

```

#define SV_SOCKET_PATH "/tmp/us_xfr"
#define BUF_SIZE 100
#define BACKLOG 5

int main(void)
{
    struct sockaddr_un addr;
    int sfd, cfd;
    ssize_t numRead;
    char buf[BUF_SIZE];

    if( (sfd = socket(AF_UNIX, SOCK_STREAM, 0)) == -1 ){
        perror("socket");
        exit(EXIT_FAILURE);
    }

    memset(&addr, 0, sizeof(struct sockaddr_un));
    addr.sun_family = AF_UNIX;
    strncpy(addr.sun_path, SV_SOCKET_PATH, sizeof(addr.sun_path) - 1);

    if( bind(sfd, (struct sockaddr *)&addr, sizeof(struct sockaddr_un)) == -1 ){
        perror("bind");
        exit(EXIT_FAILURE);
    }

    if(listen(sfd, BACKLOG) == -1){
        perror("listen");
        exit(EXIT_FAILURE);
    }

    for(;;) {
        if( (cfd = accept(sfd, NULL, NULL)) == -1 ){
            perror("accept");
            exit(EXIT_FAILURE);
        }

        while((numRead = read(cfd, buf, BUF_SIZE)) > 0)
            if(write(STDOUT_FILENO, buf, numRead) != numRead){
                perror("partial/failed write");
                exit(EXIT_FAILURE);
            }

        if(numRead == -1){
            perror("read");
            exit(EXIT_FAILURE);
        }

        if(close(cfd) == -1){
            perror("close");
            exit(EXIT_FAILURE);
        }
    }
}

```