

divert , ipfw question

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/hackers/2004-09/0371.html>

From: Zrelli Saber Ben Mohamed (zrelli_at_jaist.ac.jp)

Date: 09/28/04

Date: Tue, 28 Sep 2004 19:08:36 +0900

To: freebsd-net@freebsd.org, freebsd-hackers@freebsd.org, hackers@freebsd.org, net@freebsd.org

Hi ,

I'm interested in the "divert" mechanism and want to try it out ,
so I recompiled the kernel (FreeBSD 5.2.1-RELEASE #0) after adding the
IPDIVERT option and then added the needed lines in the rc.conf file,
after that , I set up ipfw to divert packets to some port
here is my ipfw rule set .

```
00100 allow ip from any to any via lo0
00200 deny ip from any to 127.0.0.0/8
00300 deny ip from 127.0.0.0/8 to any
65000 allow ip from any to any
65100 divert 5000 ip from any 22 to me <----- the divert rule
65535 deny ip from any to any
```

then, I wanted to monitor the diverted traffic using tcpdump :

```
$ tcpdump port 5000
```

when I do a telnet connection to the port 22 from a remote host , I was
expecting that tcpdump will display packets diverted to the port 5000 by
ipfw.

The remote host I use shows that it connects to port 22 and the ipfw
divert rule seems not to work.

I can set another rule to block the traffic in the port 22 , and it works.
only the divert rule seems to fail.

I wrote some piece of code using divert socket to read packets from the
divert port , but no result ...

I think I'm missing something ,

so please enlighten my mind ...

Many Thanks

--

divert , ipfw question

Saber

```

/*#include <stdio.h>
#include <string.h>
#include <sys/cdefs.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netinet/ip.h>
*/
#include <sys/param.h> /* NB: we rely on this for <sys/types.h> */
#include <sys/socket.h>
#include <sys/sysctl.h>
#include <sys/time.h>
#include <sys/uio.h>

#include <netinet/in.h>
#include <netinet/in_system.h>
#include <netinet/ip.h>
#include <netinet/ip_icmp.h>
#include <netinet/ip_var.h>
#include <arpa/inet.h>

#ifdef IPSEC
#include <netinet6/ipsec.h>
#endif /*IPSEC*/

#include <ctype.h>
#include <err.h>
#include <errno.h>
#include <math.h>
#include <netdb.h>
#include <signal.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/exits.h>
#include <termios.h>
#include <unistd.h>

#define BUFSIZE 65535

int
main(int argc, char **argv)
{
    int fd, rawfd, fdw, ret, n;
    int on = 1;
    struct sockaddr_in bindPort, sin;
    int sinlen;
    int port_nb;

```

freebsd-hackers: divert , ipfw question

```
struct ip *hdr;
unsigned char packet[BUFSIZE];
struct in_addr addr;
int i, direction;
struct ip_mreq mreq;

if (argc != 2) {
    fprintf(stderr, "Usage: %s <port number>\n", argv[0]);
    exit(1);
}
bindPort.sin_family = AF_INET;
bindPort.sin_port = htons(atol(argv[1]));
bindPort.sin_addr.s_addr = 0;

fprintf(stderr, "%s:Creating a socket\n", argv[0]);
/* open a divert socket */
fd = socket(AF_INET, SOCK_RAW, IPPROTO_DIVERT);

if (fd == -1) {
    fprintf(stderr, "%s:We could not open a divert socket\n", argv[0]);
    exit(1);
}
bindPort.sin_family = AF_INET;
bindPort.sin_port = htons(atol(argv[1]));
bindPort.sin_addr.s_addr = 0;

fprintf(stderr, "%s:Binding a socket\n", argv[0]);
ret = bind(fd, (struct sockaddr*)&bindPort, sizeof(struct sockaddr_in));

if (ret != 0) {
    close(fd);
    fprintf(stderr, "%s: Error bind(): %s", argv[0], strerror(ret));
    exit(2);
}
printf("%s: Waiting for data...\n", argv[0]);
/* read data in */
sinlen = sizeof(struct sockaddr_in);
while (1) {
    n = recvfrom(fd, packet, BUFSIZE, 0, (struct sockaddr*)&sin, &sinlen);
    hdr = (struct ip *) packet;

    printf("%s: The packet looks like this:\n", argv[0]);
    for (i = 0; i < 40; i++) {
        printf("%02x ", (int)*(packet + i));
        if (!(i + 1) % 16)
            printf("\n");
    };
    printf("\n");

    printf("%s: Source address: %s\n", argv[0], inet_ntoa(hdr->ip_src));
    printf("%s: Destination address: %s\n", argv[0], inet_ntoa(hdr->ip_dst));
}
```

freebsd-hackers: divert , ipfw question

```
printf("%s: Receiving IF address: %s\n", argv[0], inet_ntoa(sin.sin_addr));  
printf("%s: Protocol number: %i\n", argv[0], hdr->ip_p);  
  
    }  
}
```

freebsd-hackers@freebsd.org mailing list

<http://lists.freebsd.org/mailman/listinfo/freebsd-hackers>

To unsubscribe, send any mail to "freebsd-hackers-unsubscribe@freebsd.org"