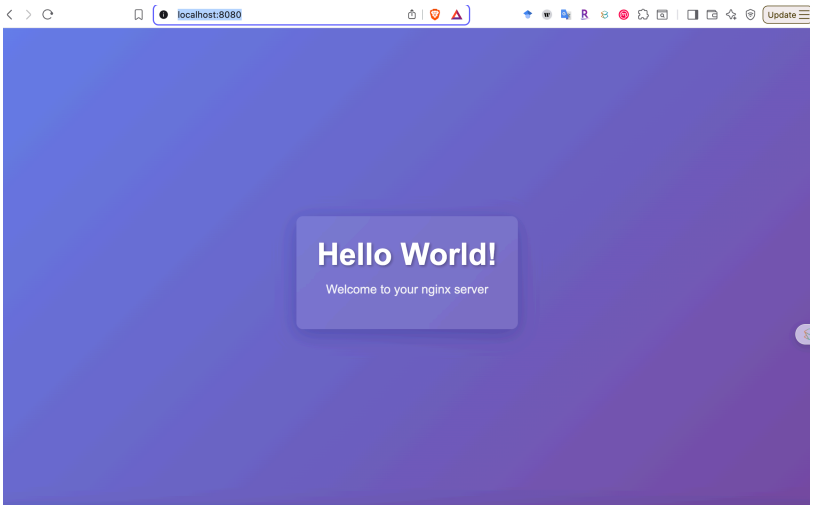
Lab 1 PKI

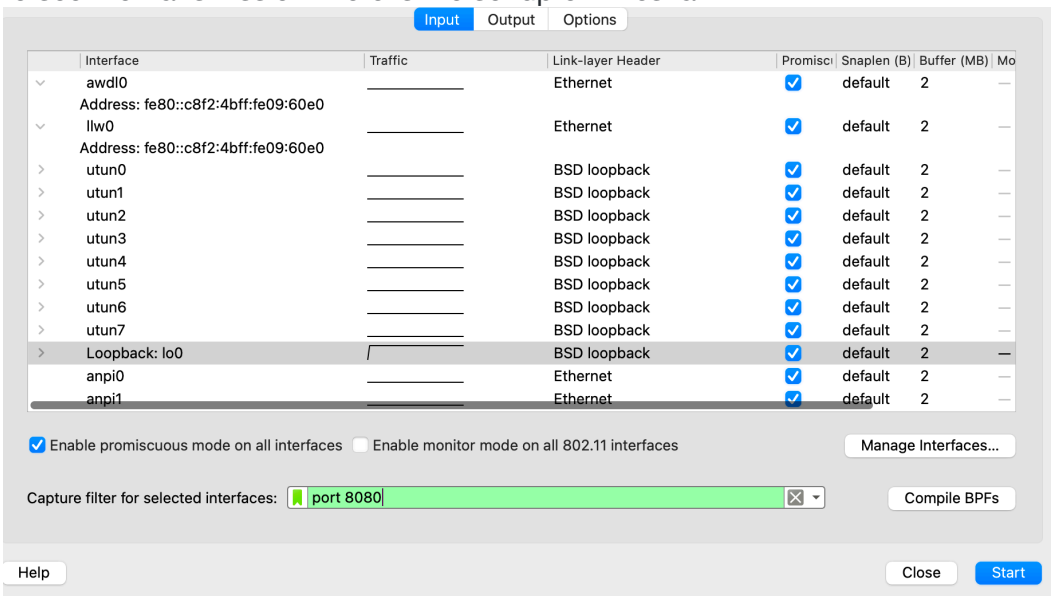
Dixi Yao, dixi@uchicago.edu

Task 1: Host a local web server

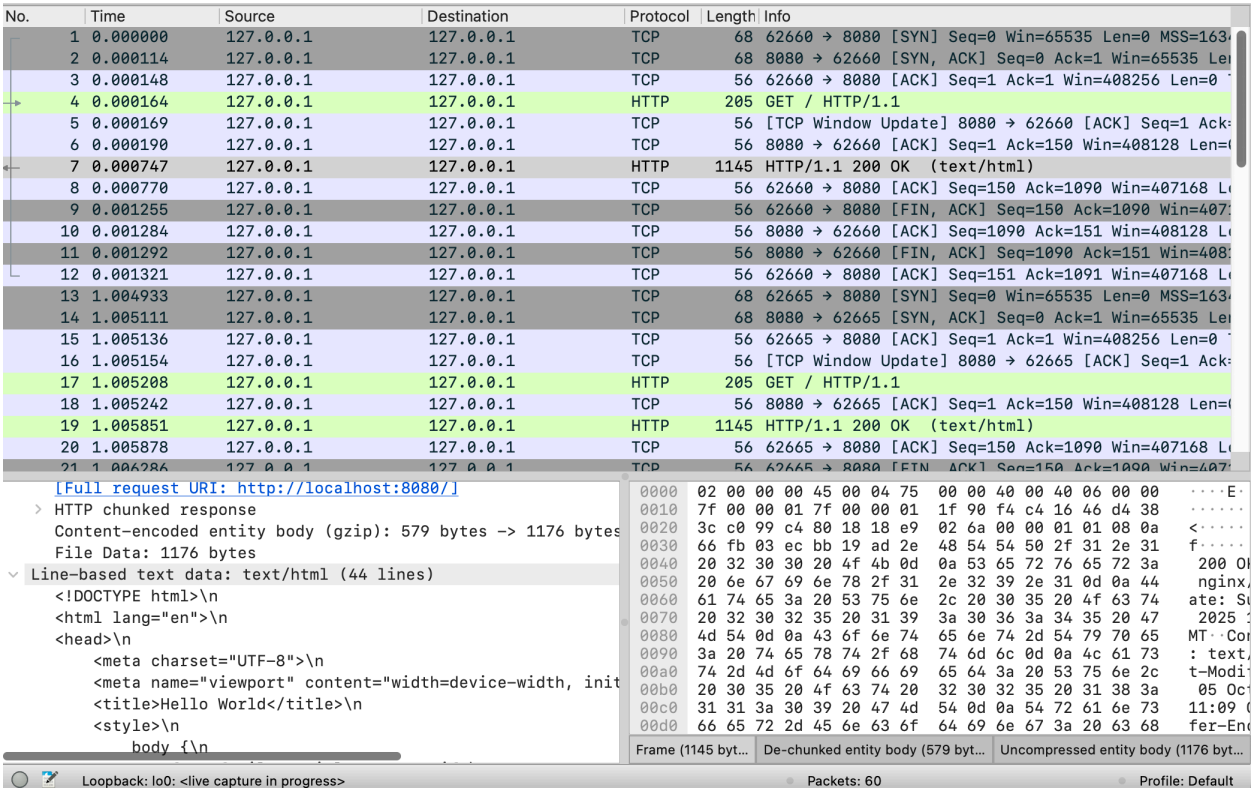
For web-server, I used nginx. I asked cursor with prompt “write me a simple webpage, with hello world , using nginx on localhost” and it generates a webpage with nginx for me. I checked the code and revised a little bit to make it running on my local host. After installing nginx and run command `sudo nginx -c "$(pwd)/nginx.conf" `, I got my localhost working on http://localhost:8080/.




2. Identify why HTTP is not secure
To answer the question, I first tried to use Wireshark to hi-jack the traffic. Because my web server is pretty simple. So the only basic command my client do is to send the simplest httrequest. I used `response = requests.get(server_url, timeout=10)` and let it repeat for 5 times to see the transmission. Here is the set up of wireshark:



So the basic idea is that as we can see in the Wireshark, an eavesdropper can capture the traffic and packets. For packets, we can see that there are a lot of TCP flows. HTTP is an application built upon the TCP flows. We put our http information into the TCP packet. As a result, if an eavesdropper gets these TCP packets. It can try to recover the HTTP content. We can see after the TCP protocol handshaking and http get send, we will have a response with HTTP/1.1 200 ok




And we can see that the original webpage of them is also encoded in the protocol (because server needs to send it to the client for get so that client and show in the browser). Then, all information about http is exposed. For eavesdropper, it first checks all TCP protocol and sort them with frame number. Then on sequence with sync, sync ACK, and ACK of TCP, HTTP protocol will be identified. And similar know which on contains the content of HTTP.

3. Create a self-signed certificate and upgrade your web server to HTTPS
So the first step is to upgrade my web server to HTTPS server. I directly put the question into cursor and let it upgrade my web server. First, we can use openssl to generate an SSL with `openssl req -x509 -newkey rsa:4096 -keyout ssl/server.key -out ssl/server.crt -days 365 -nodes -subj "/C=US/ST=Illinois/L=Chicago/O=CMSC30350/OU=Lab1/CN=localhost"`.Next, because my local root is MacOS, I directly added my SSL into the keychain so that when I connect my client with the server, it will be easier to verify. Here are some changes to the HTTPS server. So first, I need to add the configuration of using SSL in the configuration file:

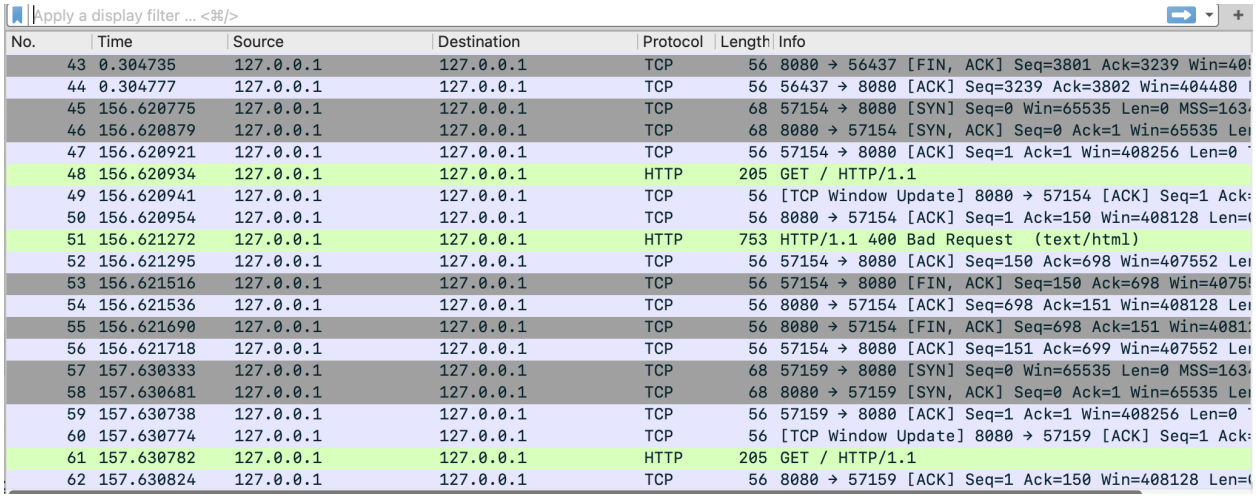
```
    # SSL Configuration
    ssl_protocols TLSv1.2 TLSv1.3;
    ssl_ciphers
    ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA384;
    ssl_prefer_server_ciphers off;
    ssl_session_cache shared:SSL:10m;
    ssl_session_timeout 10m;
```

The next thing is add the SSL certificate into my server

```
# HTTPS Server
server {
    listen 8443 ssl;
    server_name localhost;

    # SSL Certificate Configuration
    ssl_certificate "/Users/dixiyao/Desktop/Courses/CMSC 30350 1 Security, Privacy, and Consumer
    Protection/lab1/ssl/server.crt";
    ssl_certificate_key "/Users/dixiyao/Desktop/Courses/CMSC 30350 1 Security, Privacy, and Consumer
    Protection/lab1/ssl/server.key";
```

Again, we can use the wireshark to get the traffic and let's see the traffic this time. Because we have added certificates to our local host, we can also get the traffic. However, if I use the client to do it, which does not contain the certificate. I will get HTTP 1.1 /400 Bad request.




So back to our question, Why can't you obtain an SSL certificate for your local web server from a certificate authority? Because for a certificate authority, it needs a public verifiable DNS or global address so that they can give a certificate so that we can also send requests to a trustworthy identity with certificate. However, localhost is not a trustworthy certificate because everyone computer has a localhost. This is like my name is Dixi Yao and this is a identifiable information and I can be given the certificate. But there are a lot of husband and it is impossible to verify which husband is which and represents which house so it won't be able to get SSL certificate.
Another important reason is that my localhost is not logged into the certificate authority. As professor has demoed in the lecture, if I add my localhost into the authority in my computer keychain. If I save localhost into certificate authorities trusted my browser into my operating system, it will also work.