(1 – 8 번 문제는 wireshark을 이용하여 주어진 pcap 파일을 분석하는 문제입니다. Wireshark 프로그램은 packet을 capture하여 분석할 수 있는 tool입니다.)

For http-ethereal-trace-1,pcap

1. Is the browser of a client computer running HTTP version 1.0 or 1.1? What version of HTTP is the server running?
2. What is the IP address of the client computer?
3. What is the status code returned from the server to the client's browser? (for the first request and the second request)
4. When was the HTML file that the client is retrieving last modified at the server? (for the first request)
5. How many bytes of content are being returned to the client browser for the first HTTP GET?

For http-ethereal-trace-2.pcap

6. Now inspect the contents of the second HTTP GET request from the client browser to the server. Do you see an "IF-MODIFIED-SINCE:" line in the HTTP GET? If so, what information follows the "IF-MODIFIED-SINCE:" header?
7. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.

For DNS

8. 첨부된 파일 Wireshark_DNS_Sept_15_2009.pdf 파일에서 1. nslookup 과 2. Ipconfig 부분을 읽고, 그 부분에서 설명하고 있는 command 들과 command의 동작을 요약해서 정리하라. Linux에서 nslookup 대신에 사용하는 명령어는 무엇이고, ipconfig 에 준하는 명령어는 무엇인가? Linux에서 각각 명령어를 사용해 보아라. (결과 캡쳐 제출)

9. (학교 내에서 실행하시오.) 사용하는 컴퓨터에서 DNS cache를 clear 시키고, (어떤 명령을 사용하여야 하는가?) nslookup www.example.com 를 실행할 때, DNS에 관련된 패킷들을 wireshark으로 capture 한 다음, capture된 각각의 패킷에 대해서

설명하라. (무엇에 대한 query 혹은 response를 누가 누구에게 하고 있는가에 대해서.) (DNS 패킷만을 capture하기 위한 filter를 설정하는 방법을 설명하고 capture한 결과도 포함하라.)

10. 위에서 capture한 DNS 패킷에서
    A. Locate the DNS query and response messages. Are then sent over UDP or TCP?
    B. What is the destination port for the DNS query message? What is the source port of DNS response message?
    C. To what IP address is the DNS query message sent?
    D. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"? (여러 가지 종류가 있다면 각각에 대해서.)

11. 요즘 일부 서버만이 HTTP/2을 지원하고 있으며, 대부분의 서버는 여전히 HTTP/1.1을 지원하고 있다. 사용자가 서버에 접속을 할 때, 만약 HTTP/2을 지원하고 있으면 HTTP/2를 사용하고, 아닐 경우 HTTP/1.1을 사용하기를 원한다면 어떤 형태로 HTTP request를 보내는가?