Problem 1.
Three trials with vpn(ms)
605
550
600

Three trials without vpn(ms)
305
294
297

We can see clear that this method introduce more overhead. Because we have to reach vpn server first before reaching real server.

For data genereation app:
receiver：
With vpn：
Completion time 35.999000,reliable bps 290.563627, pps 0.277785.

No vpn
Completion time 0.010000,reliable bps 1046000.000000, pps 1000.000000.

In my configuration, the vpn server makes a blocking recvfrom to get feedback from server. Thus, we can see that the pps is much smaller. This can be improved by utilizing a polling method. Also, the overhead is introduced because the extra time spent to reach vpn server.

P3
TTL in our app：64

Other ping results from default ping:

```
PING www.purdue.edu (128.210.7.200) 56(84) bytes of data.
64 bytes from www.purdue.edu (128.210.7.200): icmp_seq=1 ttl=250 time=0.879 ms
64 bytes from www.purdue.edu (128.210.7.200): icmp_seq=2 ttl=250 time=0.816 ms
64 bytes from www.purdue.edu (128.210.7.200): icmp_seq=3 ttl=250 time=0.810 ms
64 bytes from www.purdue.edu (128.210.7.200): icmp_seq=4 ttl=250 time=0.736 ms
^C
--- www.purdue.edu ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3000ms
rtt min/avg/max/mdev = 0.736/0.810/0.879/0.054 ms
```

```
xinu07 86 $ ping www.cisco.com
PING e144.dscb.akamaiedge.net (23.79.213.27) 56(84) bytes of data.
64 bytes from a23-79-213-27.deploy.static.akamaitechnologies.com (23.79.213.27):
icmp_seq=1 ttl=55 time=7.95 ms
64 bytes from a23-79-213-27.deploy.static.akamaitechnologies.com (23.79.213.27):
icmp_seq=2 ttl=55 time=7.90 ms
64 bytes from a23-79-213-27.deploy.static.akamaitechnologies.com (23.79.213.27):
icmp_seq=3 ttl=55 time=7.95 ms
64 bytes from a23-79-213-27.deploy.static.akamaitechnologies.com (23.79.213.27):
icmp_seq=4 ttl=55 time=8.08 ms
64 bytes from a23-79-213-27.deploy.static.akamaitechnologies.com (23.79.213.27):
icmp_seq=5 ttl=55 time=8.00 ms
64 bytes from a23-79-213-27.deploy.static.akamaitechnologies.com (23.79.213.27):
icmp_seq=6 ttl=55 time=8.13 ms
64 bytes from a23-79-213-27.deploy.static.akamaitechnologies.com (23.79.213.27):
icmp_seq=7 ttl=55 time=8.11 ms
64 bytes from a23-79-213-27.deploy.static.akamaitechnologies.com (23.79.213.27):
icmp_seq=8 ttl=55 time=9.02 ms
64 bytes from a23-79-213-27.deploy.static.akamaitechnologies.com (23.79.213.27):
icmp_seq=9 ttl=55 time=7.97 ms
^C
--- e144.dscb.akamaiedge.net ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8010ms
rtt min/avg/max/mdev = 7.902/8.128/9.025/0.339 ms
```

```
ping www.google.com
PING www.google.com (216.58.192.164) 56(84) bytes of data.
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=1 ttl=53 time=7.06 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=2 ttl=53 time=7.27 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=3 ttl=53 time=7.16 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=4 ttl=53 time=7.15 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=5 ttl=53 time=7.09 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=6 ttl=53 time=7.24 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=7 ttl=53 time=7.18 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=8 ttl=53 time=7.21 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=9 ttl=53 time=7.12 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=10 ttl=53 time=7.18 ms
64 bytes from ord36s02-in-f164.1e100.net (216.58.192.164): icmp_seq=11 ttl=53 time=7.01 ms
^C
--- www.google.com ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time 10012ms
```

```
rtt min/avg/max/mdev = 7.015/7.155/7.274/0.102 ms
```

We can see that the ttl is different in different sites.

Different operating systems have different TTL. Windows TTL is 128. LINUX is 64. And UNIX is 255.
Thus by looking at the TTL, the attacker would be able to tell what kind of systems the machine is running. This will help the attacker to choose the proper attacking way for the specific OS.

TOS: default value, not set.
Fragmentation: is set to be don’t fragment.