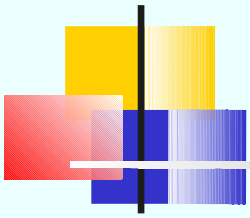
Lập trình mạng

Lập trình Socket với SSL

Giảng viên: ***TS. Nguyễn Mạnh Hùng***

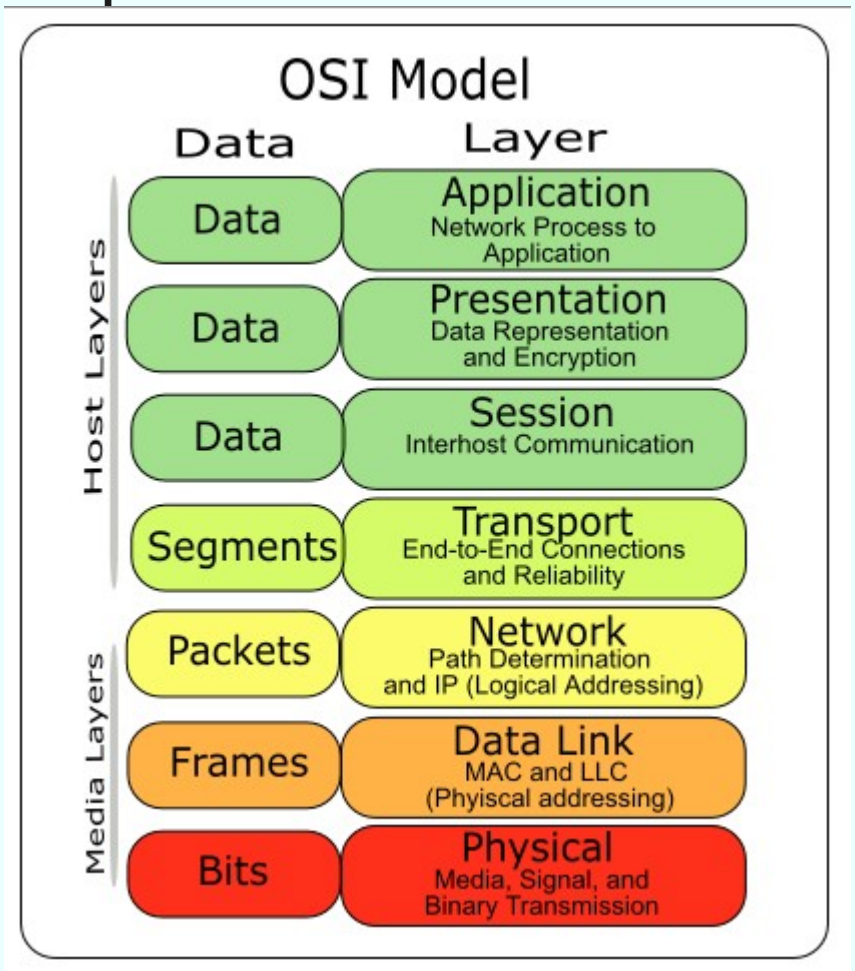
Học viện Công nghệ Bưu chính Viễn thông (PTIT)



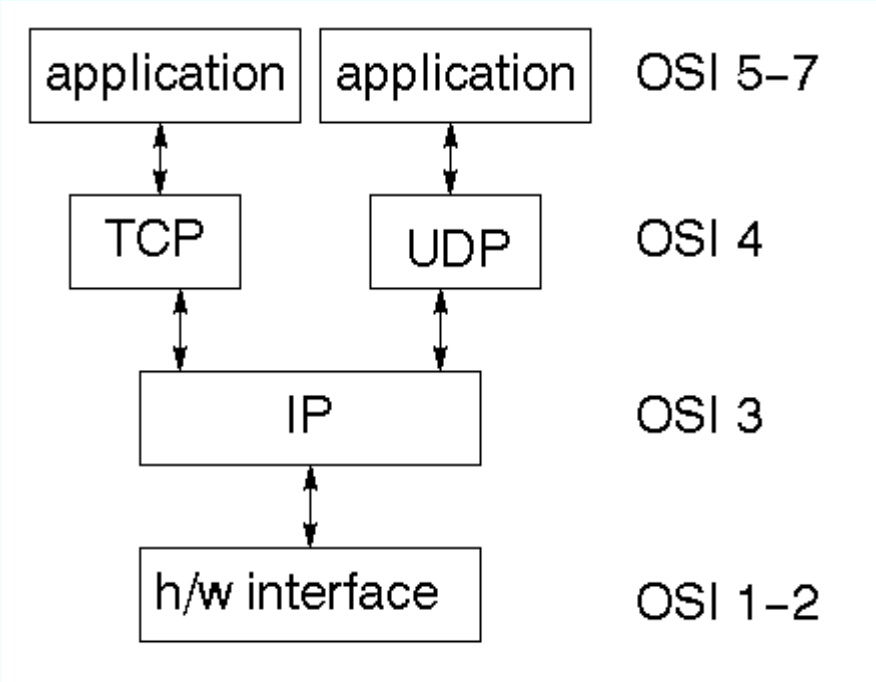
Nội dung

- Giao thức TCP/IP với SSL
- Cài đặt phía server
- Cài đặt phía client
- Ví dụ: đảo ngược chuỗi
- Bài tập

TCP/IP trong mô hình ISO

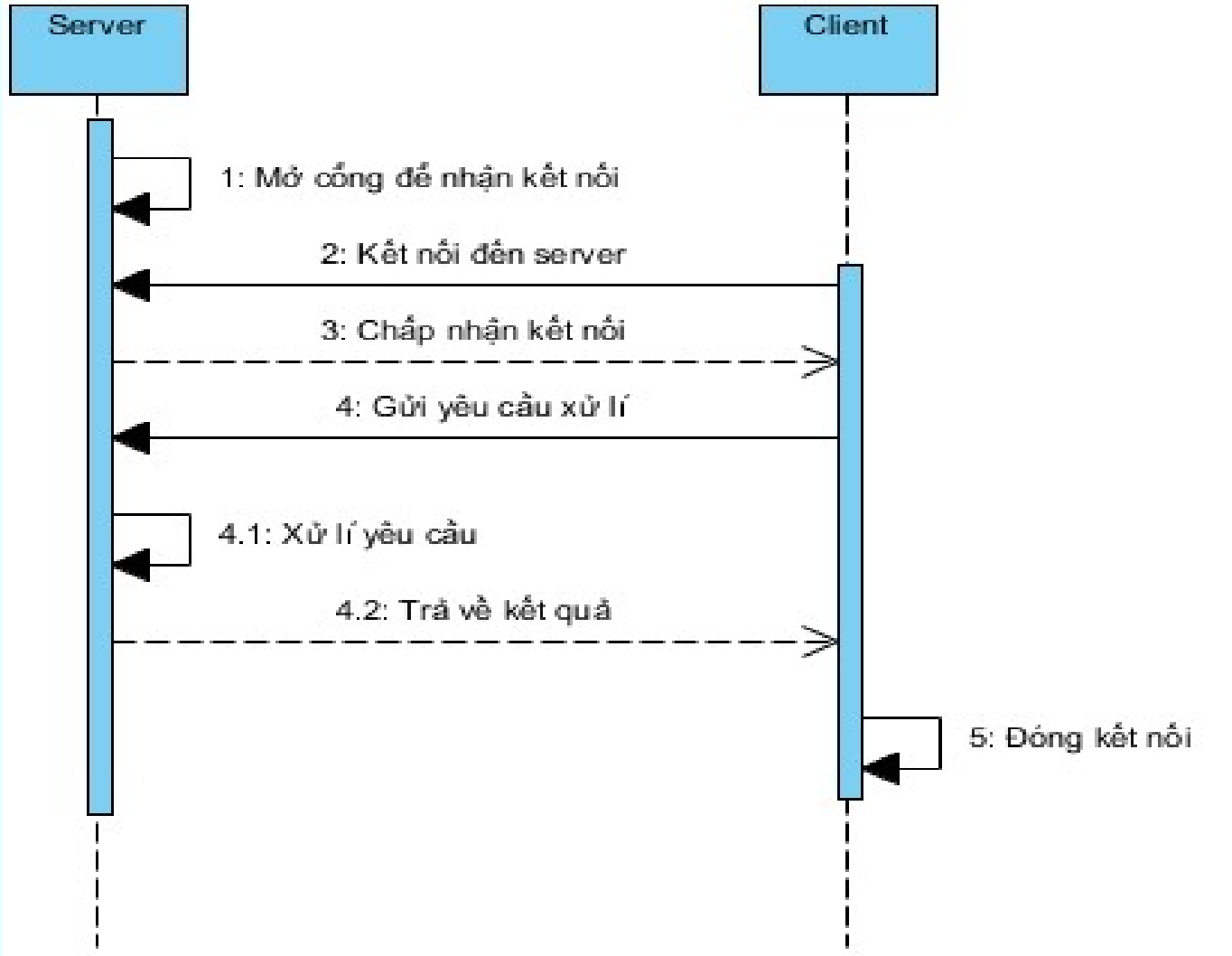


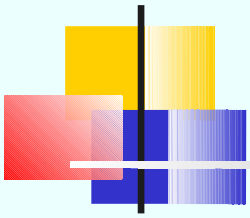
[image source: http://1.bp.blogspot.com]



[image source: http://jan.newmarch.name]

Giao thức TCP/IP

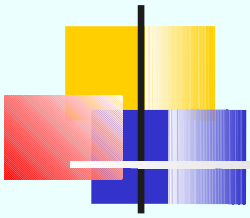




Server (1)

Bước 1: Mở một server socket tại một cổng có số hiệu xác định

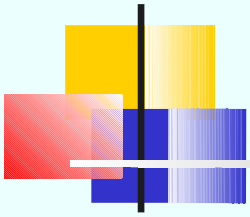
```
try {
    SSLServerSocketFactory sslserversocketfactory =
       SSLServerSocketFactory.getDefault();
    SSLServerSocket sslserversocket =
      sslserversocketfactory.createServerSocket(9999);

}
catch(IOException e) {
    System.out.println(e);
}
```



Server (2)

Bước 2: Tạo một đối tượng socket từ ServerSocket để lắng nghe và chấp nhận các kết nối từ phía client

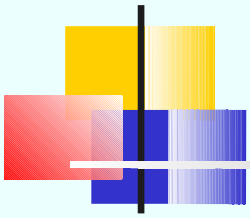
```
try {
        SSLSocket sslsocket = sslserversocket.accept();
        Scanner is = new
            Scanner(sslsocket.getInputStream());
        PrintStream os = new
            PrintStream(sslsocket.getOutputStream());
    }catch (IOException e) {
        System.out.println(e);
    }
```



Server (3)

Bước 3: Mỗi khi nhận được dữ liệu từ client, tiến hành xử lí và gửi trả về client đó

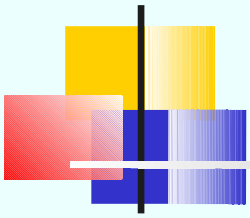
```
// Xu li du lieu nhan duoc va tra ve
while (true) {
    // doc du lieu vao
      String input = is.nextLine();

      // xu li du lieu
      ...

      // tra ve du lieu
      os.println(dữ liệu trả về);
}
```



Client (1)

Bước 1: Mở một kết nối client socket đến server có tên xác định, tại một cổng có số hiệu xác định

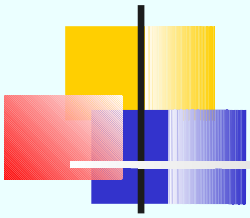
```
try {
    SSLSocketFactory sslsocketfactory =
       SSLSocketFactory.getDefault();
    SSLSocket sslsocket =
       sslsocketfactory.createSocket("localhost", 9999);

    } catch (UnknownHostException e) {
        System.err.println(e);
    } catch (IOException e) {
        System.err.println(e);
    }
```



Client (2)

Bước 2: Mở luồng kết nối vào (nhận dữ liệu) và kết nối ra (gửi dữ liệu) đến socket vừa mở

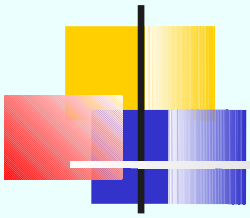
```
try {
      PrintStream os = new
          PrintStream(sslsocket.getOutputStream());
      Scanner is = new
          Scanner(sslsocket.getInputStream());
} catch (UnknownHostException e) {
      System.err.println(e);
} catch (IOException e) {
      System.err.println(e);
}
```



Client (3)

Bước 3: Gửi dữ liệu đến server

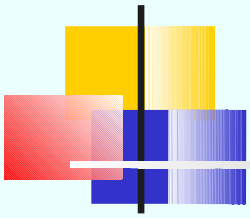
```
try {
    os.println(dữ liệu gửi đi);

}catch (UnknownHostException e) {
    System.err.println("e);
}catch (IOException e) {
    System.err.println("e);
}
```



Client (4)

Bước 4: Nhận dữ liệu đã qua xử lí từ server về

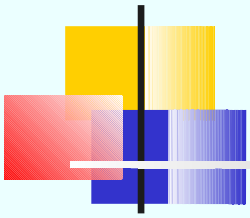
```
try {
      String responseStr = is.nextLine(); // du lieu nhan ve

}catch (UnknownHostException e) {
      System.err.println(e);
}catch (IOException e) {
      System.err.println(e);
}

return responseStr;
```



Client (5)

Bước 5: Đóng các kết nối tới server

```
try {
      os.close();
     is.close();
     mySocket.close();
} catch (UnknownHostException e) {
     System.err.println(e);
} catch (IOException e) {
     System.err.println(e);
}
```



Chạy chương trình

Chạy server:

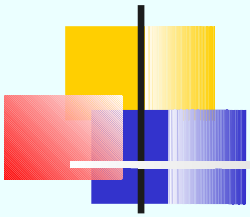
```
    >java -Djavax.net.ssl.keyStore=mySrvKeystore
-Djavax.net.ssl.keyStorePassword=123456 <tên file server>
```

Chạy client:

```
    >java -Djavax.net.ssl.trustStore=mySrvKeystore
-Djavax.net.ssl.trustStorePassword=123456 <tên file client>
```

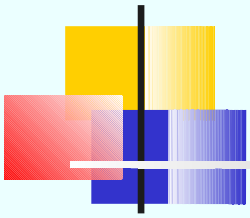
Muốn debug, thêm tùy chọn này vào cả 2 lệnh trên:

```
 -Djava.protocol.handler.pkgs=com.sun.net.ssl.internal.www.protocol
-Djavax.net.debug=ssl
```



Ví dụ: đảo chuỗi (1)

```
import java.lang.String;

public class ReverseString {
private String _string;

// khoi tao khong tham so
public ReverseString(){}

// khoi tao co tham so
public ReverseString(String _string){
    this._string = _string;
}

public String get_string() {
    return _string;
}

public void set_string(String _string) {
    this._string = _string;
}
```



Ví dụ: đảo chuỗi (2)

```
//phuong thuc dao nguoc chuoi ki tu cua lop nay
public void reverse(){
    String tmp ="";
    for(int i=_string.length() - 1; i >=0 ;i--)
            tmp += _string.substring(i, i+1);
    this._string = tmp;
}
}
```

Ví dụ: đảo chuỗi – server (1)

```
import javax.net.ssl.SSLServerSocket;
import javax.net.ssl.SSLSocket;

public class TCPServer {
// Khai bao server socket, luong vao-ra, va doi tuong socket
    SSLServerSocket myServer = null;
    String input;
    Scanner is;
    PrintStream os;
    SSLSocket clientSocket = null;

// Mo mot server socket
public void openServer(){
    try {
        SSLServerSocketFactory sslserversocketfactory =
            SSLServerSocketFactory.getDefault();
        SSLServerSocket myServer =
            sslserversocketfactory.createServerSocket(9999);
    }catch(IOException e) {
         System.out.println(e);
    }
}
```

Ví dụ: đảo chuỗi – server (2)

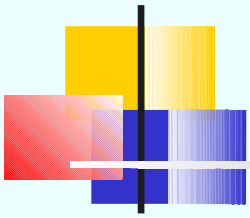
```
// Chap nhan ket noi va xu li du lieu
public void listening(){
try {
    clientSocket = myServer.accept();
    is = new Scanner(clientSocket.getInputStream());
    os = new PrintStream(clientSocket.getOutputStream());

    // Xu li du lieu nhan duoc va tra ve
    while (true) {
        // doc du lieu vao
          input = is.nextLine();

          // xu li du lieu
          ReverseString str = new ReverseString(input);
          str.reverse();

          // tra ve du lieu
          os.println(str.get_string());
    }
}catch (IOException e) {
    System.out.println(e);
}}}
```

Ví dụ: đảo chuỗi – client (1)

```
import javax.net.ssl.SSLSocket;
import javax.net.ssl.SSLSocketFactory;

public class TCPClient {
// khai bao socket cho client, luong vao-ra
SSLSocket mySocket = null;
PrintStream os = null;
Scanner is = null;

// Tao ket noi
public void connection(){
    try {
        SSLSocketFactory sslsf = SSLSocketFactory.getDefault();
        SSLSocket mySocket = sslsf.createSocket("hostname", 9999);
         os = new PrintStream(mySocket.getOutputStream());
         is = new Scanner(mySocket.getInputStream());
    } catch (UnknownHostException e) {
         System.err.println(e);
    } catch (IOException e) {
         System.err.println(e);
    }
}
```

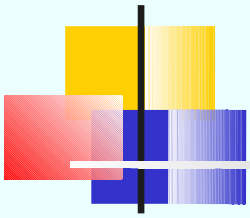
Ví dụ: đảo chuỗi – client (2)

```
public void send(String str){ // gui du lieu den server
    if (mySocket != null && os != null) {
        try {
            os.println(str);
        } catch (UnknownHostException e) {
            System.err.println(e);
        }catch (IOException e) {
            System.err.println(e);
        }
    }
}

public String receive(){ // nhan du lieu tra ve tu server
    String responseStr = null;
    if (mySocket != null && is != null) {
        try {
            responseStr = is.nextLine();
        } catch (UnknownHostException e) {
            System.err.println(e);
        }catch (IOException e) {
            System.err.println(e);
        }
    }
    return responseStr;
}
```



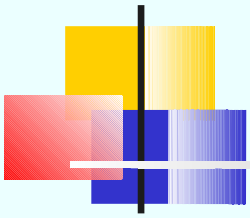
Ví dụ: đảo chuỗi – client (3)

```
// dong cac ket noi
public void close(){
    if (mySocket != null && os != null && is != null) {
        try {
           os.close();
            is.close();
            mySocket.close();
        } catch (UnknownHostException e) {
            System.err.println(e);
        } catch (IOException e) {
            System.err.println(e);
        }
    }
}
}
```



Bài tập (1)

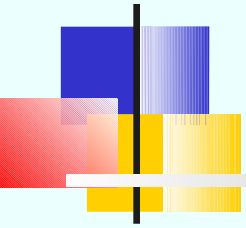
- Cài đặt theo mô hình giao thức TCP/IP và SSL cho bài toán:
- Client yêu cầu người dùng nhập từ bàn phím hai số nguyên dương a và b
- server nhận và tính BSCNN của a và b, sau đó trả về kết quả cho client
- Client nhận lại kết quả và show ra màn hình cho người dùng



Bài tập (2)

Cùng yêu cầu, nhưng cài đặt đúng mô hình MVC

- Cài đặt theo mô hình giao thức TCP/IP và SSL cho bài toán:
- Client yêu cầu người dùng nhập từ bàn phím hai số nguyên dương a và b
- server nhận và tính BSCNN của a và b, sau đó trả về kết quả cho client
- Client nhận lại kết quả và show ra màn hình cho người dùng



Questions?