Lab 8 (Week 9)
TCP Connection Analysis

CAN201

Dr. Wenjun Fan

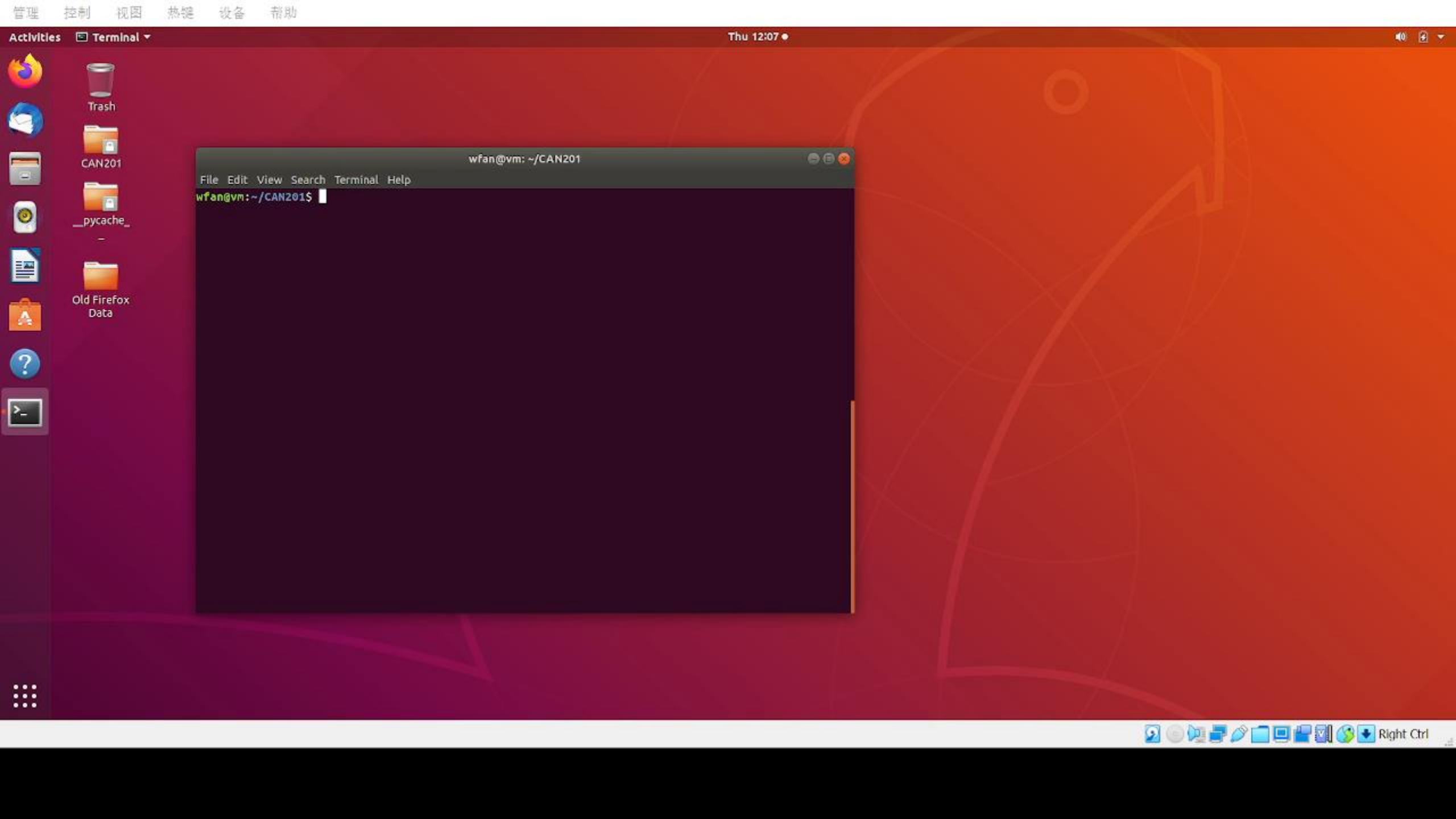
Outline

- General feedback for the last Lab
- TCP connection analysis
- Lab Exercise

General feedback the last Lab

- No module named ‘mininet’
- $sudo pip3 install or #pip3 install
- ip addresses inconsistent

https://box.xjtlu.edu.cn/f/c4c3d6cc73964c7b9db0/


管理 控制 视图 热键 设备 帮助
Activities
Terminal
Thu 12:07
Trash
CAN201
__pycache__
Old Firefox Data
wfan@vm: ~/CAN201
File Edit View Search Terminal Help
wfan@vm:~/CAN201$
Right Ctrl

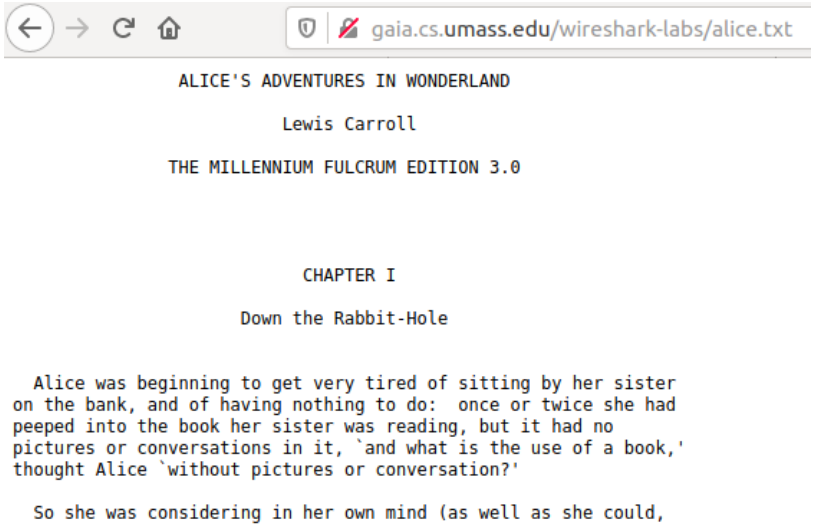
TCP connection analysis

1. We'll analyze a trace of the TCP segments sent and received in transferring a 150KB file from your computer to a remote server.

2. To this end, we'll need to use Wireshark to obtain a packet trace of the TCP transfer of a file from your computer to a remote server. You'll do so by accessing a Web page that will allow you to enter the name of a file stored on your computer, and then transfer the file to a Web server using the HTTP POST method while running wireshark during this time to obtain the trace of TCP segments.

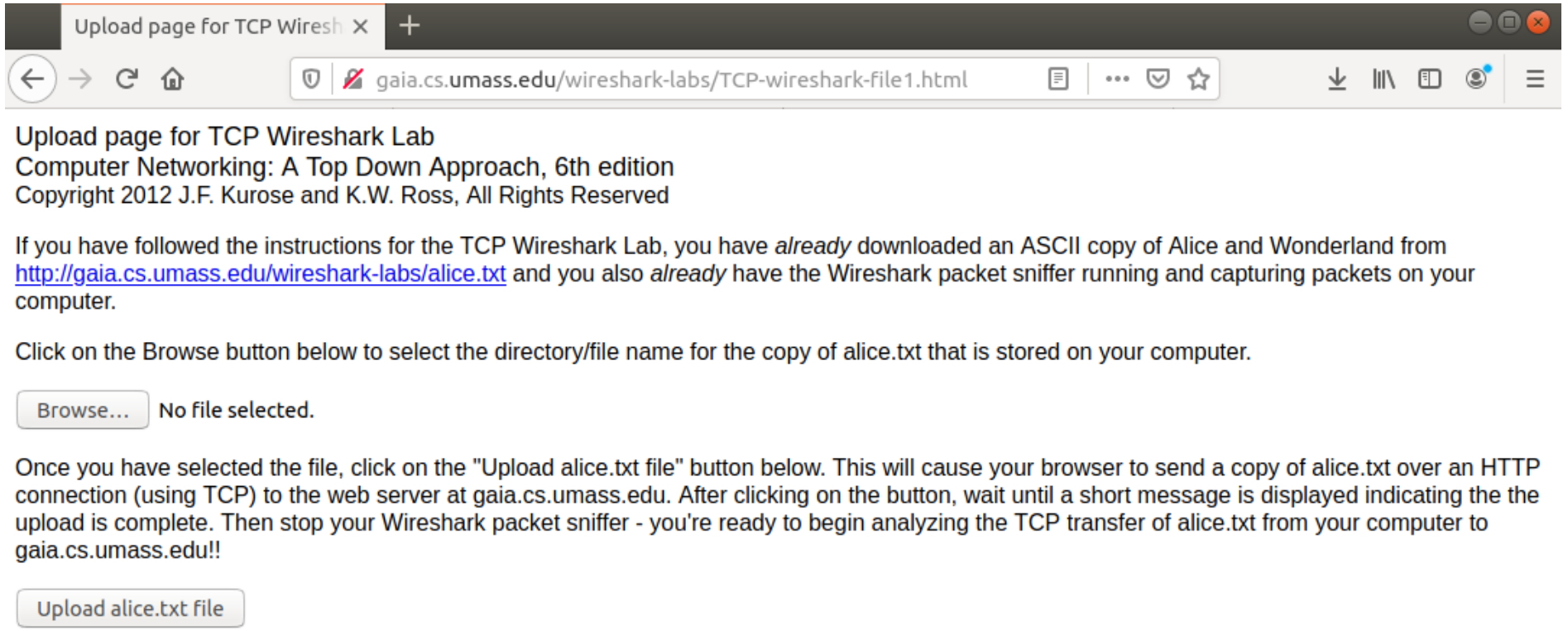
The test run for the TCP connection analysis

1. Start up your web browser. Go the http://gaia.cs.umass.edu/wireshark-labs/alice.txt and retrieve an ASCII copy of *Alice in Wonderland.* Store this file somewhere on your computer.
   - Note: you just save the page as alice.txt on your local computer (Ubuntu OS)



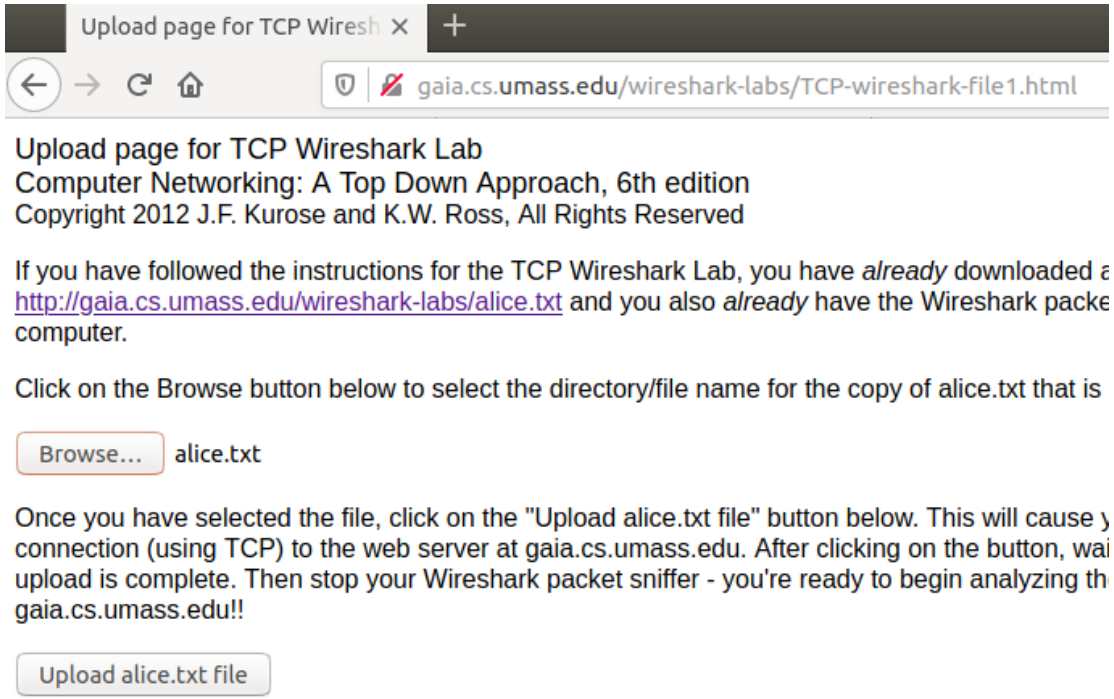
The test run for the TCP connection analysis

2. Next go to http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html. You should see a screen like this



The test run for the TCP connection analysis

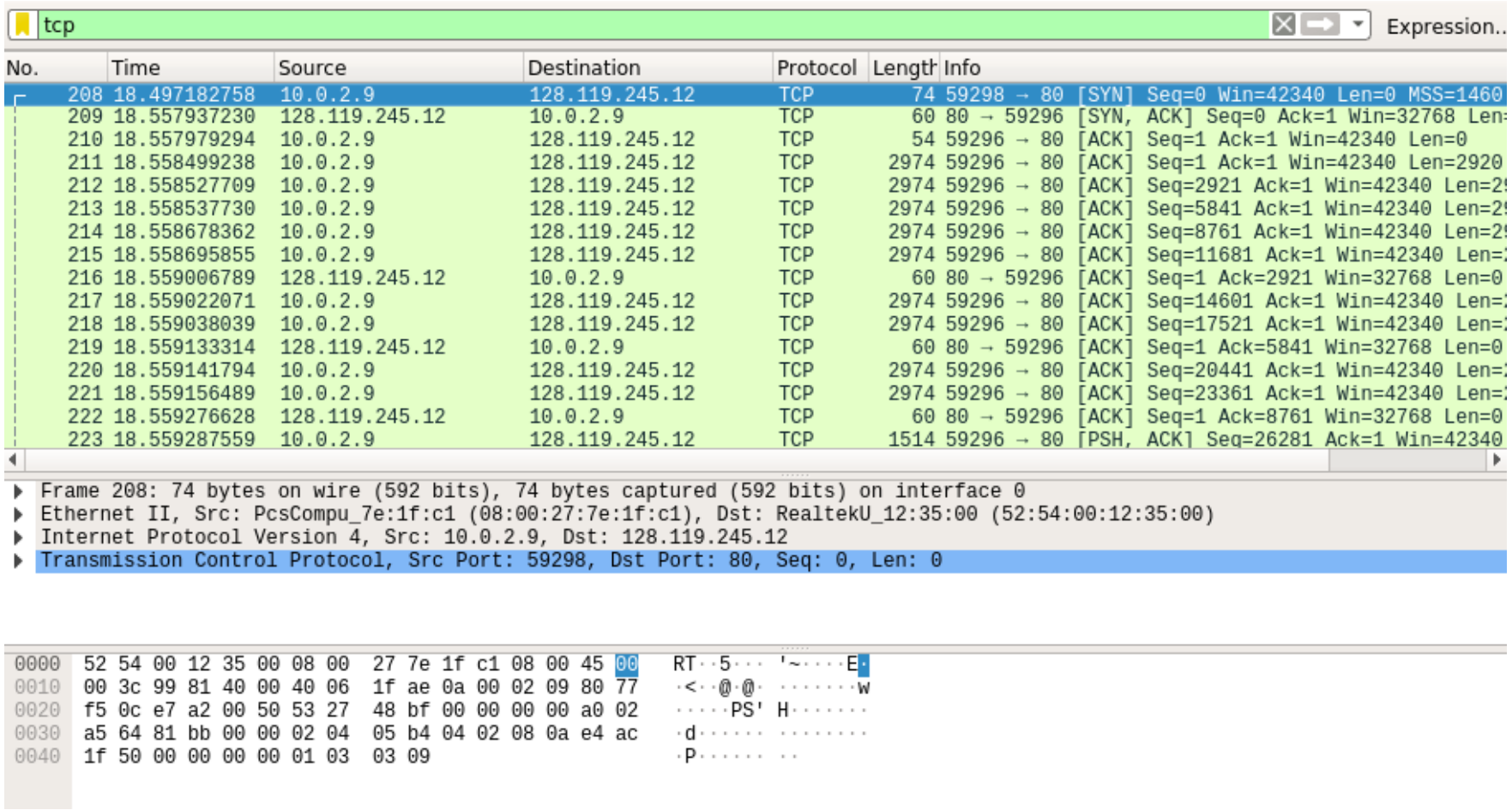
3. Use the *Browse* button in this form to enter the name of the file (alice.txt) on your computer.
   - Note: Don't yet press the "*Upload alice.txt file*" button.



The test run for the TCP connection analysis

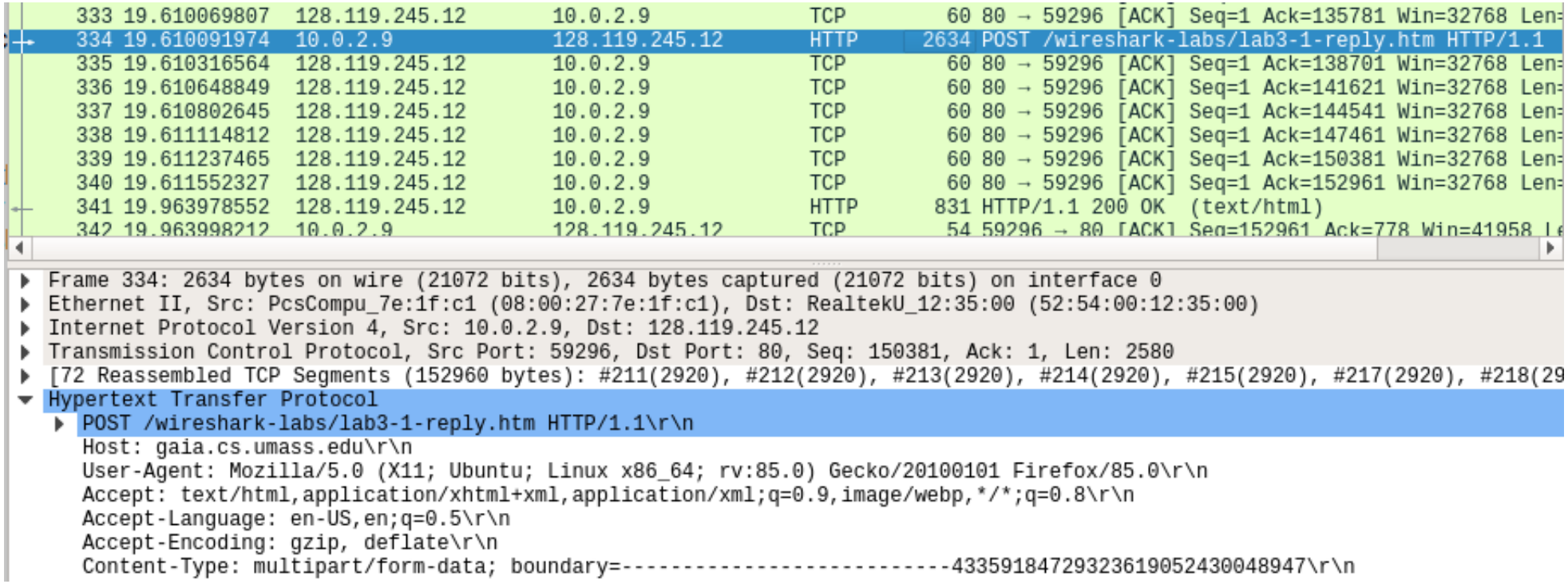
4. Now start up Wireshark and begin packet capture *(Capture->Start)* and then press *OK* on the Wireshark Packet Capture Options screen (we'll not need to select any options here).
5. Returning to your browser, press the "*Upload alice.txt file*" button to upload the file to the gaia.cs.umass.edu server. Once the file has been uploaded, a short congratulations message will be displayed in your browser window.
6. Stop Wireshark packet capture. Your Wireshark window should look similar to the window shown below.

The test run for the TCP connection analysis



The test run for the TCP connection analysis

- To find the POST command, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.



Lab Exercise

- Start your Wireshark to listen on the working interface, and then turn on your browser to open this link http://gaia.cs.umass.edu/wireshark-labs/alice.txt and retrieve an ASCII copy of *Alice in Wonderland.*
  - *To see if you can get the similar ASCII characters as follows that were captured by Wireshark when the browser sent an HTTP GET message*

```
GET /wireshark-labs/alice.txt HTTP/1.1
Host: gaia.cs.umass.edu
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:106.0) Gecko/20100101 Firefox/106.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
```

Lab Exercise

- With the above string of *ASCII characters, answer the following questions:*
  1) What is the URL of the document requested by the browser?
  2) Which version of HTTP is the browser running?
  3) Does the browser request a non-persistent or a persistent (TCP) connection?
  4) What type of browser initiates this message?