

# *UNIX Network Programming* *(an introduction)*

---

Andrea Passarella

Feb. 25, 2004



# Addressing

- `struct sockaddr_in{`  
    `sa_family_t`     `sin_family;` → **AF\_INET**  
    `u_int16_t`     `sin_port;` → **port, 16 bit**  
    `struct in_addr` `sin_addr;`  
};
- `struct in_addr{`  
    `u_int32_t`     `s_addr;` → **IPv4 address, 32 bit**  
};
- `struct sockaddr_in` `addr_a;`  
`memset(&addr_a, 0, sizeof(addr_a)); /* blank with 0 */`  
`addr_a.sin_family = AF_INET; /* IPv4 address */`  
`addr_a.sin_port = htons(1234); /* SRV_PORT, network ordered */`  
`inet_pton(AF_INET, "192.168.1.1", &addr_a.sin_addr);`



# Server side

```
#define SA struct sockaddr;  
  
struct sockaddr_in my_addr, cl_addr;  
int ret, len, sk, cl_sk;  
  
sk = socket(AF_INET, SOCK_STREAM, 0);  
memset(&my_addr, 0, sizeof(my_addr));  
my_addr.sin_family = AF_INET;  
my_addr.sin_addr.s_addr = htonl(INADDR_ANY);  
my_addr.sin_port = htons(1234);  
  
ret = bind(sk, (SA *) &my_addr, sizeof(my_addr));  
ret = listen(sk, 10);  
  
len = sizeof(cl_addr);  
cl_sk = accept(sk, (SA *) &cl_addr, &len);
```



# Client side

```
#define SA struct sockaddr;  
  
struct sockaddr_in srv_addr;  
  
int ret, sk;  
  
  
sk = socket(AF_INET, SOCK_STREAM, 0);  
memset(&srv_addr, 0, sizeof(srv_addr));  
srv_addr.sin_family = AF_INET;  
srv_addr.sin_port = htons(1234);  
ret = inet_pton(AF_INET, "192.168.1.1", &srv_addr.sin_addr);  
  
ret = connect(sk, (SA *) &srv_addr, sizeof(srv_addr));
```



# Communication

## □ To send data:

```
int ret, sk_a;
char msg[1024];
...
strcpy(msg, "something to send");
ret = send(sk_a, (void *) msg, strlen(msg), 0);
if(ret == -1 || ret < strlen(msg)){/* error */
    ...
}
```

## □ To receive data:

```
int ret, len, sk_a;
char msg[1024];
...
ret = recv(sk_a, (void *) msg, len, MSG_WAITALL);
/* len is the size of the incoming message (<= sizeof(msg))
   */
if( (ret == -1) || (ret < len) ){/* error */
    ...
}
```



# Includes

- Headers to be included

```
#include <unistd.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <arpa/inet.h>
```