第40卷第1期 中国科学技术大学学报 Vol. 40, No. 1
2010年1月 JOURNAL OF UNIVERSITY OF SCIENCE AND TECHNOLOGY OF CHINA Jan. 2010

文章编号:0253-2778(2010)01-0103-06

一类慢速拒绝服务攻击的防御方法

董 阔,杨寿保

(中国科学技术大学计算机科学技术系,安徽合肥 230026)

摘要:与高速率的拒绝服务攻击相比,慢速拒绝服务攻击难以被现有的拒绝服务攻击检测工具检测出来,其隐蔽性更高.通过分析慢速拒绝服务攻击在不同网络环境中对网络性能的影响,提出使用动态调整超时重传定时器的策略来防御此类攻击.实验表明,此类动态调整策略可有效抵御慢速拒绝服务攻击,与当前网络所使用的策略相比,在攻击周期小于2 s时,网络吞吐量提升了300%以上.

关键词:慢速拒绝服务攻击;拥塞控制;动态调整;超时重传定时器

中图分类号:TP393.08 **文献标识码**:A doi:10.3969/j.issn.0253-2778.2010.01.018

A method for defending low-rate denial of service attacks

DONG Kuo, YANG Shoubao

(*Department of Computer Science and Technology, University of Science and Technology of China, Hefei* 230026, *China*)

Abstract: Compared with high-rate denial of service attacks, low-rate denial of service attack is hard to detect by the existing intrusion detection systems, because it is much more concealed. The network performance with low-rate attacks in different environments was analyzed; two novel dynamic adjusting strategies for retransmission timeout were also proposed. Experiments indicate that the proposed method can effectively fight off low-rate denial of service attacks. Compared with the strategy currently used on the Internet, it can enhance the network throughput above 300% when the attack period is less than 2 seconds.

Key words: low-rate denial of service attack; congestion control; dynamic adjustment; retransmission timeout timer

0 引言

拒绝服务攻击的目的是恶意的阻止合法用户的服务或者严重影响其服务质量,它会消耗网络、服务器或者是终端机的资源.拒绝服务攻击中消耗的资源对象包括网络带宽、服务器或路由器的CPU时间、服务器的处理容量和专门的协议数据单元等.拒绝服务攻击的类型包括消耗服务器操作系统的协议数据单元的TCP SYN洪水攻击,以广播地址为发送目标的ICMP包反射攻击,DNS解析请求风暴攻击等等.上述攻击的共同点是利用被攻破主机为跳板,发送大量突发数据包到被攻击对象.虽然这些攻击可以产生严重的攻击效果,但是也具有明显的特征,可以通过一定的方法检测到.

收稿日期:2008-04-18;**修回日期**:2008-05-18

基金项目:国家自然科学基金(60673172)和安徽省自然科学基金(070412045)资助.

作者简介:董阔,男,1981年生,博士生.研究方向:网络安全,密码学.E-mail:dongkuo@mail.ustc.edu.cn

通讯作者:杨寿保,教授.E-mail:syang@ustc.edu.cn

Kuzmanovic 等提出了一种新颖的攻击方法——低速率拒绝服务攻击[1]，简称慢速拒绝服务攻击. 慢速拒绝服务攻击针对 TCP 协议拥塞控制中存在的漏洞，通过周期性的发送脉冲 UDP 数据流，降低共享同一瓶颈链路的 TCP 流的带宽，这种攻击利用了 TCP 协议的漏洞，使用较小的代价即可以获得很好的攻击效果. 虽然此类攻击并未广泛出现，但是由于攻击的平均流量不高，攻击持续时间短，传统的大流量数据包的拒绝服务攻击的检测工具无法有效检测；而在路由器上部署的 AQM 算法如 RED，SRED 等都无法减轻此类攻击的效果. 同时，由于攻击的发起并不需要耗费太多资源，此类攻击对网络存在较大威胁.

本文通过深入分析慢速拒绝服务攻击原理及不同网络参数对攻击效果的影响，总结攻击时网络中数据包特征，提出动态调整超时重传定时器的策略来防御慢速拒绝服务攻击，并通过实验床测试比较了几种不同策略的性能. 实验证明，我们提出的动态调整策略可以在慢速拒绝服务攻击发生很好的保证网络的性能.

1 相关工作

文献[1]最先关注慢速拒绝服务攻击的研究，文中提出了随机化超时重传定时器下限(RTO_{min})的解决方法，这种方法虽然可以减轻攻击的影响，但是在没有遭受慢速拒绝服务攻击的情况下，随机改变 RTO_{min} 会使网络性能受到影响，而且这种随机变化无法避免 TCP 发送端遭受慢速拒绝服务攻击.

文献[2]根据慢速拒绝服务攻击提出了针对 TCP 协议拥塞避免中加性增，乘性减(additive increase, multiplicative decrease，AIMD)方法的攻击，使用持续时间更短、周期更小的 UDP 脉冲攻击 AIMD 过程，并提出了两阶段检测方法. 首先使用小波变换分析网络到来的流量与发送的 TCP ACK 流量的变化，然后在第二阶段使用累积和算法根据 ACK 数据包变化的规律在线检测攻击的发生.

Guirguis 提出了降低服务质量(reduction of quality，RoQ)攻击[3]，定义了攻击强度的概念. RoQ 攻击利用系统的动态特性，通过发起一些精心安排的低强度服务请求数据流，使得被攻击的系统变得低效，不稳定. 此类攻击的一个例子即是对 AIMD 过程的攻击，另外一个例子是对采用 RED 算法的路由器进行慢速拒绝服务攻击，结果造成路由器的丢包率增加. Guirguis 在文献[4]中使用相同的模型对慢速拒绝服务攻击进行了分析，在工作展望中提出了设计可控制的拥塞控制协议的观点.

Yu Chen 提出的 HAWK 算法[5]在路由器上建立 UDP 流表，对经过的 UDP 流进行记录，如果超过预先设置的阈值则对相应的 UDP 源进行封禁，此方法可以减轻分布式慢速拒绝服务攻击的效果. 研究检测慢速拒绝服务攻击的工作还有文献[6]中提出的使用动态时间封装(dynamic time wrapping，DTW)的方法进行流量特征检测，对取样流量与预定义的攻击特征进行比较. 但是 DTW 方法检验的误报率很高，而且需要链路中的背景流量在一个较低的范围内，否则检验结果较差.

2 慢速拒绝服务攻击的原理与分析

2.1 慢速拒绝服务攻击原理

在当前的互联网中使用的传输层协议主要有可靠的 TCP 协议和不可靠的 UDP 协议，前者使用确认(acknowledgement)以及拥塞控制和流量控制机制来保证数据传输的可靠性及不同流之间的公平性，是一个面向流的协议. 随着网络技术的不断发展，TCP 协议出现了很多变种，如 TCP Reno，TCP Vegas，SACK 等，也出现了一些在 TCP 协议基础上改进的拥塞控制协议如 FAST TCP，XCP 等. 在这些协议的应用中，发送端都维持一个缓冲区，此缓冲区常被叫做拥塞窗口，不同的协议对于拥塞窗口的管理策略不同. 目前使用最广泛的 TCP Reno 协议采用慢启动和 AIMD 策略来管理发送端的缓冲区，同时要维持重传超时计时器(retransmission timeout timer，RTO). 当发送端在 RTO 超时之前没有收到接收端返回的应答，则重新进入慢启动过程，RTO 大小变为原来的两倍，慢启动阈值(slow start threshold，ssthresh)大小变为超时前拥塞窗口的一半.

RTO 的计算采用式(1)～(3)，发送端每次成功收到应答包(ACK)后重新计算 RTO 值. 从公式来看，RTO 的大小由链路 RTT 的变化决定，另外，RTO 具有常量下界 RTO_{min}，根据 RFC2988[7]建议，RTO_{min} 的取值为 1 s. 设置 RTO_{min} 常量的最初目的是为了提高网络系统性能，然而，这个常量的设置导致了慢速拒绝服务攻击的出现.

$$RTTVAR = (1-\beta)RTTVAR + \beta \mid SRTT - R' \mid \quad (1)$$

$$\mathrm{SRTT}=(1-\alpha)\mathrm{SRTT}+\alpha R' \quad (2)$$

在式(1),式(2)中 $\alpha=1/8,\beta=1/4$

$$\mathrm{RTO}=\max(\mathrm{RTO}_{\min},\mathrm{SRTT}+\max(G,4\mathrm{RTTVAR})) \quad (3)$$

慢速拒绝服务攻击基于没有流控的 UDP 协议,利用了 TCP 协议重传超时及慢启动过程的漏洞. 攻击者向攻击目标以一定速率周期性地发送 UDP 数据包,通过调节发送速率,造成 UDP 数据流量和链路上的背景流量之和超过链路所能承载的数据流量,从而造成链路拥塞,使所有共享此链路的 TCP 数据流全部由于重传超时进入慢启动阶段,链路上所有的 TCP 流发送端的拥塞窗口被减小到一个最大数据单元(maximum segment size, MSS). 攻击者的再次攻击使得在发送端的拥塞窗口仍然处于很低的水平时,链路继续发生拥塞,所有 TCP 流发送端的拥塞窗口再次被减小到一个 MSS. 如此重复攻击过程,造成网络性能的极大下降. 这种攻击还可以被推广为分布式拒绝服务攻击,由于攻击的源头不确定,更加大了检测的难度.

2.2　对慢速拒绝服务攻击的分析

文献[1]的研究主要关注链路 TCP 的吞吐量变化,而没有讨论攻击数据流持续时间和攻击周期变化时,网络中的数据发送流程以及 TCP 发送端的拥塞窗口的变化. 在本节中我们将对上述场景做进一步的分析. 我们对慢速拒绝服务攻击的分析基于以下假设:与传输延迟相比,可以忽略节点处理延迟;攻击时在靠近接收端的链路上数据包被阻断;不考虑推迟应答的影响;接收端每接收到一个发送端包就会发送 ACK 包. 令攻击的目标链路上传输的 TCP 流的 RTT 范围区间为$[t_1, t_2]$,所有 TCP 流的 $\mathrm{RTO}_{\min}$相同,攻击的周期为 T,持续时间为 L,如图 1(a)所示,在 $T>\mathrm{RTO}_{\min}$的前提下:

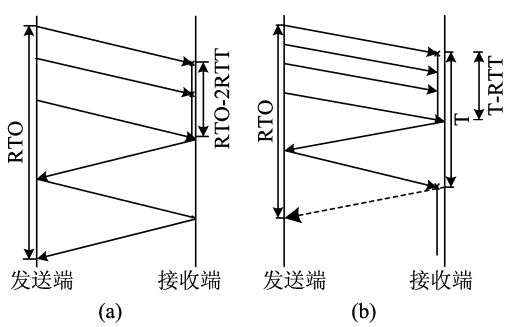

图 1　慢速拒绝服务攻击过程中数据包的流程

Fig. 1　Data packets flow path during low-rate denial of service attack

①$L<\mathrm{RTO}_{\min}-2t_2$,在这种情况下攻击是不成功的,此时从接收端发送过来的 ACK 在超时前可以到达发送端,发送端的拥塞窗口没有降低;

②$\mathrm{RTO}_{\min}-2t_2<L<\mathrm{RTO}_{\min}-2t_1$,此时有一部分 RTT 较小的 TCP 数据流仍可以正常发送,而 RTT 较大($t>\mathrm{RTO}_{\min}-T$)的 TCP 流将受攻击影响而进入慢启动过程;

③$L>\mathrm{RTO}_{\min}-2t_1$,此时链路上所有的 TCP 流均将受到攻击的影响,拥塞窗口下降,进入慢启动过程.

若 $T<\mathrm{RTO}_{\min}$,不失一般性,考虑单流环境下的攻击,并设此流的 RTT 为 t,攻击分析图如图 1(b)所示,此时:

①$L<T-t$,攻击不成功,从接收端发送过来的 ACK 在 RTO 超时前可以到达发送端,发送端的拥塞窗口没有降低;

②$L>T-t$,攻击成功,TCP 流发送端拥塞窗口下降,进入慢启动过程.

从以上的分析中可以看出,相同攻击周期条件下,攻击持续时间越长,对网络性能的影响越大. 继续分析攻击周期对网络性能的影响,此时假设每次攻击都可以使发送端进入慢启动过程:

①T 小于 $\mathrm{RTO}_{\min}$,此时如果 $T|\mathrm{RTO}_{\min}$,则网络性能将降到最低,拥塞窗口始终保持在 1 个 MSS 大小;若 $T|\mathrm{RTO}_{\min}$不成立,假设$(k-1)T<\mathrm{RTO}_{\min}<kT, T'=kT-\mathrm{RTO}_{\min}$. 此时拥塞窗口的收敛值为 $2^t+t=T-\mathrm{RTO}_{\min}$方程的解,由文献[8]可知此方程的解为

$$t=\log_2(\mathrm{lambert}W(2^{T-\mathrm{RTO}}\ln 2)/\ln 2) \quad (4)$$

②T 略大于 $\mathrm{RTO}_{\min}$,此时如图 2(a)所示,当发送端结束超时等待,仍在慢启动过程中时,又一次攻击使得发送端再次进入超时等待. 需要注意的是每次重传超时后,ssthresh 将被设为超时前拥塞窗口大小的一半,故从第二次攻击后,发送端将经历慢启动过程和拥塞避免过程,拥塞窗口的收敛值表示形式与公式(4)中相同.

③若攻击的周期间隙较长,发送端经过了慢启动过程,则由于 sshthresh 的减小,慢启动持续时间缩短,发送端在此后的攻击间隙可以再次经历慢启动过程和拥塞避免过程,如图 2(b)所示. 对图 2(b)中的 TCP 流吞吐量进行分析,遭受攻击后第一个慢启动过程持续时间为 $\log_2 \mathrm{ssthresh}$ 个 RTT,最终拥

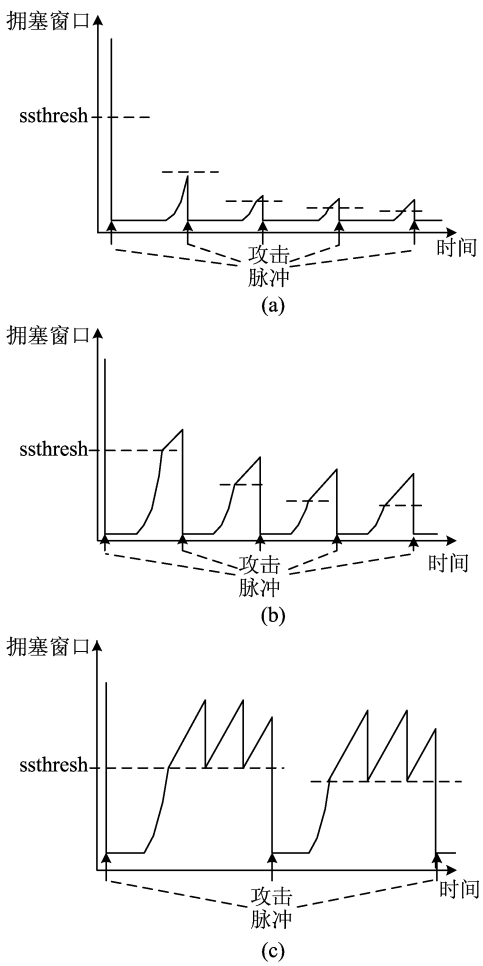

图 2 攻击中拥塞窗口的变化曲线

Fig. 2 The change curve of congestion window during attack

塞窗口可以达到 $(\text{ssthresh}+\frac{T-\text{RTO}_{\min}}{\text{RTT}}-\log_2\text{ssthresh})$ 个 MSS 大小;第二次攻击使 ssthresh′ 变成受攻击前拥塞窗口的一半大小,即 $\text{ssthresh}'=\frac{1}{2}(\text{ssthresh}+\frac{T-\text{RTO}_{\min}}{\text{RTT}})$ 个 MSS,此时慢启动持续时间为 $\lceil\log_2\text{ssthresh}'\rceil$ RTT,经过线性增长,最后发送端的拥塞窗口可以达到 $(\text{ssthresh}'+\frac{T-\text{RTO}_{\min}}{\text{RTT}}-\lceil\log_2\text{ssthresh}\rceil)$ 个 MSS 大小,此状态下拥塞窗口的收敛值与公式(4)中相同.

④如果攻击周期间隔足够长,使得拥塞窗口可以经历完整的慢启动及拥塞避免过程,此时攻击对于网络性能的影响较小,如图 2(c)所示.

由以上分析可见,慢速拒绝服务攻击可以造成网络性能的极大下降.其本质原因是由于无流控的 UDP 与 TCP 在竞争链路资源时处于优势地位.但是与防御基于其他协议如 ICMP 协议的方法不同,完全封禁 UDP 流量是不切实际的,因为虽然当前 Internet 中使用的主要是 TCP 协议,但是仍有相当一部分应用基于 UDP 协议.因此根据慢速拒绝服务攻击的特点设计出防御此类攻击的方法是非常重要的.

3 基于动态调整的慢速拒绝服务攻击防御方法

目前,对于慢速拒绝服务攻击的检测方法主要集中在频域检测方向,在数据包层次做检测的准确性不高,而且在路由器上部署的检测算法容易造成路由器处理数据包的性能下降,导致没有攻击发生时网络的性能下降.本文提出了一种数据包的修改方法及两种基于此修改方法的 $\text{RTO}_{\min}$ 动态调整策略,可以在不影响网络性能的前提下有效地减轻慢速拒绝服务攻击的影响.

3.1 $\text{RTO}_{\min}$ 的动态调整策略

在我们的方案中,在路由器上对发送给发送端的 ACK 数据包做细小的修改,将路由器上处理的 UDP 数据包与 TCP 数据包的比例用三个比特位反馈给 TCP 发送端.这三个比特使用目前没有启用的 IP 数据包头中第二个字节的前三个比特,可以表示的数字为 0～7,用来表示路由器处理的 UDP 数据包与 TCP 数据包的比值,分别可以表示的比例为 0,1∶1,2∶1,3∶1,4∶1,5∶1,6∶1,7∶1,这样可以将路由器上所处理的 UDP 数据包占全部数据包的比例分为 0, 1/2, 2/3, 3/4, 4/5, 5/6, 6/7, 7/8 这 8 种.在 TCP 发送端接收到反馈的 ACK 后,从中读取出对应的数据位,然后计算出相应的比例用于动态调整策略.

我们提出两种新的调整方法用于防御慢速拒绝服务攻击.第一种是基于阈值的调整策略,第二种是线性调整策略.

我们提出的第一个策略是基于阈值的调整方法(threshold-based adjusting strategy, TBAS).在此策略中,首先使用一个较大的 $\text{RTO}_{\min}$ 值 t_s,然后当收到的 UDP 包比例 k 增多到超过一个阈值 S 时,将 $\text{RTO}_{\min}$ 的值减小至 t_{e1},这样,$\text{RTO}_{\min}$ 的值可以表示成

$$\text{RTO}_{\min}=\begin{cases}t_s, & k<S\\ t_{e1}, & \text{其他情况}\end{cases} \tag{5}$$

k 可以由路由器反馈的信息计算得到,本文中取 $S=3/4$.

第二个调整策略是线性调整策略(linear adjusting strategy, LAS),此策略提出的基于阈值的调整策略不同. 在 RTO_{min} 随着网络中攻击数据包的增加而实时的变化. 在收到 UDP 数据包比例 $k'<\alpha$ 时,设置 RTO_{min} 的值为 t_s;当 $k'\geqslant\beta$ 时,设置 RTO_{min} 的值为一个较小的 t_{e2},在其他情况下,RTO_{min} 的值在 t_s 与 t_{e2} 之间基于 k' 的值线性变化,此时 RTO_{min} 的值可以表示成

$$RTO_{min}=\begin{cases}t_s, & k<\alpha\\ t_s+(t_{e2}-t_s)k, & \alpha\leqslant k<\beta\\ t_{e2}, & k\geqslant\beta\end{cases}\quad(6)$$

在本文中取 $\alpha=1/2,\beta=7/8$.

从对路由器性能的影响方面来看,由于这种修改 IP 包头的策略只是修改三个比特的值,带来的负载是非常小的. 由于计算数据包比例所需要的值可以很容易从 SNMP 协议的统计数据中获得,而路由器基本上都会开启 SNMP 或对应的网络管理协议来对网络进行监控和管理,所以这个修改并没有过多的增加路由器的开销.

3.2　实际测试与讨论

我们采用实际网络测试的方式来评测动态调整策略的性能. 实验床组织结构如图 3 所示,攻击方和正常用户均为 P Ⅲ 1.0 G CPU, 内存 256 M SDRAM,10/100 M 自适应网卡,安装的操作系统为 Redhat FC5,内核版本为 2.6.18-1.2239.fc5,两者通过安装了 FreeBSD 操作系统的多网卡主机(Celeron 1.0 G, 256 M SDRAM)连接到路由器(Celeron 1.0 G, 512 M SDRAM, Debian Linux 3.1,内核版本 2.6.18),而后连接到攻击对象主机(P Ⅲ 1.0 G CPU,内存 256 M SDRAM,10/100 M 自适应网卡,FC5). 在攻击方使用峰值速率 11 M/s,持续时间为 0.1 s,攻击周期在 0.5 s 至 5 s 之间变化的脉冲 UDP 流量进行攻击;在正常用户上运行网络测量工具 Iperf,测量攻击时的网络吞吐量,在 FreeBSD 主机上运行 dummynet 程序以模拟网络链路延迟. 我们只测试了单条 TCP 流的链路遭受慢速拒绝服务攻击时的网络吞吐量,因为在慢速拒绝服务攻击下多条 TCP 流的吞吐量与单条的吞吐量的变化规律类似[1].

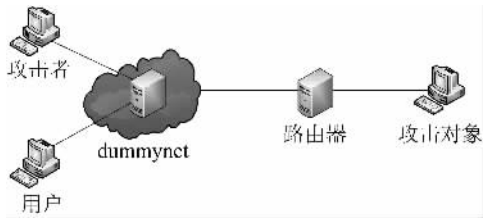

图 3　实验床拓扑结构

Fig. 3　The topology of experiment-bed

我们对比了采用动态调整策略与采用固定 RTO_{min} 策略(determined adjusting strategy, DAS)时网络性能的变化. 为了对比效果,还加入了文献[1]中所提出的随机 RTO_{min} 的策略(randomized adjusting strategy, RAS). 经过多次测试,最终选取的参数值为 $t_s=1$ s, $t_{e1}=t_{e2}=0.1$ s,图 4(a)是链路 RTT 为 50 ms 时的实验结果,图 4(b)是链路 RTT 为 200 ms 时的实验结果,两者的趋势基本一致. 可以看到,在使用了动态调整策略后,网络的吞吐量有很大提升,网络吞吐量可以达到额定吞吐量的 60% 以上;与 DAS 策略相比,TBAS 和 LAS 在攻击周期

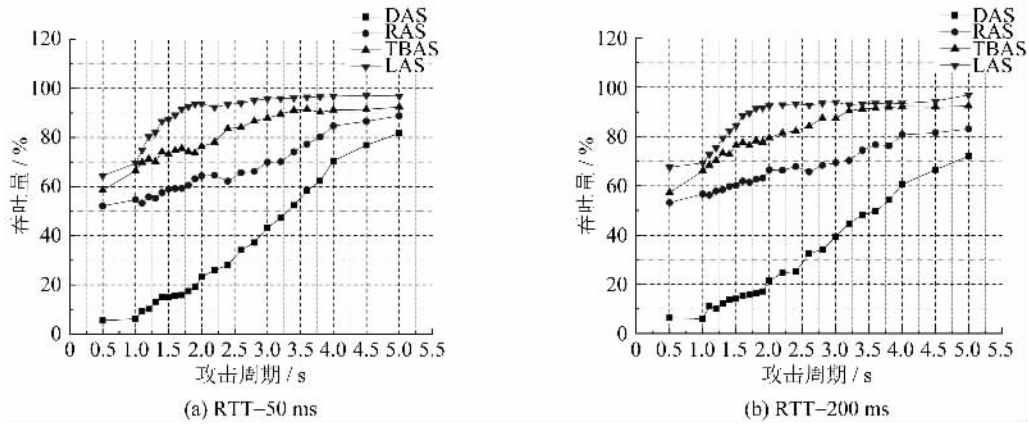

图 4　调整策略效果图

Fig. 4　The effect charts of the adjustment strategy

小于 2 秒时可以提升吞吐量 300%以上. 两种策略相比,LAS 的效果要好于 TBAS,这是因为 LAS 是在不断的调节过程当中,而 TBAS 只有在达到阈值后才能进行调节,在到达阈值之前的过程攻击对网络性能仍有很大影响.

使用动态调整 RTO_{min} 的方法来防御慢速拒绝服务攻击,可以避免影响没有出现攻击时网络的性能,这是因为动态调整策略在网络没有出现攻击时不会对 RTO_{min} 做出改变. 由于调整的前提是网络中的 UDP 数据流量比例过高,在正常的以 TCP 流量为主的网络中不会触发对 RTO_{min} 的调整,从而不会对其他 TCP 流的性能产生影响,保持了 TCP 用户间的公平性. RAS 策略在遭受慢速拒绝服务攻击时,如果随机到的 RTO_{min} 数值较大,对网络性能仍然会有很大影响. 过滤 UDP 脉冲的算法[5] 在出现正常 UDP 脉冲流量时会发生误报,从而影响了正常的基于 UDP 协议的应用,而我们提出的动态调整策略不会出现上述情况,其性能更优.

此外,本文中提出的数据包修改方案部署在靠近服务器端的路由器上,不同于其他防御分布式拒绝服务攻击的方法尤其是出口过滤(在数据包离开 ISP 之前进行过滤),ISP 部署此方案主要是为了提高本 ISP 网络服务的可用性,这与其他需要 ISP 协作防御的方法不同,是与 ISP 自身服务质量直接相关的.

4 结论

本文对慢速拒绝服务攻击进行了详细分析并提出了基于路由器反馈的动态调整超时重传定时器的方法. 实验结果证明,此类方法在防御慢速拒绝服务攻击时可以获得非常好的效果. 下一步的工作将在本文工作的基础上,在 Internet 上进行进一步的实验,调整策略的阈值,以获得更好的效果. 另外,对 TCP 的拥塞控制机制进行适当的修改,设计可控的拥塞控制方法,通过检测的结果控制数据包的发送来防御慢速拒绝服务攻击.

参考文献(References)

[1] Kuzmanovic A, Knightly E W. Low-rate TCP-targeted denial of service attacks[C]//Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications. NewYork: ACM, 2003:75-86.

[2] Luo X, Chang R K C. On a new class of pulsing denial-of-service attacks and the defense[C]//Network and Distributed System Security Symposium (NDSS'05). San Diego, CA, 2005:61-79.

[3] Guirguis M, Bestavros A, Matta I. Exploiting the transients of adaptation for RoQ attacks on Internet resources[C]//Proceedings of The 12th IEEE International Conference on Network Protocols. Los Alamitos, CA, USA: IEEE Computer Soc, 2004:184-195.

[4] Guirguis M, Bestavros A, Matta I. On the impact of low-rate attacks [C]//Proceedings of IEEE International Conference on Communications, 2006: 2 316-2 321.

[5] Kwok Y K, Tripathi R, Chen Y, et al. HAWK, halting anomalies with weighted choking to rescue well-behaved TCP sessions from shrew DDoS attacks[J]. Lecture Notes in Computer Science, 2005, 3 619: 423-432.

[6] Sun H B, Lui J C S, Yau D K Y. Distributed mechanism in detecting and defending against the low-rate TCP attack[J]. Computer Networks, 2006, 50 (13): 2 312-2 330.

[7] Paxson V, Allman M. Computing TCP's Retransmission Timer[S]. Internet RFC 2988, 2000.

[8] Corless R, Gonnet G, Hare D, et al. On the Lambert W function [J]. Advances in Computational Mathetatics, 1996, 5: 329-359.