

Reliable Server Pooling Sockets API Extensions

Michael Tüxen

tuxen@fh-muenster.de

Document status

- draft-ietf-rserpool-api-00.txt.
- First version as an WG document.
- Basics are covered.
- A lot of details (parameters, constants, structures, additional functions) are still missing.
- Comments are very welcome.

Design goals of the API

- Mimic the socket API, because
 - application writers can learn it easily.
 - Programs can be ported easily.
- Can be implemented as a user-land library using the socket API for the transport layer.
- Does only deal with applications using RSerPool.

Some functions (incomplete list)

- `rsp_socket`
- `rsp_close`
- `rsp_connect`
- `rsp_register`
- `rsp_deregister`
- `rsp_sendmsg`
- `rsp_rcvmsg`
- `rsp_select`

An Example: echo PU

```
fd = rsp_socket(AF_INET, SOCK_DGRAM, IPPROTO_SCTP);
rsp_connect(fd, "echo", 4);
done = 0;
FD_ZERO(&rset);
while(!done) {
    FD_SET(0, &rset);
    FD_SET(fd, &rset);
    rsp_select(fd + 1, &rset, NULL, NULL, NULL);
    if (FD_ISSET(0, &rset)) {
        n = Read(0, (void *) buf, sizeof(buf));
        if (n == 0)
            done = 1;
        else
            rsp_sendmsg(fd, (const void *) buf, len, NULL, 0);
    }
    if (FD_ISSET(fd, &rset)) {
        n = rsp_recvmsg(fd, (void *) buf, sizeof(buf), NULL, 0);
        if(n > 0)
            printf("%s", buf);
    }
}
rsp_close(fd);
```

Example: Echo PE

```
fd = rsp_socket(AF_INET, SOCK_DGRAM, IPPROTO_SCTP);
memset(&server_addr, 0, sizeof(server_addr));
server_addr.sin_family      = AF_INET;
server_addr.sin_len         = sizeof(struct sockaddr_in);
server_addr.sin_addr.s_addr = htonl(INADDR_ANY);
server_addr.sin_port        = htons(7);
rsp_bind(fd, (const struct sockaddr *) &server_addr,
         sizeof(server_addr));
memset(&linfo, 0, sizeof(struct rsp_loadinfo));
linfo.policy = RSP_ROUND_ROBIN;
rsp_register(fd, "echo", 4, &linfo);
while (1) {
    memset(&sinfo, 0, sizeof(struct rsp_sndrcvinfo));
    len = rsp_recvmsg(fd, (void *) buf, sizeof(buf), &sinfo, 0);
    rsp_sendmsg(fd, (const void *) buf, len, sinfo, 0);
}
rsp_deregister(fd);
rsp_close(fd);
```

Implementation

- A first version based on the ID is available at
 - <http://tdrwww.exp-math.uni-essen.de/dreibholz/rserpool/>

ToDo

- Define structures, function prototypes, constants, additional functions...
- Implement the API
- Implement some demo applications to see if the API is sufficient