

The home exams build upon the obligatory exam. Thus, if the obligatory exam has flaws, implementation work that is not directly related to the home exams may be necessary to fix these flaws. To make your life easier, we therefore provide a pre-compiled MIP daemon that you may use instead of the one from your own obligatory exam.

This MIP daemon uses three sockets: one to communicate with an application (ping), and two to communicate with the routing daemon (one for routing and one for forwarding). All the sockets used are of type SOCK_SEQPACKET (address family AF_UNIX) and the names of the sockets are provided as command line arguments.

When communicating with the ping client, the MIP daemon accepts data in the form:

Address Message

where *Address* is 1 byte containing the destination MIP address, and *Message* is the message that should be transferred to the ping server. (which is listening to a socket of the MIP daemon on the receiver side). "*Address Message*" is a single array of characters.

The Ping server receives the message from its local MIP daemon's socket at the receiver side in the form:

Address Message

where *Address* is 1 byte containing the source MIP address.

The Routing daemon uses two socket files: a socket file for the routing update, and another one for forwarding.

When a host receives a routing packet, it prefixes the src MIP address (that is the nexthop address) to the routing data and gives to the local Routing daemon. Technically, the first byte of the buffer (that the MIP daemon sends to the Routing daemon) contains the nexthop address, and the rest bytes is the routing table. The MIP daemon receives an array of characters representing the routing tables from the Routing daemon and sends it to the neighbor nodes. For forwarding, the MIP daemon sends and receives only one byte of data (it sends the destination MIP and receives the next-hop MIP).

Command line arguments for MIP daemon:

MIP_daemon64 [-d] <Socket_application> <Socket_Route> <Socket_forwarding> <Number of Interfaces> [MIP addresses ...]

-d: debugging, optional;
<Socket_application>: name of the socket for the application program;
<Socket_Route>: name of the socket for sending/receiving a routing table;
<Socket_forwarding>: name of the socket used to find the next-hop MIP for a given destination address (used during forwarding)

<Number of Interfaces>: integer value, number of possible interfaces for MIP addresses;
[MIP addresses ...]: list of local MIP addresses

Example 1:

```
MIP_daemon64 -d client_sock routing_sock forwarding_sock 4 10 20 30 40
```

Explanation: MIP daemon runs with four interfaces with MIP addresses 10, 20, 30 and 40. The socket file "client_sock" is used for communication between the ping client or ping server with the MIP daemon. The socket file "routing_sock" is used to communicate the routing table, and the MIP daemon uses "forwarding_sock" to find the next-hop address (forwarding).

Example 2:

```
MIP_daemon64 client_sock routing_sock forwarding_sock 2 10 20
```

Explanation:

Here, no debugging is enabled, and two interfaces are supported for MIP addresses 10 and 20).

The MIP daemon command lines do not specify which interface gets which address; the daemon's output shows the mapping of MIP addresses to interfaces.

IMPORTANT: note that home exam 1 needs some modifications to the MIP daemon. Therefore, if you use the provided MIP daemon, you do not obtain points for this part of home exam 1.