华东师范大学数据科学与工程学院实验报告

课程名称：计算机网络与编程	**年级：**2021	**上机实践成绩：**
指导教师：张召	**姓名：温兆和**	**学号：**10205501432
上机实践名称：UDPping 程序		**上机实践日期：**2022.05.27
上机实践编号：14	**组号：**001-432	**上机实践时间：**13：00

一、实验目的

了解系统命令 ping 的用法；
实现简单的 UDPping 程序。

二、实验任务

ping 命令的简单使用；

使用 IDEA 实现简单的 UDPping 程序。

三、使用环境

Intellij IDEA 2020.3.2
JDK 11.0.6

四、实验过程

Task1. 参考上述提供的服务器端代码，结合 Socket 编程课上学习的相关知识，利用 JAVA 编写实现客户端代码，请将实验结果附在实验报告中。

客户端代码：

```
package WEIZHI;
import java.io.IOException;
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.nio.charset.StandardCharsets;
import java.net.InetAddress;
import java.util.Calendar;

public class TCPClient {
    public static void main(String[] args) throws IOException {
        MessageUtil m1 = new MessageUtil();
        String sendBytes = m1.buildWithPort(m1.port);
        long t1,t2,t=0;
        DatagramSocket datagramSocket = new DatagramSocket(30000);
        for(int i=0;i<10;i++)
        {
            DatagramPacket sendPacket = new DatagramPacket(sendBytes.getBytes(), 0,
sendBytes.getBytes().length, InetAddress.getByName("255.255.255.255"), 9091);
            datagramSocket.send(sendPacket);
            t1 = System.currentTimeMillis();
            System.out.print(i+1);
            System.out.println(" sent");
            byte[] buf = new byte[1024];
            DatagramPacket receivePacket = new DatagramPacket(buf,buf.length);
            datagramSocket.receive(receivePacket);
            t2 = System.currentTimeMillis();
            t=t2-t1;
            System.out.println(t+" equals "+t2+" minus "+t1);
            String ip = receivePacket.getAddress().getHostAddress();
```

```
            int port = receivePacket.getPort();
            int len = receivePacket.getLength();
            String data = new String(receivePacket.getData(),0, len);
            String bb = m1.parseTag(data);
            System.out.println(bb);
            System.out.println("RTT rquals "+t);
        }
        datagramSocket.close();
    }
}
class MessageUtil {
    private static final String TAG_HEADER = "special tag:";
    private static final String PORT_HEADER = "special port:";
    public int port = 30000;
    public String buildWithPort(int port) {
        return PORT_HEADER + port;
    }
    public int parsePort(String data) {
        if (data.startsWith(PORT_HEADER)) {
            return Integer.parseInt(data.substring(PORT_HEADER.length()));
        }
        return -1;
    }
    public String buildWithTag(String tag) {
        return TAG_HEADER + tag;
    }
    public String parseTag(String data) {
        if (data.startsWith(TAG_HEADER)) {
            return data.substring(TAG_HEADER.length());
        }
        return null;
    }
}
```

服务器代码：

```
    package WEIZHI;
import java.io.IOException;
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.nio.charset.StandardCharsets;
import java.util.UUID;
public class TCPServer {
    public static void main(String[] args) throws IOException {
        MessageUtil m2 = new MessageUtil();
        DatagramSocket datagramSocket = new DatagramSocket(9091);
        byte[] buf = new byte[1024];
        for(int i=0;i<10;i++)
        {
            DatagramPacket receivePacket = new DatagramPacket(buf, 0, buf.length);
            System.out.println("Blocked and wait for the searcher...");
            datagramSocket.receive(receivePacket);
            String ip = receivePacket.getAddress().getHostAddress();
            int port = receivePacket.getPort();
            int len = receivePacket.getLength();
            String data = new String(receivePacket.getData(),0, len);
            int a = m2.parsePort(data);
            System.out.println("I'm receiver." + ip + ":"+ port + "'s sender
says:"+
                    data);
            String tag = UUID.randomUUID().toString() ;
            String responseData = m2.buildWithTag(tag);
            byte[] responseDataBytes =
```

```
responseData.getBytes(StandardCharsets.UTF_8);
            DatagramPacket responsePack = new DatagramPacket(responseDataBytes,
responseDataBytes.length, receivePacket.getAddress(), a);
            datagramSocket.send(responsePack);
        }
        datagramSocket.close();
    }
}
```

结果：

"C:\jdk-18_windows-x64_bin (1)\jdk-18.0.1.1\bin\java.exe" "-javaagent:C:\Program Files (x86)\Windows Mail\IntelliJ IDEA 2022.1\lib\idea_rt.jar=57218:C:\Program Files (x86)\Windows Mail\IntelliJ IDEA 2022.1\bin" -Dfile.encoding=UTF-8 -classpath C:\Users\HUAWEI\IdeaProjects\untitled2\out\production\untitled2 WEIZHI.TCPClient

1 sent
51 equals 1653631709083 minus 1653631709032
92ff7cda-4513-48de-a8ce-bd2eefd80f7e
RTT rquals 51
2 sent
0 equals 1653631709083 minus 1653631709083
3e8dd183-e612-41a0-ba03-2a5a15b759d8
RTT rquals 0
3 sent
0 equals 1653631709083 minus 1653631709083
159c9057-c8ea-4734-9d05-aab8798ac2ad
RTT rquals 0
4 sent
0 equals 1653631709094 minus 1653631709094
e45fc28e-c62e-4e73-954f-60b7b79cf36d
RTT rquals 0
5 sent
0 equals 1653631709094 minus 1653631709094
44c04356-8fa6-4d51-8d10-d1423013dc9c
RTT rquals 0
6 sent
0 equals 1653631709096 minus 1653631709096
b0f80ba8-636e-450c-af0d-6e3fb731ddfc
RTT rquals 0
7 sent
0 equals 1653631709096 minus 1653631709096
c35e4797-003d-4ce5-9c7b-b8eaf3b27b5e
RTT rquals 0
8 sent
1 equals 1653631709097 minus 1653631709096
65f406ac-312c-4adc-91db-f13c9dccb02e
RTT rquals 1
9 sent
0 equals 1653631709097 minus 1653631709097
7d10f290-fc80-4b63-abda-7f097ceaf5c8
RTT rquals 0
10 sent
0 equals 1653631709097 minus 1653631709097

4251781a-3a62-4fd1-ac72-5c50f2b63bc3
RTT rquals 0

Process finished with exit code 0

我们可以看到，除了第一次发送和回复的传播时延是 51 毫秒，其余次数时延均为 0。

五、总结

在本周的实验中，我们了解了系统命令 ping 的用法并实现了简单的 UDPping 程序，为后续的学习打好了基础。