



**EAST ASIA UNIVERSITY
OF TECHNOLOGY**

LẬP TRÌNH MẠNG
(Network Programming)
SOCKET – UDP/TPC

Nguyễn Anh Thơ
natho5578@gmail.com

Nội dung

Tuần	Nội dung	
01	Tổng quan lập trình Socket	
02	Lập trình Socket UDP	
03	Lập trình Socket TCP	
04	Lập trình Socket IP	

Lập trình Socket TCP

■ Cấu trúc gói tin



Cấu trúc Header

Offsets		Octet#	0								1								2								3																												
Octet	Bit	Bit#	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																					
0	0		Source port																Destination port																																				
4	32		Sequence number																																																				
8	64		Acknowledgment number (if ACK set)																																																				
12	96		Data offset				Reserved 0 0 0			N S	C W R	E C R	U R G	A C K	P C H	S S T	R S Y	F I N	Window Size																																				
16	128										Checksum																						Urgent pointer (if URG set)																						
20	160										Options																																												
...																																																				

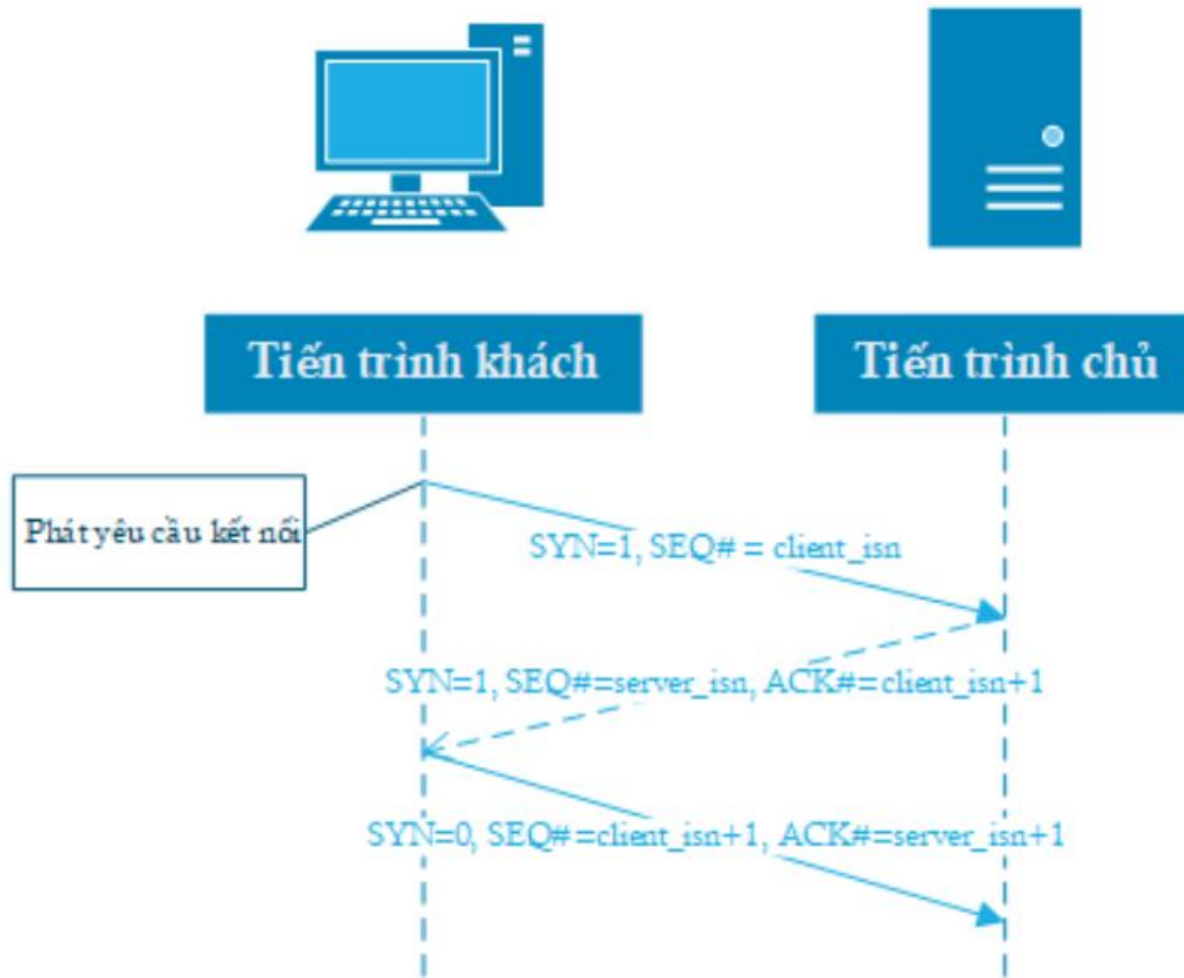
Quy trình hoạt động - TCP

Bước 1 – Thiết lập liên kết – TCP

Bước 2 – Truyền dữ liệu

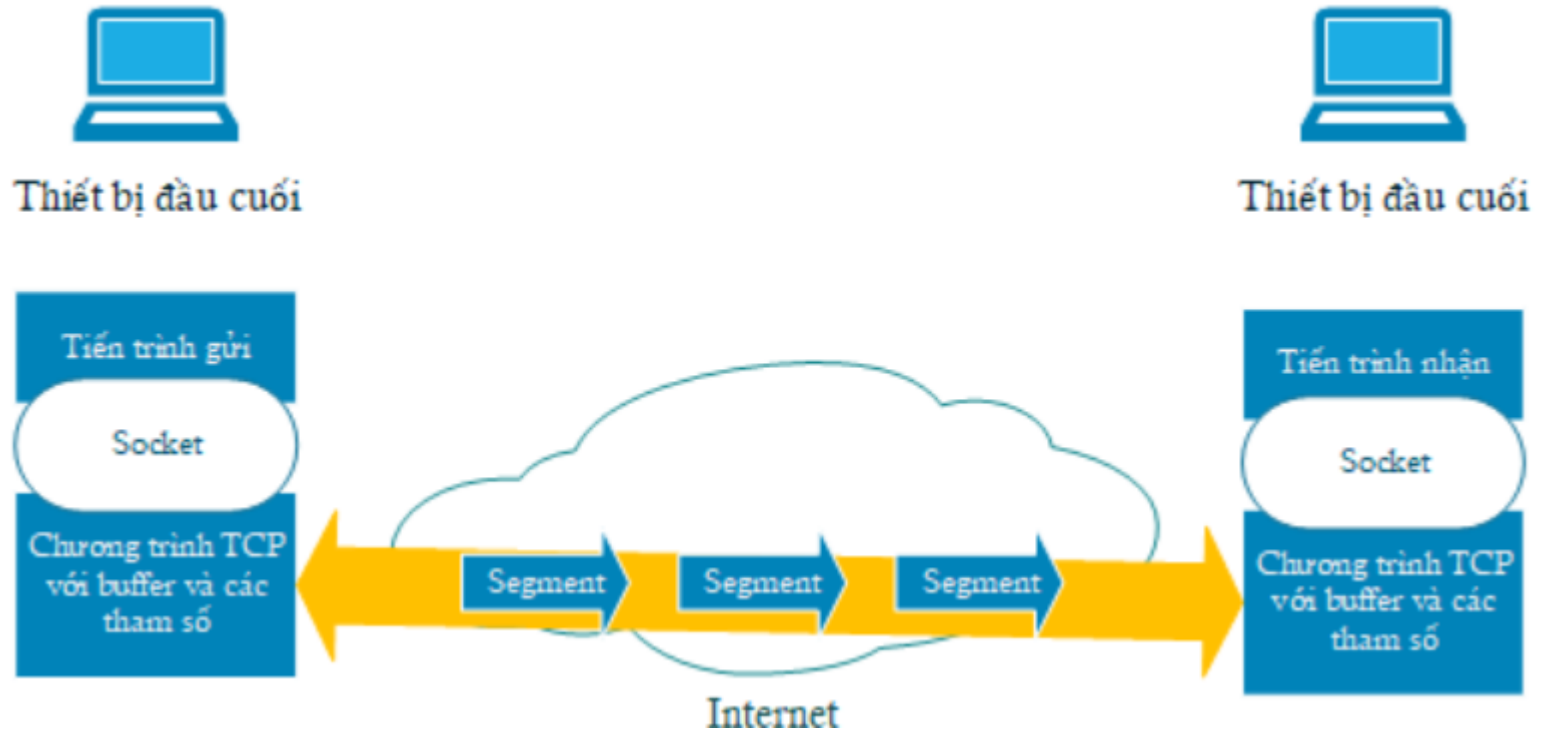
Bước 3 – Đóng liên kết

TCP – Bước 1. Thiết lập liên kết



Quy trình bắt tay ba bước

TCP – Bước 2. Truyền dữ liệu



Truyền dữ liệu qua liên kết TCP:

- + Song công: A->B -> A
- + Segment:
- + các buffer:

Lập trình Socket TCP với C#

Bước 1. Thiết lập liên kết TCP

■ 1.1. Khởi tạo Socket TCP

```
// Tại máy Client
// khởi tạo object của lớp socket để sử dụng dịch vụ Tcp
// lưu ý SocketType của Tcp là Stream
var socket = new Socket(SocketType.Stream, ProtocolType.Tcp);

//Tại Server
// tcp sử dụng đồng thời hai socket:
// một socket để chờ nghe kết nối, một socket để gửi/nhận dữ liệu
// socket listener này chỉ làm nhiệm vụ chờ kết nối từ Client
var listener = new Socket(AddressFamily.InterNetwork,
                           SocketType.Stream, ProtocolType.Tcp);

// tcp đòi hỏi một socket thứ hai làm nhiệm vụ gửi/nhận dữ liệu
// socket này được tạo ra bởi lệnh Accept
var socket = listener.Accept();
```

Lập trình Socket TCP với C#

Bước 1. Thiết lập liên kết TCP

...

1.2. Liệt kê Server với cổng và lắng nghe

```
// yêu cầu hệ điều hành cho phép chiếm dụng cổng tcp 1308
// server sẽ nghe trên tất cả các mạng mà máy tính này kết nối tới
// chỉ cần gói tin tcp đến cổng 1308, tiến trình server sẽ nhận được
listener.Bind(localEndPoint);

// bắt đầu lắng nghe chờ các gói tin tcp đến cổng 1308
listener.Listen(10);
```

Lập trình Socket TCP với C#

Bước 1. Thiết lập liên kết TCP



..

1.3. Tạo kết nối TCP

```
// Tại Server
// tcp đòi hỏi một socket thứ hai
// làm nhiệm vụ gửi/nhận dữ liệu
// socket này được tạo ra bởi lệnh Accept
var socket = listener.Accept();

// Tại Client
// tạo kết nối tới Server
socket.Connect(serverEndpoint);
```

Lập trình Socket TCP với C#

Bước 2. Truyền dữ liệu

2.1. Gửi dữ liệu qua TCP

```
// Tại Server
var sendBuffer = Encoding.ASCII.GetBytes(result);
// gửi kết quả lại cho client
socket.Send(sendBuffer);
// không tiếp tục gửi dữ liệu nữa
socket.Shutdown(SocketShutdown.Send);

// Tại Client
// biến đổi chuỗi thành mảng byte
var sendBuffer = Encoding.ASCII.GetBytes(text);
// gửi mảng byte trên đến tiến trình server
socket.Send(sendBuffer);
// không tiếp tục gửi dữ liệu nữa
socket.Shutdown(SocketShutdown.Send);
```

Lập trình Socket TCP với C#

Bước 2. Truyền dữ liệu

2.2. Nhập dữ liệu

```
// nhận mảng byte từ dịch vụ Tcp và lưu vào bộ đệm  
var length = socket.Receive(receiveBuffer);  
// chuyển đổi mảng byte về chuỗi  
var result = Encoding.ASCII.GetString(receiveBuffer, 0, length);  
// không tiếp tục nhận dữ liệu nữa  
socket.Shutdown(SocketShutdown.Receive);
```

Lập trình Socket TCP với C#

Bước 3. Đóng liên kết

```
// đóng socket và giải phóng tài nguyên  
socket.Close();
```

Chú ý: Hoạt động của TCP theo chế độ blocking:

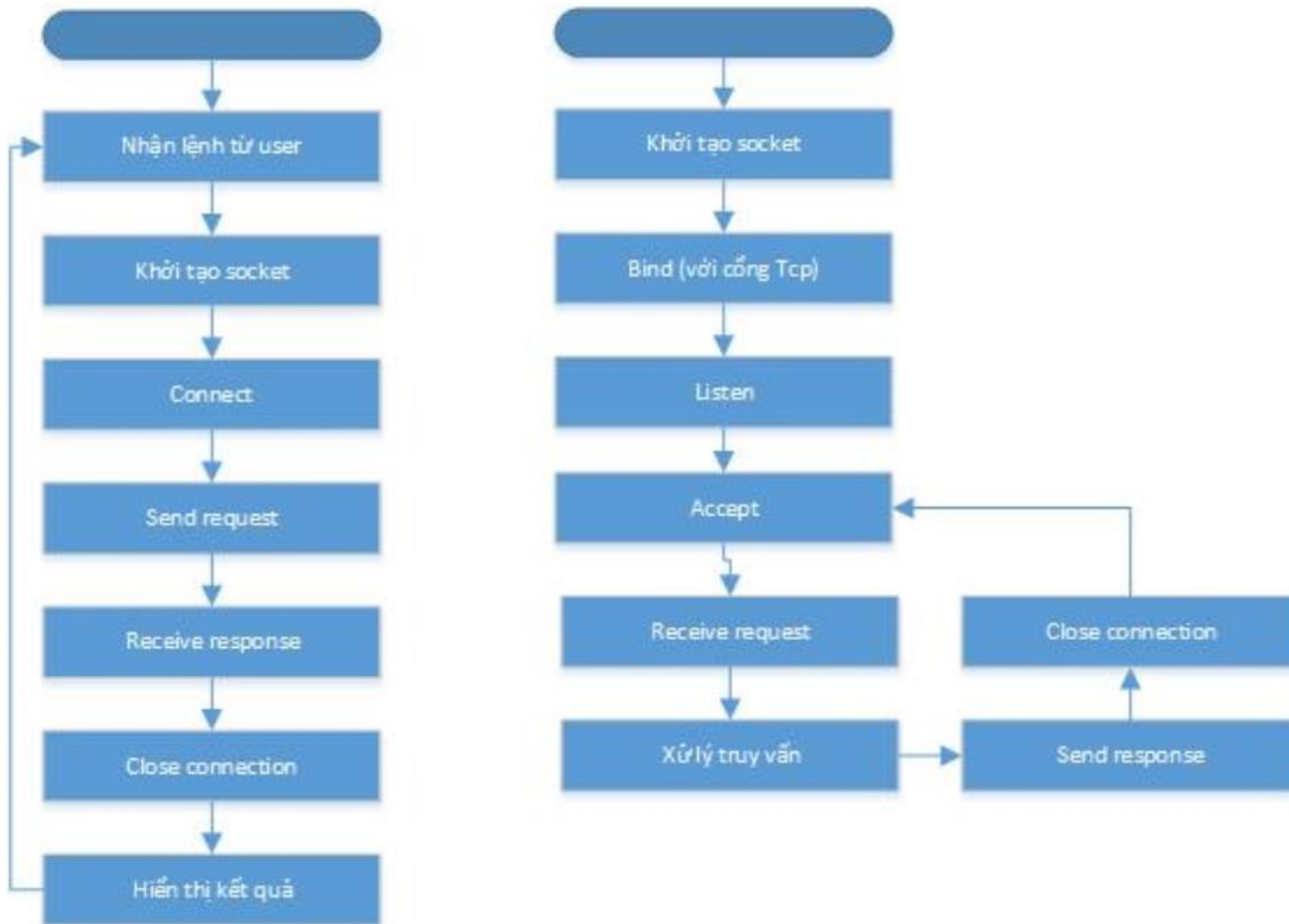
Bài tập thực hành

Bài toán

Viết một ứng dụng dòng lệnh đơn giản (console application) theo mô hình client/server đáp ứng các yêu cầu:

1. Client cho phép người dùng nhập một chuỗi ký tự từ bàn phím và gửi chuỗi ký tự cho server;
2. Server nhận chuỗi ký tự, chuyển đổi tất cả ký tự thành dạng in hoa và gửi chuỗi kết quả lại cho client;
3. Client nhận kết quả và hiển thị lại cho người dùng.

Sơ đồ thực hiện



Bài tập

- Viết chương trình theo mô hình client- server sử dụng UDP Socket thực hiện yêu cầu sau:
 - Tại Client nhập vào danh sách Điểm học phần của sinh viên gồm thông tin: DiemSV(ID, Hovaten, DiemHP1, DiemHP2). Việc nhập danh sách chỉ dừng khi nhập ID=0;
 - Tại Server Nhận được thông tin DiemSV thực hiện tạo và cập nhật vào danh sách theo thông tin sau:
ID, HovaTen, DiemGPA
Biết rằng: DiemGPA được xác định như sau:
Nếu $0 < DTBHP < 5 \rightarrow DiemGPA = "D"$
Nếu $5 < DTBHP < 7 \rightarrow DiemGPA = "C"$
Nếu $7 < DTBHP < 9 \rightarrow DiemGPA = "B"$
Nếu $9 < DTBHP < 10 \rightarrow DiemGPA = "A"$
 $DTBHP = (HP1 + 2*HP2)/3;$
 - Server gửi lại Client dữ liệu đã xử lý + Thời gian xử lý (Thời gian hệ thống)
 - Client in lại kết quả nhận lại từ Server

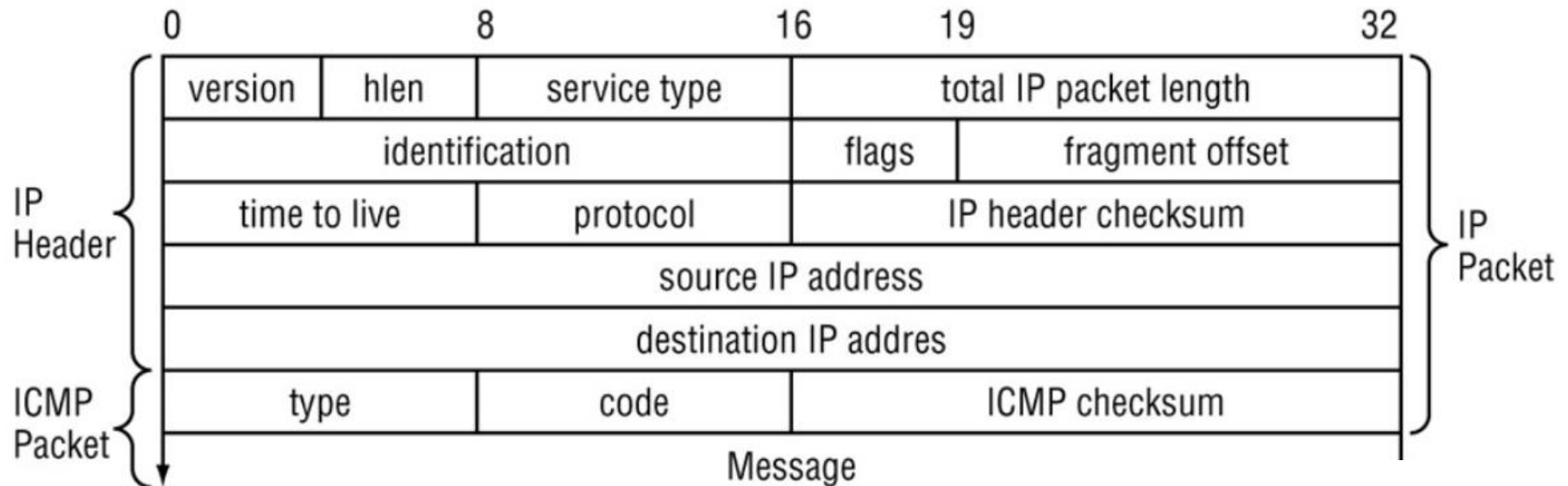
Bài tập - Sử dụng giao thức UDP/TCP

1. Xây dựng ứng dụng chat theo mô hình host-to - host
2. Xây dựng ứng dụng chatbox (trả lời tự động)
3. Xây dựng thi trắc nghiệm gồm 2 chức năng
 - Ra đề thi
 - Chấm thi tự động

Socket IP

- Socket IP (raw socket) – Truyền thông host – to – host
- Sử dụng giao thức ICMP – Internet Control Message Protocol
- Công cụ mạng dữ liệu ICMP là giao thức IP

Giao thức ICMP



Type = 1 byte : Giá trị được gọi type code

Ví dụ:

+ type code 0 – echo reply: gói tin phản hồi cho echo

Code=1 byte

Checksum=2 byte

Lập trình Socket IP

```
// Socket IP
// Bước 1 - Tạo Socket IP

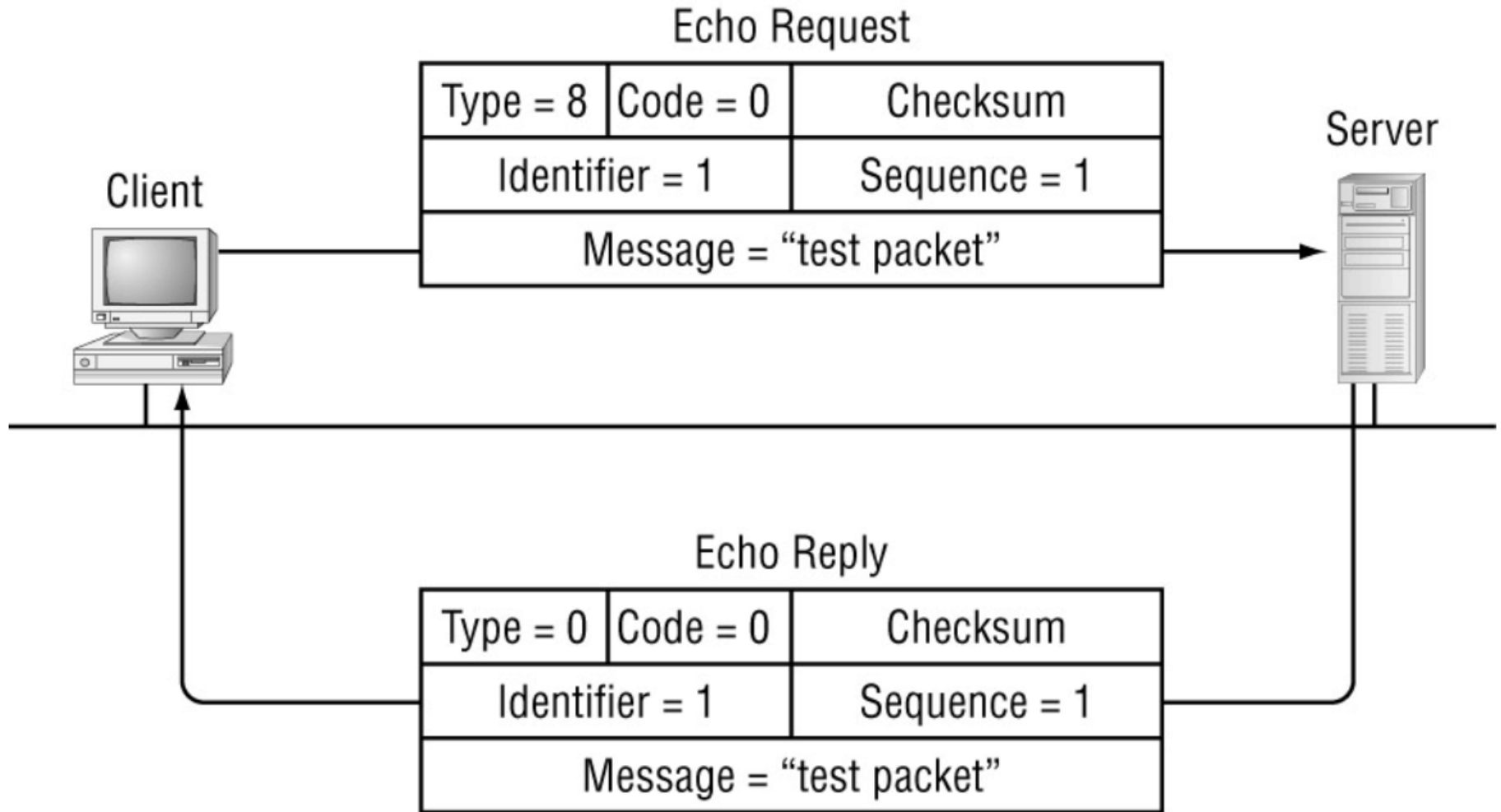
Socket socket = new Socket(AddressFamily.InterNetwork,
                             SocketType.Raw, ProtocolType.Icmp);

// Bước 2 - Truyền dữ liệu
EndPoint endpoint = new EndPoint(IPAddress.Parse("192.168.1.2"), 0);
// Gửi gói tin
socket.SendTo(packet, endpoint);
// Nhận gói tin
public int ReceiveFrom (byte[] buffer, ref System.Net.EndPoint remoteEP);
```

Chú ý:

- Phần header socket IP cố định 20byte (buffer[20])
- Dgram bắt đầu từ byte thứ 21

Xây dựng chương trình Ping



QUESTION ?