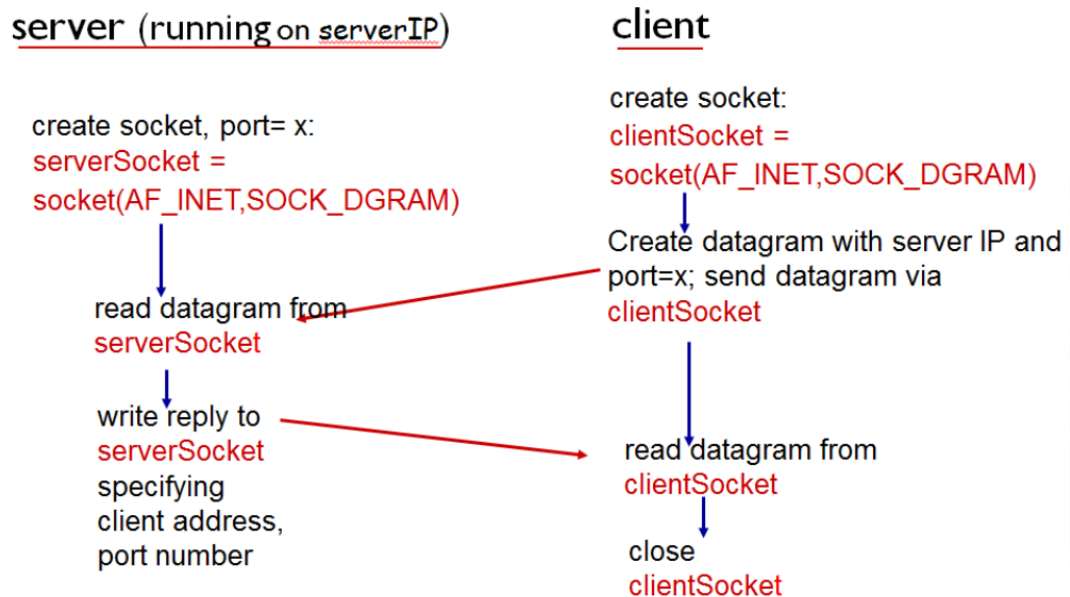


Socket 编程示例 (Python 版)

1、UDP Socket 编程



1) 客户端代码:

```
1 from socket import *
2 serverName = '191.101.232.165' # 服务器地址, 本例中使用一台远程主机
3 serverPort = 12000 # 服务器指定的端口
4 clientSocket = socket(AF_INET, SOCK_DGRAM) # 创建UDP套接字, 使用IPv4协议
5 message = input('Input lowercase sentence:').encode() # 用户输入信息, 并编码为bytes以便发送
6 clientSocket.sendto(message, (serverName, serverPort)) # 将信息发送到服务器
7 modifiedMessage, serverAddress = clientSocket.recvfrom(2048) # 从服务器接收信息, 同时也能得到服务器地址
8 print(modifiedMessage.decode()) # 显示信息
9 clientSocket.close() # 关闭套接字
```

2) 服务器端代码:

```
1 from socket import *
2 serverPort = 12000 # 服务器指定的端口
3 serverSocket = socket(AF_INET, SOCK_DGRAM) # 创建UDP套接字, 使用IPv4协议
4 serverSocket.bind(('', serverPort)) # 将套接字绑定到之前指定的端口
5 print("The server is ready to receive")
6 while True: # 服务器将一直接收UDP报文
7     message, clientAddress = serverSocket.recvfrom(2048) # 接收客户端信息, 同时获得客户端地址
8     modifiedMessage = message.upper() # 将客户端发来的字符串变为大写
9     serverSocket.sendto(modifiedMessage, clientAddress) # 通过已经获得的客户端地址, 将修改后的字符串发回客户端
```

打开命令指示符或者 powershell 下到代码的根目录,先运行服务器, 然后再运行客户端。

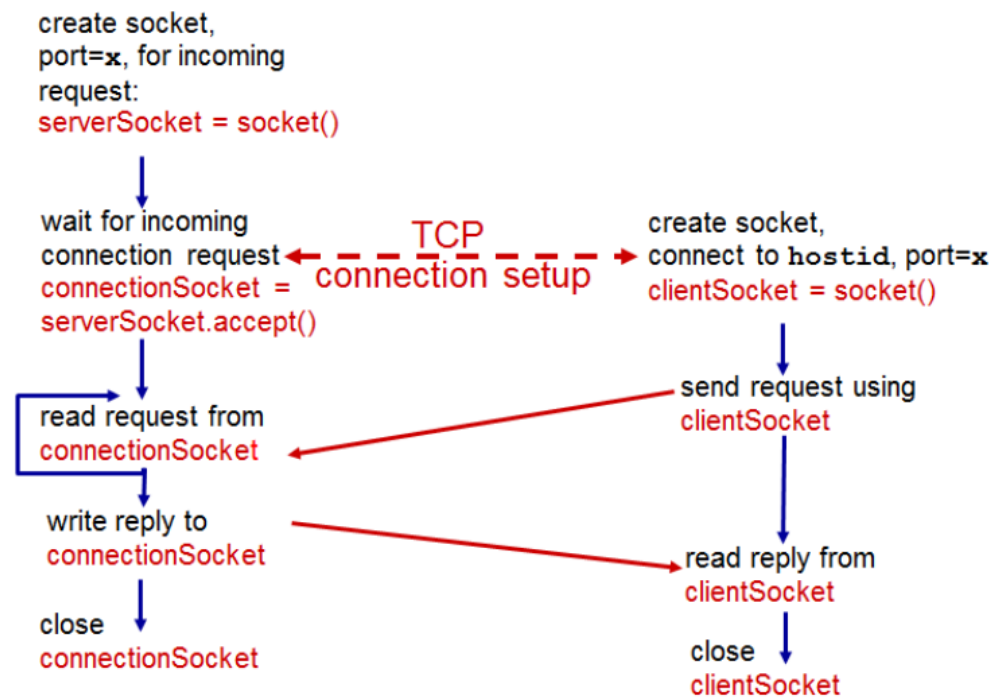
```
PS D:\work\zhujiao\socket\code> python .\UDPServer.py
The server is ready to receive
```

```
PS D:\work\zhujiao\socket\code> python .\UDPClient.py
Input lowercase sentence:jello
JELLO
```

2、TCP Socket 编程

server (running on hostid)

client



1) 客户端代码:

```
1 from socket import *
2 serverName = '191.101.232.165' # 指定服务器地址
3 serverPort = 12000
4 clientSocket = socket(AF_INET, SOCK_STREAM) # 建立TCP套接字, 使用IPv4协议
5 clientSocket.connect((serverName, serverPort)) # 向服务器发起连接
6
7 sentence = input('Input lowercase sentence:').encode() # 用户输入信息, 并编码为bytes以便发送
8 clientSocket.send(sentence) # 将信息发送到服务器
9 modifiedSentence = clientSocket.recvfrom(1024) # 从服务器接收信息
10 print(modifiedSentence[0].decode()) # 显示信息
11 clientSocket.close() # 关闭套接字
```

2) 服务器端代码

```

1  from socket import *
2  serverPort = 12000
3  serverSocket = socket(AF_INET, SOCK_STREAM) # 创建TCP欢迎套接字, 使用IPv4协议
4  serverSocket.bind(('',serverPort)) # 将TCP欢迎套接字绑定到指定端口
5  serverSocket.listen(1) # 最大连接数为1
6  print("The server is ready to receive")
7
8  while True:
9      connectionSocket, addr = serverSocket.accept() # 接收到客户连接请求后, 建立新的TCP连接套接字
10     print('Accept new connection from %s:%s...' % addr)
11     sentence = connectionSocket.recv(1024) # 获取客户发送的字符串
12     capitalizedSentence = sentence.upper() # 将字符串改为大写
13     connectionSocket.send(capitalizedSentence) # 向用户发送修改后的字符串
14     connectionSocket.close() # 关闭TCP连接套接字

```

打开命令指示符或者 powershell 下先运行服务器, 然后再运行客户端。

```

PS D:\work\zhujiao> python .\socket\code\TCPServer.py
The server is ready to receive
Accept new connection from 127.0.0.1:58712...

```

```

PS D:\work\zhujiao\socket\code> python .\TCPClient.py
Input lowercase sentence:hello
HELLO

```

3、Web 服务器

Web服务器 - hints

- 典型http请求报文包头格式

GET /somedir/page.html HTTP/1.1

Host: www.someschool.edu

Connection: close

User-agent: Mozilla/5.0

Accept-language:fr

- 典型http响应报文包头格式

HTTP/1.1 200 OK

Connection: close

Content-Type: text/html

Content-Length: length of data

Data data data ...

- 1) 把编好的 WebServer.py 和 HelloWorld.html 两个文件放在一台主机的同一个目录下, 并运行 WebServer.py, 作为服务器

2) 在另一台主机上打开浏览器，并输入：

<http://xxx.xxx.xxx.xxx:PortNumber/HelloWorld.html>

以获取服务器上的 HelloWorld.html 文件。其中 xxx.xxx.xxx.xxx 是服务器的 IP 地址。如果在同一台主机上运行，IP 地址则为 127.0.0.1

3) 一切正常的话，将看到：



4、相关软件/示例代码说明

文件说明：

- UDP 套接字代码示例：UDPServer.py UDPCClient.py
- TCP 套接字代码示例：TCPServer.py TCPClient.py
- Python 安装文件：python-3.6.3.exe
- 实验文件：WebServer_1.py (代码框架) HelloWorld.html (网页文件)