Lập trình với socket

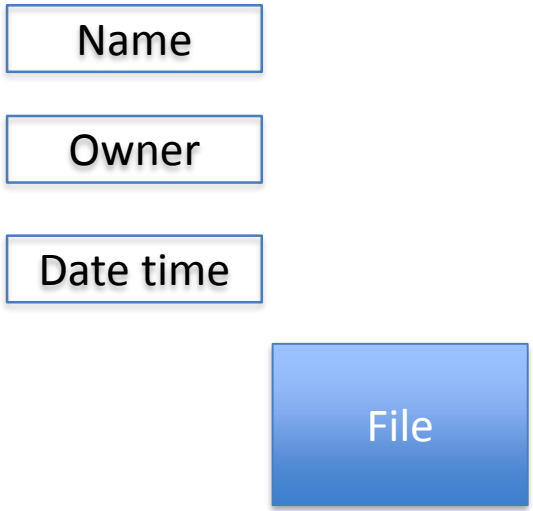
Giảng Viên: Lưu An Phú

Agenda

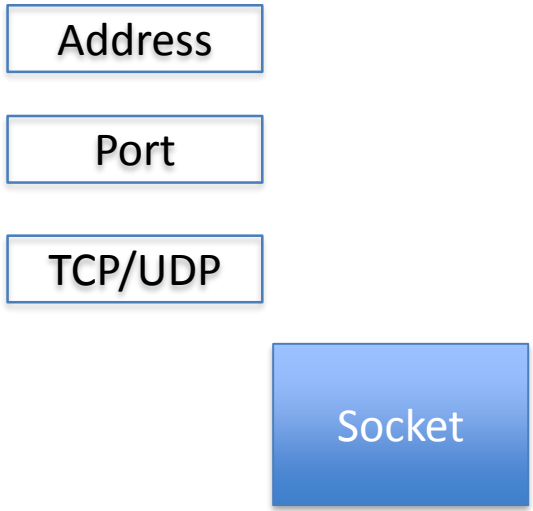
- Tổng quan về socket.
- Luồng gói tin đi trong mạng.
- Lập trình với socket
- Viết chương trình chat trong mạng Lan.

Socket overview

File



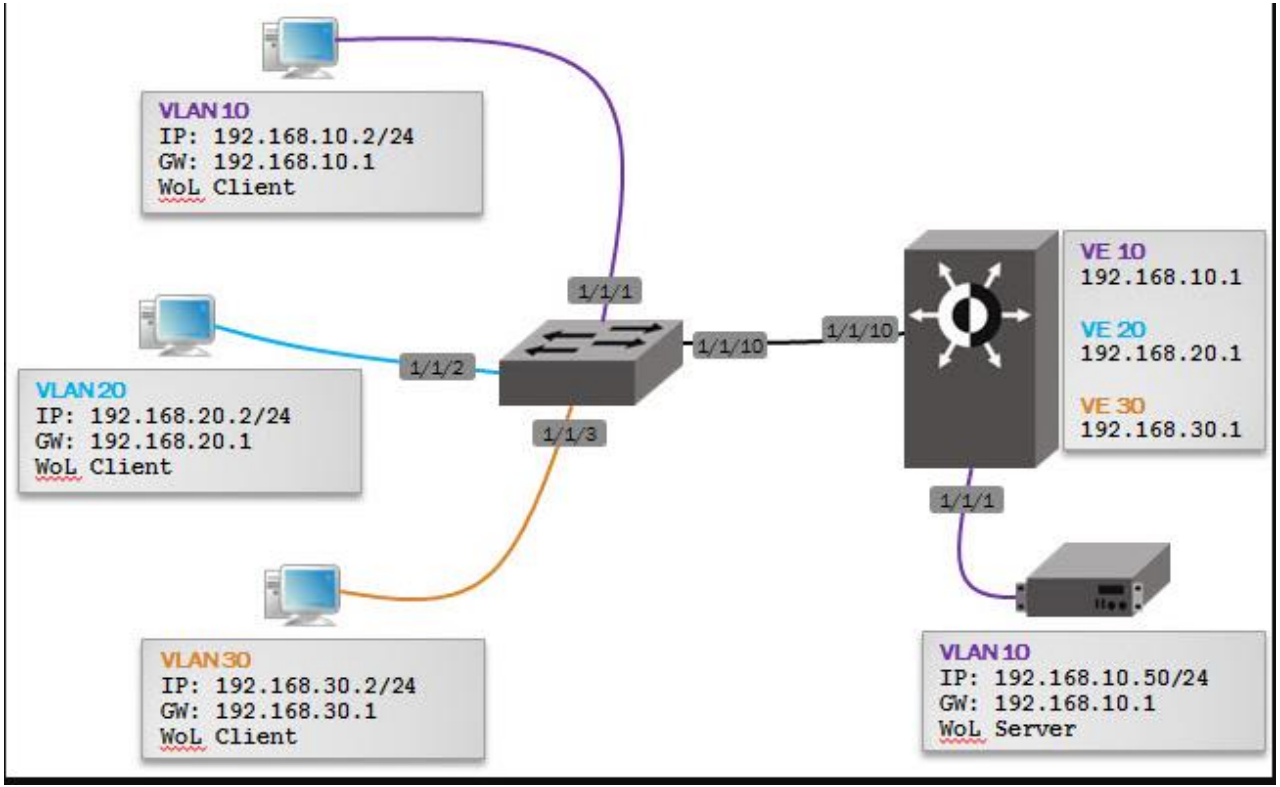
Socket



Định danh của một máy tính trong mạng internet

- Địa chỉ IP
  - 192.168.1.1
- Global IP address
- Local IP address
- Port
- Client and sever

Luồng gói tin đi trong mạng



Open file – create a socket

- #include <sys/socket.h>
- int socket(int domain, int type, int protocol);
- domain
  - AF_INET: IPv4 Internet domain
  - AF_UNIX: UNIX domain
- Type
  - SOCK_DGRAM: fixed-length, connectionless, unreliable messages
  - SOCK_STREAM: sequenced, reliable, bidirectional, connection-oriented byte streams

Protocol
IPPROTO_IP
IPPROTO_IPV6
IPPROTO_ICMP
IPPROTO_RAW
IPPROTO_TCP
IPPROTO_UDP

Thông tin của socket

- Struct socketadd_in

```
struct sockaddr_in {
    sa_family_t    sin_family; /* address family: AF_INET */
    in_port_t      sin_port;   /* port in network byte order */
    struct in_addr sin_addr;   /* internet address */
};

address.sin_family = AF_INET;
address.sin_addr.s_addr = INADDR_ANY;
address.sin_port = htons( PORT );
```

Setup socket cho sever

- Attach a socket to port
  - const char *inet_ntop(int domain, const void *restrict addr, char *restrict str, socklen_t size);
  - int bind(int sockfd, const struct sockaddr *addr, socklen_t len);

```
address.sin_family = AF_INET;
address.sin_addr.s_addr = INADDR_ANY;
address.sin_port = htons( PORT );

// Forcefully attaching socket to the port 8080
if (bind(server_fd, (struct sockaddr *)&address,
                                    sizeof(address))<0)
{
```

Setup socket cho sever

- Lắng nghe kết nối từ port
  - int listen(int sockfd, int backlog);

```
if (listen(server_fd, 3) < 0)
{
    perror("listen");
    exit(EXIT_FAILURE);
}
```

Setup socket cho sever

- Chấp nhận kết nối
  - int accept(int sockfd, struct sockaddr *restrict addr, socklen_t *restrict len);

```
if ((new_socket = accept(server_fd, (struct sockaddr *)&address,
                         (socklen_t*)&addrlen))<0)
{
    perror("accept");
    exit(EXIT_FAILURE);
}
```

Setup socket cho client

- Kết nối đến sever
  - int connect(int sockfd, const struct sockaddr *addr, socklen_t len);
  - const char *inet_ntop(int domain, const void *restrict addr, char *restrict str, socklen_t size);

```
// Convert IPv4 and IPv6 addresses from text to binary form
if(inet_pton(AF_INET, "127.0.0.1", &serv_addr.sin_addr)<=0)
{
    printf("\nInvalid address/ Address not supported \n");
    return -1;
}

if (connect(sock, (struct sockaddr *)&serv_addr, sizeof(serv_addr)) < 0)
{
    printf("\nConnection Failed \n");
    return -1;
}
```

Truyền nhận dữ liệu

- ssize_t send(int sockfd, const void *buf, size_t nbytes, int flags);
- ssize_t read(int fd, void *buf, size_t count);

```
send(sock , hello , strlen(hello) , 0 );
printf("Hello message sent\n");
valread = read( sock , buffer, 1024);
printf("%s\n",buffer );
```

Sơ đồ setup socket trên sever và client

Sever

- Socket()
- Init sockaddr_in
- Bind()
- Listen()
- accept()

client

- Socket()
- Init sockaddr_in
- connect()

- send()
- read()
- close()

Thực hành

- Viết 1 chương trình chat theo mô hình client server. Cả 2 chạy cùng trên 1 máy tính

Thank you