**西安电子科技大学**

**组网与运维综合实验 课程实验报告**

**实验名称 TCP/IP报文分析**

网络与信息安全 学院 2118021 班

成 绩

姓名 学号

同作者

实验日期 2023 年 11 月 30 日

|  |
| --- |
| 指导教师评语：  指导教师：  年 月 日 |
| **实验报告内容基本要求及参考格式**  一、实验目的  二、实验所用仪器（或实验环境）  三、实验基本原理及步骤（或方案设计及理论计算）  四、实验数据记录（或仿真及软件设计）  五、实验结果分析及回答问题（或测试环境及测试结果） |

# TCP/IP报文分析

## 一、实验目的

1. 掌握H3C设备Ping和Tracert命令的使用。

2. 掌握H3C设备的系统调试功能。

3. 掌握ICMP报文在Ping操作下的工作原理。

4. 掌握H3C设备TCP参数的设置。

5. 在H3C设备上进行TCP报文分析。

6. 在H3C设备上进行UDP报文分析。

7. 进一步熟悉debug命令的使用。

## 二、实验要求

1. 3台具有24个以太网接口的交换机；

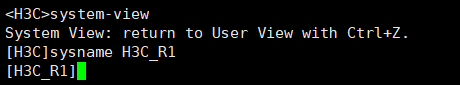
2. 2台装有Windows系列操作系统的PC（台式机或笔记本）；

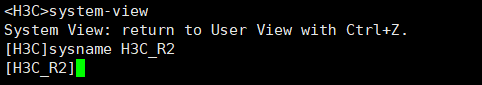
3. 2条双绞跳线（交叉线）；

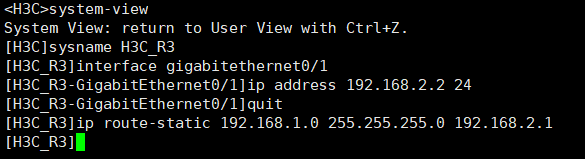
## 三、实验步骤

1. 按实验1要求配置H3C路由器基本参数

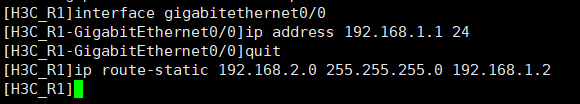
（1）首先将实验所用的3台路由器分别重命名成H3C\_R1、H3C\_R2、H3C\_R3，以便后续区分。

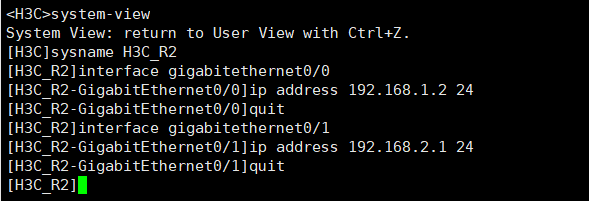


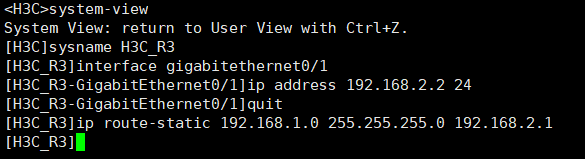




（2）之后按照实验步骤对三台路由器进行参数配置，参考表7.1中的信息，按照实物连接的接口信息配置路由器接口的ip地址，并设置静态路由，经由R2连通R1和R3.

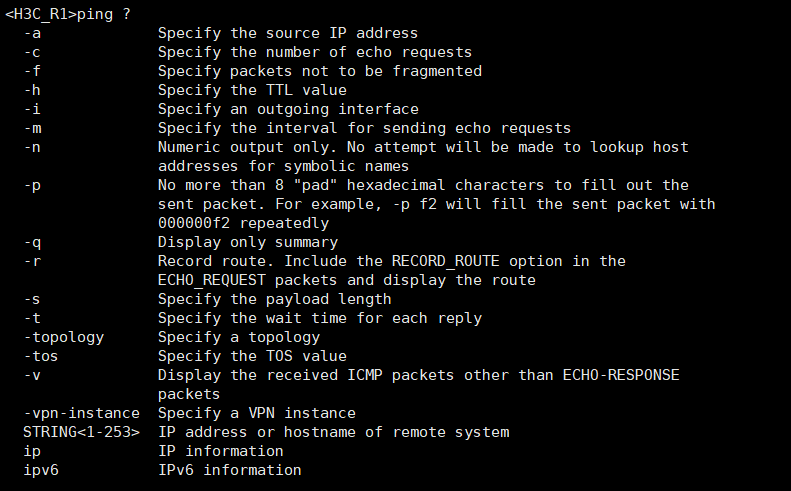






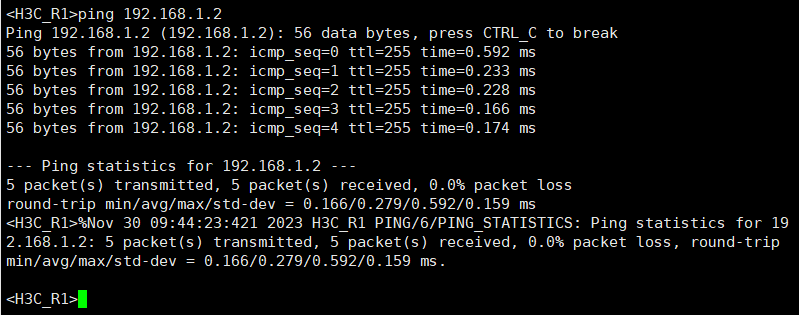
2. 掌握Ping调试工具

（1）首先根据帮助信息，可以看到，ping后接对应ip，可以发送测试信息来测试路由对指定ip的连通性；而-r参数，其功能为显示记录的报文路线，因此可以通过在ping指令后加入-r参数来具体查看报文的输送轨迹，进而验证路由间的连通性。



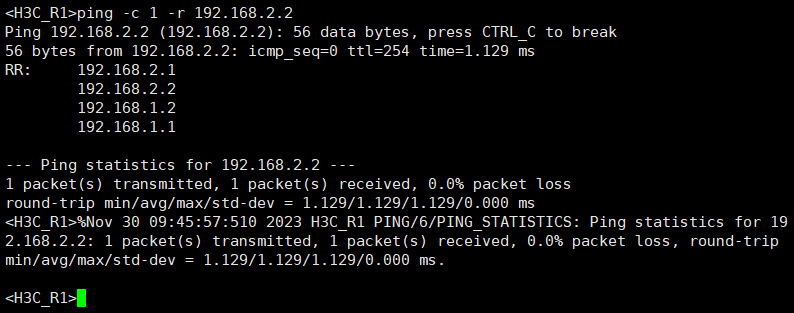
（2）首先在R1上测试R2的IP地址192.168.1.2是否可达，即验证R1与R2之间的连通性。R1与R2之间未经路由的中转，且端口ip正确配置，最后也观察到R1可以到达R2的IP地址192.168.1.2，R1发出的ICMP回显请求报文均能得到回应。

传输的最短时间、平均时间、最长时间分别为0.166ms、0.279ms、0.592ms。



（3）之后在R1上测试R3的IP地址192.168.2.2是否可达，即验证R1与R3之间的连通性。如下图所示，R1可以到达R3的IP地址192.168.2.2，R1发出的ICMP回显请求报文均能得到回应。R1与R3之间经过了R2配置的静态路由路线，如RR（record route）所示，其经过的端口倒序排列，符合我们设置的静态路由的预期。

传输的最短时间、平均时间、最长时间分别为1.129ms、1.129ms、1.129ms。



（4）ping -r原理

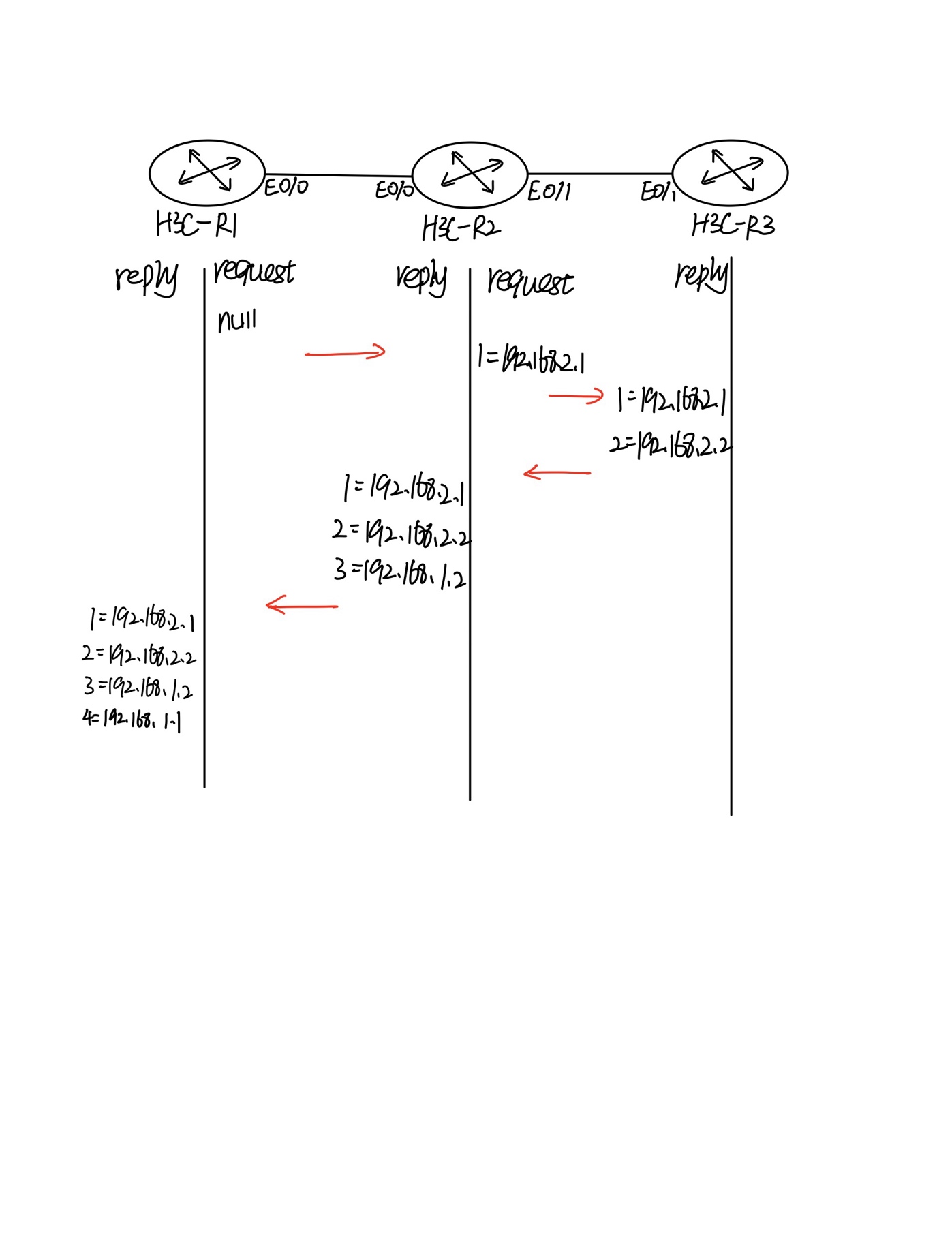
Ping -r原理图如下，首先R1发送RR选项为空的ICMP回显请求，给目的端设备R3.

回显请求报文到R2时，R2将自己的出接口IP地址（192.168.2.1）添加到ICMP回显请求报文的RR选项中，并转发该报文到目的端R3.

目的端R3收到请求报文后，发送ICMP回显响应报文，响应报文会拷贝请求报文的RR选项，并将自己的出接口IP地址（192.168.2.2）添加到RR选项中，之后将回显报文再传回R2.

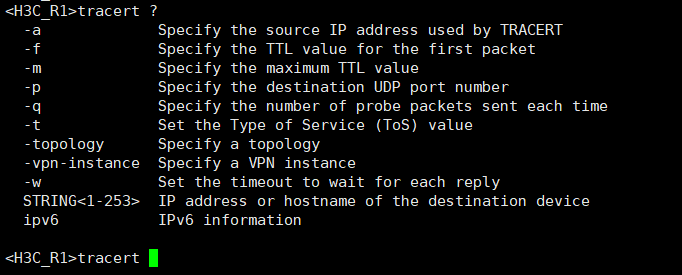
中间设备R2将报文传回R1，并将自己的出接口IP地址（192.168.1.2）添加到RR选项中。

当源端设备R1收到ICMP回显响应报文后，将自己的入接口IP地址（192.168.1.1）添加到RR选项中，最终得到实验结果所示的RR报文。



3. 掌握Tracert调试工具

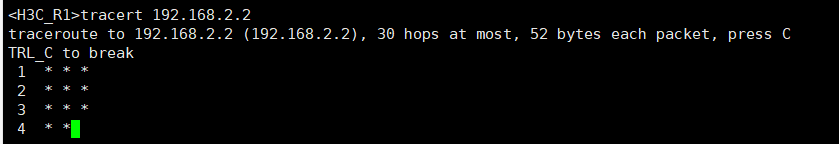
（1）首先查看tracert指令的帮助，可以发现输入对应的ip以后，可以查看IP报文从源端到达目的端所经过的三层设备，从而检查网络连接是否可用。



（2）在 R1 上使用 tracert 命令查看报文从源端到目的端（IP：192.168.2.2）

所经过的路径。发现路由器超时“ \* \* \* ”，是由于默认情况下，H3C 设备的

ICMP 超时报文发送功能关闭状态，CTRL+C 终止此次 tracert 操作。

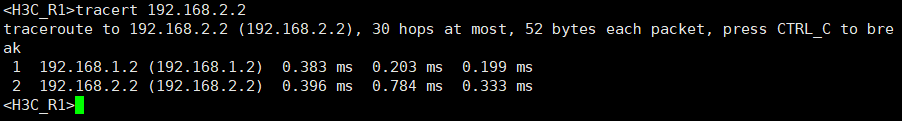


（3）为了解决超时情况，我们需要在中间设备（R2）上开启ICMP超时报文发送功能；并在目的端开启ICMP目的不可达报文发送功能





（4）之后再次使用 tracert 命令查看报文从源端到目的端所经过的路径。可以看到结果给出了1和2两个TTL，TTL值为1显示的是中间设备R2-E0/0端口的IP地址192.168.1.2；TTL值为2显示的是目的端设备R3-E0/1端口的IP地址192.168.2.2。



（5）tracert原理

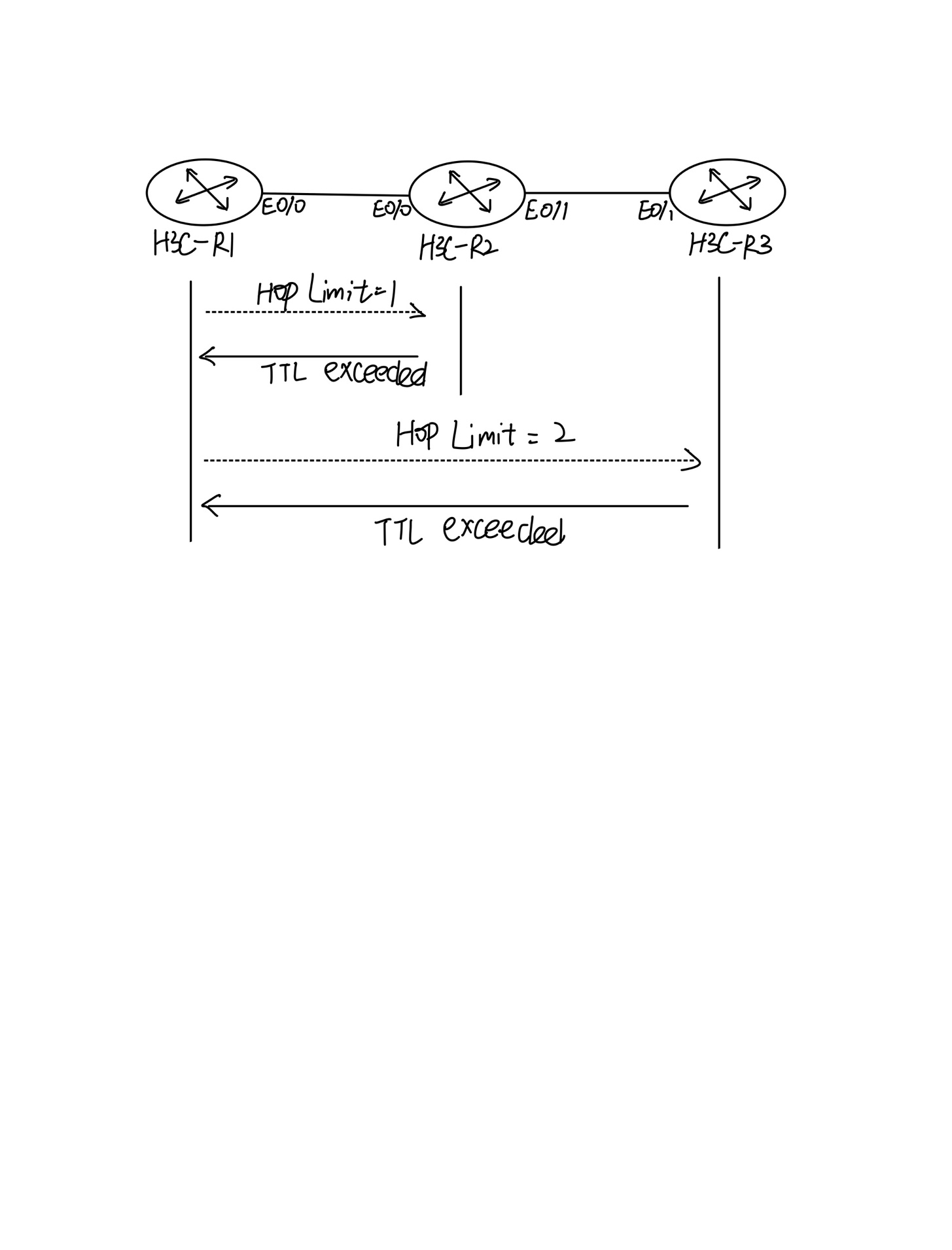
Tracert命令的原理如下图所示。

首先R1向目的端R3发送一个IP数据报文，TTL为1，报文的UDP端口号是目的端的任何一个应用程序都不可能使用的端口号。

第一跳后，R2回应一个TTL超时的ICMP错误信息，该报文含有第一跳的目标IP地址（192.168.1.2），这样源端就得到了第一个三层设备的地址。

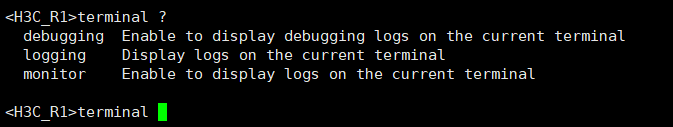
之后源端重新向目的端发送一个IP数据报文，TTL值为2，直到该报文到达目的端，由于目的端没有应用程序使用该UDP端口，因此目的端返回一个端口不可达的ICMP错误消息，并携带了该端口的IP地址（192.168.2.2）.

当源端R1收到端口不可达的ICMP错误消息之后，就可以知道报文已经到达了目的端R3，从而的得到了报文所经历的路径。

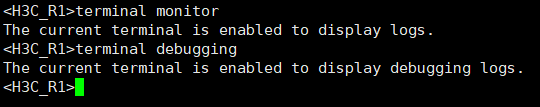


4. 配置系统调试功能——Ping

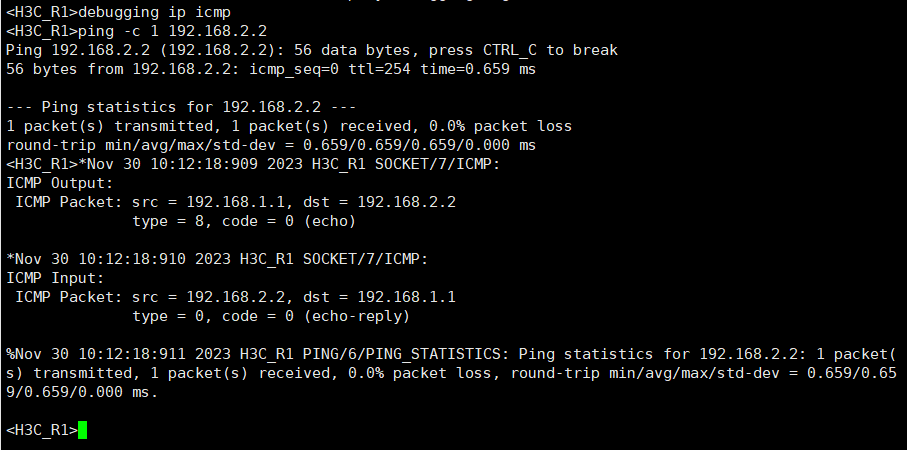
（1）在用户视图下使用terminal ？查看指令帮助，可以发现本设备支持debugging和monitor来实现调试功能



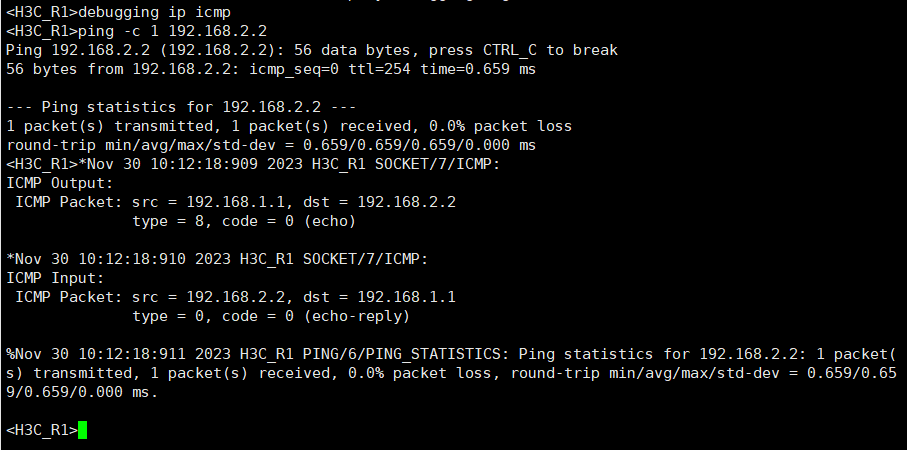
（2）开启系统信息监视和调试信息显示功能，配置系统调试功能，来达到调试的效果



（3）在R1上打开ICMP报文信息调试开关，并使用ping命令向R3-E0/1终端（192.168.2.2）发送一个ping报文，来观察调试信息。



（4）观察调试信息，发现R1向R3的目的地址192.168.2.2发出ICMP echo（Type=8，code=0）报文，R3收到报文以后，以192.168.2.2为源地址，以R1（192.168.1.1）为目的地址，发送了ICMP echo-reply，且被R1收到，表明R1与R3的192.168.2.2是连通的。

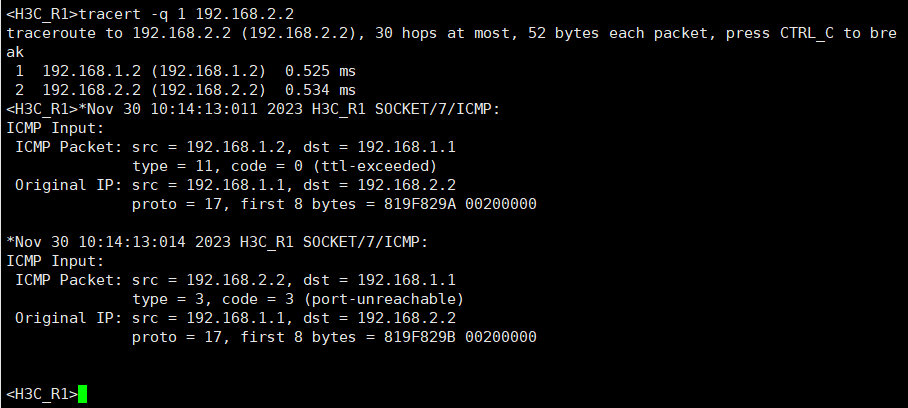


5. 配置系统调试功能——tracert

Tracert命令的debug调试可以看到R1接收的ICMP信息。

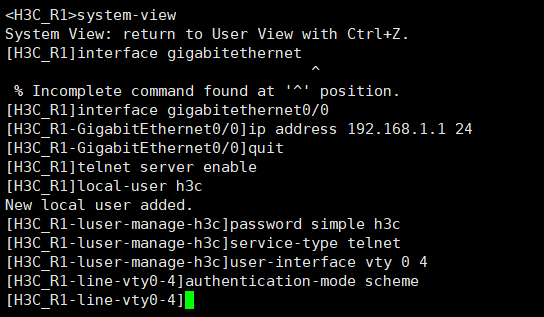
R1发送的一个目的地址为192.168.2.2，TTL值为1的IP数据报文实质，当报文到达R2时，R2发现报文的目的地址不是本地，且报文的TTL字段为1，则发送“TTL超时”的ICMP差错报文ttl-exceeded（type=11， code=0）。

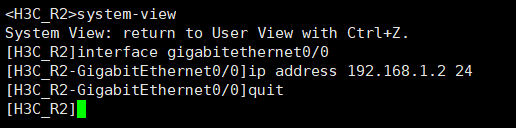
R1根据ICMP差错报文获得网关地址的下一跳为192.168.1.2，因此R1会重新向R3发送一个IP数据报文，其目的地址为192.168.2.2，TTL值为2。R3收到该报文以后，根据报文的目的端口无法找到对应的进程，则又向报文源端R1发送了一个ICMP不可达报文port-unreachable（type=3， code=3）。根据不可达报文可知，TTL=2的报文到达了R3。



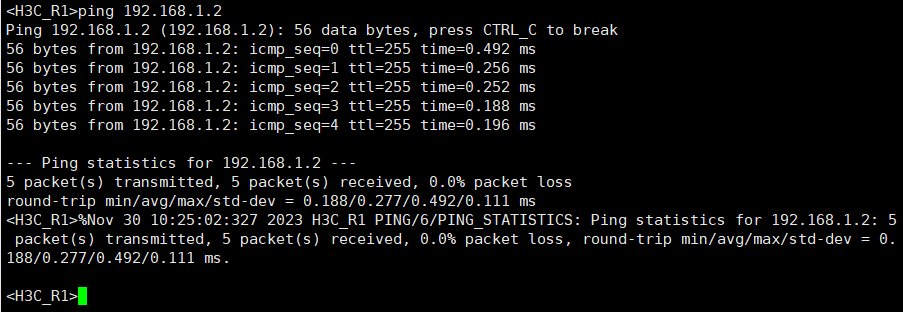
6. 按实验2配置H3C路由器基本参数

（1）根据表7-3的两个路由器端口的对应IP地址，按照参考资料给出的代码，配置路由器R1和R2的基本参数



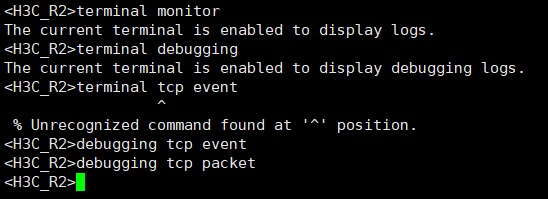


（2）之后ping测试R1与R2的连通性，可以看到连通性正常。

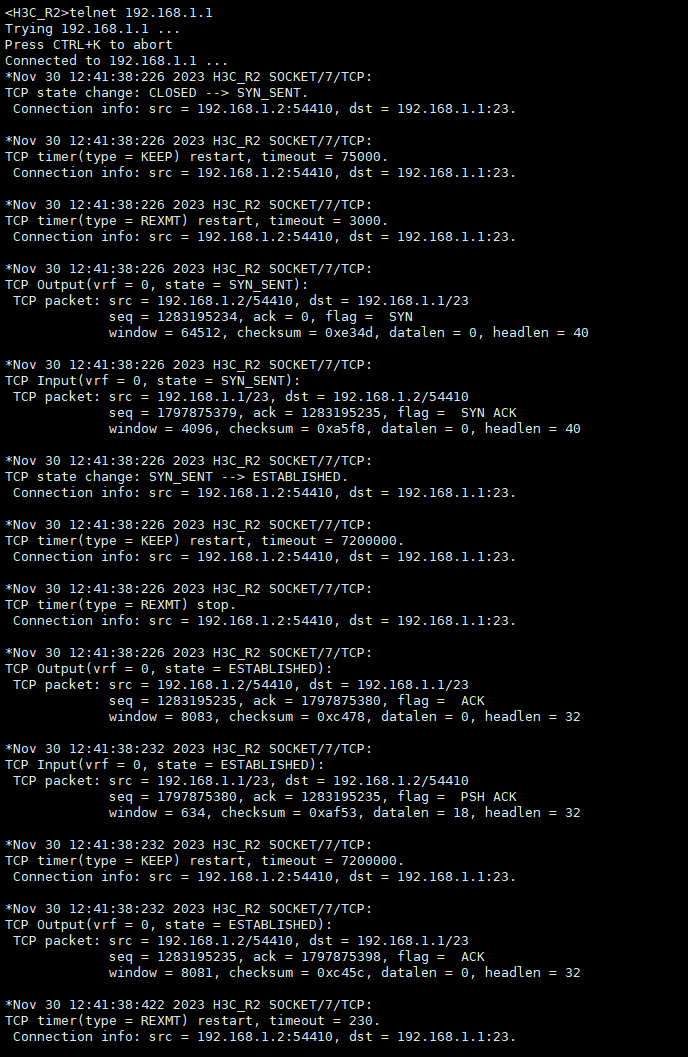


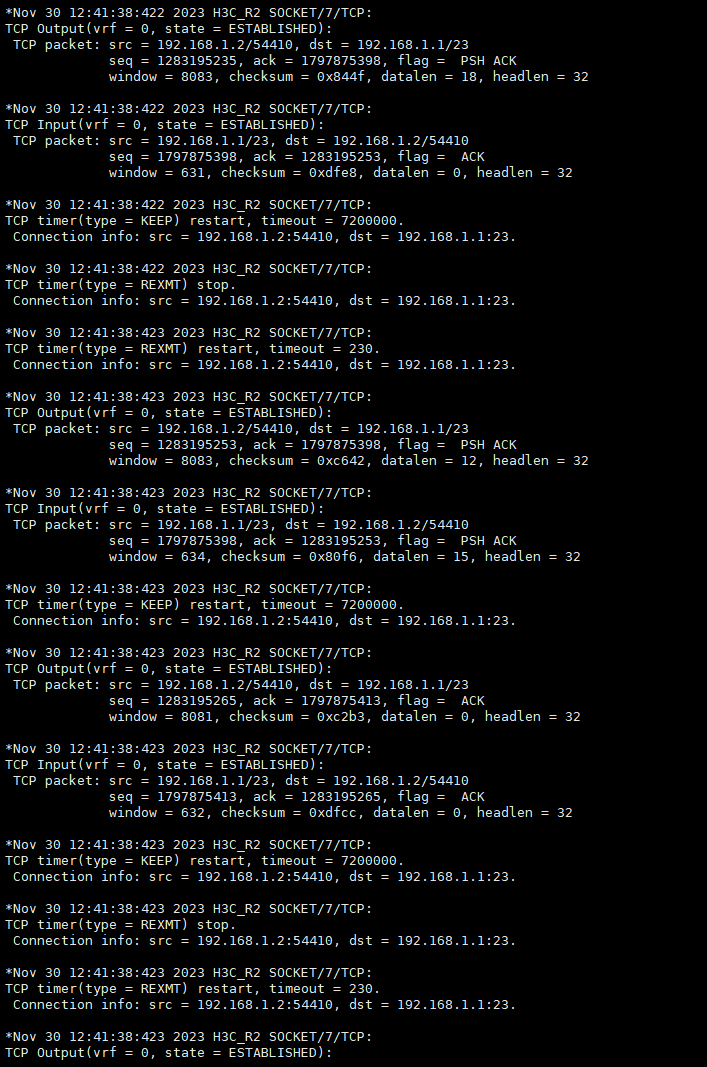
7. 测试TCP信息

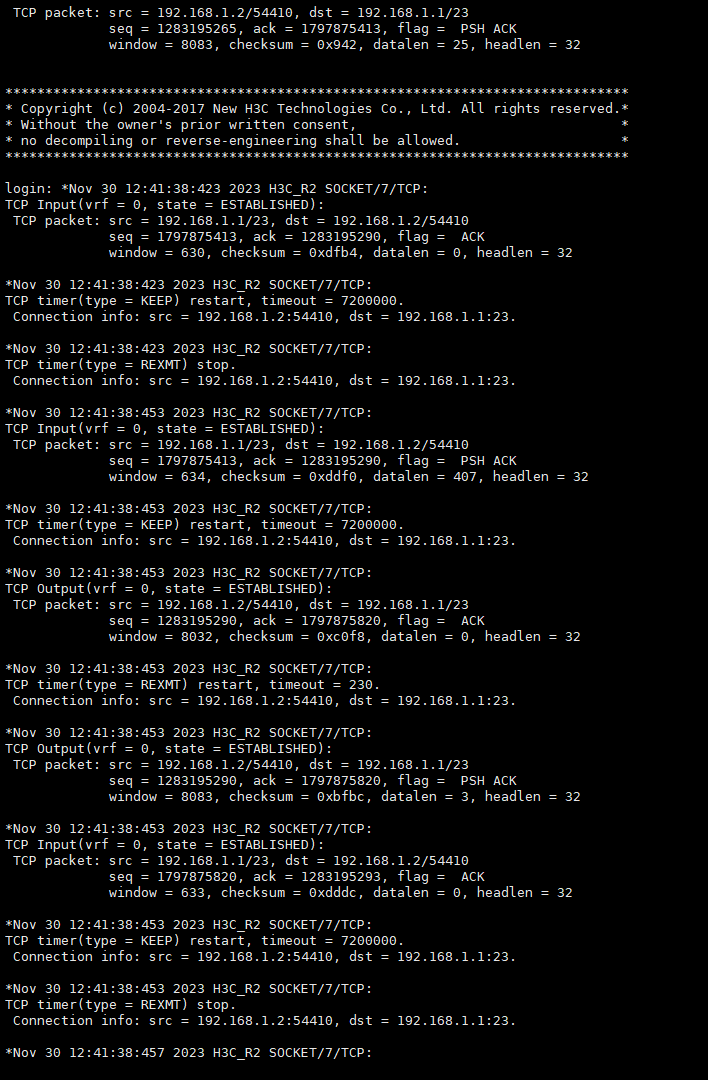
（1）首先打开R2的debugging信息，关闭ip icmp的调试信息，打开tcp event和tcp packet信息的调试。

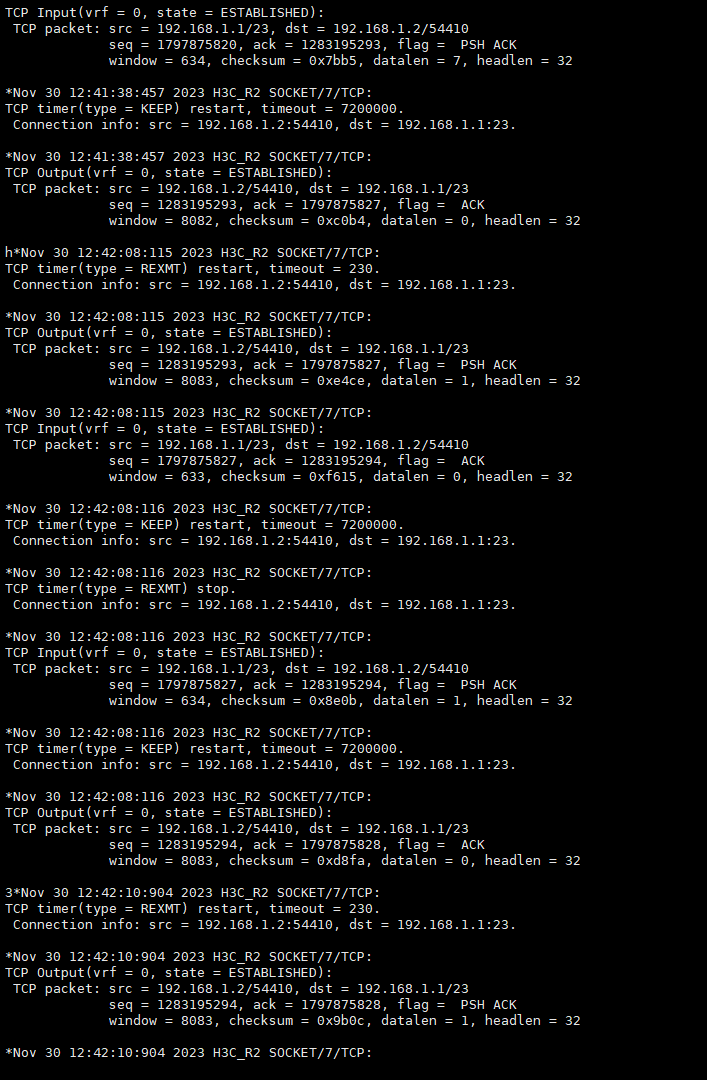


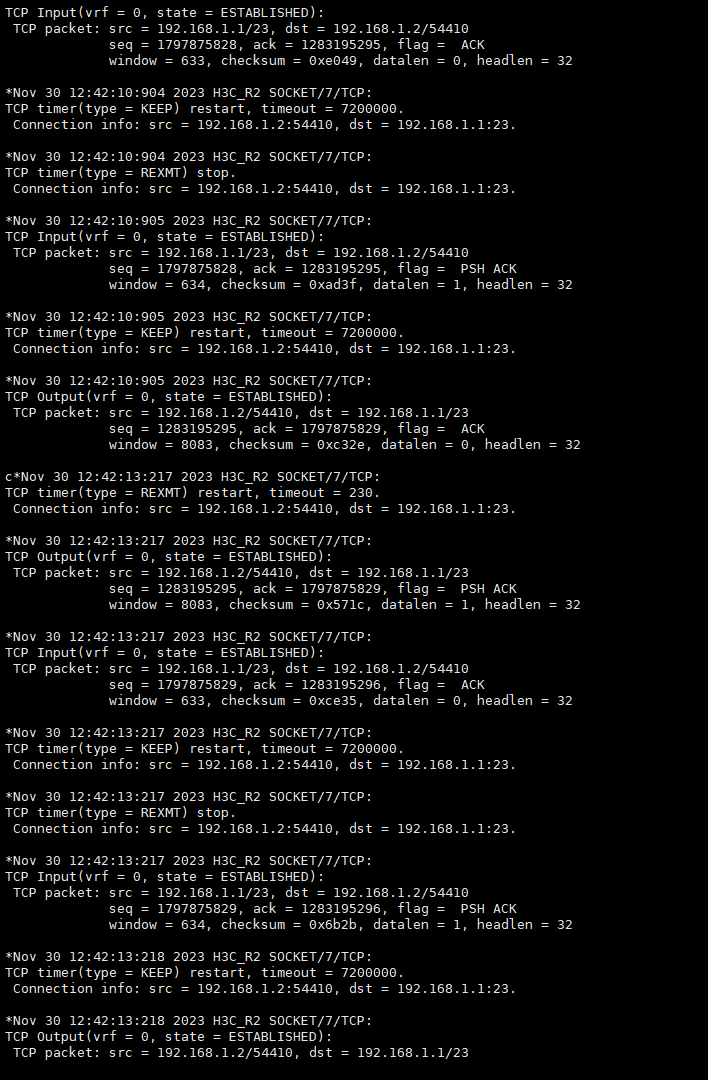
（2）之后在R2中telnet R1-E0/0（192.168.1.1），由于每次数据输入进行传输时，都会进行三次握手，因此以下12页张图包含了建立连接和关闭连接的所有过程，之后我选取一次通信过程来分析三次握手，并分析断开连接时的四次握手过程。

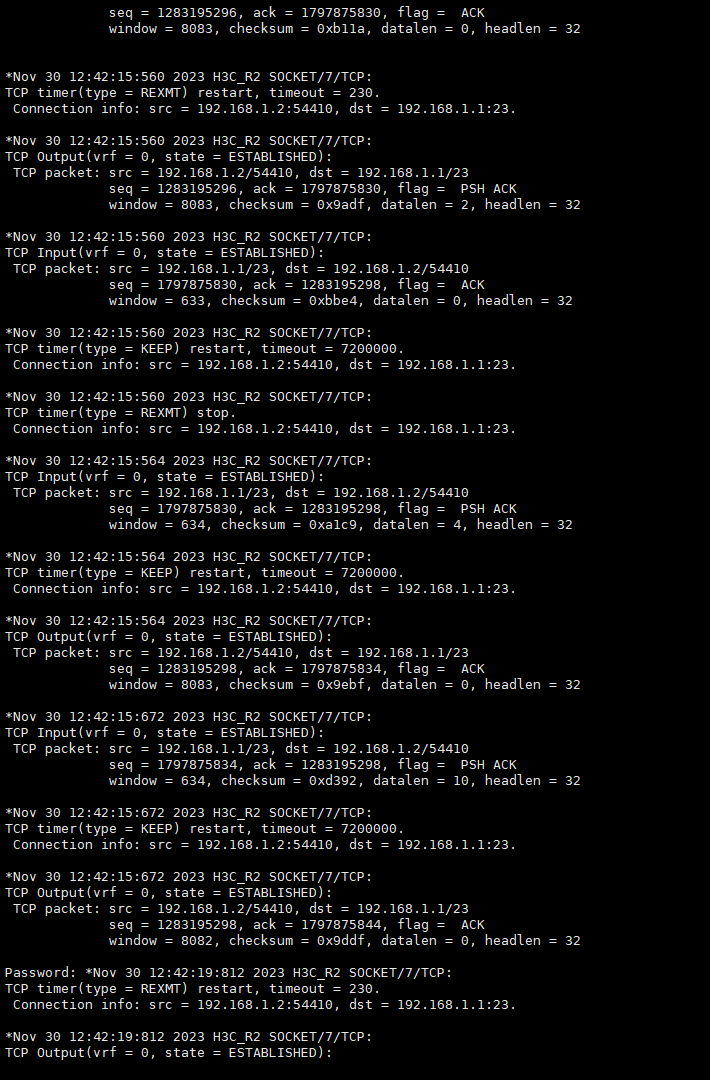


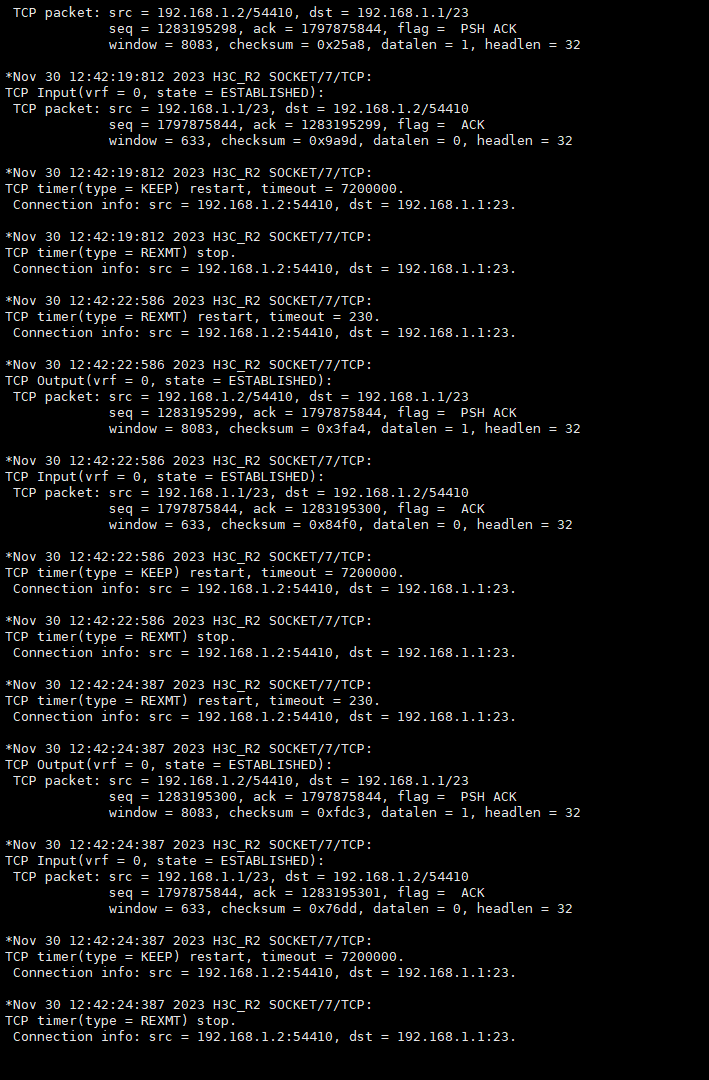


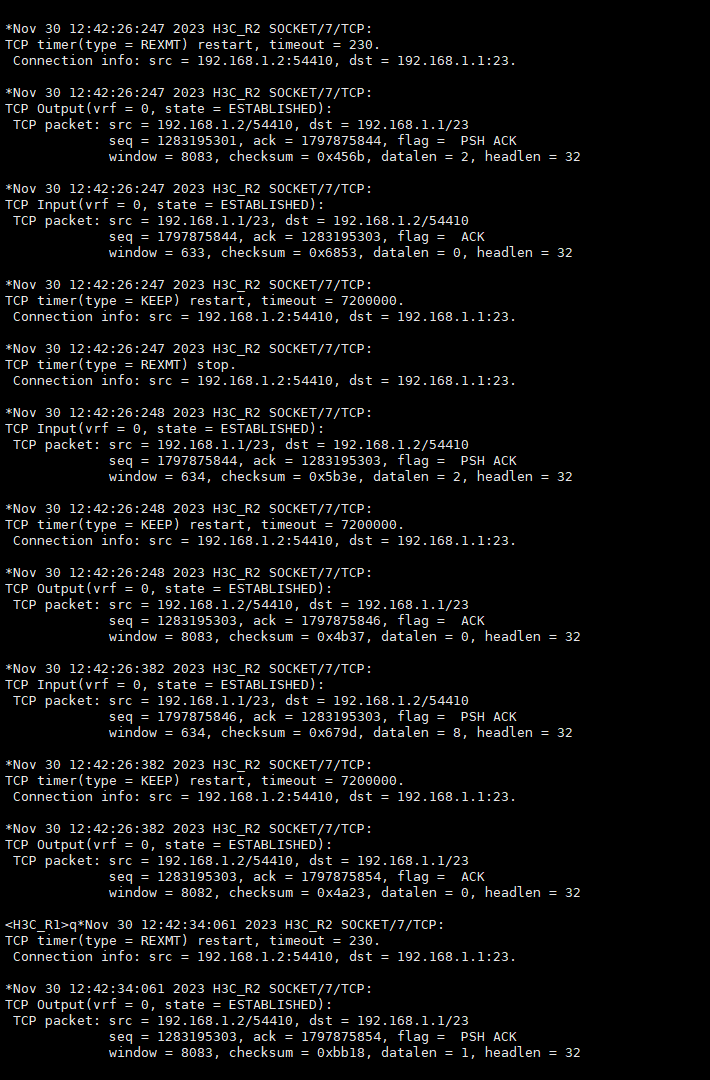


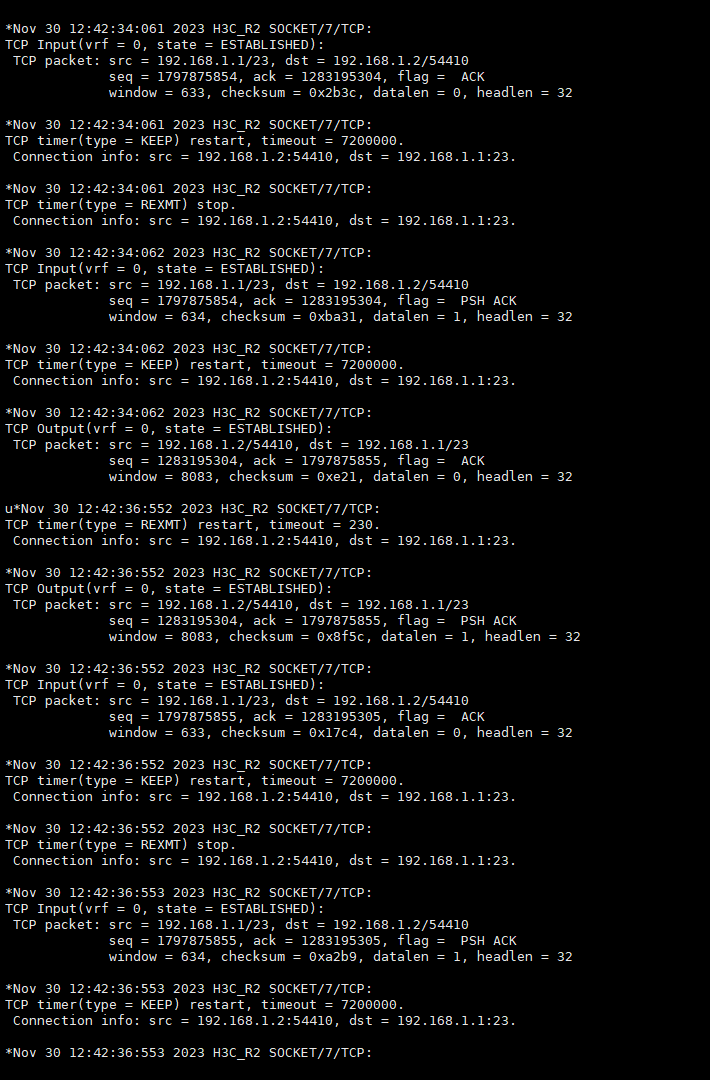


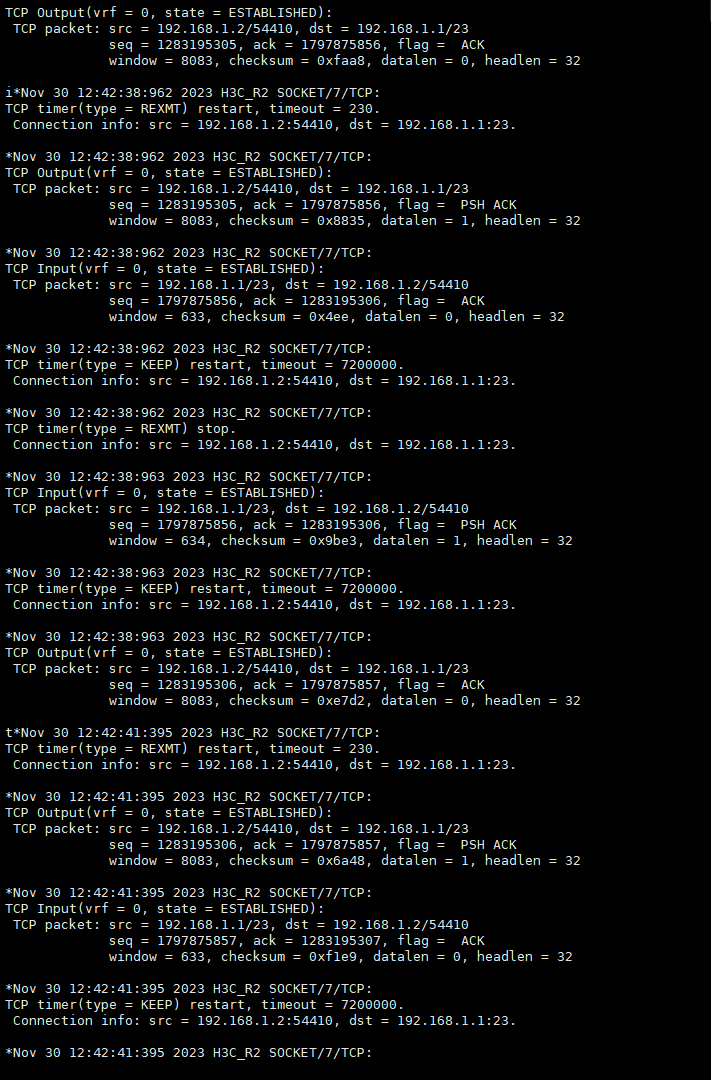


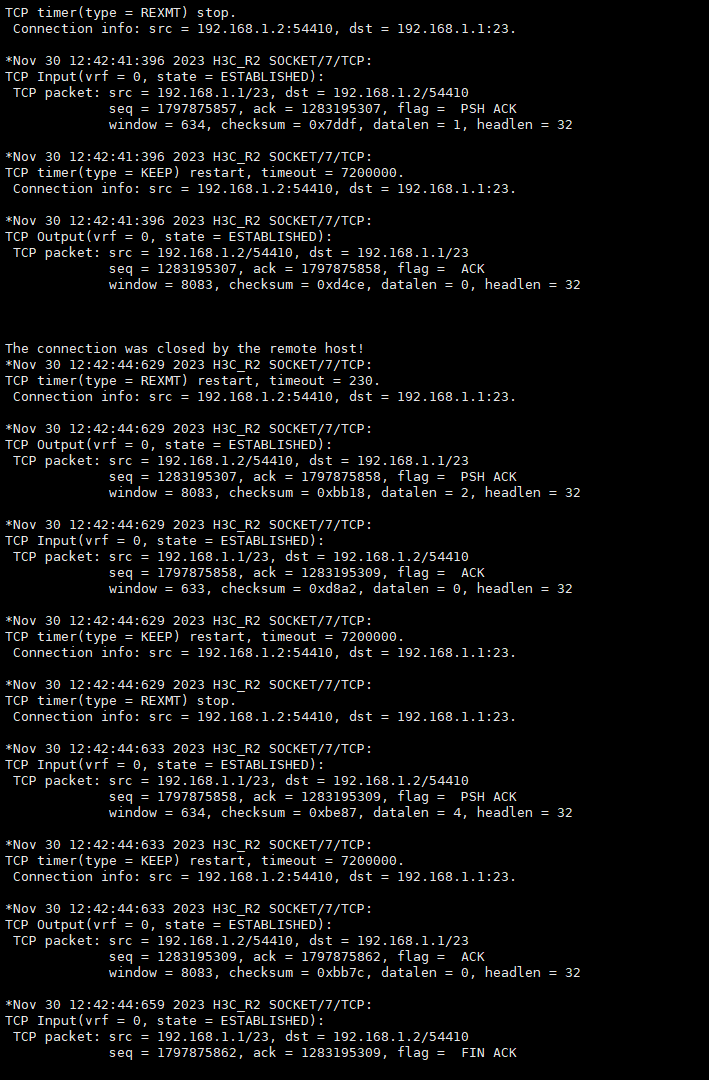


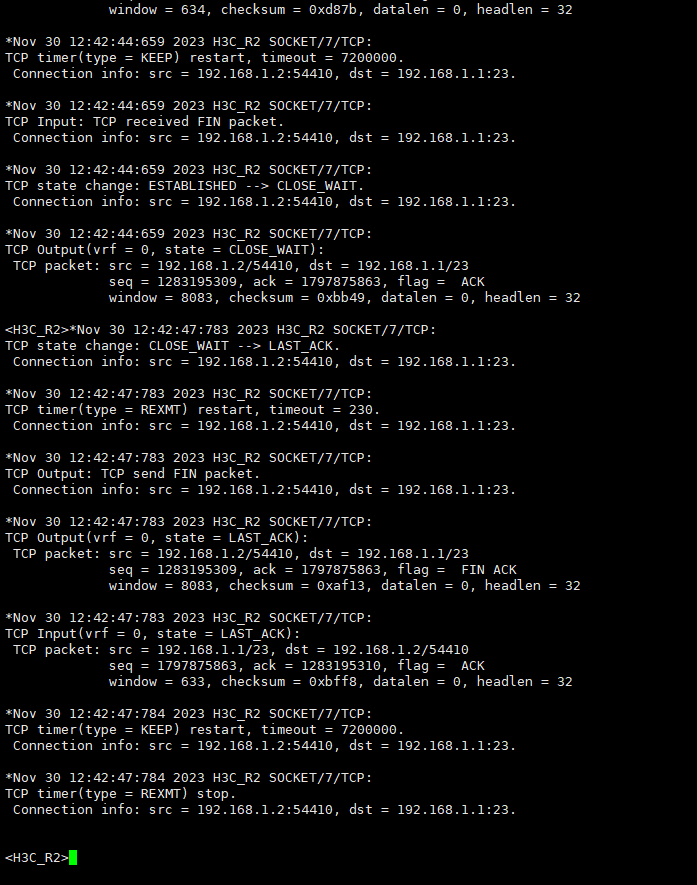












（3）**三次握手**

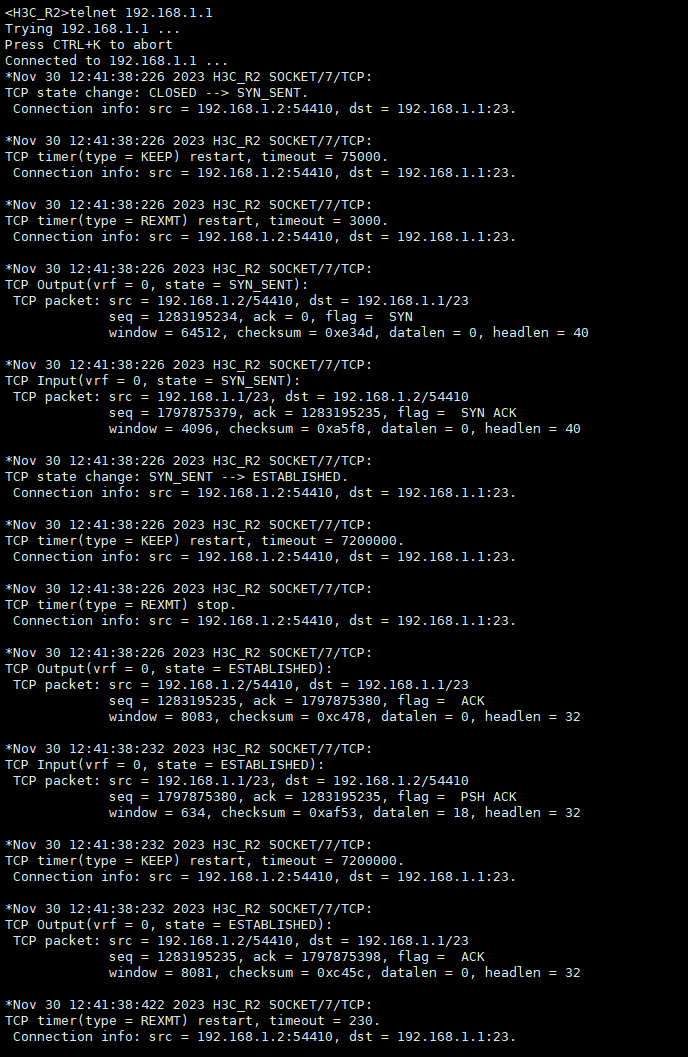
根据tcp packet的调试结果，可以分析出三次握手的过程如下：

1. state改变为SYN\_SENT，发送SYN信息来建立连接。R2要与R1建立连接，所以首先**output** SYN报文段，可以看到此报文段的seq = a = 1283195234，flag为SYN表示SYN报文段。

2.R1接收到R2发送的SYN报文段以后，发送ACK SYN报文段来表示自己接收到了R1的连接请求。因此R2接收到了**input**的SYN报文段，可以看到此报文段的seq = b = 1797875379，ack = a + 1 = 1283195235，为R1的SYN报文段的seq + 1；flag为SYN ACK表示已经R1接收到了R2发送的SYN报文段。

3.最后R2向R1 **input**最后一次握手信息SYN报文段。其中报文段的seq = a+1 = 1283195235，ack = b+1 = 1797875380，flag = ACK，表示R2接收到了R1已经接收到了连接请求。

至此，三次握手完成，连接成功建立。



（4）**四次握手**

由于TCP信道的半关闭特性，且TCP是全双工连接，因此每个方向必须都单独对信道进行关闭，一方传输结束以后发送FIN = 1的报文段来单方向终止连接，TCP不再受理本方应用进程的数据发送。另一段收到FIN之后，对FIN进行确认，之后发送ACK报文来表示接收到关闭请求，并发送FIN来关闭另一个方向的连接，最后关闭方发送ACK报文确定连接关闭，即四次握手。

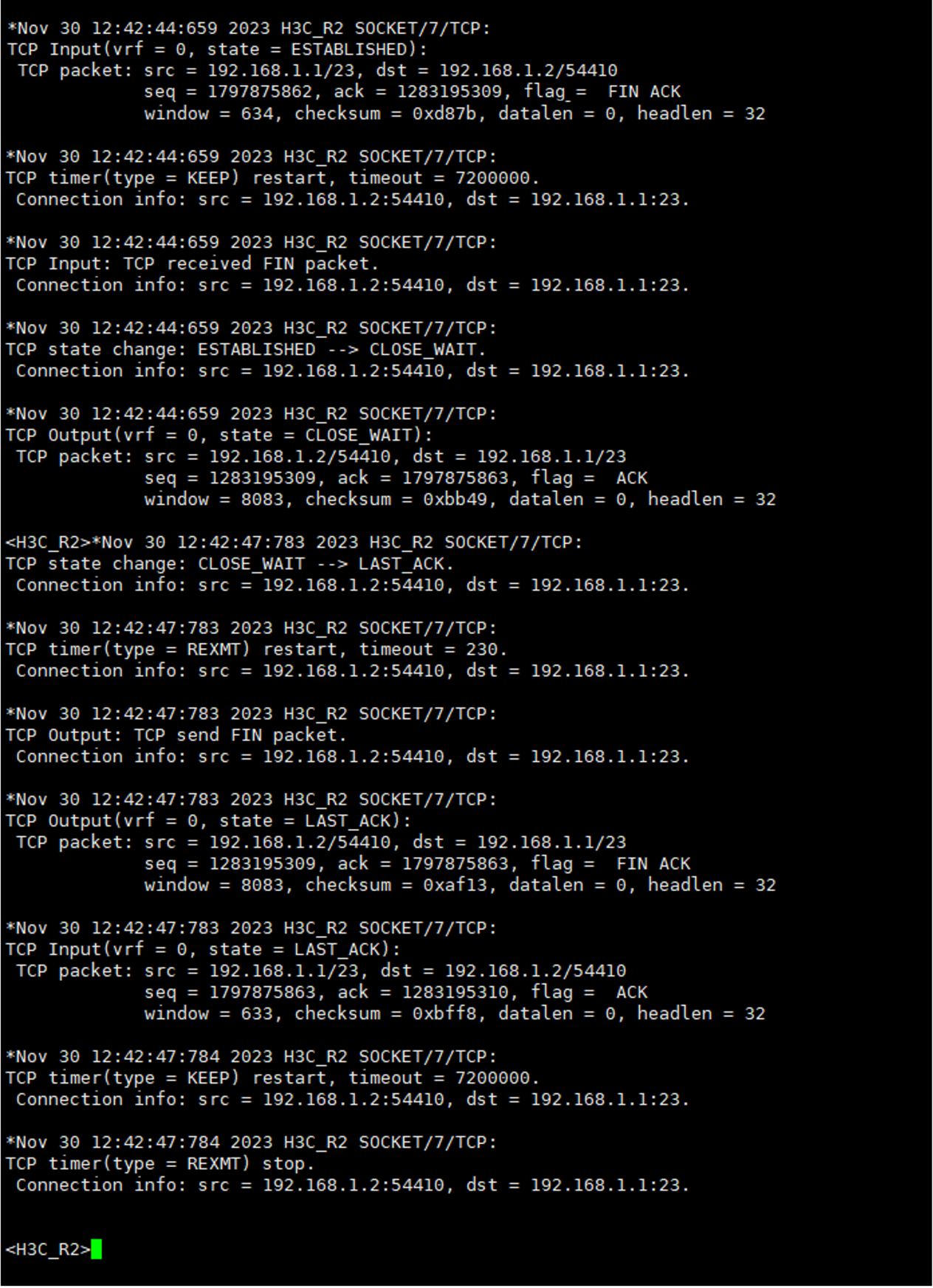
此时由于是从R2远程控制R1，因此关闭连接请求从R1发出。根据tcp packet信息的调试结果，可以得出：

1.首先R2接收到了R1发送的FIN报文，R1等待R2的ACK，其中seq = p =1797875862，flag = FIN ACK，表示接收到了R1发送的关闭tcp连接请求。

2.之后R1向R2发送ACK报文，表示接收到了关闭tcp请求。其中state = CLOSE\_WAIT，ack = p+1 = 1797875863，为FIN报文的seq+1.

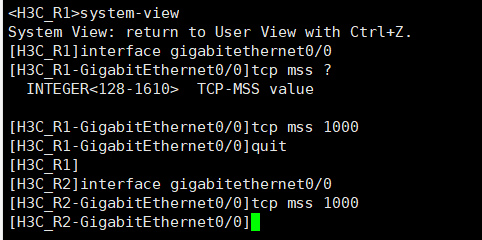
3.同时，R2也向R1发出关闭TCP连接的FIN报文，并等待R1的ACK。其中state = LAST\_ACK，seq = q = 1283195309，flag = FIN ACK。

4.最后，R1向R2发送ACK报文，完成四次握手。其中state = LAST\_ACK， ack = q+1 = 1283195310， flag = ACK。

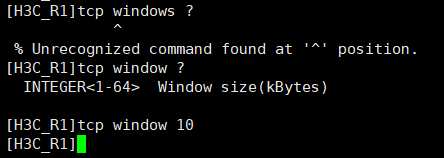


8.配置TCP属性

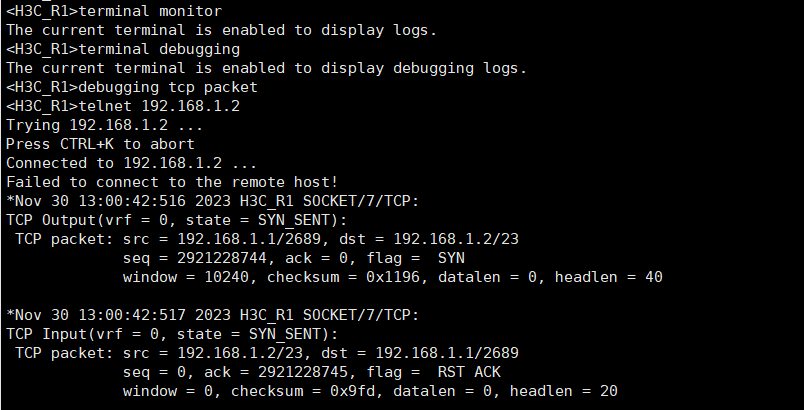
（1）首先对R1和R2进行相同配置，tcp mss 1000表示将对应接口配置接口的TCP最大报文段长度为1000.



（2）之后使用tcp window 10，把R1的TCP连接的收发缓冲区大小设置为10KB。

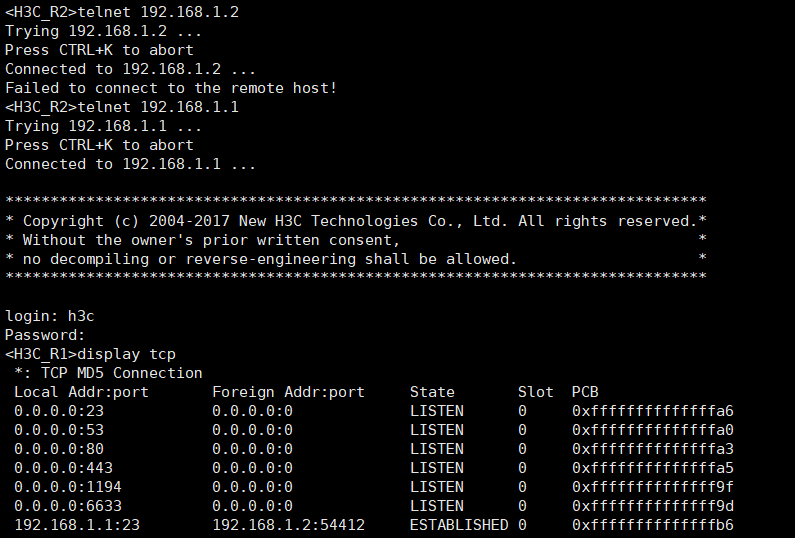


（3）最后使用debugging tcp packet命令调试TCP报文信息验证，window = 10240，即10KB。



9. 查看TCP相关的状态属性

（1）首先使用telnet从R2连接到R1，在R1上使用display tcp来查看tcp相关信息，此时可以看到state为ESTABLISHED的来自R2（192.168.1.2）的连接



（2）之后使用display tcp statistics来显示TCP连接的流量统计信息。可以看到接收包总数为639，发送包总数为599，发起连接次数为3，接受连接次数为12，已建立连接数为12。



五、实验结果及分析

1. 我们第一个实验对应的图6-7所示的拓扑图和132页下面给的节本参数配置命令之间出现了什么错误？

在配置所有端口时，应该使用千兆端口配置命令，即“interface gigabitethernet”，否则会报错。

在 H3C-R3的配置中，第一条命令应为“interface gigabitethernet0/1”，

而非配置 GE0/0端口。

2. 整个实验过程中遇到什么问题（有截图最好），如何解决的？通过该实验有何收获？

在三次握手、四次握手的报文消息验证时，发现给出的tcp event和tcp packet调试信息过多（截图12页），经过老师询问发现，每次使用tcpip传输信息时，两端都会进行三次握手，因此当远程连接输入账号密码时（账号h3c，密码h3c），每传输一个字符都会进行传输确认，保证传输稳定性。