IERG4180 Network Software Design and Programming
Project 4 Report

Name: Wang ZiFeng
SID: 1155194663
GitHub Repository: https://github.com/Catalpa1maple/IERG4180-Project

Requirement: **C++11** and **ws2_32.lib** (for windows)

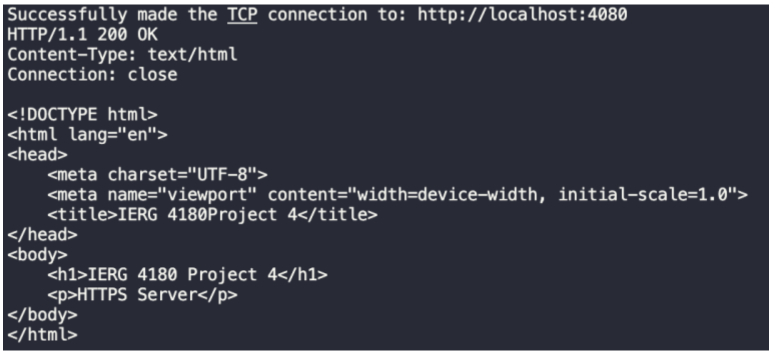
Command for compile(MacOS):

```
g++ -std=c++11 NetProbeServer.cpp -o NetProbeServer \
    -I$(brew --prefix openssl@3)/include \
    -L$(brew --prefix openssl@3)/lib \
    -lssl -lcrypto
```

Feature:

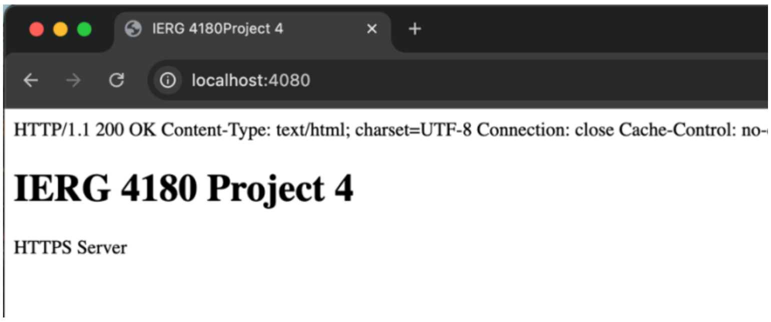
HTTP:

NetProbe:



```
Successfully made the TCP connection to: http://localhost:4080
HTTP/1.1 200 OK
Content-Type: text/html
Connection: close

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>IERG 4180Project 4</title>
</head>
<body>
    <h1>IERG 4180 Project 4</h1>
    <p>HTTPS Server</p>
</body>
</html>
```

Broswer:



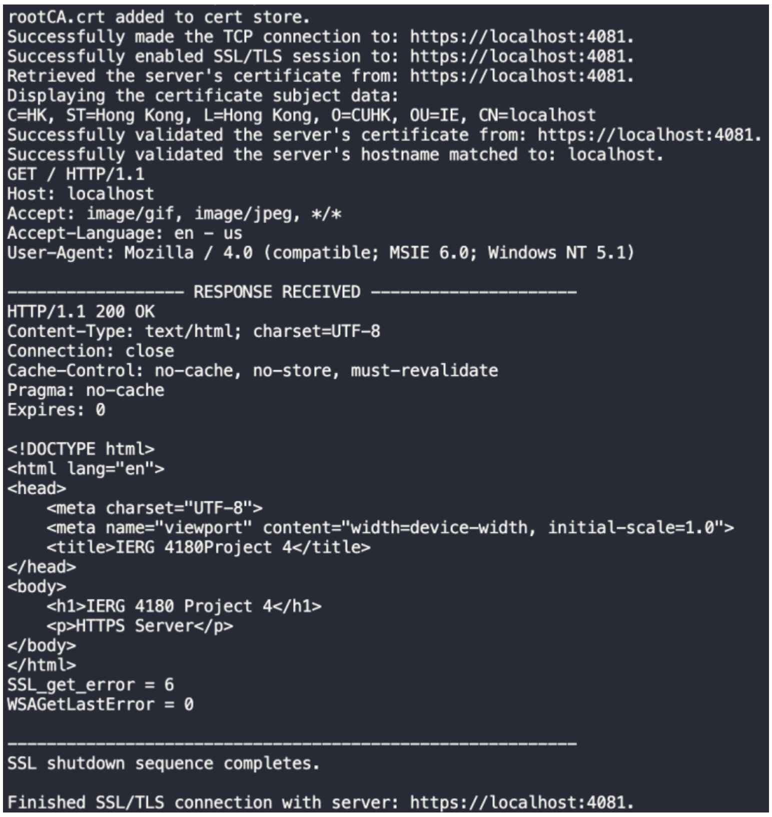
HTTP/1.1 200 OK Content-Type: text/html; charset=UTF-8 Connection: close Cache-Control: no-

IERG 4180 Project 4

HTTPS Server

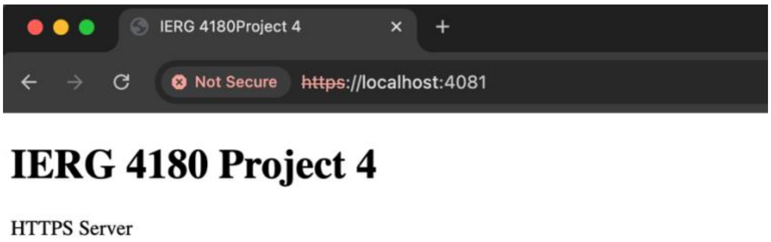
HTTPS:

NetProbe:



```
rootCA.crt added to cert store.
Successfully made the TCP connection to: https://localhost:4081.
Successfully enabled SSL/TLS session to: https://localhost:4081.
Retrieved the server's certificate from: https://localhost:4081.
Displaying the certificate subject data:
C=HK, ST=Hong Kong, L=Hong Kong, O=CUHK, OU=IE, CN=localhost
Successfully validated the server's certificate from: https://localhost:4081.
Successfully validated the server's hostname matched to: localhost.
GET / HTTP/1.1
Host: localhost
Accept: image/gif, image/jpeg, */*
Accept-Language: en - us
User-Agent: Mozilla / 4.0 (compatible; MSIE 6.0; Windows NT 5.1)

------------------ RESPONSE RECEIVED ---------------------
HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8
Connection: close
Cache-Control: no-cache, no-store, must-revalidate
Pragma: no-cache
Expires: 0

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>IERG 4180Project 4</title>
</head>
<body>
    <h1>IERG 4180 Project 4</h1>
    <p>HTTPS Server</p>
</body>
</html>
SSL_get_error = 6
WSAGetLastError = 0

----------------------------------------------------------
SSL shutdown sequence completes.

Finished SSL/TLS connection with server: https://localhost:4081.
```

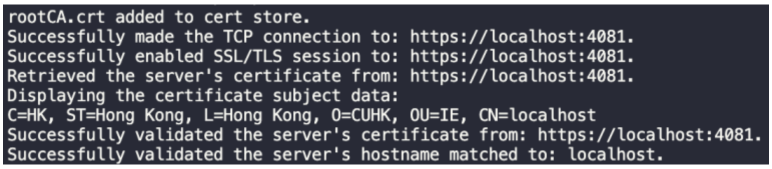
Browser:



IERG 4180 Project 4

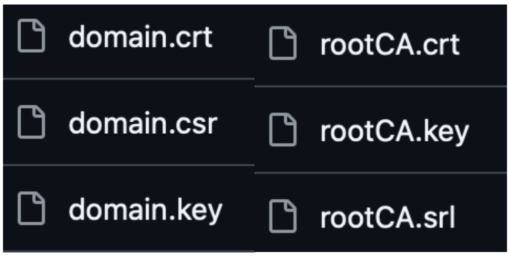
HTTPS Server

SNI:



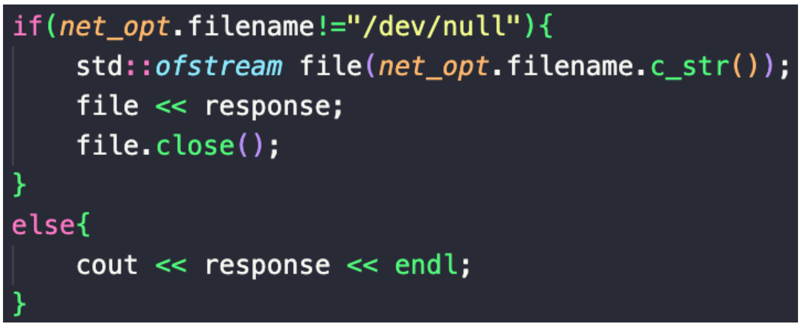
```
rootCA.crt added to cert store.
Successfully made the TCP connection to: https://localhost:4081.
Successfully enabled SSL/TLS session to: https://localhost:4081.
Retrieved the server's certificate from: https://localhost:4081.
Displaying the certificate subject data:
C=HK, ST=Hong Kong, L=Hong Kong, O=CUHK, OU=IE, CN=localhost
Successfully validated the server's certificate from: https://localhost:4081.
Successfully validated the server's hostname matched to: localhost.
```

Authentication:



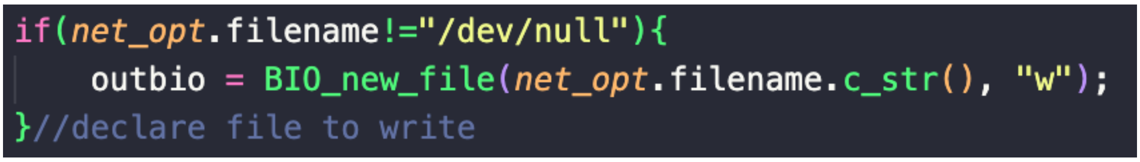
domain.crt
domain.csr
domain.key
rootCA.crt
rootCA.key
rootCA.srl

Implementation for parameter -file (code snippets):
HTTP:



```
if(net_opt.filename!="/dev/null"){
    std::ofstream file(net_opt.filename.c_str());
    file << response;
    file.close();
}
else{
    cout << response << endl;
}
```

HTTPS:



```
if(net_opt.filename!="/dev/null"){
    outbio = BIO_new_file(net_opt.filename.c_str(), "w");
}//declare file to write
```

Web Performance Measurement (HTTP):
For running host locally:
connection time is extremely small ~around 3ms ~ 5ms
reply time is roughly double of connection: ~8ms

For remote connection:
Connection time: ~ 50ms
Reply time: ~80ms

Experiments:

1. HTTPS needs time is around double of HTTP in preparation period of connection
We found that the reason mainly due to TLS setup and time of extra handshaking
For CPU usage, let say HTTP requires 30%(Mac) ~ 60%(Ubuntu) and HTTPS requires 60% ~ 70% which increased around 10%.

Arbitrary 5 times measurement on CPU usage and connection time(Ubuntu)

	1	**3**	**5**	**10**	**20**
HTTP	**56%** **68ms**	**45%** **55ms**	**53%** **52ms**	**64%** **51ms**	**61%** **49ms**
HTTPS	**71%** **104ms**	**77%** **79ms**	**69%** **72ms**	**68%** **86ms**	**72%** **81ms**

2. Arbitrary 5 times measurement on CPU usage during non-HTTP data via TCP

	1	3	5	10	20
TCP	74%	78%	71%	81%	67%

3. Arbitrary 5 times measurement on CPU usage during HTTP via TCP(as Exp.1)

	1	3	5	10	20
TCP	56%	45%	53%	64%	61%