了，即若上次处理的结果为真，便继续下一个参数，否则通知其他处理函数不必再处理此规则头的其他条件。

其次，采用同样的原理对规则选项链表进行改进，在选项条件按规则分类的基础上，再按照插件进行分类构建十字链表结构。首先在规则选项节点的结构中增加一个指针，用于指向本规则对应的选项参数链表，具体的参数结构除有参数内容外还要有横向指针用于指向本规则中的下一项参数和竖向指针用于指向本插件的下一个参数，当遇到一个选项关键字时，找到对应插件后，构建此条件的参数结构并同时放入到按规则分类的横向参数列表中和按照插件分类的竖向参数列表中。最后对于同一个匹配子函数下的参数链表可以按照某种方式进行分类或排序。改进的规则选项结构如图 4 所示。

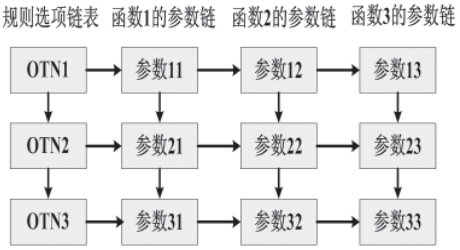

图 4 改进的规则选项链表结构

在进行规则选项匹配时，首先按照插件的顺序依次处理每个插件，然后在具体的插件中对参数列表中的每个参数进行处理。如果结果为真，继续下一个参数，否则通知其他插件不用处理此规则的其他条件了。对于规则中同时含有重型选项条件（需较多检测时间）和轻型选项条件（需较少的检测时间），一般把轻型插件放到重型之前处理，若轻型插件检测出此条件为假，后面的重型条件就没有必要处理，显然这样会减少规则匹配的时间。

2.2 模式匹配算法的改进

BM 算法是 Boyer 和 Moore 在 1977 年提出来的，其特点是考虑到在匹配比较的过程中，不少情形是前面的许多字符都匹配而最后的若干个字符不匹配，从而采用自右至左的方式扫描匹配模式。这样，一旦发现正文中出现模式中没有的字符时，就可以将匹配模式大幅度地滑过一段距离。改进的 BM 算法通过建立移位表、哈希表和前缀表

三张表来加速模式匹配速度。其中，移位表和BM算法中的类似，但和BM算法的处理有所不同。改进的模式匹配算法根据最后二个（甚至三个）字符来计算移位距离，当移位距离大于0的时候，可以移动位置继续扫描；但是当匹配发生时，我们需要知道到底是哪一个模式发生了匹配。为了避免逐个地和每个模式串进行比较，需要使用哈希技术来进行加速[3,4]。

为此，算法第一阶段的主要任务就是构造三张表：移位表、哈希表和前缀表。首先，我们假设所有的模式长度都相同，通过计算模式串的最短长度min，在进行匹配的过程中就只考虑每个模式的前min个字符。算法并不针对文本中的每个字符进行比较，而是将它们看作是长度为B（通常取值为2或3）的块，假定所有模式的总长度为M，那么M=mnt*min，其中mnt为模式的个数，移位表的作用与BM 算法中一样，只是它是根据最后的B个字符决定移位的位数而不是根据最后一个字符。在移位表中，每个长度为B的字符串都被映射成一个整数，作为移位表的索引。移位表中的值给出了它在扫描文本的过程中可以向前移动的距离。移位表可以根据需要进行压缩。当匹配发生时，我们需要知道到底是哪一个模式发生了匹配。为了避免逐个地与每个模式串进行比较，需要使用哈希表进行加速。通过计算B个字符到一个整数的映射，将其用作移位表的索引。然后将同样的数据也用于哈希表中。哈希表的第i个入口记作hash[i]，它包含了一个指向最后B位字符的哈希值为i的模式列表的指针。当发生多模式匹配时，可以根据该指针快速定位到匹配的模式串列表中，提高算法的效率。

在匹配过程中，有些后缀不只是在文本中经常出现，而且还非常有可能在几个模式中同时出现，这会带来哈希表冲突的问题，即具有相同后缀的所有模式被映射到哈希表的相同入口。因此，不得不分别检查具有这种后缀的所有模式，看它们是否与文本匹配。为了加速这个处理，引入了另外一张表格，叫做前缀表。当发现移位表哈希值为0，并需要利用哈希表确定是否有匹配的时候，可以先检查前缀表中的值与文本中相应的前缀（通过左移动min-F个字符，F是前缀表中字符块的大小）是否相符，过滤掉大量的模式。但是，只有当移位表的值经常为0，也就是规则条数很多，且具有很高的冲突可能性的时候，使用前缀表才有意义。

算法的第二阶段主要是进行实际匹配操作，主要有以下循环步骤：(1)根据文本中当前的B个字符，计算哈希值；(2)检查SHIFT[h]的值，如果该值大于0，移动文本，转向1，否则转向3；(3)计算文本前缀的哈希值，（从第min个字符开始，到当前位置的左侧），记为text_Prefix；(4)检查每个p，其中hash[h]≤p≤hash[h+1]，PREFIX[p]=test_prefix是否成立。如果它们相等，使用真正的模式（由PAT_PIONT[p]得到）对文本直接进行检查。

3 结束语

入侵检测作为一种重要的信息安全技术在计算机网络可靠运行过程中起到了非常重要的保证作用，但是，目前入侵检测系统的应用效果并不理想；特别是高速网络环境下入侵监测系统应该具有更强的处理能力和更快的响应速度[5~7]。本文通过对基于Snort的入侵检测系统进行规则的RTN和OTN结构进行十字链表的改进，同时采用一种更高效的模式匹配算法，提高系统检测效率，从而使其能更好地适应高速网络环境。

参考文献:

[1]李涛.网络安全概论[M].电子工业出版社,2004.8,p474-490.

[2]谷晓刚，江荣安，赵铭伟. Snort的高效规则匹配算法[J].计算机工程,06.9, Vol.32No.18.

[3]袁晖. 基于snort的入侵检测系统安全性研究[J].计算机科学,08, Vol.35No.4 .

[4]Zhuowei Li, Das A, Jianying Zhou. Theoretical basis for intrusion detection Information Assurance Workshop, 2005. IAW '05. Proceedings from the Sixth Annual IEEE SMC 15-17 June 2005 Page(s):184 - 192 Digital Object Identifier 10.1109/IAW.2005.1495951.

[5]Garuba M, Chunmei Liu, Fraites D. Intrusion Techniques: Comparative Study of Network Intrusion Detection Systems Information Technology: New Generations, 2008. ITNG 2008. Fifth International Conference on 7-9 April 2008 Page(s):592 - 598 Digital Object Identifier 10.1109/ITNG.2008.231.

[6]Youchan Zhu, Ying Zheng. Research on Intrusion Detection System Based on Pattern Recognition Networked Computing and Advanced Information Management, 2008. NCM '08. Fourth International Conference on Volume 1, 2-4 Sept. 2008 Page(s):609 - 612 Digital Object Identifier 10.1109/NCM.2008.13.

[7]Seungyong Yoon, Byoungkoo Kim, Jintae Oh. High-Performance Stateful Intrusion Detection System Computational Intelligence and Security, 2006 International Conference on Volume 1, Nov. 2006 Page(s):574 - 579 Digital Object Identifier 10.1109/ICCIAS.2006.294201.

作者简介：**韩忠秋**(1985-)，男，硕士研究生，研究方向：网络安全技术及应用。

收稿日期：2008-11-24