The intention of this assignment is to give you some insight into the operation of HTTP, and some of the options that it has.

You will need access to something with which you can run the "curl" command and a packet tracing utility such as "tcpdump", "wireshark" or "ethereal". Note that many current Linux distributions which with curl version 7.64, which has a minor bug which means that you will get a lot of lines looking like

```
* Expire in 0 ms for 6 (transfer 0x55e8c3097f50)
* Expire in 1 ms for 1 (transfer 0x55e8c3097f50)
* Expire in 0 ms for 1 (transfer 0x55e8c3097f50)
```

You can ignore the "Expire in..." lines for all purposes.

Task 1 (40%)

This task is intended to familiarise you with the format of HTTP requests and responses.

Use "curl -v" to fetch a variety of webpages, such as www.facebook.com, (Links to an external site.) www.cs.bham.ac.uk, (Links to an external site.) www.batten.eu.org, (Links to an external site.) www.bbc.co.uk. (Links to an external site.)

Fetch both the http: and the https: versions when they are available.

The webpages you fetch will contain a variety of headers. Compare and contrast the sets of headers you get from the various sites you try, and explain why some sites provide radically different sets of headers to others. For sites where the http: version returns different content to the https: version, explain why and how this operates.

Task 2 (40%)

Curl operates by sending and receiving HTTP requests based on the command-line arguments. As with many Internet protocols, the actual requests and responses are in plain text. You can instead construct these requests yourself. This task gets you to do this.

You can make a connection to an unencrypted webserver with "telnet www.dom.ain (Links to an external site.) 80", for example "telnet www.batten.eu.org (Links to an external site.) 80". You can make a connection to an encrypted webserver with "openssl s_client -connect dom.ain:443", for example "openssl s_client -connect www.batten.eu.org:443"Links to an external site.. Using this method, or code written in any language of your choice in which you construct the request directly, fetch the index page and one further content page. Provide a screen shot or other explanation of how you did this.

Task 3 (20%)

This task is a prelude to more detailed analysis of TCP which will form the basis of the next exercise.

Repeat task 1, while monitoring the connection using a packet tracing tool like tcpdump. You will see the establishment of a TCP connection, then the exchange of HTTP protocol data (obviously, only do this with http, rather than https). Briefly describe what you see. For a small exchange, such as a 301 redirection, what problems can you foresee?