

Multicast Example Programs

Sender Program

```
/*
 * sender.c -- multicasts "hello, world!" to a multicast group once a second
 *
 * Antony Courtney,      25/11/94
 */

#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <time.h>
#include <string.h>
#include <stdio.h>

#define HELLO_PORT 12345
#define HELLO_GROUP "225.0.0.37"

main(int argc, char *argv[])
{
    struct sockaddr_in addr;
    int fd, cnt;
    struct ip_mreq mreq;
    char *message="Hello, World!";

    /* create what looks like an ordinary UDP socket */
    if ((fd=socket(AF_INET,SOCK_DGRAM,0)) < 0) {
        perror("socket");
        exit(1);
    }

    /* set up destination address */
    memset(&addr,0,sizeof(addr));
    addr.sin_family=AF_INET;
    addr.sin_addr.s_addr=inet_addr(HELLO_GROUP);
    addr.sin_port=htons(HELLO_PORT);

    /* now just sendto() our destination! */
    while (1) {
        if (sendto(fd,message,sizeof(message),0,(struct sockaddr *) &addr,
                    sizeof(addr)) < 0) {
            perror("sendto");
            exit(1);
        }
        sleep(1);
    }
}
```

Listener Program

```
/*
 * listener.c -- joins a multicast group and echoes all data it receives from
 *               the group to its stdout...
 *
 * Antony Courtney,      25/11/94
 * Modified by: Frédéric Bastien (25/03/04)
 * to compile without warning and work correctly
 */

#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <time.h>
#include <string.h>
#include <stdio.h>

#define HELLO_PORT 12345
#define HELLO_GROUP "225.0.0.37"
#define MSGBUFSIZE 256

main(int argc, char *argv[])
{
    struct sockaddr_in addr;
    int fd, nbytes, addrlen;
    struct ip_mreq mreq;
    char msgbuf[MSGBUFSIZE];

    u_int yes=1;          /*** MODIFICATION TO ORIGINAL */

    /* create what looks like an ordinary UDP socket */
    if ((fd=socket(AF_INET,SOCK_DGRAM,0)) < 0) {
        perror("socket");
        exit(1);
    }

    /**** MODIFICATION TO ORIGINAL */
    /* allow multiple sockets to use the same PORT number */
    if (setsockopt(fd,SOL_SOCKET,SO_REUSEADDR,&yes,sizeof(yes)) < 0) {
        perror("Reusing ADDR failed");
        exit(1);
    }
    /*** END OF MODIFICATION TO ORIGINAL */

    /* set up destination address */
    memset(&addr,0,sizeof(addr));
    addr.sin_family=AF_INET;
    addr.sin_addr.s_addr=htonl(INADDR_ANY); /* N.B.: differs from sender */
    addr.sin_port=htons(HELLO_PORT);

    /* bind to receive address */
    if (bind(fd,(struct sockaddr *) &addr,sizeof(addr)) < 0) {
        perror("bind");
        exit(1);
    }
}
```

```
/* use setsockopt() to request that the kernel join a multicast group */
mreq.imr_multiaddr.s_addr=inet_addr(HELLO_GROUP);
mreq.imr_interface.s_addr=htonl(INADDR_ANY);
if (setsockopt(fd,IPPROTO_IP,IP_ADD_MEMBERSHIP,&mreq,sizeof(mreq)) < 0) {
    perror("setsockopt");
    exit(1);
}

/* now just enter a read-print loop */
while (1) {
    addrlen=sizeof(addr);
    if ((nbytes=recvfrom(fd,msgbuf,MSGBUFSIZE,0,
                        (struct sockaddr *) &addr,&addrlen)) < 0) {
        perror("recvfrom");
        exit(1);
    }
    puts(message);
}
}
```

Multicast Example / antony@scrg.cs.tcd.ie