

文章编号:1001-9081(2005)12-2745-03

基于 SYN Cookie 下防分布式拒绝服务攻击算法的分析与实现

沈清¹, 金心宇¹, 周绮敏²

(1. 浙江大学 信息与电子工程学系,浙江 杭州 310027;

2. 浙江大学 通信与信息工程研究所,浙江 杭州 310027)

(myqsq@sohu.com)

摘要:介绍了分布式拒绝服务(Distributed Denial of Service, DDoS)根据 TCP/IP 缺陷的攻击原理,在分析了数据包流量与系统资源使用率检测的基础上,提出了在 SYN Cookie 中引入 RSA 公钥加密过滤 TCP/IP 数据包的方法,用来检测与降低 DDoS 攻击的危害,该方法在实验中的测试阶段取得了较好的效果。

关键词:分布式拒绝服务;SYN Cookie;RSA;传输控制协议;洪流攻击

中图分类号: TP393.08 文献标识码:A

Analysis and realization of anti-DDoS attack algorithm based on the SYN Cookie mechanism

SHEN Qing¹, JIN Xin-yu¹, ZHOU Qi-min²

(1. Department of Information Science and Electronic Engineering, Zhejiang University, Hangzhou Zhejiang 310027, China;

2. Institute of Communication and Information Engineering, Zhejiang University, Hangzhou Zhejiang 310027, China)

Abstract: The principle of DDoS(Distributed Denial of Service) using the limitation of TCP/IP was introduced. Based on the monitor on the network packet traffic and the usage of the system resource, a method that leads the RSA algorithm into the SYN Cookie mechanism to encrypt and filtrate the IP packets was proposed, which can detect and reduce the damage of DDoS. The method have made a good performance in the experimentation.

Key words: Distributed Denial of Service(DDoS); SYN Cookie; RSA; TCP; flooding attacks

1 DDoS 利用 TCP/IP 缺陷攻击原理概述

1.1 DoS 与 DDoS

从网络攻击的各种方法和所产生的破坏情况来看,拒绝服务(Denial of Service, DoS)是一种很简单但又很有效的进攻方式。它的目的就是拒绝用户的服务访问,破坏组织的正常运行,最终使用户的部分 Internet 连接和网络系统失效。

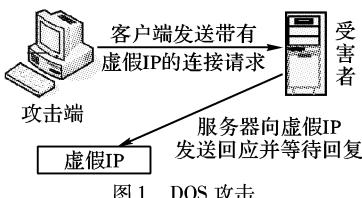


图 1 DOS 攻击

DoS 的攻击方式有很多,最基本的 DoS 攻击就是利用合理的服务请求来占用过多的服务资源,从而使合法用户无法得到服务。DoS 攻击的原理如图 1 所示。

分布式拒绝服务(Distributed Denial of Service, DDoS)是一种基于 DoS 的特殊形式的拒绝服务攻击,是一种分布、协作的大规模攻击方式,主要目标是比较大的站点,如企业网站、搜索引擎和政府部门的站点。基本的 DoS 攻击只要一台能连接 Internet 的单机就可实现,DDoS 攻击则是利用一批受控制的机器向一台机器发起攻击,具有较大的破坏性。DDoS 的攻击原理如图 2。

据研究统计,DDoS 攻击主要是通过 TCP/IP 协议来实现

的^[2](如表 1 所示),所以对 DDoS 中 TCP 攻击的监测预防成为了研究的热点。

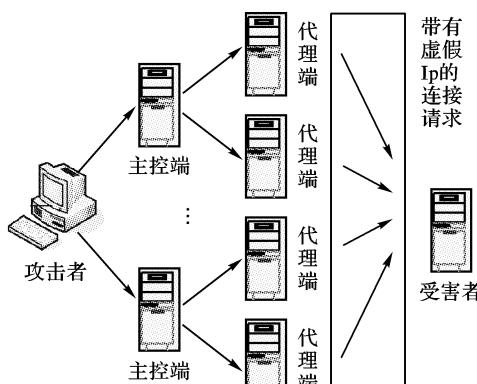


图 2 DDoS 攻击

表 1 常见 DDoS 攻击几种协议所占的比例

协议	DDoS 攻击所占比例
TCP	90% ~ 94%
UDP	2.4% ~ 5%
ICMP	2.1% ~ 2.6%
其他	2.06% ~ 2.9%

1.2 传统的 TCP/IP 三次握手规则

假设 Server 为客户端, Client 为被 Server 授权的合法客户,能获得 Server 授权的某些操作。Server 与 Client 之间的

收稿日期:2005-06-13;修订日期:2005-08-19

作者简介:沈清(1981-),男,安徽巢湖人,硕士研究生,主要研究方向:网络安全、网络视频; 金心宇(1956-),男,上海人,教授,主要研究方向:网络安全、人工智能、系统集成; 周绮敏(1963-),女,浙江杭州人,高工,主要研究方向:网络协议、信号处理。

Internet 通信遵守传统的 TCP/IP 的三次握手规则。其操作步骤为：

(1) 如果用户 Client 需要某种远程服务,他首先向中心主机 Server 发送一个请求,带有随机序列号的 SYN 包,即:Client 呼叫 Server: <SYN(序列号 = seq_{client})>

(2) 主机 Server 确认此用户 ID 是否是在数据库中,若在,则响应,并向用户 Client 发送一个带有应答序列号的 SYN + ACK 包,其中应答号等于原序列号加 1。同时主机产生自己的发送序列号,并将其与应答号一起发送给客户 Client,即:Server->Client <SYN(序列号 = seq_{server}),ACK(应答序列号 = seq_{client}) + 1>

(3) 用户 Client 再向主机 Server 回送一个应答包,其应答号等于主机 Server 向客户 client 发送的序列号加 1。即:Client->Server: <ACK(应答序列号:seq_{server} + 1)>,这样就完成了三次握手规则。

1.3 TCP SYN 洪流的原理

攻击者就是利用了“三次握手”的漏洞,假设一个用户向服务器发送了 SYN 报文后突然死机或掉线,那么服务器在发出 SYN + ACK 应答报文后无法收到客户端的 ACK 报文,这时服务器端一般会重试并等待一段时间后再丢弃这个未完成的连接,这段时间我们称为 SYN Timeout,一般为 30s 至 2min 不等。如果恶意的攻击者大量模拟这种情况,服务器端将为了维护一个非常大的半连接列表而消耗非常多的资源,即使是简单的保存并遍历也会消耗非常多的 CPU 时间和内存,何况还要不断对这个列表中的 IP 进行 SYN + ACK 的重试。如果服务器的 TCP/IP 栈不够强大,最后的结果就是堆栈溢出崩溃,即使服务器端的系统足够强大,服务器端也将忙于处理攻击者伪造的 TCP 连接请求而无暇处理客户的正常请求。

由此可以看出一个攻击者发起 TCP SYN 洪流攻击主要出于以下两个动机:

1) 消耗主机资源:TCP SYN 洪流攻击可以使服务器因为大量的半开连接而耗尽系统资源;

2) 淹没正常 TCP 报文:由于大量的非法 TCP SYN 攻击报文(伪造源 IP 地址)和来自于合法用户的 TCP SYN 报文是无法区分的,所以当 TCP SYN 报文过多必须丢弃一部分时,大量合法 TCP SYN 报文将同时丢失。严重时合法报文流被非法报文流完全淹没,用户无法与服务器建立 TCP 连接。

无论是哪种目的,对服务商以及合法用户都是一个不可估量的损失。

1.4 DDoS TCP SYN 的易攻击性

对于一个网络知识认识不太的人来说,想发动一起 DDoS 攻击也不是什么难事,而且还有如 Trinoo, Tribal Flood Network, TFN, TFN2K 等这些专业攻击软件,利用 Raw SOCKET 编写一个 DDoS 攻击程序也不是什么难事^[4]。这就使得 DDoS 攻击事件频频发生,增加了防范 DDoS 的难度。

2 数据包量监测过滤防范机制

2.1 数据包量异常监测机制

经过上述详细的分析,不难发现 DoS 攻击技术的核心是攻击端使用伪 IP 向目的主机发送垃圾数据包。

由于伪 IP 的随意性^[6],很难通过源 IP 来判定一个数据包是否合法,但 DDoS 攻击就是发送大量的数据包来消耗主机的资源。我们需要确定是通过网络流量(KB/S),还是网络数据包流量每单位时间通过网络的数据包数量(个/S)来确

定是否有大量数据包传输。通过笔者在实际网络的测量(见图 3,4),在相似流量(约 400kb/s)的情况下,正常网络的数据包大概是 400~500 个,而有 TCP 洪流的攻击下,其数据包量竟达 4000~5000 个,所以笔者认为,从网络数据包流量的监测,更能反映网络的状况。所以有理由这样假设,当主机在单位时间收到的数据包数大于一个阈值(由管理员根据统计得到),而且此时主机资源(CPU,内存,L/O)的使用率也超过一定的阈值并持续了一段时间,我们就认为主机收到了 DDoS 攻击。

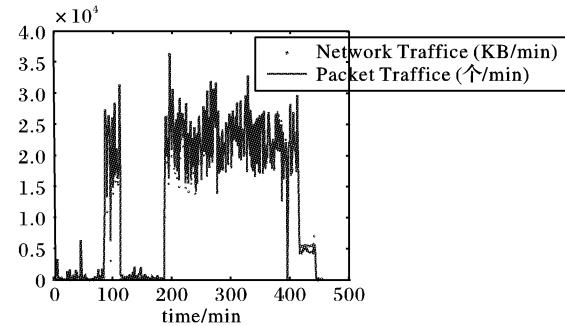


图 3 正常网络流量与数据包流量关系

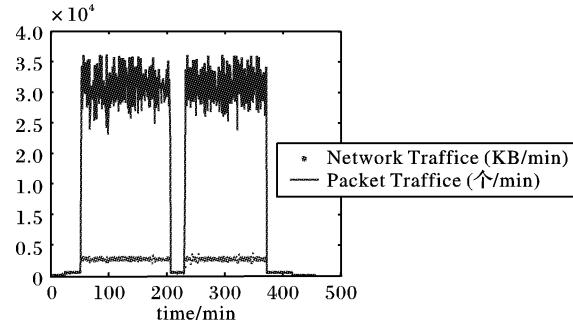


图 4 TCP 洪流攻击时网络流量与数据包流量关系

2.2 基于 RSA 密钥的过滤机制

当主机发现有 DDoS 攻击的迹象的时候,马上启用 RSA 密钥机制来过滤数据包,剔除数据包中可能存在的不合法数据包。该过滤机制主要是利用 SYN Cookie 对 TCP 服务器端的三次握手协议作一些修改。它的原理是:在 TCP 服务器收到 TCP SYN 包并返回 TCP SYN + ACK 包时,不分配一个专门的数据区,而是根据这个 SYN 包计算出一个 Cookie 值。在收到 TCP ACK 包时,TCP 服务器在根据 Cookie 值检查这个 TCP ACK 包的合法性。如果合法,再分配专门的数据区处理未来的 TCP 连接。

所以要求这个 Cookie 不但带有这次连接的属性,而且必须是无法伪造的。RSA 加密算法的理论基础是可逆模指数运算^[8],它的安全性是基于数论中大整数分解的困难性,即加密函数 $E_k(m) = m^e \bmod n$ 的单向性,对攻击者来说,解密在计算上是不可行的。目前它仍然安全并且被广泛应用。

服务器收到一个陌生源 IP 的 TCP 报文,利用 SYN Cookie 机制,生成该源 IP 的 Cookie,并用该 IP 发送过来的随机数进行 RSA 加密生成 ACK,此时服务器端并不为此次连接存储任何信息,属于无状态的握手。此时客户端收到服务器的加密序列,解密并用 Server 端的随机数进行加密,再回送给 Server,Server 判断此 IP 是否合法,如合法,则分配数据区进行 TCP 连接,反之,则拒绝。

其具体过程如下:

Client 端随机选取两个素数 m_c, n_c , 且 $t_c = m_c * n_c$,

$\varphi(t_c) = (m_c - 1)(n_c - 1)$ 。随机选取 $e_c, 1 < e_c < \varphi(t_c)$, 且 $\gcd(\varphi(t_c), e_c) = 1$, 即 e_c 与 $\varphi(t_c)$ 互为素数。求出解密密钥 $d_c, d_c = e_c^{-1} \bmod \varphi(t_c)$ 。然后, Client 将序列号加密后的 SYN 包和加密密钥 e_c, t_c 发送给主机 Server。同时主机 Server 也选取两个素数 m_s, n_s , 计算 $t_s = m_s * n_s, \varphi(n_s) = (m_s - 1) * (n_s - 1)$, 然后随机选取 $e_s, 1 < e_s < \varphi(t_s)$, 且 $\gcd(\varphi(t_s), e_s) = 1$, 即 e_s 与 $\varphi(t_s)$ 互为素数。求出解密密钥 $d_s, d_s = e_s^{-1} \bmod \varphi(t_s)$ 。产生一个序列号加密, 并用 Client 发来的加密密钥加密, 应答序列号等于原序列号加 1, 加上自己的加密密钥 m_s, n_s , 将其一起发送给客户 Client, 并生产 $\text{Cookie} = seq_s + 1$, 客户 Client 用自己的解密密钥解密, 解出 Server 发送过来的序列号, 加上 1, 然后用 Server 的加密密钥加密, 再回送给 Server。Server 解密求出应答序列号, 可以认证是否为合法用户。由此完成三次握手(如图 5 所示)。

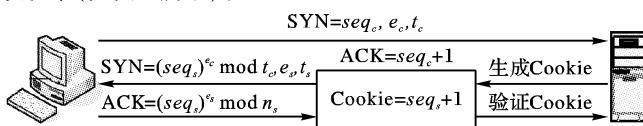


图 5 基于 SYN Cookie 的 RSA 过滤 TCP 报文

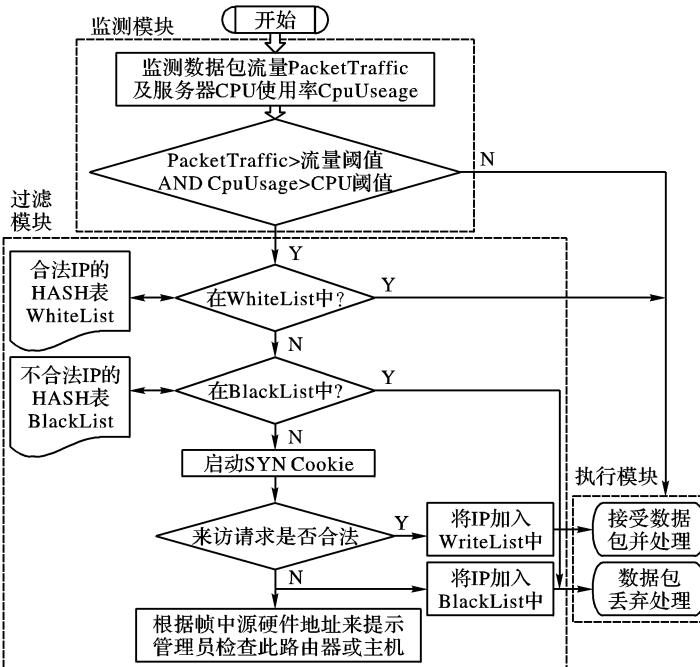


图 6 本机制流程

- A) Client→Server: <SYN(序列号 = seq_c), e_c, t_c >
- B) Server→Client: <ACK(应答序列号 $seq_c + 1$), SYN(加密序列号 = $((seq_s)^{e_c} \bmod n_s, e_s, t_s)$), 并产生 $\text{Cookie} = seq_s + 1$ 。
- C) Client→Server: 先解密出 Server 的序列号: $seq_s = ((seq_s)^{e_c} \bmod n_s, e_s, t_s) < ACK(\text{加密应答序列号} = ((seq_s)^{e_s} \bmod n_s))$ 。

Server 通过验证 $\text{Cookie} = ((seq_s)^{e_s}) \bmod n_s$ 的正确与否即可知道 Client 是否为合法用户。

定理 Client 或 Server 如上选取 e, d, n , 总是存在 $m = (m^e)^d \bmod n$ (其中 m 为某个明文)。

证明: 设 $r_1, r_2, \dots, r_{\varphi(n)}$ 是与 n 互素的模 n 剩余类集。 a 为与 n 互素的没个整数, 所以 $ar_1, ar_2, \dots, ar_{\varphi(n)}$ 也是和 n 互素, 且两两不同余。

否则若 $ar_i = ar_j \bmod n$

由 a 与 n 互素, 有 $r_i = r_j \bmod n$ 与假设矛盾。

$$\text{所以: } \prod_{i=1}^{\varphi(n)} (ar_i) = \sum_{i=1}^{\varphi(n)} r_i \bmod n$$

$$\text{即: } a^{\varphi(n)} \prod_{i=1}^{\varphi(n)} (r_i) = \sum_{i=1}^{\varphi(n)} r_i \bmod n$$

由于 $r_1, r_2, \dots, r_{\varphi(n)}$ 是与 n 互素, 得 $a^{\varphi(n)} = 1 \bmod n$, 则有 $a^{k\varphi(n)} = 1 \bmod P$, 对于任意得 $m \in Z_n$, 如果 $p + m$, 则 p 与 m 互素, 有 $m^{p-1} = 1 \bmod p$, 又 $\varphi(p) = (p-1) | \varphi(n)$, 所以:

$$m^{k\varphi(n)} + 1 = m^{k\varphi(n)} * m = m \bmod p$$

如果 $p | m$, 则 $m = 0 \bmod p$, 有 $m^{k\varphi(n)+1} = m \bmod p$ 成立。

同理可证恒有: $m^{k\varphi(n)+1} = m \bmod q$, 而 $\gcd(p, q) = 1$, 可得 $m^{k\varphi(n)+1} = m \bmod pq$ 。因为 $d = e^{-1} \bmod \varphi(n)$, 即 $ed = 1 \bmod \varphi(n)$, 所以可设 $ed = k\varphi(n) + 1$, 其中 k 是一个整数且 $k \geq 1$ 。所以同对任意的 m , 有 $(m^e)^d = m^{ed} = m^{k\varphi(n)+1} = m \bmod n$ 。

证毕。

同时先建立两张表, 如图 6 中的两个 Hash 表: whitelist, blacklist, 一张用来存放最近一段时间访问服务器的主机的合法 IP, 命名为 WriteList; 另一张用来存放被过滤掉的不合法的 IP 记录, 命名为 BlackList, 用来屏蔽其中的 IP。根据 Hash 算法在快速寻找目标中的优势, 采用 Hash 算法来存储两张表。

3 结语

TCP 报文过滤机制的难点在于准确区分攻击报文和合法报文。本文提出的基于数据包流量与被攻击主机资源使用异常检测技术, 通过结合在 SYN Cookie 技术中运用 RSA 密钥算法来鉴别 TCP 报文, 从中剔除不合法的报文, 利用 Hash 算法建立两个 Hash 表来更快的确定一个 TCP 包的合法性。并进一步的综合主机资源使用率的监测来更有效地监测异常行为, 减少误报率。根据不合法 IP 数据帧中的来源硬件地址来确定此数据报的上一级位置(路由器或主机), 并向管理员给予提示。这些机制使得系统在一定强度的 TCP 洪流下仍然能提供较好的服务。此机制在一个基于 Linux 平台的校园网网关里得以实现, 很好地融合了网关连接请求部分, 在测试阶段表现了良好的稳定性和健壮性。

参考文献:

- [1] 吴虎, 刘云超, 陈挺. 对 DDoS 攻击防范策略的研究及若干实现 [J]. 计算机应用研究, 2002, 19(8): 34–36.
- [2] 曹玥, 李晖, 吕东亚. 基于 DDoS 的 TCP SYN 攻击与防范 [J]. 电子科技, 2004, 173(2): 19–23.
- [3] SCHUBA C. Analysis of a Denial of Service Attack on TCP [A]. IEEE Security and Privacy Conference[C]. 1997. 208–230.
- [4] STEVENS W. TCP/IP Illustrated, Volume1 [S]. Addison-Wesley Professional, 1994.
- [5] TCP SYN Flooding and IP Spoofing Attacks CERT Advisory CA [EB/OL]. http://www.cert.org.tw, jan. 2000.
- [6] Denial of service Attacking with TCP SYN flooding [EB/OL]. http://www.cert.org.tw, 2000-01.
- [7] 吴文森. 公开密钥密码体制 RSA 算法的实现与应用 [J]. 计算机工程, 1998, 12(2): 28–32.
- [8] SCHNEIER B. 应用密码学 [M]. 北京: 机械工业出版社, 2000. 334–340.