

网络编程第三次作业

吴桐

57119108

2022 年 9 月 27 日

1 基于 UDP 的 ping 实现

ping 程序的逻辑非常简单明了：发送 ICMP 报文-接收 ICMP 报文-统计信息。

在发送 ICMP 报文之前，程序需要将用户输入的信息进行预处理，例如域名解析。在得到目的 IP 地址后，我们需要构造报文，其中主要包括报头设置和校验和计算。之后通过原始套接字发送 ICMP 报文。我们一次性发出若干 ICMP 报文后，统一接收 ICMP 回复。在接收数据报时，我们需要判断数据报是否完整，以及接收到的数据报是否为之前发送的 ICMP 报文的回复。在统计阶段，我们统计发送的数据报数量以及收到的回复报文数量，进而得到丢包率。

在运行程序的过程中，可能会出现段错误。借助 gdb 和输出调试等方法，我们发现代码在域名解析部分出现了错误。将代码修改如下即可正常运行。

```
if( (inaddr=inet_addr(argv[1]))==INADDR_NONE)
{
    if((host=gethostbyname(argv[1]))==NULL)
    {
        perror("gethostbyname error");
        exit(1);
    }
    memcpy( (char *)&dest_addr.sin_addr,host->h_addr,host->h_length);
}
else
{
    memcpy( (char *)&dest_addr.sin_addr, &inaddr, sizeof(inaddr));
}
```

我们在这里需要使用原始套接字，然而这种套接字只能由 root 产生。我利用容器创建了一个很简单的内网环境：主机一（10.9.0.5）和主机二（10.9.0.11），并获得两主机的 root 权限终端。这样我们就可以非常方便地直接输入指令运行程序。如图 1 和图 2 所示，我们可以通过 IP 地址或者域名 ping 其他主机。

```
root@13cfc4e25acc:/volumes# uping 10.9.0.11
PING 10.9.0.11(10.9.0.11): 56 bytes data in ICMP packets.
64 byte from 10.9.0.11: icmp_seq=1 ttl=64 rtt=3014.000 ms
64 byte from 10.9.0.11: icmp_seq=2 ttl=64 rtt=2006.000 ms
64 byte from 10.9.0.11: icmp_seq=3 ttl=64 rtt=1002.000 ms

-----PING statistics-----
3 packets transmitted, 3 received , %0 lost
```

图 1: ping 同网段 IP 地址

```
root@13cfc4e25acc:/volumes# uping www.baidu.com
PING www.baidu.com(180.101.49.11): 56 bytes data in ICMP packets.
64 byte from 180.101.49.11: icmp_seq=1 ttl=48 rtt=3012.000 ms
64 byte from 180.101.49.11: icmp_seq=2 ttl=48 rtt=2011.000 ms
64 byte from 180.101.49.11: icmp_seq=3 ttl=48 rtt=1010.000 ms

-----PING statistics-----
3 packets transmitted, 3 received , %0 lost
```

图 2: ping 域名