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

姓名 学号

同作者

实验日期 2023 年 11 月 30 日

成 绩

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。

```
<H3C>system-view
System View: return to User View with Ctrl+Z.
[H3C]sysname H3C_R1
[H3C_R1]
```

```
<H3C>system-view
System View: return to User View with Ctrl+Z.
[H3C]sysname H3C_R2
[H3C_R2]
```

```
<H3C>system-view
System View: return to User View with Ctrl+Z.
[H3C]sysname H3C_R3
```

（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.

```
[H3C_R1]interface gigabitethernet0/0
[H3C_R1-GigabitEthernet0/0]ip address 192.168.1.1 24
[H3C_R1-GigabitEthernet0/0]quit
[H3C_R1]ip route-static 192.168.2.0 255.255.255.0 192.168.1.2
[H3C_R1]
```

```
<H3C>system-view
System View: return to User View with Ctrl+Z.
[H3C]sysname H3C_R2
[H3C_R2]interface gigabitethernet0/0
[H3C_R2-GigabitEthernet0/0]ip address 192.168.1.2 24
[H3C_R2-GigabitEthernet0/0]quit
[H3C_R2]interface gigabitethernet0/1
[H3C_R2-GigabitEthernet0/1]ip address 192.168.2.1 24
[H3C_R2-GigabitEthernet0/1]quit
[H3C_R2]
```

```
[H3C_R3]interface gigabitethernet0/1
[H3C_R3-GigabitEthernet0/1]ip address 192.168.2.2 24
[H3C_R3-GigabitEthernet0/1]quit
[H3C_R3]ip route-static 192.168.1.0 255.255.255.0 192.168.2.1
[H3C_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。

```
<H3C_R1>ping ?
  -a               Specify the source IP address
  -c               Specify the number of echo requests
  -f               Specify packets not to be fragmented
  -h               Specify the TTL value
  -i               Specify an outgoing interface
  -m               Specify the interval for sending echo requests
  -n               Numeric output only. No attempt will be made to lookup host
                   addresses for symbolic names
  -p               No more than 8 "pad" hexadecimal characters to fill out the
                   sent packet. For example, -p f2 will fill the sent packet with
                   000000f2 repeatedly
  -q               Display only summary
  -r               Record route. Include the RECORD_ROUTE option in the
                   ECHO_REQUEST packets and display the route
  -s               Specify the payload length
  -t               Specify the wait time for each reply
  -topology        Specify a topology
  -tos             Specify the TOS value
  -v               Display the received ICMP packets other than ECHO-RESPONSE
                   packets
  -vpn-instance    Specify a VPN instance
  STRING<1-253>    IP address or hostname of remote system
  ip               IP information
  ipv6             IPv6 information
```

（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。

```
<H3C_R1>ping 192.168.1.2
Ping 192.168.1.2 (192.168.1.2): 56 data bytes, press CTRL_C to break
56 bytes from 192.168.1.2: icmp_seq=0 ttl=255 time=0.592 ms
56 bytes from 192.168.1.2: icmp_seq=1 ttl=255 time=0.233 ms
56 bytes from 192.168.1.2: icmp_seq=2 ttl=255 time=0.228 ms
56 bytes from 192.168.1.2: icmp_seq=3 ttl=255 time=0.166 ms
56 bytes from 192.168.1.2: icmp_seq=4 ttl=255 time=0.174 ms

--- Ping statistics for 192.168.1.2 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.166/0.279/0.592/0.159 ms
<H3C_R1>%Nov 30 09:44:23:421 2023 H3C_R1 PING/6/PING_STATISTICS: Ping statistics for 19
2.168.1.2: 5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss, round-trip
min/avg/max/std-dev = 0.166/0.279/0.592/0.159 ms.

<H3C_R1>
```

（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。

```
<H3C_R1>ping -c 1 -r 192.168.2.2
Ping 192.168.2.2 (192.168.2.2): 56 data bytes, press CTRL_C to break
56 bytes from 192.168.2.2: icmp_seq=0 ttl=254 time=1.129 ms
RR:     192.168.2.1
        192.168.2.2
        192.168.1.2
        192.168.1.1

--- Ping statistics for 192.168.2.2 ---
1 packet(s) transmitted, 1 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 1.129/1.129/1.129/0.000 ms
<H3C_R1>%Nov 30 09:45:57:510 2023 H3C_R1 PING/6/PING_STATISTICS: Ping statistics for 19
2.168.2.2: 1 packet(s) transmitted, 1 packet(s) received, 0.0% packet loss, round-trip
min/avg/max/std-dev = 1.129/1.129/1.129/0.000 ms.

<H3C_R1>
```

（4）ping -r 原理

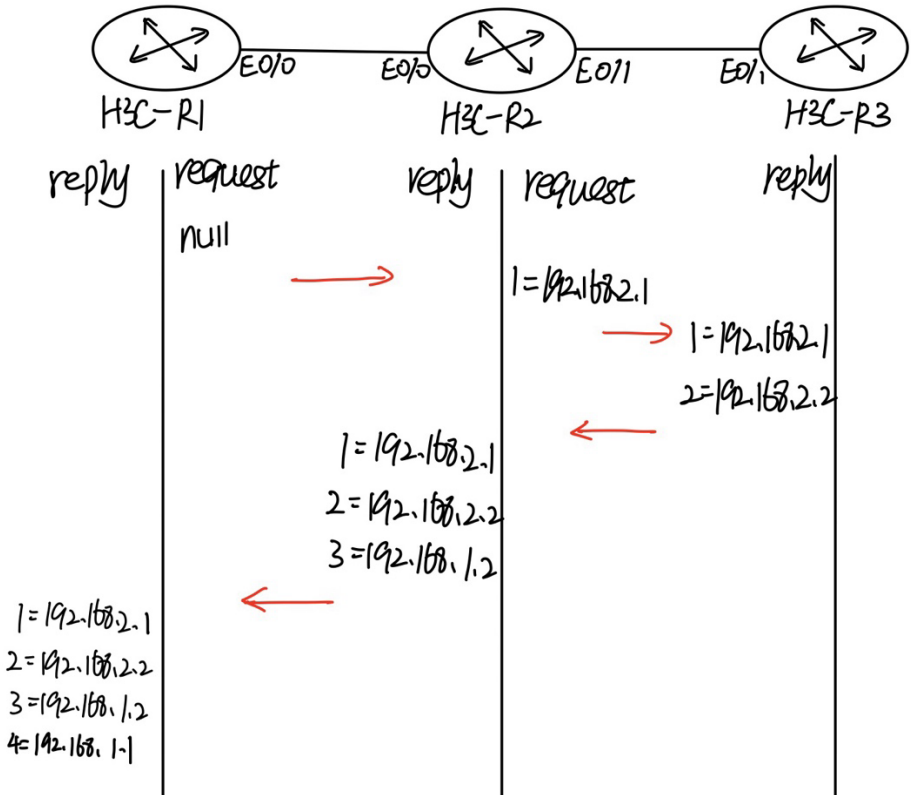
Ping -r 原理图如下，首先 R1 发送 RR 选项为空的 ICMP 回显请求，给目的端设备 R3.

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。

```
<H3C_R1>tracert ?
  -a              Specify the source IP address used by TRACERT
  -f              Specify the TTL value for the first packet
  -m              Specify the maximum TTL value
  -p              Specify the destination UDP port number
  -q              Specify the number of probe packets sent each time
  -t              Set the Type of Service (ToS) value
  -topology       Specify a topology
  -vpn-instance   Specify a VPN instance
  -w              Set the timeout to wait for each reply
  STRING<1-253>   IP address or hostname of the destination device
  ipv6            IPv6 information

<H3C_R1>tracert
```

（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。

```
<H3C_R1>tracert 192.168.2.2
traceroute to 192.168.2.2 (192.168.2.2), 30 hops at most, 52 bytes each packet, press C
TRL_C to break
 1  * * *
 2  * * *
 3  * * *
 4  * *
```

（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

```
[H3C_R2]ip ttl-expires enable
[H3C_R2]
```

```
[H3C_R3]ip unreachables enable
[H3C_R3]
```

（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。

```
<H3C_R1>tracert 192.168.2.2
traceroute to 192.168.2.2 (192.168.2.2), 30 hops at most, 52 bytes each packet, press CTRL_C to bre
ak
 1  192.168.1.2 (192.168.1.2)  0.383 ms  0.203 ms  0.199 ms
 2  192.168.2.2 (192.168.2.2)  0.396 ms  0.784 ms  0.333 ms
<H3C_R1>
```

（5）tracert 原理

Tracert 命令的原理如下图所示。

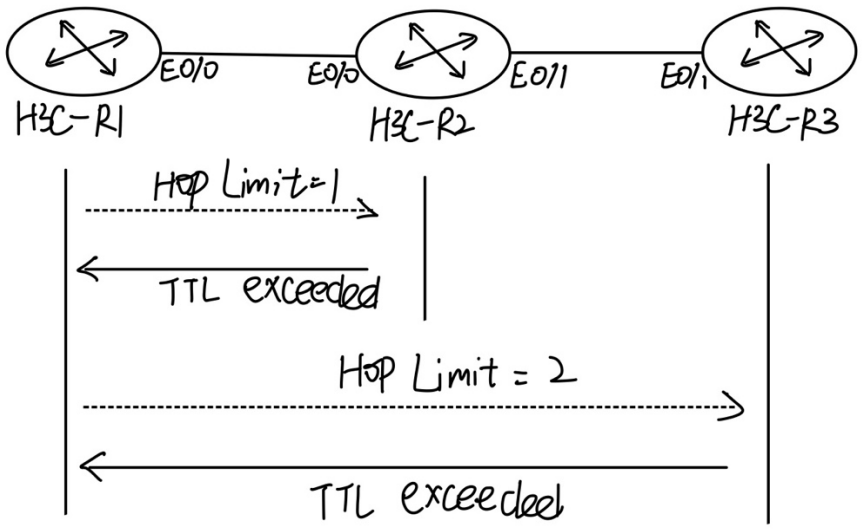
首先 R1 向目的端 R3 发送一个 IP 数据报文，TTL 为 1，报文的 UDP 端口号是目的端的任何一个应用程序都不可能使用的端口号。

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

```
<H3C_R1>terminal ?
  debugging  Enable to display debugging logs on the current terminal
  logging    Display logs on the current terminal
  monitor    Enable to display logs on the current terminal

<H3C_R1>terminal
```

（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

```
<H3C_R1>terminal monitor
The current terminal is enabled to display logs.
<H3C_R1>terminal debugging
The current terminal is enabled to display debugging logs.
<H3C_R1>
```

（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。

```
<H3C_R1>debugging ip icmp
<H3C_R1>ping -c 1 192.168.2.2
```

（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。

```
<H3C_R1>ping -c 1 192.168.2.2
Ping 192.168.2.2 (192.168.2.2): 56 data bytes, press CTRL_C to break
56 bytes from 192.168.2.2: icmp_seq=0 ttl=254 time=0.659 ms

--- Ping statistics for 192.168.2.2 ---
1 packet(s) transmitted, 1 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.659/0.659/0.659/0.000 ms
<H3C_R1>*Nov 30 10:12:18:909 2023 H3C_R1 SOCKET/7/ICMP:
ICMP Output:
 ICMP Packet: src = 192.168.1.1, dst = 192.168.2.2
              type = 8, code = 0 (echo)

*Nov 30 10:12:18:910 2023 H3C_R1 SOCKET/7/ICMP:
ICMP Input:
 ICMP Packet: src = 192.168.2.2, dst = 192.168.1.1
              type = 0, code = 0 (echo-reply)

%Nov 30 10:12:18:911 2023 H3C_R1 PING/6/PING_STATISTICS: Ping statistics for 192.168.2.2: 1 packet(
s) transmitted, 1 packet(s) received, 0.0% packet loss, round-trip min/avg/max/std-dev = 0.659/0.65
9/0.659/0.000 ms.

<H3C_R1>
```

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。

```
<H3C_R1>tracert -q 1 192.168.2.2
traceroute to 192.168.2.2 (192.168.2.2), 30 hops at most, 52 bytes each packet, press CTRL_C to bre
ak
 1  192.168.1.2 (192.168.1.2)  0.525 ms
 2  192.168.2.2 (192.168.2.2)  0.534 ms
<H3C_R1>*Nov 30 10:14:13:011 2023 H3C R1 SOCKET/7/ICMP:
ICMP Input:
 ICMP Packet: src = 192.168.1.2, dst = 192.168.1.1
              type = 11, code = 0 (ttl-exceeded)
 Original IP: src = 192.168.1.1, dst = 192.168.2.2
              proto = 17, first 8 bytes = 819F829A 00200000

*Nov 30 10:14:13:014 2023 H3C R1 SOCKET/7/ICMP:
ICMP Input:
 ICMP Packet: src = 192.168.2.2, dst = 192.168.1.1
              type = 3, code = 3 (port-unreachable)
 Original IP: src = 192.168.1.1, dst = 192.168.2.2
              proto = 17, first 8 bytes = 819F829B 00200000


<H3C_R1>
```

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

```
<H3C_R1>system-view
System View: return to User View with Ctrl+Z.
[H3C_R1]interface gigabitethernet
                                  ^
 % Incomplete command found at '^' position.
[H3C_R1]interface gigabitethernet0/0
[H3C_R1-GigabitEthernet0/0]ip address 192.168.1.1 24
[H3C_R1-GigabitEthernet0/0]quit
[H3C_R1]telnet server enable
[H3C_R1]local-user h3c
New local user added.
[H3C_R1-luser-manage-h3c]password simple h3c
[H3C_R1-luser-manage-h3c]service-type telnet
[H3C_R1-luser-manage-h3c]user-interface vty 0 4
[H3C_R1-line-vty0-4]authentication-mode scheme
[H3C_R1-line-vty0-4]
```

```
<H3C_R2>system-view
System View: return to User View with Ctrl+Z.
[H3C_R2]interface gigabitethernet0/0
[H3C_R2-GigabitEthernet0/0]ip address 192.168.1.2 24
[H3C_R2-GigabitEthernet0/0]quit
[H3C_R2]
```

（2）之后 ping 测试 R1 与 R2 的连通性，可以看到连通性正常。

```
<H3C_R1>ping 192.168.1.2
Ping 192.168.1.2 (192.168.1.2): 56 data bytes, press CTRL C to break
56 bytes from 192.168.1.2: icmp_seq=0 ttl=255 time=0.492 ms
56 bytes from 192.168.1.2: icmp_seq=1 ttl=255 time=0.256 ms
56 bytes from 192.168.1.2: icmp_seq=2 ttl=255 time=0.252 ms
56 bytes from 192.168.1.2: icmp_seq=3 ttl=255 time=0.188 ms
56 bytes from 192.168.1.2: icmp_seq=4 ttl=255 time=0.196 ms

--- Ping statistics for 192.168.1.2 ---
5 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss
round-trip min/avg/max/std-dev = 0.188/0.277/0.492/0.111 ms
<H3C_R1>%Nov 30 10:25:02:327 2023 H3C_R1 PING/6/PING_STATISTICS: Ping statistics for 192.168.1.2: 5
 packet(s) transmitted, 5 packet(s) received, 0.0% packet loss, round-trip min/avg/max/std-dev = 0.
188/0.277/0.492/0.111 ms.

<H3C_R1>
```

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。

```
<H3C_R2>terminal monitor
The current terminal is enabled to display logs.
<H3C_R2>terminal debugging
The current terminal is enabled to display debugging logs.
<H3C_R2>terminal tcp event
                  ^
 % Unrecognized command found at '^' position.
<H3C_R2>debugging tcp event
<H3C_R2>debugging tcp packet
<H3C_R2>
```

（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。

```
<H3C R2>telnet 192.168.1.1
Trying 192.168.1.1 ...
Press CTRL+K to abort
Connected to 192.168.1.1 ...
*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP state change: CLOSED --> SYN_SENT.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 75000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 3000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = SYN_SENT):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195234, ack = 0, flag =  SYN
             window = 64512, checksum = 0xe34d, datalen = 0, headlen = 40

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = SYN_SENT):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875379, ack = 1283195235, flag =  SYN ACK
             window = 4096, checksum = 0xa5f8, datalen = 0, headlen = 40

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP state change: SYN_SENT --> ESTABLISHED.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195235, ack = 1797875380, flag =  ACK
             window = 8083, checksum = 0xc478, datalen = 0, headlen = 32

*Nov 30 12:41:38:232 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875380, ack = 1283195235, flag =  PSH ACK
             window = 634, checksum = 0xaf53, datalen = 18, headlen = 32

*Nov 30 12:41:38:232 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:232 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195235, ack = 1797875398, flag =  ACK
             window = 8081, checksum = 0xc45c, datalen = 0, headlen = 32

*Nov 30 12:41:38:422 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.
```

```
*Nov 30 12:41:38:422 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195235, ack = 1797875398, flag =  PSH ACK
             window = 8083, checksum = 0x844f, datalen = 18, headlen = 32

*Nov 30 12:41:38:422 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875398, ack = 1283195253, flag =  ACK
             window = 631, checksum = 0xdfe8, datalen = 0, headlen = 32

*Nov 30 12:41:38:422 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:422 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195253, ack = 1797875398, flag =  PSH ACK
             window = 8083, checksum = 0xc642, datalen = 12, headlen = 32

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875398, ack = 1283195253, flag =  PSH ACK
             window = 634, checksum = 0x80f6, datalen = 15, headlen = 32

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195265, ack = 1797875413, flag =  ACK
             window = 8081, checksum = 0xc2b3, datalen = 0, headlen = 32

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875413, ack = 1283195265, flag =  ACK
             window = 632, checksum = 0xdfcc, datalen = 0, headlen = 32

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
```

```
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195265, ack = 1797875413, flag =  PSH ACK
             window = 8083, checksum = 0x942, datalen = 25, headlen = 32


******************************************************************************
* Copyright (c) 2004-2017 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

login: *Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875413, ack = 1283195290, flag =  ACK
             window = 630, checksum = 0xdfb4, datalen = 0, headlen = 32

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:423 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:453 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875413, ack = 1283195290, flag =  PSH ACK
             window = 634, checksum = 0xddf0, datalen = 407, headlen = 32

*Nov 30 12:41:38:453 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:453 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195290, ack = 1797875820, flag =  ACK
             window = 8032, checksum = 0xc0f8, datalen = 0, headlen = 32

*Nov 30 12:41:38:453 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:453 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195290, ack = 1797875820, flag =  PSH ACK
             window = 8083, checksum = 0xbfbc, datalen = 3, headlen = 32

*Nov 30 12:41:38:453 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875820, ack = 1283195293, flag =  ACK
             window = 633, checksum = 0xdddc, datalen = 0, headlen = 32

*Nov 30 12:41:38:453 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:453 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:457 2023 H3C_R2 SOCKET/7/TCP:
```

```
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875820, ack = 1283195293, flag =  PSH ACK
             window = 634, checksum = 0x7bb5, datalen = 7, headlen = 32

*Nov 30 12:41:38:457 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:457 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195293, ack = 1797875827, flag =  ACK
             window = 8082, checksum = 0xc0b4, datalen = 0, headlen = 32

h*Nov 30 12:42:08:115 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:08:115 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195293, ack = 1797875827, flag =  PSH ACK
             window = 8083, checksum = 0xe4ce, datalen = 1, headlen = 32

*Nov 30 12:42:08:115 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875827, ack = 1283195294, flag =  ACK
             window = 633, checksum = 0xf615, datalen = 0, headlen = 32

*Nov 30 12:42:08:116 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:08:116 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:08:116 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875827, ack = 1283195294, flag =  PSH ACK
             window = 634, checksum = 0x8e0b, datalen = 1, headlen = 32

*Nov 30 12:42:08:116 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:08:116 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195294, ack = 1797875828, flag =  ACK
             window = 8083, checksum = 0xd8fa, datalen = 0, headlen = 32

3*Nov 30 12:42:10:904 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:10:904 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195294, ack = 1797875828, flag =  PSH ACK
             window = 8083, checksum = 0x9b0c, datalen = 1, headlen = 32

*Nov 30 12:42:10:904 2023 H3C_R2 SOCKET/7/TCP:
```

```
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875828, ack = 1283195295, flag =  ACK
             window = 633, checksum = 0xe049, datalen = 0, headlen = 32

*Nov 30 12:42:10:904 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:10:904 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:10:905 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875828, ack = 1283195295, flag =  PSH ACK
             window = 634, checksum = 0xad3f, datalen = 1, headlen = 32

*Nov 30 12:42:10:905 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:10:905 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195295, ack = 1797875829, flag =  ACK
             window = 8083, checksum = 0xc32e, datalen = 0, headlen = 32

c*Nov 30 12:42:13:217 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:13:217 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195295, ack = 1797875829, flag =  PSH ACK
             window = 8083, checksum = 0x571c, datalen = 1, headlen = 32

*Nov 30 12:42:13:217 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875829, ack = 1283195296, flag =  ACK
             window = 633, checksum = 0xce35, datalen = 0, headlen = 32

*Nov 30 12:42:13:217 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:13:217 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:13:217 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875829, ack = 1283195296, flag =  PSH ACK
             window = 634, checksum = 0x6b2b, datalen = 1, headlen = 32

*Nov 30 12:42:13:218 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:13:218 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
```

```
            seq = 1283195296, ack = 1797875830, flag =  ACK
            window = 8083, checksum = 0xb11a, datalen = 0, headlen = 32


*Nov 30 12:42:15:560 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:15:560 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
            seq = 1283195296, ack = 1797875830, flag =  PSH ACK
            window = 8083, checksum = 0x9adf, datalen = 2, headlen = 32

*Nov 30 12:42:15:560 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
            seq = 1797875830, ack = 1283195298, flag =  ACK
            window = 633, checksum = 0xbbe4, datalen = 0, headlen = 32

*Nov 30 12:42:15:560 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:15:560 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:15:564 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
            seq = 1797875830, ack = 1283195298, flag =  PSH ACK
            window = 634, checksum = 0xa1c9, datalen = 4, headlen = 32

*Nov 30 12:42:15:564 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:15:564 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
            seq = 1283195298, ack = 1797875834, flag =  ACK
            window = 8083, checksum = 0x9ebf, datalen = 0, headlen = 32

*Nov 30 12:42:15:672 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
            seq = 1797875834, ack = 1283195298, flag =  PSH ACK
            window = 634, checksum = 0xd392, datalen = 10, headlen = 32

*Nov 30 12:42:15:672 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:15:672 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
            seq = 1283195298, ack = 1797875844, flag =  ACK
            window = 8082, checksum = 0x9ddf, datalen = 0, headlen = 32

Password: *Nov 30 12:42:19:812 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:19:812 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
```

```
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195298, ack = 1797875844, flag =  PSH ACK
             window = 8083, checksum = 0x25a8, datalen = 1, headlen = 32

*Nov 30 12:42:19:812 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875844, ack = 1283195299, flag =  ACK
             window = 633, checksum = 0x9a9d, datalen = 0, headlen = 32

*Nov 30 12:42:19:812 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:19:812 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:22:586 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:22:586 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195299, ack = 1797875844, flag =  PSH ACK
             window = 8083, checksum = 0x3fa4, datalen = 1, headlen = 32

*Nov 30 12:42:22:586 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875844, ack = 1283195300, flag =  ACK
             window = 633, checksum = 0x84f0, datalen = 0, headlen = 32

*Nov 30 12:42:22:586 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:22:586 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:24:387 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:24:387 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195300, ack = 1797875844, flag =  PSH ACK
             window = 8083, checksum = 0xfdc3, datalen = 1, headlen = 32

*Nov 30 12:42:24:387 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875844, ack = 1283195301, flag =  ACK
             window = 633, checksum = 0x76dd, datalen = 0, headlen = 32

*Nov 30 12:42:24:387 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:24:387 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.
```

```
*Nov 30 12:42:26:247 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:26:247 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195301, ack = 1797875844, flag =  PSH ACK
             window = 8083, checksum = 0x456b, datalen = 2, headlen = 32

*Nov 30 12:42:26:247 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875844, ack = 1283195303, flag =  ACK
             window = 633, checksum = 0x6853, datalen = 0, headlen = 32

*Nov 30 12:42:26:247 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:26:247 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:26:248 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875844, ack = 1283195303, flag =  PSH ACK
             window = 634, checksum = 0x5b3e, datalen = 2, headlen = 32

*Nov 30 12:42:26:248 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:26:248 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195303, ack = 1797875846, flag =  ACK
             window = 8083, checksum = 0x4b37, datalen = 0, headlen = 32

*Nov 30 12:42:26:382 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875846, ack = 1283195303, flag =  PSH ACK
             window = 634, checksum = 0x679d, datalen = 8, headlen = 32

*Nov 30 12:42:26:382 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:26:382 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195303, ack = 1797875854, flag =  ACK
             window = 8082, checksum = 0x4a23, datalen = 0, headlen = 32

<H3C_R1>q*Nov 30 12:42:34:061 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:34:061 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195303, ack = 1797875854, flag =  PSH ACK
             window = 8083, checksum = 0xbb18, datalen = 1, headlen = 32
```

```
*Nov 30 12:42:34:061 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875854, ack = 1283195304, flag =  ACK
             window = 633, checksum = 0x2b3c, datalen = 0, headlen = 32

*Nov 30 12:42:34:061 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:34:061 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:34:062 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875854, ack = 1283195304, flag =  PSH ACK
             window = 634, checksum = 0xba31, datalen = 1, headlen = 32

*Nov 30 12:42:34:062 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:34:062 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195304, ack = 1797875855, flag =  ACK
             window = 8083, checksum = 0xe21, datalen = 0, headlen = 32

u*Nov 30 12:42:36:552 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:36:552 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195304, ack = 1797875855, flag =  PSH ACK
             window = 8083, checksum = 0x8f5c, datalen = 1, headlen = 32

*Nov 30 12:42:36:552 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875855, ack = 1283195305, flag =  ACK
             window = 633, checksum = 0x17c4, datalen = 0, headlen = 32

*Nov 30 12:42:36:552 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:36:552 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:36:553 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875855, ack = 1283195305, flag =  PSH ACK
             window = 634, checksum = 0xa2b9, datalen = 1, headlen = 32

*Nov 30 12:42:36:553 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:36:553 2023 H3C_R2 SOCKET/7/TCP:
```

```
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195305, ack = 1797875856, flag =  ACK
             window = 8083, checksum = 0xfaa8, datalen = 0, headlen = 32

i*Nov 30 12:42:38:962 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:38:962 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195305, ack = 1797875856, flag =  PSH ACK
             window = 8083, checksum = 0x8835, datalen = 1, headlen = 32

*Nov 30 12:42:38:962 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875856, ack = 1283195306, flag =  ACK
             window = 633, checksum = 0x4ee, datalen = 0, headlen = 32

*Nov 30 12:42:38:962 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:38:962 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:38:963 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875856, ack = 1283195306, flag =  PSH ACK
             window = 634, checksum = 0x9be3, datalen = 1, headlen = 32

*Nov 30 12:42:38:963 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:38:963 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195306, ack = 1797875857, flag =  ACK
             window = 8083, checksum = 0xe7d2, datalen = 0, headlen = 32

t*Nov 30 12:42:41:395 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:41:395 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195306, ack = 1797875857, flag =  PSH ACK
             window = 8083, checksum = 0x6a48, datalen = 1, headlen = 32

*Nov 30 12:42:41:395 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875857, ack = 1283195307, flag =  ACK
             window = 633, checksum = 0xf1e9, datalen = 0, headlen = 32

*Nov 30 12:42:41:395 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:41:395 2023 H3C_R2 SOCKET/7/TCP:
```

```
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:41:396 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875857, ack = 1283195307, flag =  PSH ACK
             window = 634, checksum = 0x7ddf, datalen = 1, headlen = 32

*Nov 30 12:42:41:396 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:41:396 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195307, ack = 1797875858, flag =  ACK
             window = 8083, checksum = 0xd4ce, datalen = 0, headlen = 32



The connection was closed by the remote host!
*Nov 30 12:42:44:629 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:629 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195307, ack = 1797875858, flag =  PSH ACK
             window = 8083, checksum = 0xbb18, datalen = 2, headlen = 32

*Nov 30 12:42:44:629 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875858, ack = 1283195309, flag =  ACK
             window = 633, checksum = 0xd8a2, datalen = 0, headlen = 32

*Nov 30 12:42:44:629 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:629 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:633 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875858, ack = 1283195309, flag =  PSH ACK
             window = 634, checksum = 0xbe87, datalen = 4, headlen = 32

*Nov 30 12:42:44:633 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:633 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195309, ack = 1797875862, flag =  ACK
             window = 8083, checksum = 0xbb7c, datalen = 0, headlen = 32

*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875862, ack = 1283195309, flag =  FIN ACK
```

```
            window = 634, checksum = 0xd87b, datalen = 0, headlen = 32

*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP Input: TCP received FIN packet.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP state change: ESTABLISHED --> CLOSE_WAIT.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = CLOSE_WAIT):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195309, ack = 1797875863, flag =  ACK
             window = 8083, checksum = 0xbb49, datalen = 0, headlen = 32

<H3C_R2>*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP state change: CLOSE_WAIT --> LAST_ACK.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP Output: TCP send FIN packet.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = LAST_ACK):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195309, ack = 1797875863, flag =  FIN ACK
             window = 8083, checksum = 0xaf13, datalen = 0, headlen = 32

*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = LAST_ACK):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875863, ack = 1283195310, flag =  ACK
             window = 633, checksum = 0xbff8, datalen = 0, headlen = 32

*Nov 30 12:42:47:784 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:47:784 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.


<H3C_R2>
```

（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。

```
<H3C_R2>telnet 192.168.1.1
Trying 192.168.1.1 ...
Press CTRL+K to abort
Connected to 192.168.1.1 ...
*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP state change: CLOSED --> SYN_SENT.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 75000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 3000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = SYN_SENT):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195234, ack = 0, flag =  SYN
             window = 64512, checksum = 0xe34d, datalen = 0, headlen = 40

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP Input vrf = 0, state = SYN_SENT):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875379, ack = 1283195235, flag =  SYN ACK
             window = 4096, checksum = 0xa5f8, datalen = 0, headlen = 40

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP state change: SYN_SENT --> ESTABLISHED.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:41:38:226 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195235, ack = 1797875380, flag =  ACK
             window = 8083, checksum = 0xc478, datalen = 0, headlen = 32
```

（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。

```
*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = ESTABLISHED):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875862, ack = 1283195309, flag =  FIN ACK
             window = 634, checksum = 0xd87b, datalen = 0, headlen = 32

*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP Input: TCP received FIN packet.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP state change: ESTABLISHED --> CLOSE_WAIT.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:44:659 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = CLOSE_WAIT):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195309, ack = 1797875863, flag =  ACK
             window = 8083, checksum = 0xbb49, datalen = 0, headlen = 32

<H3C_R2>*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP state change: CLOSE_WAIT --> LAST_ACK.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) restart, timeout = 230.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP Output: TCP send FIN packet.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP Output(vrf = 0, state = LAST_ACK):
 TCP packet: src = 192.168.1.2/54410, dst = 192.168.1.1/23
             seq = 1283195309, ack = 1797875863, flag =  FIN ACK
             window = 8083, checksum = 0xaf13, datalen = 0, headlen = 32

*Nov 30 12:42:47:783 2023 H3C_R2 SOCKET/7/TCP:
TCP Input(vrf = 0, state = LAST_ACK):
 TCP packet: src = 192.168.1.1/23, dst = 192.168.1.2/54410
             seq = 1797875863, ack = 1283195310, flag =  ACK
             window = 633, checksum = 0xbff8, datalen = 0, headlen = 32

*Nov 30 12:42:47:784 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = KEEP) restart, timeout = 7200000.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.

*Nov 30 12:42:47:784 2023 H3C_R2 SOCKET/7/TCP:
TCP timer(type = REXMT) stop.
 Connection info: src = 192.168.1.2:54410, dst = 192.168.1.1:23.


<H3C_R2>
```

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.

```
<H3C_R1>system-view
System View: return to User View with Ctrl+Z.
[H3C_R1]interface gigabitethernet0/0
[H3C_R1-GigabitEthernet0/0]tcp mss ?
  INTEGER<128-1610>  TCP-MSS value

[H3C_R1-GigabitEthernet0/0]tcp mss 1000
[H3C_R1-GigabitEthernet0/0]quit
[H3C_R1]
[H3C_R2]interface gigabitethernet0/0
[H3C_R2-GigabitEthernet0/0]tcp mss 1000
[H3C_R2-GigabitEthernet0/0]
```

（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。

```
[H3C_R1]tcp windows ?
            ^
 % Unrecognized command found at '^' position.
[H3C_R1]tcp window ?
  INTEGER<1-64>  Window size(kBytes)

[H3C_R1]tcp window 10
[H3C_R1]
```

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

```
<H3C_R1>terminal monitor
The current terminal is enabled to display logs.
<H3C_R1>terminal debugging
The current terminal is enabled to display debugging logs.
<H3C_R1>debugging tcp packet
<H3C_R1>telnet 192.168.1.2
Trying 192.168.1.2 ...
Press CTRL+K to abort
Connected to 192.168.1.2 ...
Failed to connect to the remote host!
*Nov 30 13:00:42:516 2023 H3C_R1 SOCKET/7/TCP:
TCP Output(vrf = 0, state = SYN_SENT):
 TCP packet: src = 192.168.1.1/2689, dst = 192.168.1.2/23
             seq = 2921228744, ack = 0, flag =  SYN
             window = 10240, checksum = 0x1196, datalen = 0, headlen = 40

*Nov 30 13:00:42:517 2023 H3C_R1 SOCKET/7/TCP:
TCP Input(vrf = 0, state = SYN_SENT):
 TCP packet: src = 192.168.1.2/23, dst = 192.168.1.1/2689
             seq = 0, ack = 2921228745, flag =  RST ACK
             window = 0, checksum = 0x9fd, datalen = 0, headlen = 20
```

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

```
<H3C_R2>telnet 192.168.1.2
Trying 192.168.1.2 ...
Press CTRL+K to abort
Connected to 192.168.1.2 ...
Failed to connect to the remote host!
<H3C_R2>telnet 192.168.1.1
Trying 192.168.1.1 ...
Press CTRL+K to abort
Connected to 192.168.1.1 ...

******************************************************************************
* Copyright (c) 2004-2017 New H3C Technologies Co., Ltd. All rights reserved.*
* Without the owner's prior written consent,                                 *
* no decompiling or reverse-engineering shall be allowed.                    *
******************************************************************************

login: h3c
Password:
<H3C_R1>display tcp
 *: TCP MD5 Connection
 Local Addr:port       Foreign Addr:port     State       Slot  PCB
 0.0.0.0:23            0.0.0.0:0             LISTEN      0     0xffffffffffffa6
 0.0.0.0:53            0.0.0.0:0             LISTEN      0     0xffffffffffffa0
 0.0.0.0:80            0.0.0.0:0             LISTEN      0     0xffffffffffffa3
 0.0.0.0:443           0.0.0.0:0             LISTEN      0     0xffffffffffffa5
 0.0.0.0:1194          0.0.0.0:0             LISTEN      0     0xffffffffffffff9f
 0.0.0.0:6633          0.0.0.0:0             LISTEN      0     0xffffffffffffff9d
 192.168.1.1:23        192.168.1.2:54412     ESTABLISHED 0     0xffffffffffffb6
```

（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。

```
<H3C_R1>display tcp statistic
Received packets:
    Total: 639
    packets in sequence: 288 (973 bytes)
    window probe packets: 0, window update packets: 12
    checksum error: 0, offset error: 0, short error: 0
    packets dropped for lack of memory: 0
    packets dropped due to PAWS: 0
    duplicate packets: 0 (0 bytes), partially duplicate packets: 0 (0 bytes)
    out-of-order packets: 0 (0 bytes)
    packets with data after window: 0 (0 bytes)
    packets after close: 0
    ACK packets: 308 (6768 bytes)
    duplicate ACK packets: 27, ACK packets for unsent data: 0

Sent packets:
    Total: 599
    urgent packets: 0
    control packets: 14
    window probe packets: 0, window update packets: 0
    data packets: 285 (6745 bytes), data packets retransmitted: 0 (0 bytes)
    ACK-only packets: 300 (0 delayed)
    unnecessary packet retransmissions: 0

Syncache/syncookie related statistics:
    entries added to syncache: 12
    syncache entries retransmitted: 0
    duplicate SYN packets: 0
    reply failures: 0
    successfully build new socket: 12
    bucket overflows: 0
    zone failures: 0
    syncache entries removed due to RST: 0
    syncache entries removed due to timed out: 0
    ACK checked by syncache or syncookie failures: 0
    syncache entries aborted: 0
    syncache entries removed due to bad ACK: 0
    syncache entries removed due to ICMP unreachable: 0
    SYN cookies sent: 0
    SYN cookies received: 0

SACK related statistics:
    SACK recoveries: 0
    SACK retransmitted segments: 0 (0 bytes)
    SACK blocks (options) received: 0
    SACK blocks (options) sent: 0
    SACK scoreboard overflows: 0

Other statistics:
    retransmitted timeout: 0, connections dropped in retransmitted timeout: 0
    persist timeout: 0
    keepalive timeout: 28, keepalive probe: 28
    keepalive timeout, so connections disconnected: 0
    fin wait 2 timeout, so connections disconnected: 0
    initiated connections: 3, accepted connections: 12, established connections: 12
    closed connections: 14 (dropped: 0, initiated dropped: 3)
    bad connection attempt: 0
    ignored RSTs in the window: 0
    listen queue overflows: 0
    RTT updates: 308(attempt segment: 299)
    correct ACK header predictions: 0
    correct data packet header predictions: 166
    resends due to MTU discovery: 0
    packets dropped with MD5 authentication: 0
    packets permitted with MD5 authentication: 0
```

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。