实验报告

姓名：王苑铮 学号：2015K8009922002

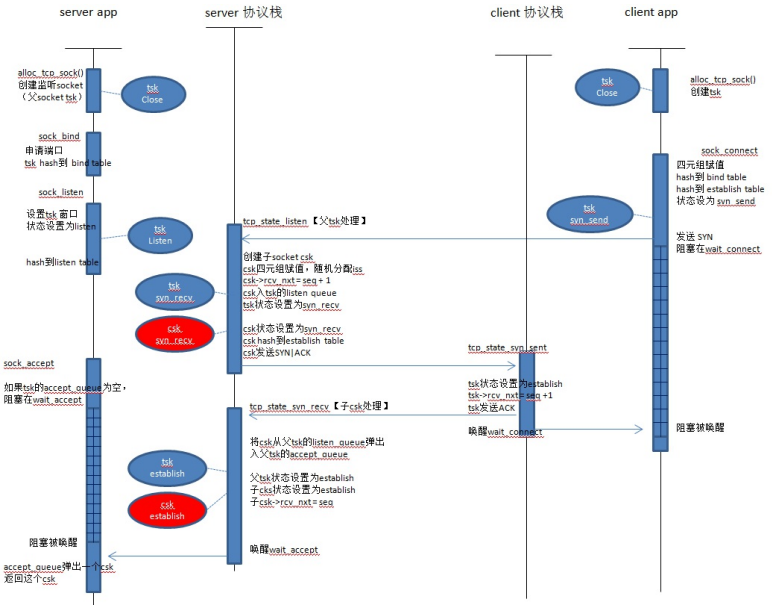
1.实验题目： 网络传输实验一

2.实验内容：

- 实现tcp状态转换、实现socket api

3.实验过程

仿照uml时序图的方式画了示意图(不是标准的uml时序图)
正常状态下的连接建立过程：



断开连接时，因为不涉及sleep和wakeup，时序图比较简单，就不画了

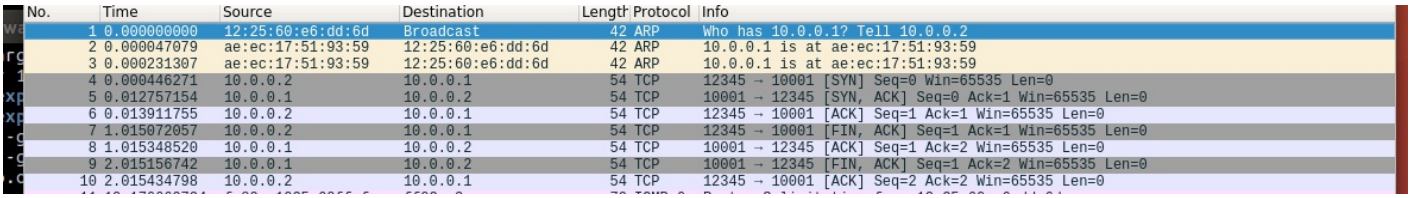
备注：

- 刚申请的tsk是closed状态，sock_listen()把tsk设置为listen状态，以及close()会依据当前状态不同，设置到不同的关闭状态。只有这些状态转移是由socket api函数完成的，除此之外所有状态转移都是由协议栈（tcp_processing线程）完成的

- sock_accept()如果accept_queue不空(意味着之前有握手成功但还没有使用的子csk)，则不进入阻塞状态，直接从accept_queue弹出一个csk返回

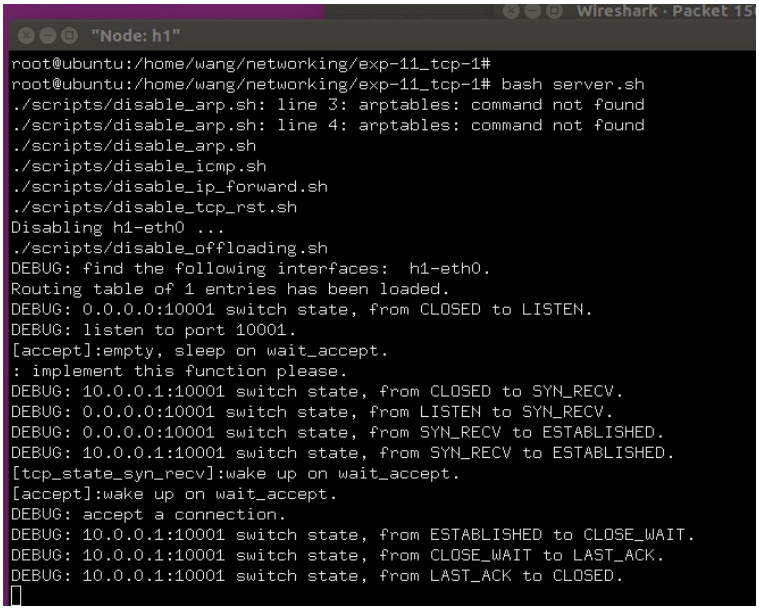
4.实验结果

抓包结果：



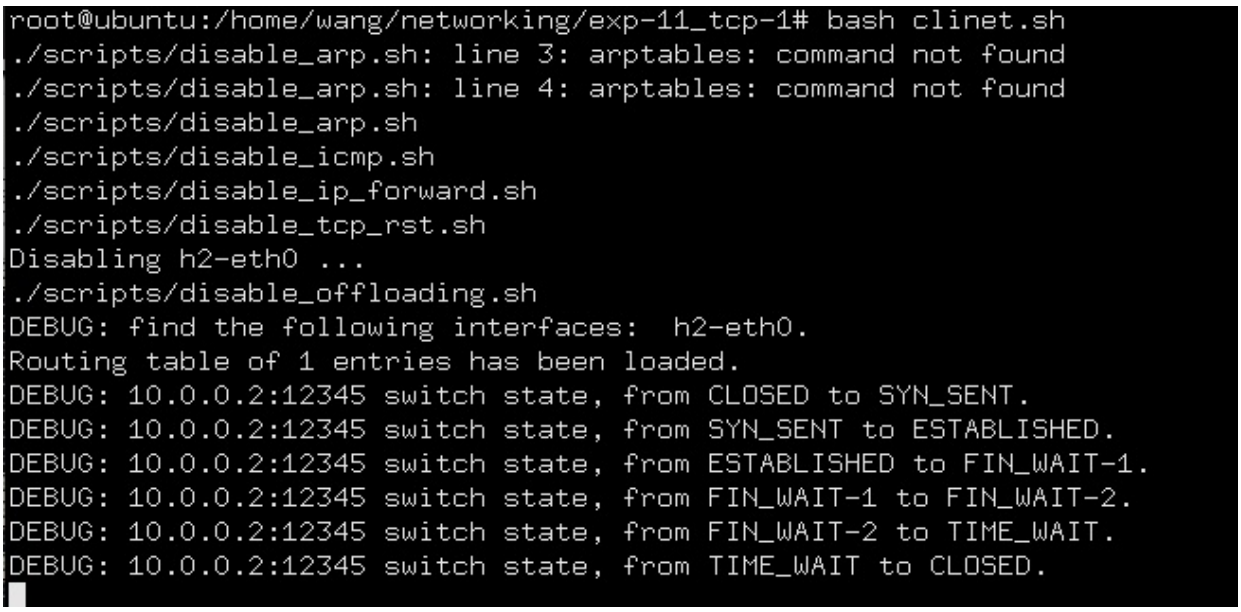
No.	Time	Source	Destination	Length	Protocol	Info
1	0.000000000	12:25:60:e6:dd:6d	Broadcast	42	ARP	Who has 10.0.0.1? Tell 10.0.0.2
2	0.000047079	ae:ec:17:51:93:59	12:25:60:e6:dd:6d	42	ARP	10.0.0.1 is at ae:ec:17:51:93:59
3	0.000231307	ae:ec:17:51:93:59	12:25:60:e6:dd:6d	42	ARP	10.0.0.1 is at ae:ec:17:51:93:59
4	0.000446271	10.0.0.2	10.0.0.1	54	TCP	12345 → 10001 [SYN] Seq=0 Win=65535 Len=0
5	0.012757154	10.0.0.1	10.0.0.2	54	TCP	10001 → 12345 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0
6	0.013911755	10.0.0.2	10.0.0.1	54	TCP	12345 → 10001 [ACK] Seq=1 Ack=1 Win=65535 Len=0
7	1.015072057	10.0.0.2	10.0.0.1	54	TCP	12345 → 10001 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
8	1.015348520	10.0.0.1	10.0.0.2	54	TCP	10001 → 12345 [ACK] Seq=1 Ack=2 Win=65535 Len=0
9	2.015156742	10.0.0.1	10.0.0.2	54	TCP	10001 → 12345 [FIN, ACK] Seq=1 Ack=2 Win=65535 Len=0
10	2.015434798	10.0.0.2	10.0.0.1	54	TCP	12345 → 10001 [ACK] Seq=2 Ack=2 Win=65535 Len=0

h1状态：



```
"Node: h1"
root@ubuntu:/home/wang/networking/exp-11_tcp-1#
root@ubuntu:/home/wang/networking/exp-11_tcp-1# bash server.sh
./scripts/disable_arp.sh: line 3: arptables: command not found
./scripts/disable_arp.sh: line 4: arptables: command not found
./scripts/disable_arp.sh
./scripts/disable_icmp.sh
./scripts/disable_ip_forward.sh
./scripts/disable_tcp_rst.sh
Disabling h1-eth0 ...
./scripts/disable_offloading.sh
DEBUG: find the following interfaces:  h1-eth0.
Routing table of 1 entries has been loaded.
DEBUG: 0.0.0.0:10001 switch state, from CLOSED to LISTEN.
DEBUG: listen to port 10001.
[accept]:empty, sleep on wait_accept.
: implement this function please.
DEBUG: 10.0.0.1:10001 switch state, from CLOSED to SYN_RECV.
DEBUG: 0.0.0.0:10001 switch state, from LISTEN to SYN_RECV.
DEBUG: 0.0.0.0:10001 switch state, from SYN_RECV to ESTABLISHED.
DEBUG: 10.0.0.1:10001 switch state, from SYN_RECV to ESTABLISHED.
[tcp_state_syn_recv]:wake up on wait_accept.
[accept]:wake up on wait_accept.
DEBUG: accept a connection.
DEBUG: 10.0.0.1:10001 switch state, from ESTABLISHED to CLOSE_WAIT.
DEBUG: 10.0.0.1:10001 switch state, from CLOSE_WAIT to LAST_ACK.
DEBUG: 10.0.0.1:10001 switch state, from LAST_ACK to CLOSED.
```

h2状态：



```
root@ubuntu:/home/wang/networking/exp-11_tcp-1# bash clinet.sh
./scripts/disable_arp.sh: line 3: arptables: command not found
./scripts/disable_arp.sh: line 4: arptables: command not found
./scripts/disable_arp.sh
./scripts/disable_icmp.sh
./scripts/disable_ip_forward.sh
./scripts/disable_tcp_rst.sh
Disabling h2-eth0 ...
./scripts/disable_offloading.sh
DEBUG: find the following interfaces:  h2-eth0.
Routing table of 1 entries has been loaded.
DEBUG: 10.0.0.2:12345 switch state, from CLOSED to SYN_SENT.
DEBUG: 10.0.0.2:12345 switch state, from SYN_SENT to ESTABLISHED.
DEBUG: 10.0.0.2:12345 switch state, from ESTABLISHED to FIN_WAIT-1.
DEBUG: 10.0.0.2:12345 switch state, from FIN_WAIT-1 to FIN_WAIT-2.
DEBUG: 10.0.0.2:12345 switch state, from FIN_WAIT-2 to TIME_WAIT.
DEBUG: 10.0.0.2:12345 switch state, from TIME_WAIT to CLOSED.
```

5.结果分析

- 看抓包结果可知，三次握手与四次挥手都正确
- 看服务器和客户端的状态转移打印结果，状态转移正确
- 服务器、客户端都分别用自己的程序、reference互相验证，得到的都是上述结果

  结论： tcp连接的建立与断开正确