一种分布式拒绝服务攻击的检测模型

李目海[1,2] 李 明[1] 吴新星[1] 李旭宏[2]
(华东师范大学信息科学技术学院 上海 200241)[1] (枣庄学院计算机科学系 山东 277160)[2]

摘 要 提出了一种实时检测网络是否受到DDoS攻击的新模型,解决了传统检测方法难以区分突变正常流量与异常流量的问题。结合网络正常流量的特点,提出了检测DDoS攻击的新度量和检测算法。该算法不仅结构简洁、运算速度快;而且能够充分利用已知信息,具有较强的抗干扰能力。实际检测结果表明,本模型可实现对DDoS攻击的实时检测。

关键词 分布式拒绝服务,攻击,检测,模型,算法

A Kind of Detecting Model against DDoS Attacks

LI Mu-hai[1,2] LI Ming[1] WU Xin-xing[1] LI Xu-hong[2]
(School of Information Science & Technology, East China Normal University, Shanghai 200241, China)[1]
(Department of Computer Science, Zaozhuang University, Shandong 277160, China)[2]

Abstract This paper presented a run-time model that can efficiently detect network attacks. The model resolves the problem that classic method can't distinguish anomalies from attacks. Combining the characteristics of normal traffic, an efficient measure and algorithm which can detect DDoS attacks were given. The algorithm not only has simple structure, high speed to meet the needs of real-time detection, but also can take full advantage of known information, and has more strong anti-jamming capability. The actual test results indicate that this model can be used to achieve real-time detection of DDoS attacks.

Keywords Distribution denial-of-service, Attack, Detection, Model, Algorithm

1 引言

DDoS攻击具有易实现、大流量、分布式、隐蔽性等特征,并已成为Internet的首要威胁[1]。通常,在DDoS发动攻击前,攻击者已控制了数量庞大的计算机(又称僵尸机),这些计算机将在攻击者的指挥下,通过发送大量垃圾包,对一个或多个目标同时实施DDoS攻击,使被攻击目标所在网络链路拥塞,导致目标机无法正常工作甚至瘫痪,从而给受害者带来极大损失。本文针对这种攻击给出了一种实时检测方法。

目前常用的攻击检测方法主要有两种[2],一种是误用检测,这种方法通过建立知识库来检测攻击。这种方法显然存在明显缺陷,因为它无法抵御未知特征的网络攻击。另一种方法是异常检测,这类检测方法的优点是可以检测未知特征的DDoS攻击,缺点是误警率高。异常检测通常使用两种检测技术[3],一种是阈值检测技术,该技术通过对系统正常状态的研究,给出一个阈值,一旦被检测系统检测超出设定阈值,则认为受到攻击。但阈值大小如何设置是个难点。二是基于流量的统计检测技术,目前这一技术的研究比较活跃。对于检测技术,我们目前已开展较深入的研究工作[4],通过自相关函数建立了检测模型,给出了低误报和低漏报的概率检测算法。我们目前另一项研究工作是通过Hurst参数估计来检测网络是否受到攻击[5]。随着小波技术应用的不断深入,2001年以后,使用小波分析方法检测网络攻击的文献不断出现[6-8]。文献[9,10]通过熵和协方差阵等方法在攻击检测方面也做了一些有益的探索。

上述文献虽然提供了一些较好的检测方法,但普遍存在无法将突变的正常流量(如服务器早8点突然增加的正常流量)与攻击流量区分的问题。产生这一现象的原因是缺乏对已知信息(正常流量先验知识)的充分利用,这是本文解决问题的特色之一。

本文的结构是:第2节给出DDoS检测模型和检测方法;第3节通过实际数据对检测模型进行验证;最后总结全文。

2 DDoS检测模型及检测方法

假设 $y(t)$,$n(t)$ 和 $a(t)$ 分别表示 t 时刻到达被攻击目标机(以下称服务器)的网络总流量、正常流量和攻击流量。这样,$y(t)$ 可写成:

$$y(t) = n(t) + a(t) \tag{1}$$

为方便建模分析,将上关系式变为:

$$a(t) = y(t) - n(t) \tag{2}$$

从式(2)可见,当系统没受到攻击时,$a(t)=0$,即 $y(t)$ 与 $n(t)$ 的值相等;当网络受到攻击时,$a(t)$ 值将迅速变得异常庞大。下面基于此种思想建立检测攻击模型。

在给出检测模型前,需要了解网络流量的基本特征:在正

到稿日期:2008-04-29 本文的研究工作得到国家自然科学基金(项目编号:60573125),上海重点学科建设项目(项目编号:B411)和山东省教育厅科技项目(项目编号:J07WJ29)的部分支持。

常情况下，服务器在固定时间的正常流量在一定时期内相对稳定[11]。

图1是从枣庄学院中心服务器采集到的数据流量，通过图示可以看出，曲线每天变化幅度较大，但不同天的相同时刻的流量变化及幅度却非常相似，。如每天早上8点前后的流量数据等。产生这一现象的原因是在一定时期内服务器提供的服务基本不变，从而使它的正常用户及其数量也基本稳定。又因为每个用户的工作习惯和对服务器的需求基本稳定，在大量用户的组合效应下，使得在一定时期的固定时间上，网络流量基本保持不变。文献[9]通过计算熵来判定系统是否受到攻击正是依据这一特征。文献[8]提供的实际流量图也同样证实了这一特征。

根据网络正常流量具有的稳定特征，检测攻击时，使用最新统计的正常流量作为检测时的正常流量是合适的，图1中的"统计值"就用于此。

定理1 在正常情况下，$a(t)/N(t)$将服从均值几乎为0的正态分布，且此分布与流量的大小无关。其中$N(t)$为在t时刻的流量统计值。

证明：假设服务器在t时刻检测时与统计$N(t)$时有m_t个相同的常用用户。

显然，当服务器没有受到攻击时，$y(t)$就是正常流量，则$y(t)=n_{m_t}(t)+n_{r_t}(t)$。其中$n_{m_t}(t)$，$n_{r_t}(t)$分别表示$t$时刻$m_t$个常用用户和$r_t$个随机用户产生的正常流量。同样，$N(t)=N_{m_t}(t)+N_{S_t}(t)$，这里的$N_{m_t}(t)$，$N_{S_t}(t)$表示$t$时刻统计流量包含的$m_t$个常用用户和$s_t$个随机用户产生的流量。则

$$\frac{a(t)}{N(t)}=\frac{n_{m_t}(t)-N_{m_t}(t)}{N(t)}+\frac{n_{r_t}(t)-N_{s_t}(t)}{N(t)}$$

在正常情况下，网络流量主要是常用用户产生的，随机用户的访问流量只是总流量的极小部分。根据流量的稳定性，则有$n_{r_t}(t)=n_{m_t}(t)<N(t)$且$N_{s_t}(t)=N_{m_t}(t)<N(t)$，因此$\frac{n_{r_t}(t)-N_{s_t}(t)}{N(t)}$可看成是一个近似为0的常量，所以$a(t)/N(t)$的分布由$\frac{n_{m_t}(t)-N_{m_t}(t)}{N(t)}$来确定。又因正常情况下的随机用户流量仅是总流量极小部分，所以常用用户流量占总流量的比值可看成是与用户个数无关的量，即$\frac{n_{m_t}(t)}{N(t)}-\frac{N_{m_t}(t)}{N(t)}$与$t$时刻的常用用户个数$m_t$无关。又由于同一组用户的流量具有相同的分布，即$n_{m_t}(t)$与$N_{m_t}(t)$的分布相同。因此$\frac{n_{m_t}(t)}{N(t)}$与$\frac{N_{m_t}(t)}{N(t)}$也有相同的分布，它们的差$\frac{n_{m_t}(t)}{N(t)}-\frac{N_{m_t}(t)}{N(t)}$将服从均值为0的正态分布。且其最大振幅$\left|\frac{n_{m_t}(t)}{N(t)}-\frac{N_{m_t}(t)}{N(t)}\right|\leqslant\max\left(\frac{n_{m_t}(t)}{N(t)},\frac{N_{m_t}(t)}{N(t)}\right)\leqslant\frac{N(t)}{N(t)}=1$。故在正常情况下，$a(t)/N(t)$是一个与流量大小无关、均值几乎为0的正态分布。

通过实际测试的数据（如图2），当服务器没有受到攻击时，$a(t)/N(t)$在各时段样本均值的绝对值全在0.5以内，为方便叙述，记$a(t)/N(t)$为$A(t)$，并记其均值为η。

根据定理1，当网络没有受到攻击时，$A(t)$服从均值为η的正态分布，且与检测起始时间和网络流量的大小无关，从而克服了突变的正常流量（正常流量迅速激增或下降）与攻击流量的误判问题。而当系统受到攻击时，流量中的随机部分$n_{r_t}(t)$将变得异常庞大，其值通常超出正常流量的数倍或更多，这样$A(t)$的分布将由$\frac{n_{r_t}(t)-N_{s_t}(t)}{N(t)}$决定，因此$A(t)$将不再是接近于0均值的正态分布。因此，只要判定$A(t)$不服从均值近似为0的正态分布，即可判定系统受到异常流量的攻击。

下面给出具体的检测模型及检测方法。

设检测总次数为T，由于$A(t)$的方差未知，故用它的无偏估计S_A作为方差，则$A(t)$的均值可通过如下方法估计：

根据统计学知识，$\frac{u_A(T)-\eta}{S_A(T)/\sqrt{T}}$服从自由度为$T-1$的$t$分布。这里$T$为采样次数；$S_A(T)$，$u_A(T)$分别表示$A(t)$检测到第$T$次时方差无偏估计的平方根和样本均值。

给定置信度为P，则η的置信区间为$\left(u_A(T)+t_{\frac{\delta}{2}}\frac{S_A(T)}{\sqrt{T}},u_A(T)+t_{1-\frac{\delta}{2}}\frac{S_A(T)}{\sqrt{T}}\right)$。其中$t_{\frac{\delta}{2}}$，$t_{1-\frac{\delta}{2}}$分别为$t$分布的分位点，$\delta=1-P$。通常情况下$T$的值大于20，故$t$分位点可用标准正态分布的分位点来近似。

若η的置信区间在$[-0.5,0.5]$以内。则可以确定系统没有受到异常流量的攻击。否则认为受到攻击，至此我们得到一个用于攻击检测且不受突变正常流量干扰的数学模型和判别方法。

3 模型验证

图1是通过Sniffer软件从枣庄学院校园网服务器上采集的流量数据图示（每10秒采集一次）。图1由统计值和2007年12月18、19日两天的实时数据组成。从图1中可明显看出流量数据在不同日期、相同时间的相似特征。另外，图1中的部分异常流量是使用WinArpAttack软件攻击服务器形成的。其中，在18日下午2时33分和3点36分进行了持续10分和8分钟的两次模拟攻击，19日上午8点55分了也进行了持续11分钟的模拟攻击。从图1中可明显看出攻击时刻流量的明显变化。图2是$A(t)$在18日的数据图示，从图2可明显看出，在正常情况下，$A(t)$分布与流量大小的无关性及异常流量发生的时间点。

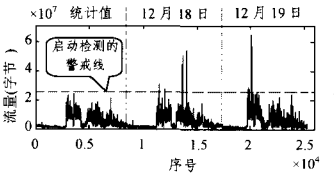

图1 统计流量及实时检测流量

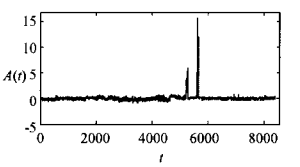

图2 $A(t)$在18日的曲线图

序号	12月18日 开始时间	置信区间(95%)	受到攻击	12月19日 开始时间	置信区间(95%)	受到攻击
1	8:30	(−0.0151,0.1001)	否	8:05	(0.2973,0.3697)	否
2	10:01	(−0.1409,0.0883)	否	8:22	(0.0449,0.1684)	否
3	14:33	(2.3799,3.0245)	是	8:55	(3.2534,3.7432)	是
4	15:36	(3.6149,5.0761)	是	9:05	(0.2954,0.7259)	是
5				10:29	(0.1439,0.2060)	否

为提高检验效率，设置2.5×10^7时为检测警戒线，当流量超过此值时，则连续检测10分钟。正常情况下，$A(t)$均值应在$[-0.5,0.5]$区间内。按照本文的检测算法，18日共自动检测4次，19日检测5次。上表是18，19日的实时检测结果。

在上表中，19日检测到两次可能受到攻击，这与攻击时长有关，由于检测时长为10分钟，而19日的攻击时长为11分钟，因此第四次检测的数据中有一分钟的攻击流量数据，所

以影响了置信区间。

结束语 本文通过对网络流量的分析，提出了一个实时检测 DDoS 攻击的检测模型，此模型具有算法简洁、易于实现等特点，适用于实时检测。通过实际数据检验表明，此方法可快速准确判别服务器是否受到 DDoS 攻击。由于该算法使用了正常流量的先验知识，从而提高了算法的适应能力，解决了突变正常流量与异常流量无法区分的难题，提高了服务器可能受到 DDoS 攻击的预报准确率。

我们下一步将研究如何利用路由器的流量分析与控制功能，建立集检测、控制为一体的全自动网络安全管理系统。

参 考 文 献

[1] Background on DDoS. http://www.ddos.com/index.php?content= products/ background.html December, 2007

[2] Rohrmair G T, Lowe G. Using data-independence in the analysis of intrusion detection systems. Theoretical Computer Science, 2005, 340: 82-101

[3] Carl G, Kesidis G. Denial-of-Service attack detection techniques. IEEE Internet Computing, 2006, 10(1): 82-89

[4] Li Ming. An approach to reliably identifying signs of DDOS Flood attacks based on LRD traffic pattern recognition. Computers & security, 2004, 23: 549-558

[5] Li Ming. Change trend of averaged Hurst parameter of traffic under DDOS flood attacks, Computers & Security, 2006, 25 (3): 213-220

[6] Carl G, Brook R R, Rai S. Wavelet based of Service detection. Computers & Security, 2006, 25: 600-615

[7] Hamdi M, Boudriga N. Detecting Denial-of-Service attacks using the wavelet transform. Computer Communications, 2007, 30: 3203-3213

[8] 肖志新，杨岳湘，杨霖. 基于小波技术的网络异常流量检测与实现. 计算机科学，2006，33(10)

[9] Feinstein L, et al. Statistical approaches to DDoS attack detection and response // DARPA information survivability conference and exposition proceedings. 2003, 1: 303-14

[10] 魏向荣，李之棠. DDoS 的协方差检测模型. 通信学报，2006，27 (11A): 72-75

[11] 李德全. 拒绝服务攻击. 电子工业出版社，2007: 1-16

[12] 帕普里斯 A，等. 概率、随机变量和随机过程 . 西安交通大学出版社，2004: 220-222

(上接第 287 页)

集成点信息表中相关项；当集成 Agent 出错时，管理 Agent 根据错误状况，选择重新分配请求或者恢复集成 Agent 重新执行。

集成 Agent 利用消息传递机制实现不同企业应用的集成，它是模型的核心组成部分。集成 Agent 所包含的功能模块有：集成 Agent 核心引擎、标准(消息标准机制)、服务编排、知识库等。集成 Agent 的核心引擎消息转换模块则包含消息接收器、消息发送器、消息连接器、消息转换器等模块。消息路由则包含输入路由和输出路由两个部分。其详细功能模块示意图 4 所示。

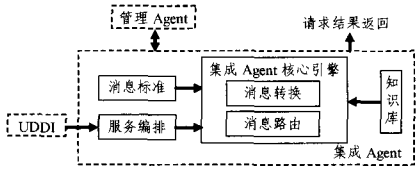

图 4 集成 Agent 功能模块示意图

集成 Agent 的工作机制：集成 Agent 接收到管理 Agent 的转发的应用请求消息，消息转换模块读取应用请求消息，并将消息标准化。集成 Agent 分析应用请求类型，根据集成 Agent 知识库(包含集成 Agent 以前编排的服务的历史记录)中包含的集成 Agent 提供常用服务，选择对应的 Web Services，若没有对应的服务则通过服务编排模块生成对应的 Web Services，并将其加入到知识库中。调用 Web services 获得返回结果，最后将结果非标准化发送给应用请求者。

3.3 模型实现

为了证明基于服务架构的多 Agent 企业应用集成模型的设计思想，进行了原型平台的实现。利用该原型平台对某企业电子商务协商系统和 ERP 进行集成[7]。平台以 Web Services 为基础，将企业 ERP 系统中的生产计划模块、库存管理模块中某些功能通过 SOAP/XML 适配器转换为 Web Servi-ces 发布到企业私有 UDDI 注册中心。管理 Agent 和集成 Agent 之间的通信采用 JMS[8] 异步通信机制。管理 Agent 的集成点分配模块初始化时将各个集成点分配同样的优先级 20，每次选择优先级最高的集成点进行分配。集成 Agent 的核心引擎消息转换模块则实现了针对不同协议(JMS, SOAP, HTTP 等)的消息接收器、消息发送器、消息连接器和消息转换器。消息路由实现了输入路由和输出路由两个部分。利用 BPEL4WS[9] 表示的 Web Services 业务流程作为集成 Agent 知识库。通过对电子商务系统与 ERP 系统之间的数据访问，检验了智能企业应用集成平台模型的可行性，结果显示模型能较好地实现异构应用之间互连互通、数据共享、业务流程统一的集成。

结束语 结合基于以服务为集成架构的设计理念，本文给出了一种基于服务架构的多 Agent 企业应用集成模型，用智能 Agent 技术设计了模型中两个重要的部分：集成服务端和集成点，最后实现了模型的原型，原型能实现异构应用的连接、数据共享、业务流程统一。在未来的工作中，我们将从以下几个方面逐步完善模型：集成 Agent 核心引擎静态调用 Web Services，为了增加集成中心的灵活性考虑采用动态调用的方式实现；管理 Agent 对应用请求仅仅做简单的身份验证，应解决消息在网络传递过程中安全性不高等问题。

参 考 文 献

[1] Mann J. Approaches to Enterprise Application Integration [EB /OL]. http://eai.ebizq.net, 2004-08-14

[2] SOA-解决中小企业应用集成的良药. http://www.sagapio.com/techsubject/soaplan.htm, 2006-04-1

[3] Newcomer E, Lomow G. Understanding SOA with Web Services 中文版. 电子工业出版社，2006

[4] W3C. SOAP Version 1.2 Part 1 Messaging Framework —— W3C Recommendation[EB/OL]. (2003-06). http://www.w3.org/TR/soap12-part1/

[5] Keen M, Bishop S, Hopkins A, et al. http://publib blboulder-1ibmlcom /Redbooks1nsf /Red PieceAbstracts/ sg246346. html? Open Patterns: Im Plementing an SOA using an Enter Prise Service Bus EB /OL1200417125

[6] UDDI.org White Paper. The Evolution of UDDI. http://www.uddi.org/specification.html. 7/19/2002

[7] 冯文辉. 面向服务架构的协商系统与企业 ERP 集成研究. 华南师范大学硕士学位论文. 2007

[8] Monson-Haefel R, Chappell D A. Java Message Service. O'Reilly. 2001

[9] Chafle G, Chandra S, Mann V, et al. Decent ralized Orchest ration of Composite Web Services. WWW2004, New York, USA, 2004: 134-143