

check0 实验报告

221900006 耿天成

2.1 Fetch a Web Page

```
221900006 耿天成 ~ check0.mmd
~/playground/minnow | main >1 ?1 telnet cs144.keithw.org http
Trying 104.196.238.229...
Connected to cs144.keithw.org.
Escape character is '^]'.
GET /hello HTTP/1.1
Host: cs144.keithw.org
Connection: close

HTTP/1.1 200 OK
Date: Tue, 24 Sep 2024 15:43:17 GMT
Server: Apache
Last-Modified: Thu, 13 Dec 2018 15:45:29 GMT
ETag: "e-57ce93446cb64"
Accept-Ranges: bytes
Content-Length: 14
Connection: close
Content-Type: text/plain

Hello, CS144!
Connection closed by foreign host.
```

Figure 1: web page fetch

2.2 Send yourself an email

```
~/playground/minnow | main >1 ?5 telnet smtp.exmail.qq.com smtp
Trying 183.2.143.59...
Connected to smtp.exmail.qq.com.
Escape character is '^]'.
220 smtp.qq.com Esmtplib QQ QMail Server
EHLO
501 Syntax: ehlo hostname
EHLO qq.com
250-smtp.qq.com
250-PIPELINING
250-SIZE 73400320
250-STARTTLS
250-AUTH LOGIN PLAIN
250-AUTH=LOGIN
250-MAILCOMPRESS
250 8BITMIME
auth login
334 VXNlcm5hbWU6
MjIxOTAwMDA2QHNTYWlsLm5qdS5lZHUuY24=
334 UGFzc3dvcmQ6
ajZaZkhkZHUcEtXalVFbw==
235 Authentication successful
mail from:221900006@smail.nju.edu.cn
250 Ok
rcpt to:221900006@smail.nju.edu.cn
250 Ok
data
354 End data with <CR><LF>.<CR><LF>
from:221900006@smail.nju.edu.cn
to:221900006@smail.nju.edu.cn
subject: lab0 test telnet and smtp

hello telnet
hello smtp
.
250 Ok: queued as
```

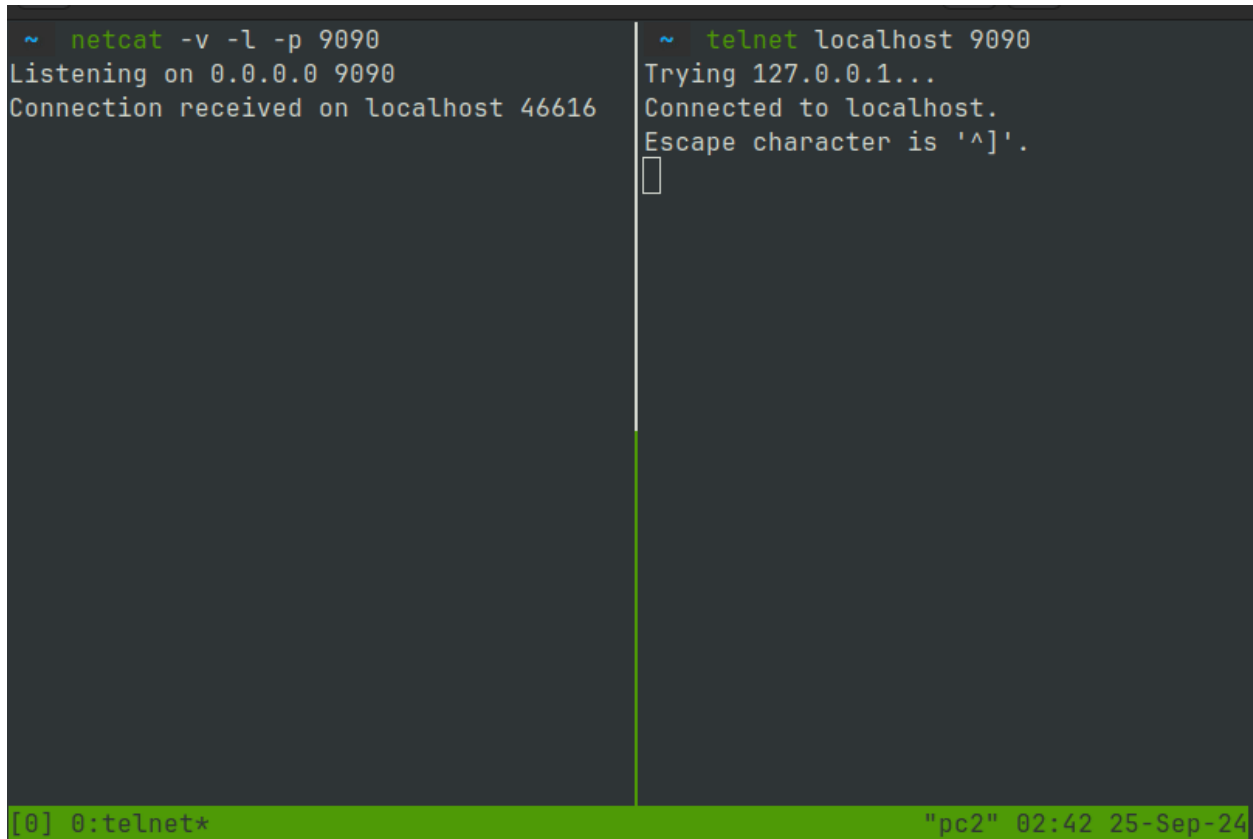
Figure 2: email terminal



Figure 3: email mailbox

Note: The email passcode is disabled and you cannot login through it.

2.3 Listening and connecting



```
~ netcat -v -l -p 9090
Listening on 0.0.0.0 9090
Connection received on localhost 46616

~ telnet localhost 9090
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
[0] 0:telnet* "pc2" 02:42 25-Sep-24
```

Figure 4: netcat and telnet

Hint: this is a gif, check it in markdown file.

3 C++ network program (ver. OS stream socket)

Code:

```
void get_URL( const string& host, const string& path )
{
+ TCPSocket tcp;
+ tcp.connect( Address( host, "http" ) );
+ tcp.write(
+   vector<string_view>( { "GET ", path, " HTTP/1.1\r\nHost: ", host, "\r\nConnection: close\r\n\r\n" } )
+   while ( !tcp.eof() ) {
+     string buffer;
+     tcp.read( buffer );
+     cout << buffer;
+   }
}
```

Result:

4 byte stream

Result:

```

~/playground/minnow | main >1 ?5 cmake --build build --target check_webget
Test project /home/isapo/playground/minnow/build
  Start 1: compile with bug-checkers
1/2 Test #1: compile with bug-checkers ..... Passed    2.89 sec
  Start 2: t_webget
2/2 Test #2: t_webget ..... Passed    1.06 sec

100% tests passed, 0 tests failed out of 2

Total Test time (real) = 3.96 sec
Built target check_webget

```

Figure 5: cmake check webget

```

~/playground/minnow | main >1 ?5 cmake --build build --target check0
Test project /home/isapo/playground/minnow/build
  Start 1: compile with bug-checkers
1/10 Test #1: compile with bug-checkers ..... Passed    0.12 sec
  Start 2: t_webget
2/10 Test #2: t_webget ..... Passed    1.06 sec
  Start 3: byte_stream_basics
3/10 Test #3: byte_stream_basics ..... Passed    0.01 sec
  Start 4: byte_stream_capacity
4/10 Test #4: byte_stream_capacity ..... Passed    0.01 sec
  Start 5: byte_stream_one_write
5/10 Test #5: byte_stream_one_write ..... Passed    0.01 sec
  Start 6: byte_stream_two_writes
6/10 Test #6: byte_stream_two_writes ..... Passed    0.01 sec
  Start 7: byte_stream_many_writes
7/10 Test #7: byte_stream_many_writes ..... Passed    0.06 sec
  Start 8: byte_stream_stress_test
8/10 Test #8: byte_stream_stress_test ..... Passed    0.06 sec
  Start 37: compile with optimization
9/10 Test #37: compile with optimization ..... Passed    1.29 sec
  Start 38: byte_stream_speed_test
  ByteStream throughput: 19.11 Gbit/s
10/10 Test #38: byte_stream_speed_test ..... Passed    0.05 sec

100% tests passed, 0 tests failed out of 10

```

Figure 6: cmake check0

Wow, 19Gbps. Super fast, right?

I use `queue` of `string` storing the data, and one `string_view` (an reference to `data.front()`) to accelerate peek and pop.

environment

```
lscpu | grep "Model name" | sort -u
Model name: AMD Ryzen 5 7500F 6-Core Processor
sudo dmidecode -t memory | grep Speed
      Speed: 4800 MT/s
      Configured Memory Speed: 6000 MT/s
      Speed: 4800 MT/s
      Configured Memory Speed: 6000 MT/s
cat /etc/issue
Ubuntu 24.04.1 LTS \n \l
cat /proc/version
Linux version 6.8.0-45-generic (buildd@lcy02-amd64-115) (x86_64-linux-gnu-gcc-13
(Ubuntu 13.2.0-23ubuntu4) 13.2.0, GNU ld (GNU Binutils for Ubuntu) 2.42) #45-Ubuntu
SMP PREEMPT_DYNAMIC Fri Aug 30 12:02:04 UTC 2024
```