一种基于 Hurst 参数的 SYN Flooding 攻击实时检测方法*)

严 芬[1,2] 王佳佳[2] 殷新春[2] 黄 皓[1,2]
(南京大学计算机软件新技术国家重点实验室 南京 210093)[1]
(扬州大学信息工程学院计算机科学与工程系 扬州 225009)[2]

摘 要 提出了一种轻量级的源端 DDoS 攻击检测的有效方法。基于 Bloom Filter 技术提取网络数据包中新的可疑源 IP 地址出现的次数,然后使用实时在线 VTP 方法进行异常检测,不仅能够实时检测出 DDoS 攻击的存在,而且能够避免因为网络数据流量的正常突变引起的误报。从实验结果可以看出,该方法还能够发现大流量背景下,攻击流量没有引起整个网络流量显著变化的 DDoS 攻击。

关键词 DDoS,源端检测,Bloom Filter,实时检测,Hurst 参数

Real-time SYN Flooding Attacks Detection Method Based on Hurst Parameter

YAN Fen[1,2] WANG Jia-jia[2] YIN Xin-chun[2] HUANG Hao[1,2]
(State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China)[1]
(Department of Computer Science and Engineering, Technology Institute, Yangzhou University, Yangzhou 225009, China)[2]

Abstract An efficient light-weight method for defending against DDoS attacks at the source end was designed. The Bloom Filter was used to pick up the amount of doubtful new source IP of network packets. Then, on line VTP technology was used to detect abnormity. Our method can not only detect the existence of DDoS attacks on line, but also avoid the false alarm of normal break. According to experiments, the method can find out the DDoS intrusion against the large scale network, which does not arouse the sharp changes of the network traffic.

Keywords DDoS, Detection at the source-end, Bloom filter, On-line detection, Hurst parameter

1 引言

DDoS(Distributed Denial of Service)攻击是利用足够数量的傀儡机产生数目巨大的攻击数据包对一个或多个目标实施 DoS 攻击,耗尽受害端的资源,使受害主机丧失提供正常网络服务的能力。DDoS 攻击已经是当前网络安全最严重的威胁之一,是对网络可用性的挑战。就目前的网络状况而言,世界的每一个角落都有可能受到 DDoS 攻击。但是,只要能尽早检测到这种攻击并及时作出反应,就能将损失减到最小程度。因此,DDoS 攻击检测方法的研究一直受到广泛关注。

目前,DDoS 攻击中约有 90% 是 SYN Flooding 攻击[1],因而 SYN Flooding 攻击的检测也成为当前研究重点之一。SYN Flooding 攻击是 DDoS 攻击的一种,又称半开式连接攻击,主要是利用 TCP 连接的三次握手信息造成的。当攻击者恶意地快速连续送出许多 TCP SYN 包给被攻击端,却没有后续确认包传出时,被攻击端可用的 TCP 连接队列会因为存储太多正在等待连接的信息而超过其容许量,从而导致暂停服务。当前对 SYN Flooding 攻击的检测手段主要包括统计、模式预测、人工智能等方法。统计的方法一般会受到阈值的限制,并且不能够区分正常的网络拥塞和 DDoS 攻击;模式预测和人工智能的方法很难适应攻击研究的大流量背景的需要,同时也存在误报率高的问题。

研究表明,真实的网络流量具有自相似特性。Leland[2] 对 Bellcore 的测试分析结果表明,真实的网络流量具有统计自相似性,这与传统的网络流量的泊松或贝努利过程是完全不同的。Paxson[3] 对 WAN 网络进行测量,也发现网络业务量表现出长相关的特性。Hu[4] 在光纤上研究了突发数据流对网络数据流自相似性的影响。近年来,这种特性也被应用到了 DDoS 攻击检测之中,利用攻击发生前后网络流量特征不同的原理检测攻击的发生。文献[5]和[6]都利用了网络流量自相似的原理。文献[5]采用了相关系数法检测 DDoS 攻击、首先对网络流量进行高频统计,然后对其相邻时刻流量进行相似度分析,根据相似度的变化来发现异常。首先,算法不对数据包的信息进行提取,只是简单地从网络流量上进行操作,阈值很难确定;其次,算法必须首先对网络流量进行统计,并需保留统计结果进行后续分析,因此不能够实时在线检测攻击;再次,正常的网络拥塞与 DDoS 攻击均能够引起原有网络流量的变化,导致不相似,该算法不能够区分正常的网络拥塞和真正的 DDoS 攻击,误报率较高。文献[6]采用基于 Hurst 参数的 VTP 分析法来检测 DDoS 攻击,该方法也仅仅是对网络流量进行分析计算,同样存在不能够区分正常的网络拥塞和真正的 DDoS 攻击、误报率较高的问题。目前,关于自相似过程的描述有很多,主要有 R/S 统计分析[2]、Whittle 估计法[7]、小波分析法[8]、周期图估计法和方差-时间图(Variance-Time Plots, VTP)分析法[9]。其中,VTP 分析法通过 Hurst 参数的变化反映攻击的发生,具有计算复杂度低、能够

*)基金项目:国家高技术研究发展计划("863"计划)基金资助项目(2003AA142010),国家自然科学基金资助项目(60473093)和江苏省高技术研究计划基金资助项目(BG2004030)。严 芬 博士研究生,主要研究方向为网络与信息安全;王佳佳 硕士研究生,主要研究方向为信息安全;殷新春 教授,主要研究方向为密码学与信息安全;黄 皓 教授,博士生导师,主要研究方向为计算机信息系统安全、网络与信息安全。

快速检测出攻击等特点。

我们提出了一种基于 Hurst 参数的快速 SYN Flooding 攻击检测方法，在攻击的源端检测 DDoS 攻击，不需要设置阈值，并且适合于大规模网络背景流量。本方法具有准确性高、实时检测、能区分正常的网络拥塞与攻击情况、更适合大流量背景攻击等特点。算法主要采用了两种核心技术：一是采用了基于 Bloom Filter 的数据结构提取数据包的特征信息。该结构使用静态的固定存储空间和静态存储方法，能够用较少的存储空间存储大量的数据包信息；使用 Hash 函数能够进一步使得数据在存储空间内平均分布；二是使用了基于 Hurst 参数的、实时在线的方差-时间图分析方法（On Line VTP，文中简称 OL-VTP）对提取的特征信息进行检测，判断是否发生了 DDoS 攻击。Hurst 参数能够快速准确地反映出数据包特征时间序列的自相似性的变化情况。

本文第 2 节分析了 SYN Flooding 攻击的特征及其与正常网络拥塞的区别；第 3 节给出了使用 Bloom Filter 提取攻击特征，从而判断数据包是否具有伪造的源 IP 地址的方法；第 4 节阐述了使用基于 Hurst 参数的 VTP 分析法检测 DDoS 攻击的具体方法；第 5 节给出了实验结果及性能分析，说明了算法的优点；最后对全文进行总结并展望未来的工作。

2 SYN Flooding 攻击特征分析

为了及时检测出攻击，需要了解 SYN Flooding 攻击发生时与正常状态下网络行为的区别。正常情况下，如果主机 A 向主机 B 发送一个 SYN 包请求建立 TCP 连接，那么主机 B 接到该请求后会向主机 A 发送一个 SYN+ACK 包，最后主机 A 再向主机 B 发送一个 ACK 包，此时三次握手连接成功，可以进行数据传输了。当攻击发生时，网络中将会出现大量的 SYN 包，受害主机 B 的连接队列资源很快被耗尽，从而不能及时发送 SYN+ACK 包。即使 B 发送了 SYN+ACK 包，由于主机 A 假冒了主机 C 的源地址，A 和 C 都不可能回送 ACK 包。此时，网络中 SYN 包的数量远远大于 ACK 包的数量（如图 1 所示）。而当正常情况下网络出现拥塞时，也可能出现 SYN 包的数量大于 ACK 包数量的情况，或者 SYN 包到达速率发生变化的情况。因此，仅仅根据网络整体流量的自相似性来判断攻击的发生是不够的，需要区别正常网络拥塞与 DDoS 攻击的情况。因此，我们考虑挖掘攻击发生时更深层次的有用信息。

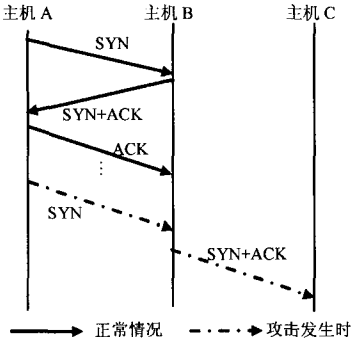

图 1　正常与非正常三次握手连接示意图

DDoS 攻击往往采取伪造攻击包源 IP 地址的方法。攻击发生与正常网络拥塞的一个主要区别就在于网络中是否出现了大量伪造的源地址。攻击发生时会出现大量以前从未出现过的源 IP 地址，而网络在正常拥塞的情况下出现的源地址大部分都是以前出现过的[10]。另外，DDoS 攻击一般借助工具产生，研究发现，这些工具伪造攻击包的源 IP 地址，而且一个伪造的源 IP 被多次使用，即向目标发出多个源 IP 相同的攻击包。现有的以被伪造的 IP 出现的个数为检测依据的检测方法，虽然当大量的攻击包出现时也能检测到攻击，但显然不能做到快速有效的检测。因此，为了区分正常网络拥塞和 DDoS 攻击，同时考虑对重复出现的攻击包的处理，尽早地检测到攻击，我们不以伪造的新 IP 地址的个数为检测依据，而以网络中出现的大量新的、伪造的源 IP 地址的次数为依据，并根据这些新的、伪造的源地址出现的次数构成的时间序列的自相似性来判断是否发生了 SYN Flooding 攻击。

3 基于 Bloom Filter 的信息提取

Bloom Filter 最早诞生于 1970 年，原本用于依次测试一系列信息是否属于给定的信息集以确定其成员资格，1980 年开始用于降低不同文件对磁盘访问的速度以及其它方面，现在被扩展应用于 DDoS 攻击检测中。在 Bloom Filter 结构中，一个二维向量表由 k 级向量组成，每一级向量对应于一个 Hash 函数，并且包含 m 位向量。每一位向量包含一个计数器 $C_{i,j}$（$1\leqslant i\leqslant k, 1\leqslant j\leqslant m$），如果被击中则加 1。由于存在 Hash 冲突，该结构存在一定程度的误差，这种误差可以通过调整 Hash 函数和 m 值来减小。使用 Bloom Filter 能够节约存储空间，也便于对源地址信息进行下一步处理，Hash 函数的使用能够使数据在向量表中的分布更分散。我们的算法基于 Bloom Filter 数据结构，并做了一定的改进，用一个 Hash 函数对应于两级向量。

DDoS 攻击发生时，会出现大量伪造源地址的数据包，即源 IP 中出现大量的、以前从未出现过的新地址。这些包的出现势必使与网络地址相关的统计特性发生变化。基于这种特性，通过合适的算法肯定能检测出攻击的存在。首先，说明如何提取被伪造的源 IP 出现的次数。使用 Hash 函数将数据包源 IP 地址映射到 Bloom Filter 结构中，每一个源 IP 地址对应于 Bloom Filter 中的一个计数器。为了确定一个源 IP 地址是否为新出现的被伪造的地址，我们改进 Bloom Filter 的结构，用一个 Hash 函数对应于两级向量 T_A 和 T_S（如图 2 所示）。改进后的结构能够准确方便地统计伪造源地址出现的次数。Hash 函数用于计算数据包源 IP 地址对应的向量计数器 C_j 的位置 j（$1\leqslant j\leqslant m$）。T_A 和 T_S 均为一维 m 位向量，计数器的初始值均为 0。T_A 用来保存经过路由器的数据包的 ACK 标志位的信息，其值只可能为 0 或 1。若 T_A 中的某计数器值为 0，则说明与该计数器相对应的源地址尚未成功建立连接；若 T_A 中的某计数器值为 1，则说明与该计数器相对应的源地址已经成功建立了连接。T_S 用来保存最近一段时间经过路由器的、源地址疑似被伪造的数据包的数量信息。若 T_S 中某计数器值为 0，则说明与该计数器对应的源 IP 还没有出现伪造包的情况；若 T_S 中的某计数器值为非 0，则说明与其对应的源 IP 地址的数据包尚未被确认，该 IP 可能是伪造的。

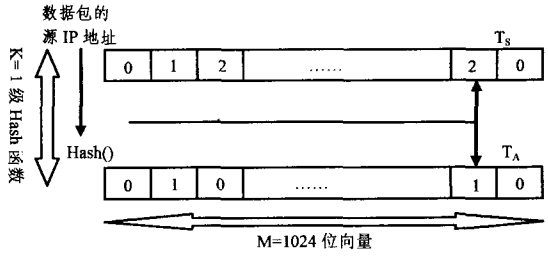

图 2　算法中使用的 Bloom Filter 结构

当一个数据包到来时，基于 Bloom Filter 的信息提取算法描述如下。

步骤 1　提取数据包的源 IP 地址，进行 Hash 运算并找出与之对应的计数器 $C_j(1\leqslant j\leqslant m)$；

步骤 2　提取该数据包的 ACK 标志位的信息，计算并重置 T_A 中计数器 C_j 的值：

· 若该数据包的 ACK 标志位为 0，且 T_A 中 C_j 原本为 0，则 T_A 中 C_j 仍为 0。这说明此源 IP 还没有成功建立过连接，可能是合法的新 IP，也可能是伪造的新 IP；

· 若该数据包的 ACK 标志位为 0，且 T_A 中 C_j 原本为 1，则 T_A 中 C_j 仍为 1。这说明该源地址曾经成功建立了三次握手连接，不属于新的源地址；

· 若该数据包的 ACK 标志位为 1，则 T_A 中 C_j 置 1，此时可能有两种情况：

① 此数据包可能是在建立三次握手连接中的第三步，此时三次握手连接已建立成功，说明该地址是合法的新源 IP；

② 此数据包可能是数据传输过程中的包，由于 TCP/IP 协议规定传输数据之前必须先建立连接，因此三次握手连接一定已经成功建立了，该地址不是新的源 IP。

步骤 3　计算并重置 T_S 中计数器 C_j 的值：

· 如果 T_A 中的计数器 C_j 为 1，则 T_S 中 C_j 置 0，因为 T_A 中 C_j 为 1 说明对应的 IP 地址是合法的，而我们只需要统计新的可疑的源地址出现的次数，不考虑已经确定的正确的源地址；

· 如果 T_A 中 C_j 为 0，则 T_S 中 C_j 加上 a（$a>0$，a 用于标识某伪造源 IP 地址出现了一次），因为 T_A 中 C_j 为 0 说明对应的 IP 地址尚未成功建立三次握手连接，可能是伪造的源地址，需要进行统计。

在正常情况下，三次握手是能够成功的。因此，T_S 中 m 个计数器基本都为 0（除非 Hash 函数引起冲突出现了脏计数器）。即使在正常网络发生拥塞时，三次握手也不能完全成功，但是并不会出现伪造的源地址。因为拥塞发生前很多地址都应该出现过，并已成功建立过连接，这些地址对应的 T_A 中的向量计数器为 1。因此，T_S 中的 m 个计数器仍然基本都为 0。当 SYN Flooding 攻击发生时，网络中会出现大量以前从未出现的源地址，这些地址对应的 T_A 中 C_j 为 0，而 T_S 中向量计数器会出现很多非 0 值，即发生了异常。又由于网络中不可避免地偶尔会发生错误，因此我们将 T_S 周期性地重置，每隔时间 t，T_S 中的向量计数器按 $C_t=(1-\lambda)C_{t-1}(\lambda>0)$ 进行重置。由于 T_A 中的计数器用于记录源 IP 地址的连接历史信息，这些历史信息需要保留下来用作后续的判断，因此 T_A 不参与重置。

我们的检测方法属于源端检测方法。在一个有限的源端局域网范围内，IP 地址的变化范围是有限的，而攻击发生时伪造的源 IP 的范围会远远大于局域网内 IP 地址的变化范围。因此，伪造的源地址被误认为是真实的源地址的概率是很小的。以一个 C 类网络为例，一个伪造的源地址被误认为是真实的源地址的概率为 $P=\frac{1}{n^k}$，设 $n=256$，$k=3$，则 $P\approx 5.96*10^{-8}$，这样的误报率是很低的。真实的源地址互相冲突或者伪造的源地址互相冲突都是可以接受的。因为，真实的源地址 ACK 标志位均为 1，因此出现的次数不进行累计。而伪造源地址的包由于 ACK 标志位均为 0，因此这些伪造地址出现的次数要进行累计。由于目前存在一些 DDoS 攻击工具（如 TFN，TFN2K 等），某个伪造的源 IP 地址会重复出现在多条攻击数据包中，此时一个伪造的源地址对应多个攻击数据包。需要结合伪造的地址以及大量的数据包，快速而有效地检测攻击发生，并且算法也需要避免针对不同攻击工具产生的检测结果的误差。所以，我们的方案是统计单位时间内伪造的源地址出现的总次数，而非总个数，从而将相同源 IP 地址的多个伪造攻击包看成是多个攻击。

4　基于 Hurst 参数的 VTP 法分析

4.1　基于 Hurst 参数的 VTP 分析算法

基于 Hurst 参数的 VTP 分析法具体求解过程如下：

假设时间序列 $X=\{X_t,t=0,1,2,\cdots\}$ 是自相似的。

(1)将序列 X 中的值每 m 个进行分块，并求得每块的平均值 $X_k^{(m)}$ 以形成新的序列 $X^{(m)}=\{X_0^{(m)},X_1^{(m)},X_2^{(m)},\cdots\}$。其中，$X_k^{(m)}=(X_{km-m+1}+\cdots+X_{km})/m$，$k=0,1,2,\cdots,m$ 是连续的整数值，k 用于标记块的序号。例如：未经处理的时间序列即为 $m=1$ 的序列，即 $m=1$ 时 $X^{(1)}=X$，$m=2$ 时 $X^{(2)}=\{(X_0+X_1)/2,(X_2+X_3)/2,\cdots\},\cdots$；

(2)计算时间序列 $X^{(m)}$ 的样本方差，$Var(X^{(m)})=\frac{1}{N/m}\sum_{k=1}^{N/m}(X_k^{(m)})^2-(\frac{1}{N/m}\sum_{k=1}^{N/m}X_k^{(m)})^2$；

(3)选用另外一个 m 值重复步骤(2)；

(4)根据计算出的 $Var(X^{(m)})$ 和 m 值，以 $\log(m)$ 为横坐标，$\log(Var(X^{(m)}))$ 为纵坐标作图，根据图中曲线的斜率 γ 的负值 β（即 $\beta=-\gamma$），计算 $H=1-\frac{\beta}{2}$，这样就可以估算出自相似时间序列 X 的 H 参数（即 Hurst 参数）值。

4.2　序列模型及检测方法

在使用基于 H 参数的 VTP 方法检测 DDoS 攻击时，先利用前面介绍的基于 Bloom Filter 结构的算法提取由新的源 IP 地址出现的次数构成的序列。在表 T_S 中记录了网络中出现的新源 IP 地址的次数信息，T_S 中的每个计数器在监测正常的网络数据时都保持 0 或接近 0。攻击一旦发生后，原有的关于新源地址出现次数的序列模型将被打破，网络中会出现大量原来没有出现过的新的伪造源 IP 地址，此时对应的计数器的值不再保持 0 值，而会变成一个正数。算法中，表 T_S 的值每隔时间 t 刷新一次，每次统计出 T_S 中各个计数器的值之和 X_t。这样，可以形成一个根据时间 t 变化的、T_S 中各个计数器值之和的时间序列，记为 X，$X=\{X_t,t=0,1,2,\cdots\}$。由于 X 具有自相似性（将在 5.1 节给出关于时间序列 X 具有自相似性的证明），符合基于 H 参数的 VTP 分析方法的基本条件，因此我们可以使用 VTP 分析法来求解 H 参数。

为了在较短的周期内计算出 H 参数，实时反映网络流量的变化过程，尽早地发现 DDoS 攻击，借鉴文献[11]中提出的算法，实时在线求解 H 参数。采用由新的源 IP 地址出现的次数构成的时间序列作为被检测序列，设当 $t=t_0$ 时，T_S 中各个计数器值的和为 X_{t_0}；当 $t=t_1$ 时，T_S 中各个计数器值的和为 X_{t_1}；当 $t=t_2$ 时，T_S 中各个计数器值的和为 X_{t_2}，…，则由新源 IP 地址出现的次数构成的时间序列（即被检测的时间序列）为 X，$X=\{X_t,t=t_0,t_1,t_2,\cdots\cdots\}$。设每计算一次 H 参数所需的时间序列长度为 l，则 $X=\{X_t,t=t_{c+1},t_{c+2},\cdots,t_{c+l}\}$。对 X 进行 H 参数求解结束之后，将该序列前端时间长度为 $a(0<a<l)$ 的子序列丢弃，并

在剩余序列的后面添加新采样的时间长度为 a 的子序列作为新的计算序列 X'，即 $X'=\{X_t, t=t_{c+a+1}, t_{c+a+2}, \cdots, t_{c+a+l}\}$。一旦计算出的 H 值发生了明显的变化，则说明发生了攻击。该方法在计算 H 参数时仅仅需重新采样长度为 a 的序列，而不是 l，节约了时间，提高了效率，反映了当前网络流量的变化过程。因此，该方法不需要进行太复杂的计算就能够在较短的时间内计算出 H 参数，并能够准确地反映网络特征实时变化的情况。这种方法的具体工作过程如图 3 所示。

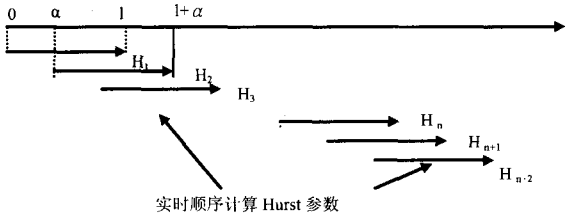

图 3　实时计算 Hurst 参数示意图

通过选择最佳参数 l 和 a 可以实现低算法复杂度和短检测时间。a 用来将上一个时间段的检测序列偏移到当前时间序列，选择的 a 越小，算法复杂度越大，但是检测到攻击的时间越短。l 是计算一次 H 参数所需要的时间序列长度，l 越大，算法复杂度越小，但检测时间越长。

5　实验结果及分析

实验采用 MIT 林肯实验室的 DARPA 入侵检测评估数据集。实验中的正常数据集来自该实验室于 1999 年提供的正常数据集 inside[12]，攻击数据集来自该实验室于 2000 年提供的攻击场景测试集 LLS_DDOS_1.0[12]。

5.1　真实网络中 X 序列的自相似性

使用基于 H 参数的 VTP 分析法的前提条件是被检测的时间序列具有自相似性，因此本文算法成立的前提是真实网络环境中由新的源地址出现的次数构成的时间序列具有自相似性。当仅仅需要知道时间序列是否具有自相似特性时，VTP 分析法是有效的[13]。下面采用 VTP 分析法来说明真实网络中由新的源 IP 地址出现的次数构成的时间序列 X 具有自相似性。针对不同的 $m(m=1,2,3,\cdots,9,10,20,\cdots,100)$，求出其平均流量的方差，然后分别以 $\log(m)$ 和 $\log(Var(X^{(m)}))$ 作为横、纵坐标绘制 V-T 图。若图形的斜率 $\gamma\in(-1,0)$，则表明该时间序列具有自相似性。而由 $H=1-\frac{\beta}{2}, \left(\beta=-\gamma, \frac{1}{2}<H<1\right)$，即可估算出自相似时间序列的 H 参数值。对正常数据集 X 进行自相似性测试的结果如图 4 所示。从图中看出，X 的斜率 γ 在正常的范围内小幅度波动。

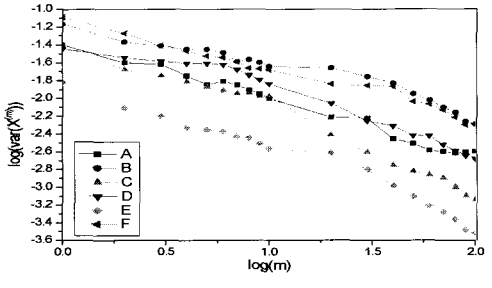

图 4　正常网络环境中 X 序列的 $\log(m)-\log(Var(X^{(m)}))$ 图

采集的正常网络流量中由新的源 IP 地址出现的次数构成的时间序列，记为 X_A。而图 4 中的曲线 A 是根据 X_A 计算出相应的 $\log(m)$ 和 $\log(Var(X^{(m)}))$ 值绘制的。同理，曲线 B,C,D,E,F 均是根据正常数据集 inside 提供的其它正常网络流量处理得到的。从图 4 可以看出，这些曲线的斜率变化是基本一致的，各个序列对应的斜率 γ 如表 1 所示。

表 1　正常网络环境中 X 序列的斜率值

检测序列	A	B	C	D	E	F
斜率 γ	−0.6	−0.56	−0.86	−0.62	−0.85	−0.6

由表 1 可以看出，正常网络中，各序列 X_i 的斜率 γ 都有 $-1<\gamma_i<0, i=A,B,C,D,E,F$，即由正常网络得到 X 序列具有自相似的特性。可见，本文方法符合基于 H 参数的 VTP 方法的计算的前提条件。因而，我们采集初始序列，使用第 4 节所述算法更新序列，同时计算各序列的 H 参数值，通过 H 参数值的变化检测 DDoS 攻击的发生。检测过程见 5.2 节。

5.2　对 SYN Flooding 攻击的实时在线检测

由上述可知，由正常网络中的 X 序列求出的 $\log(m)-\log(Var(X^{(m)}))$ 图形的斜率 $\gamma_i\in(-1,0)$，说明 X 序列具有自相似的特性，因而用 OL-VTP 方法求出的 H 参数仍然应该在正常范围内小幅度波动，求解结果如图 5 所示。设序列的时间长度为 l，序列丢弃前端时间长度为 a，聚合度为 m，实验中取 $l=1000$ms，$a=100$ms，m 取值为 1,2,3,…,9,10,20,…,100 。A 序列作为初始序列，B 序列是由 A 序列丢弃前端时间长度为 a 的数据，再在后面重新添加长度为 a 的新采集数据构成的新时间序列，也即每次更新掉原来序列的 10%。同理，C,D,E 序列分别由 B,C,D 经处理所得。图 5 中各序列对应的曲线是通过分别计算出该序列对应的 $\log(m)$ 和 $\log(Var(X^{(m)}))$ 值以后绘制的。从图 5 可以看出，这些曲线的斜率变化趋势基本是一致的，即斜率 γ 值基本差不多，只在小幅度内波动。各个序列的 H 参数值如表 2 所示。可见，正常网络行为下所产生的 X 序列的 H 参数值是基本稳定的，其值也都在正常的自相似模型范围内，即有 $\frac{1}{2}<H<1$。

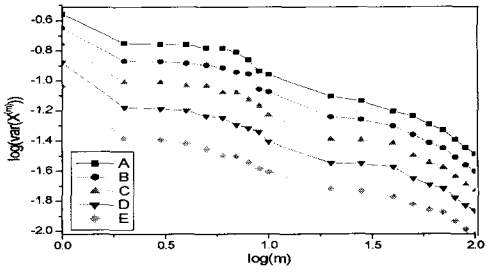

图 5　用 OL-VTP 方法求解的正常网络环境中 X 序列的 $\log(m)-\log(Var(X^{(m)}))$ 图

表 2　用 OL-VTP 方法计算正常网络环境中 X 序列的 Hurst 参数值

检测序列	A	B	C	D	E
H 参数值	0.768	0.762	0.756	0.753	0.753

下面分析当 DDoS 攻击发生时的网络行为对 X 序列的 H 参数的影响，以及 H 参数值的变化情况。由于 DDoS 攻击流量是巨大的，为了更详细地说明 DDoS 攻击流量对 H 参数的影响，我们取 $l=1000$ms，$a=10$ms，m 取值为 1,2,3,…,9,10,20,…,100，列出 $A,B,C,\cdots,J$ 共 10 个具有代表性的序列，每次更新前一个序列的 1%得到新序列。图 6 是用含有 DDoS 攻击的异常数据集计算出的部分序列的 $\log(m)-\log$

($Var(X^{(m)})$)图。

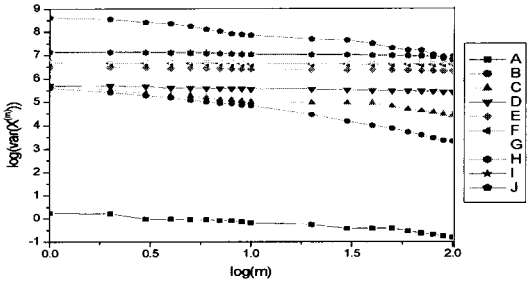

图6 含有DDoS攻击流量的X序列的$\log(m)-\log(Var(X^{(m)}))$图

在图6中，各个序列中DDoS攻击流量占整个网络流量的比例情况是：序列A为0%，序列B约为3%，序列C约为8%，序列D约为14%，…，序列E约为88%，序列F约为94%，序列G约为95%，序列H约为98%，序列I约为99%，序列J则完全由攻击流量构成，即100%。从图中可以看出序列B和序列J的曲线斜率变化要明显不同于其它的曲线。

X_A为初始序列，攻击刚开始发生时，如序列X_B，由于攻击流量的出现，整体上X序列的倾斜程度变大、斜率变小，即整体网络中当前的统计序列变得和以往不相似。但是随着攻击的持续，$\log(m)-\log(Var(X^{(m)}))$图的曲线逐渐缓和，$H$参数值越来越大，直到网络中完全充斥着攻击流量，这时X序列会因正常流量的消失而变得和以前有差异，相似程度变小，即图6中序列J的斜率变小。

攻击刚开始发生时，由于掺杂了攻击流量，对新序列的H参数值的影响是很大的。然后，随着DDoS攻击流量逐步增大，对H参数的影响逐渐变小。如表3所示，当攻击刚刚开始，攻击流量仅占网络流量的3%时，H参数会由正常的0.74骤减到0.43，降到了0.5以下，这说明网络中有大量以前从未出现过的新IP地址涌入，打破了原有的相似状态。攻击开始时，网络中X序列的相似程度呈下降趋势。随着攻击的继续进行，网络中新源地址在整个网络IP地址中持续占有较高的比例，导致相似程度的不断增大，H参数值也迅速增大。当攻击流量占整体网络流量的8%时，H参数已经达到了0.68，这个数值已经接近正常情况下的H参数值了，如表3所示C序列的H参数值。这表明攻击开始一段时间后，网络中X序列的相似程度呈上升趋势。D序列和E序列之间还存在着其它序列，这里不一一列出结果，这些被省序列的H参数值都介于0.93～0.96之间，并且也呈现出上升趋势。当攻击流量占整体网络流量的95%时，H参数上升到了0.99，此时网络中X序列的相似程度最大，也是本实验中最大的H参数值，网络中X序列的稳定性达到了最高峰。当攻击流量占整体网络流量的比例超过95%时（如H序列、I序列等），H参数值并没有继续上升，而是略有下降，这是由于随着DDoS攻击流量的不断增大，网络流量的突发性被逐渐削弱，从而X序列的相似程度不再增大，而是逐渐处于稳定状态。当网络中攻击流量达到100%时，出现已经没有正常流量引起新源IP地址出现的情况，导致不再具有突发性，从而相似程度下降很快，H参数的值由0.94下降到了0.6。但是，由之前攻击流量引起的新源IP地址已经占据了网络中新源地址的绝大部分，所以大体上还是相似的，H参数值仍然大于0.5。

表3 用OL-VTP方法计算含DDoS攻击流量的X序列的Hurst参数值

检测序列	A	B	C	D	…	E
H参数值	0.74	0.43	0.68	0.93	…	0.96
检测序列	F	G	H	I	J	
H参数值	0.98	0.99	0.96	0.94	0.6	

从以上序列分析可知，当正常网络环境中开始掺杂DDoS攻击流时，X序列的H参数值会突降。随着DDoS攻击引起的新源IP地址出现的次数所占比例的持续增大，H参数的值是逐步增大的，并且是在初期变化明显，而后变化逐渐趋小。但当网络中完全是攻击数据包时，X序列的H参数值会有一个突降的过程。因此，根据H参数值的变化，可以判断出DDoS攻击的发生。

相比较已有的一些解决方案，本算法的优点主要体现在两个方面：一是算法不需要设定阈值，能够实时在线检测攻击，做到早期检测，并节省了检测的时间；二是在前期采用的攻击包特征提取方法的保证下，算法在检测过程中不会将由于正常网络拥塞出现的不完整连接看成是伪造源IP地址的DDoS攻击，误报率低。因为，算法的误差仅在攻击数据包与正常数据包出现冲突时才可能产生。而攻击包与正常包会在两种情况下冲突：情况1，同一个被伪造的IP源地址被多次使用，则伪造的包被多次累加，计算结果完全正确；情况2，伪造的源IP与合法的源IP被Hash后，击中同一个位置，若正常的IP先出现，伪造的IP后出现，则不影响结果，若伪造的IP先出现，而正常的IP后出现，则会产生误判，而根据前面分析可知，这种误判的可能性是很小的。

结束语 本文提出了一种有效的、快速的、轻量级的、适合大流量背景的SYN Flooding攻击源端检测方法，可以实时在线发现攻击，节约了检测时间，少量攻击包的出现也能通过Hurst参数的明显变化表现出来，因而更适合大流量背景的攻击。我们采用的算法可以在只出现少量伪造数据包的情况下准确发现攻击，适用于源端检测，也为监控攻击数据包的来源以及通知受害端进行防御赢得了宝贵的时间。另外，算法消耗的计算资源较少，所以对网络的性能也不会有太大的影响。由于正常的网络拥塞和DDoS攻击具有很多共同的特点，如何快速准确地区分攻击和正常的网络拥塞一直是个难题。而我们的算法不会对正常的网络拥塞产生误报，具有一定的现实意义。

现有的DDoS攻击检测算法一般都是在攻击已经发生的情况下才能检测成功，而此时的攻击流量或多或少都已经对目标主机或目标网络造成了危害。并且绝大多数时候，即使在检测成功的情况下，仍然无法区分正常数据流和攻击数据流。因此，结合对DDoS攻击检测的研究，还需要进一步做好如何在攻击尚未成熟前进行正确预警的研究工作，以及如何开展在攻击被检测出来后，具体确定l和a的值，并准确地过滤攻击包，阻止后续攻击的研究工作。

参考文献

[1] Moore D, Volker G, Savage S. Inferring Internet Denial-of-Service Activity[C]//Proceeding of the 2001 USENIX Security Symposium. Aug. 2001: 9-22

[2] Leland W E, Taqqu M S, Willinger W, et al. On the self-similar nature of Ethernet traffic (extended version)[J]. IEEE/ACM Trans on Networking, 1994, 2(1): 1-15

（下转第162页）

其中 $E_y=\{x\in X|R_L(A(x),B(y))<\alpha\}$。我们通过对此公式的分析，参照前文解决基于算子 R_L 的三Ⅰ支持度算法连续性问题的方法，可以得到以下的结论：

定理 7 FMP 问题的基于算子 R_L 的反向三Ⅰ支持度算法在 A 处具有连续性。

因为 FMP 问题的基于算子 R_L 的反向三Ⅰ算法是还原算法[14]，所以同样有下面的推论：

推论 7 FMP 问题的基于 R_L 的反向三Ⅰ算法在 A 处具有逼近性。

接下来我们讨论 FMT 问题的基于 R_L 的反向三Ⅰ支持度算法的连续性问题。FMT 问题的基于 R_L 的反向三Ⅰ支持度算法的计算公式[14]，如下：

$$A^*(x)=\bigvee_{y\in K_x}(B^*(y)-R_L(A(x),B(y))+\alpha),x\in X$$

其中：$K_x=\{y\in Y|R_L(A(x),B(y))<\alpha\}$。同理可以得到：

定理 8 FMT 问题的基于 R_L 的反向三Ⅰ支持度算法在 B 处具有连续性。

因为 FMT 问题的基于 R_L 的反向三Ⅰ算法是还原算法[14]，所以下面的推论也成立：

推论 8 FMT 问题的基于 R_L 的反向三Ⅰ算法在 B 处具有逼近性。

根据 FMP 和 FMT 问题的基于 R_0 的反向三Ⅰ算法的计算公式[7]，我们有：

推论 9 FMP 问题的基于 R_0 的反向三Ⅰ支持度算法在 A 处具有连续性。

推论 10 FMT 问题的基于 R_0 的反向三Ⅰ支持度算法在 B 处具有连续性。

关于模糊控制系统中使用较多的几个蕴涵算子的 FMP 和 FMT 问题的反向三Ⅰ算法的计算公式，已经得出[15]。我们从这些计算公式可以得到下面两个结论：

推论 11 FMP 问题的基于 R_G，Yager 算子($R(x,y)=y^x,x,y\in[0,1]$)，Reichenbach 算子($R(x,y)=(1-x)+xy,x,y\in[0,1]$)，Kleene-Dienes 算子($R(x,y)=(1-x)\vee y,x,y\in[0,1]$)的反向三Ⅰ算法在 A 处具有连续性。

推论 12 FMT 问题的基于 R_G，Yager 算子、Reichenbach 算子、Kleene-Dienes 算子的反向三Ⅰ算法在 B 处具有连续性。

参 考 文 献

[1] 王国俊. 模糊推理的全蕴涵三Ⅰ算法. 中国科学(E 辑)，1999，29(1)：43-53

[2] 王国俊. 非经典数理逻辑与近似推理. 北京：科学出版社，2000

[3] 裴道武. FMT 问题的两种三Ⅰ算法及其还原性. 模糊系统与数学，2001，15(4)：1-7

[4] 裴道武. 模糊推理全蕴涵算法及其还原性. 数学研究与评论，2004，24(2)：359-368

[5] Pei D W. Unified full implication algorithms of fuzzy reasoning. Information Sciences，2008，178(2)：520-530

[6] 裴道武. 关于模糊逻辑与模糊推理逻辑基础问题的十年研究综述. 工程数学学报，2004，21(2)：249-258

[7] 宋士吉，吴澄. 模糊推理的反向三Ⅰ算法. 中国科学(E 辑)，2002，32(2)：230-246

[8] 徐蔚鸿，谢中科，等. 两类模糊推理算法的连续性和逼近性. 软件学报，2004，15(10)：1485-1492

[9] 于鹏，王国俊. 基于正则蕴涵算子的三Ⅰ算法的性质. 陕西师范大学学报(自)，2007，35(2)：14-17

[10] 吴望名. 模糊推理的原理和方法. 贵阳：贵州科技出版社，1994

[11] 李绍稳，熊范纶，等. 专家系统中的特征展开三Ⅰ算法模糊推理模型及其应用. 模式识别与人工智能，2001，14(3)：272-275

[12] 马盈仓，何华灿. 一类剩余格上的三Ⅰ算法. 计算机科学，2004，31(5)：127-129

[13] 彭家寅，侯健，李洪兴. 基于某些常用蕴涵算子的反向三Ⅰ算法. 自然科学进展，2005，15(4)：404-410

[14] 侯健，尤飞，李洪兴. 由三Ⅰ算法构造的一些模糊控制器及其响应能力. 自然科学进展，2005，15(1)：29-37

[15] Song S J，Feng C，Lee E S. Triple I method of fuzzy reasoning. Computers and Mathematics with Applications，2002，44(2)：1567-1579

[16] 何映思，全海金，邓辉文. 具有还原性的多重多维模糊推理算法. 计算机科学，2007，34(4)：145-148

[17] Zhao Z H，Li Y J. Reverse triple I method of fuzzy reasoning for the implication operator R_L. Computers and Mathematics with Applications，2007，53：1020-1028

(上接第 113 页)

[3] Paxson V，Floyd S. Wide area traffic：the failure of poisson modeling[C]// Proc. ACM Sigcomm'94. 1994：257-268

[4] Hu G，Dolzer K，Gauger C M. Does Burst Assembly Really Reduce the self-similarity[J]// Conference on Optical Fiber Communication(OFC 2003). Technical Digest Series，2003，86：124-126

[5] 何慧，张宏莉，张伟哲，等. 一种基于相似度的 DDoS 攻击检测方法[J]. 通信学报，2004，25(7)：176-184

[6] 李金明，王汝传. 基于 VTP 方法的 DDoS 攻击实时检测技术研究[J]. 电子学报，2007，35(4)：791-796

[7] Garrett M . Contribution toward real - time service on packet switched networks[D]. New York：Columbia University，1993

[8] Wornell G W , Oppenheim A V . Estimation of fractal signals from noisy measurements using wavelets[J]. IEEE Trans on Signal Processing，1992，40(3)：611-623

[9] Zhang H F，Shu Y T，Yang O. Estimation of Hurst parameter by variance-time plots Communcations[C]// IEEE Pacific Rim Conference on Computers and Signals Proceeding' 10 years PACRIM 1987-1997-Networking the Pacific Rim'. 1997(2)：883-886

[10] Jung J，Krishnamurthy B，Rabinovich M. Flash Crowds and Denial of Service Attacks：Characterization and Implications for CDNs and Web Sites. Honolulu，Hawaii，USA 2002

[11] Hagiwara T，Doi H，et al. High-speed calculation method of the Hurst parameter based on real traffic[C]// LCN 2000，Proceedings 25th Annual IEEE Conference on Local Computer Network. 2000：662-669

[12] http://www.ll.mit.edu/IST/ideval/data/2000/2000_data_index.htm

[13] Paxon V. Fast Approximation of Self-similar Network Traffic [R]. Technical Report，LBL36750. University of California，Berkeley，Apr. 1995