Lab 1

Step1:

1. What version of HTTP is the server running?

HTTP/1.1



- 2. How is the beginning of the content sent by the server recognized by the client? Because both the client and server are following the HTTP protocol, and there is a HTTP response header that contains various of information about this HTTP request, such as a header field called 'Content-Length', the client could parse this value, then better recognize its content.
- 3. How does the client know what type of content is returned? The HTTP response header contains header field called Content-Type, which specifies the type of content that is returned.



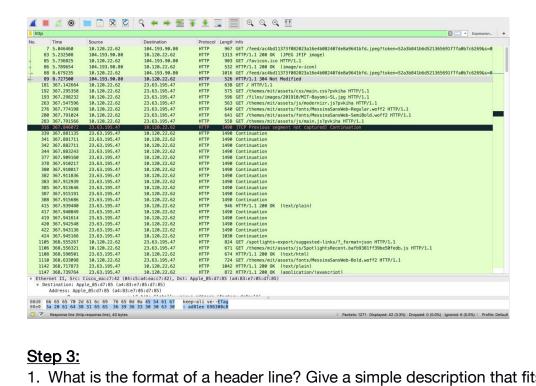
Step2:

1. find two HTTP URL:

http://pics3.baidu.com/feed/ac4bd11373f082023a16e4b082407de8a9641bfd.jpeg?token= 52a3b841b6d521365691f7fa0b7c6269&s=08606984CE33B9CE4AEC049B03004093

http://www.mit.edu

2. clear the browser history, and send the first HTTP request for two times one next another, to see how the browser cached the content, then send the second request.



Step 3:

- 1. What is the format of a header line? Give a simple description that fits the headers you see?
- 1). For HTTP request header, the first line is like:

HTTP-Method + SP + Request Target(such as URL) + SP + HTTP-Version + '\r\n', The rest lines are header fields as Key-Value pairs.

A key that represents a field of the request and its value is to the right side of the colon. Every line ends up with '\r\n'.

2). For HTTP response header, the first line is in a format:

HTTP-Version' + SP + Status-Code + \r\n'.

The rest lines are also header fields as Key-Value pairs.

```
GET /feed/ac4bd11373f082023a16e4b082407de8a9641bfd.jpeg?token=52a3b841b6d521365691f7fa0b7c62696s=08606984CE33B9CE4AEC049B03004093 HTTP/1.1\r\n
  Host: pics3.baidu.com\r\n
  Connection: keep-alive\r\n
  Upgrade-Insecure-Requests: 1\r\n
  User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.90 Safari/537.36\r\n
  Accept: \ text/html, application/xhtml+xml, application/xml; q=0.9, image/webp, image/appg, */*; q=0.8, application/signed-exchange; v=b3\r\n
   Accept-Encoding: gzip, deflate\r\n
  Accept-Language: en-US.en:g=0.9\r\n
   [truncated]Cookie: BAIDUID=145CF32FB3062E9B7808F2E903E73340:FG=1; BDUSS=XhxbTFpdDFLQm5kemxYYkl-cVRDMmFSNHFxelp5MENycUFT0G81emNySHlxNDlkRVFBQUFBJCQAAAAAAAA
Hypertext Transfer Protocol
  HTTP/1.1 200 OK\r\n
   Server: JSP3/2.0.14\r\n
   Date: Fri, 04 Oct 2019 18:16:25 GMT\r\n
   Content-Type: image/jpeg\r\n
  Content-Length: 63449\r\n
   Connection: keep-alive\r\n
   ETag: ad81ee696300c043c711cad0cb1601b3\r\n
   Age: 278994\r\n
   Accept-Ranges: bytes\r\n
   Cache-Control: 259200\r\n
   Tracecode: 22754150010194568970100120\r\n
   Traceid: 22754150010194568970100120\r\n
   Ohc-File-Size: 63449\r\n
   Timing-Allow-Origin: *\r\n
   Ohc-Cache-HIT: iad01-sys-jomo6.iad01.baidu.com [4], zhuzuncache92 [1], qdix170 [3]\r\n
```

2. What headers are used to indicate the kind and length of content that is returned in a response?

Content-Type and Content-Length



Step4:

1. What is the name of the header the browser sends to let the server work out whether to send fresh content?

If-None-Match: ad81ee696300c043c711cad0cb1601b3\r\n

The value to the right of the colon is ETag of the picture that is cached, it marks the version of a resource. If the resource is unchanged since last visit, then the cached content should have an ETag the same as what server side has. For our case, there is no change, so the server does not send a fresh content, and the response is 304 Not Modified.



2. Where exactly does the timestamp value carried by the header come from? It should come from the 'Last-Modified' header field of the most recent download of the content, which is timestamp given by the server when the content last modified.