基于行为分布的 DDoS 攻击检测方法

赵 英 黄文宇
(北京化工大学 信息科学与技术学院, 北京 100029)

摘 要: 分布式拒绝服务(distributed denial of service, DDoS)攻击能够在短时间内产生巨量的数据包耗尽目标主机或网络的资源,经过研究发现这些伪造的数据包在一个特定的时间内有着合法数据包所不具备的函数特点。因此,本文提出了行为分布的模型,一旦有可疑流流入服务器,则开始计算这些可疑流的行为分布差异,如果该差异小于一个设定的阈值,则判断有 DDoS 攻击发生;反之则为合法的数据访问。根据 NS-3 的模拟实验,证明该模型能够有效的从合法访问中区分出 DDoS 攻击流,对提前控制 DDoS 攻击的发生具有重要的意义。

关键词: DDoS 攻击; 行为分布; 行为分布差异; 熵检测
中图分类号: TP393

引 言

随着计算机网络技术的高速发展以及人们对网络依赖的持续增长,网络安全成为亟待解决的问题。DDoS 攻击是利用大量被控制的计算机向目标机发起的攻击,而 DDoS 攻击软件的出现则使 DDoS 攻击实施更加简单,攻击者并不需要了解黑客知识或者系统的漏洞,仅仅通过攻击软件向受害主机发送大量的数据包即可完成攻击。在美国计算机安全研究所(CSI)和 FBI 于 2002 年春季做的关于计算机犯罪与安全的一项调查中,接受调查的各部门有 40% 承认在过去 1 年内检测到了 DDoS 攻击。而在 2003 年的报告中,DDoS 攻击给接受调查的 530 家机构带来的经济损失在各种攻击方式中仅次于信息窃取,远远高出排在第 3 位的病毒[1]。

由于硬件与并行技术的协同进步,现今的 DDoS 攻击更加倾向于暴风式攻击[2]受害主机的网络资源来造成网络拥塞以达到拒绝服务的目的,因此如何从大量的合法网络数据中区分出 DDoS 攻击成为网络安全人员的首要问题。Li 等[3]提出了 TCM-KNN 算法来检测网络中的流量异常,但该算法需要大量的数据进行数据融合。Mao 等[4]中对 DDoS 攻击时的各个属性进行分析,得到了一些非常规手段可以得到的数值,然而该文章没有对如何利用这些属性进行检测进行详细说明。赵继俊等[5]提出采用基于流连接信息熵的 DDoS 攻击检测算法,但该算法需要根据不同的情况来设定数据包异常的判定规则,对于 DDoS 攻击来说又显得不够灵活。

本文研究的目标是识别出那些模拟正常网络访问模式的流量,为此引入信息论的相关概念来勘测网络流的分布。这种方法适用于社区网络之中,因为社区网络可以协同工作从而在早期发现 DDoS 攻击。现今的 DDoS 检测算法的准确率较低,如熵检测算法[6]在发出攻击警报时,有可能是一个错误的警报。比如当有爆炸性新闻事件时网络流量会大幅度增加,而熵检测算法会认定为 DDoS 攻击。本文的方法是在 DDoS 攻击报警后(如熵检测报警)各个汇聚路由分别计算可疑流的行为分布差异,如果可疑流行为分布差异小于系统设定的阈值,则断定为 DDoS 攻击。本文首先对系统进行分析与阐述模型,并基于该模型提出行为分布检测算法,然后用 NS-3 仿真软件进行 DDoS 攻击的模拟实验与相关分析,最后对全文进行总结,并对进一步工作进行展望。

1 系统分析

因为检测 DDoS 攻击需要对本网段内的网络设备进行监控,而社区网络中能够对本网段的路由进行管理和配置并可以相互协作,因此本文以社区网络为前提进行讨论。

为了使讨论更容易理解,本文提出了如下几个假设:

收稿日期: 2010-04-12
基金项目: 国家自然科学基金(20671010)
第一作者: 男,1966 年生,教授
E-mail: Zhaoy@ mail. buct. edu. cn

(1)攻击者在各个攻击端使用相同的控制函数产生器进行发包,如多项式函数、幂函数、泊松分布函数或卡方分布函数等;

(2)同一时间内有且仅有一台服务器遭受攻击;

(3)网络系统是线性的,稳定的。

图 1 是一个简单的攻击模型,本文不考虑 DDoS 的产生及路径,仅对可操控的路由一端进行辨别。首先定义社区网络中各个路由上有着相同目的地址的包称为流,并定义在流中不同时刻的包的分布为行为分布,形象的理解为这些包在统一的行为下(函数下)所产生的分布。当 DDoS 警报响起时,社区网络中的各个路由分别在一定的时间 T 内对可疑流中的包进行抽样统计,计算出可疑流中包的行为分布。

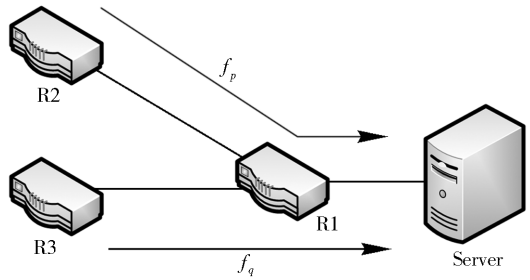


图 1　简单的 DDoS 攻击图

Fig. 1　A simple DDoS attack diagram

在攻击场景中攻击者使用一个随机变量 X 来控制攻击包的产生速度。可能使用的方法为:

(1)用一个持续的速度产生包,即 $X=C$,C 为常数;

(2)通过攻击时间 t 增加攻击包的数量,即 $X=at+b$,a 和 b 为常数;

(3)用泊松分布模拟网络流量,即 $P(X)=\frac{\lambda^k e^{-\lambda}}{k!}$,$k=0,1\cdots$,$\lambda$ 为常数;

(4)其他模拟函数。

如果在图 1 的网络中一旦发出 DDoS 攻击报警,并发现了 2 个可疑流 f_p 和 f_q,则可以在一定的时间 T 内对任意的路由 R1、R2 或 R3 完成取样的工作。一旦取样工作完成,分别取得了时间 T 内可疑流 f_p 和 f_q 的分布函数:

$$\begin{cases} P(x)=p(x_1,x_2,x_3,\cdots,x_n) \\ Q(x)=q(x_1,x_2,x_3,\cdots,x_n) \end{cases} \tag{1}$$

通过等式(1)的分布函数,计算这两个分布的 KL 距离[7]:

$$D(p\parallel q)=\sum_{x\in\chi}p(x)\log\frac{p(x)}{q(x)} \tag{2}$$

这里的 x 是 X 的样本空间,本文将 $D(p\parallel q)$ 称作 $p(x)$ 和 $q(x)$ 的行为分布差异。

如果 f_p 和 f_q 是攻击流,那么他们被不同攻击机的同一函数 $f(x)$ 所生成,理想情况下 $p(x)$ 和 $q(x)$ 是同一函数或呈线性关系的话,$D(p\parallel q)=0$。由于傀儡机本身系统时间和网络条件等环境的不同,所产生的实际结果和假设的结果不一定总是相同的,但是两个攻击流的行为分布差异应该比一个攻击流和一个正常流的行为分布差异要小的多。本文用下面的公式判断 2 个流 p 和 q 是否为相同流:

$$A(p,q)=\begin{cases} 0 & D(p\parallel q)>\xi \\ 1 & D(p\parallel q)\leqslant\xi \end{cases} \tag{3}$$

其中 ξ 是给定的阈值。当两个流的行为分布差异大于给定的阈值时,则认为有 DDoS 产生,反之则为合法的数据访问。其中 ξ 的取值是通过多组行为分布差异比较得出的一个估算值,为了保证突发越变时的准确性,引入自主学习的方法,即每次比较后的估算值与历史学习值进行平均加权求均作为本次的阈值。当然任何设定阈值的判断都会有误差的,所以现在将攻击流的数目扩展为 $n(n>2)$,并从这 n 个可疑流中随机抽取 $m(2m<n)$ 对进行匹配。标记为 p_i,q_i,$i=1,2,\cdots m$。因此经过计算得到 m 对可疑流的行为分布差异 $A_i(p_i,q_i)$,$i=1,2,\cdots,m$。如果每一对的可能误差为 p,则真值的表达式为:

$$A(p_1,p_2,\cdots,p_m,q_1,q_2,\cdots q_m)=1-p^m \tag{4}$$

假设通过分析得到的误差 $p=0.5$,那么如果希望 99% 以上正确率时解得 $m\geqslant 7$。

基于上面的分析,阐述算法模型如下:

(1)在警报发起时在本地路由上识别可疑流;

(2)在时间 T 内对可疑流中包的数目进行取样并找到样本空间 $x_1,x_2,\cdots,x_n$;

(3)用 $p(x_i)=f_i/\sum_{i=1}^{n}f_i$,$i=1,2,\cdots,n$ 计算可疑流的行为分布;

(4)将各个可疑流的行为分布提交给所对应的汇聚路由,并由该汇聚路由通过等式(2)进行计算任意两个可疑流的行为分布差异;

(5)在不同的汇聚路由上通过等式(3)比较可疑流的行为分布差异,并进行判断 ξ 的取值;

(6)通过等式(3)、(4)判定是否有 DDoS 攻击。

2 DDoS 攻击模拟

为了确认上述算法的准确性,作者在 NS-3 上模拟了一次 DDoS 攻击。模拟环境包含 1 个服务器,13 个路由器和 256 个客户机。模拟中包含了 1 条卡方攻击流,1 条泊松攻击流,2 条混合攻击流(攻击流和合法流的混合),1 条合法流和 3 条延迟流。并在该攻击场景中对比混合攻击流与合法流,混合攻击流和卡方攻击流。下面给出了简要的模拟代码。

```
//加入调试命令:
CommandLine cmd;
cmd. Parse (argc, argv);
……
//创建所有节点:
NodeContainer n;
n. Create(270);
……
//配置直连信道的属性:
PointToPointHelper p2p;
p2p. SetDeviceAttribute ("DataRate", StringValue ("50Mbps"));
p2p. SetChannelAttribute ("Delay", StringValue ("2ms"));
//设置直连信道两端节点:
NetDeviceContainer netserr0 = p2p. Install (serr0);
NetDeviceContainer netr1r11 = p2p. Install (r1r11);
……
//设置平行信道的属性:
CsmaHelper csma;
csma. SetChannelAttribute ("DataRate", StringValue ("100Mbps"));
csma. SetChannelAttribute ("Delay", StringValue ("1ms"));
//设置平行信道内的节点:
NetDeviceContainer netr0Tor4 = csma. Install (r0Tor4);
NetDeviceContainer netr11ton132 = csma. Install (r11ton132);
……
//设置 IP 地址:
Ipv4AddressHelper ipv4;
ipv4. SetBase ("10. 1. 1. 0", "255. 255. 255. 0");
Ipv4InterfaceContainer i = ipv4. Assign (netserr0);
……
//建立模拟应用,设置服务器的工作时间和服务类型:
uint16_t port = 9;
UdpEchoServerHelper server (port);
ApplicationContainer apps = server. Install (n. Get(0));
apps. Start (Seconds (1. 0));
apps. Stop (Seconds (10. 0));
……
//日志:
CsmaHelper:: EnablePcapAll (" DDoSSimulation", false);
……
```

上面给出的是简要的拓扑代码中因为节点数量太大,只是作为参考的一部分,而具体的模拟代码通过内部的和自定义的函数作为模拟依据,也不在过多的进行算法说明了。通过实验捕捉到的 pcap 文件中的数据依据前文的模型进行计算后得出结果如图 2,其中横坐标是采样所用的时间,纵坐标是该时刻采样数据的行为分布差异,通过对比得到信息如下:

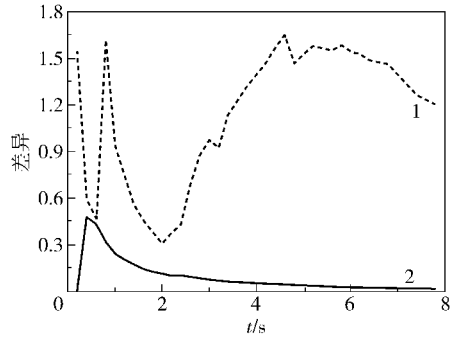
(1)在抽样开始运行 25 个时间单元后系统达到稳定。如果每 s 抽取 10 个样本,则仅需 2. 5 s 即可进行判断;

(2)一旦系统达到稳定,攻击流和合法流的行为分布差异值要大于 1;

(3)攻击流之间的行为分布差异小于 0. 1,这个差异值和一个攻击流一个合法流的差异值相差很多。

图 2 对比的是单数据源的情形,下面考虑多组数据源的情形[4]。假设网络中有 20 条数据流,其中有 16 条攻击流,那么基于本文算法检测出行为分布差异大于阈值的至少有 6 对,若该算法失效的概率是 20%,则判断本次为 DDoS 攻击的准确率则为 99. 99% 以上。

图 3 是采用同一数据源的熵检测算法图,通过图 3 分析得出合法流在 15 个时间单位、20 个时间单位和 35 个时间单位发生越变,混合攻击流在 13

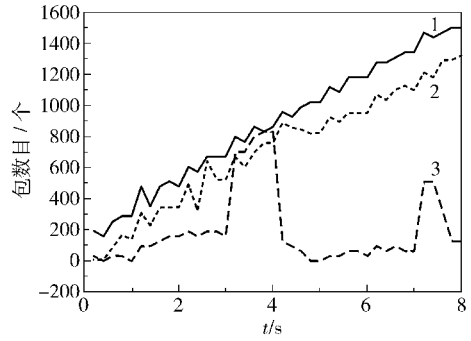


1—混合攻击流和攻击流；2—混合攻击流和正常流

图 2　行为分布差异比较图

Fig. 2　Comparison of the behaviour distribution diagrams

个时间单位时发生越变，攻击流在数据采集的 40 个时间单位无越变。如只采用原始的熵检测算法判断 DDoS 攻击（在越变时认为有攻击产生）则可以得出本数据源下该算法的准确率较低的结论。



1—攻击流；2—混合攻击流；3—合法流

图 3　熵检测图

Fig. 3　Entropy detection diagram

本方法的关键点在于一次攻击在短时间内共用同一个攻击模式，而这个模式是合法流所不具备的。因此一旦 DDoS 攻击检测算法（如熵检测）警报有潜在的攻击，分析器就开始在网络中计算不同可疑流之间的行为分布差异，如果该值小于分析得到的阈值（如基于实验的 0.1）则认定为一次 DDoS 攻击，反之则是合法的数据访问。

3　结束语

提出了一个高效的、精确的 DDoS 攻击检测方法，具有如下优点：独立性强，只需要社区网络环境而不需要多方 ISP 的支持；通用性高，可以鉴别各种伪装的数据包；效率高，可以在几秒钟甚至更短的时间内进行辨别；资源占用率低，不需要大量的存储空间和运算。虽然理论分析和实验检测都表明这是一个有效的 DDoS 检测方法，然而该方法暂时处于基础阶段，还有许多需要进一步完成的工作：如何在准确检测和快速检测中折中选择一直以来都是个问题，需要在实践中获取最优解决方案；在大量合法数据访问服务器时算法的正确性保证，即 根据实际情况进行取值所应用的算法选择；攻击者可能会用多种函数产生器进行发包来使本文的检测算法失效。然而任何攻击背后都会隐藏着人造的潜在规则，这是下一阶段研究的主要方向。

参考文献：

[1] 李德全. 拒绝服务攻击[M]. 北京：电子工业出版社，2007.
Li D Q. Distributed denial of service[M]. Beijing: Publishing House of Electronics Industry, 2007. (in Chinese)

[2] Xiao B, Chen W, He Y X. An autonomous defense against SYN flooding attacks: Detect and throttle attacks at the victimside independently[J]. Journal of Parallel and Distributed Computing, 2008, 68(4): 456-470.

[3] Li Y, Guo L, Tian Z H, et al. A lightweight web server anomaly detection method based on transductive scheme and genetic algorithms[J]. Computer Communications, 2008, 31(17): 4018-4025.

[4] Mao Z M, Sekar V, Spatscheck O, et al. Analyzing Large DDoS Attacks Using Multiple Data Sources[C]// Proceedings of the 2006 SIGCOMM workshop on Large-scale attack defense, Pisa, Italy, 2006: 161-168.

[5] 赵继俊，胡志刚，张健. 基于流连接信息熵的 DDoS 攻击检测算法[J]. 计算机工程，2007，33(16)：139-141.
Zhao J J, Hu Z G, Zhang J. DDoS attacks detection algorithm based on flow connection entropy[J]. Computer Engineering, 2007, 33(16): 139-141. (in Chinese)

[6] Kumar K, Singh J K. A distributed approach using entropy to detect DDoS attacks in ISP domain[C]// International Conference of Signal Processing, Communications and Networking, Feb 22 - 24, 2007, Chennai, India. vol 1: 331-337.

[7] Cover T M, Thomas J A. Elements of information theory [M]. 2nd Ed. Hoboken: John Wiley & Sons Inc, 2007.

A method for detecting distributed denial of service attacks based on behavior distribution

ZHAO Ying HUANG WenYu

(College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China)

Abstract: A distributed denial of service (DDoS) attack is a common network attack and it is difficult to prevent. A DDoS attack usually generates a huge amount of packages in a very short time and exhausts the resources of the host and network which are attacked. Consequently, DDoS attack is a great threat to the stability of high - speed networks. Many studies have shown that the attack packages are generated by one or several functions. Therefore, the attack packages always share some features that valid packages do not have. This paper introduces the concept of behavior distribution. When suspicious flows arrive at a server, the software calculates the differences in their behavior distribution. If the difference is lower than the threshold, it is deemed a DDoS attack. Otherwise, it is a valid access. The NS-3 experimental results indicate that this method can effectively distinguish a DDoS attack from a valid access and thus contain an attack as soon as possible.

Key words: DDoS attack; behaviour distribution; difference in behaviour distribution; entropy detection