Ssl2\_final

**12141163 이욱진**

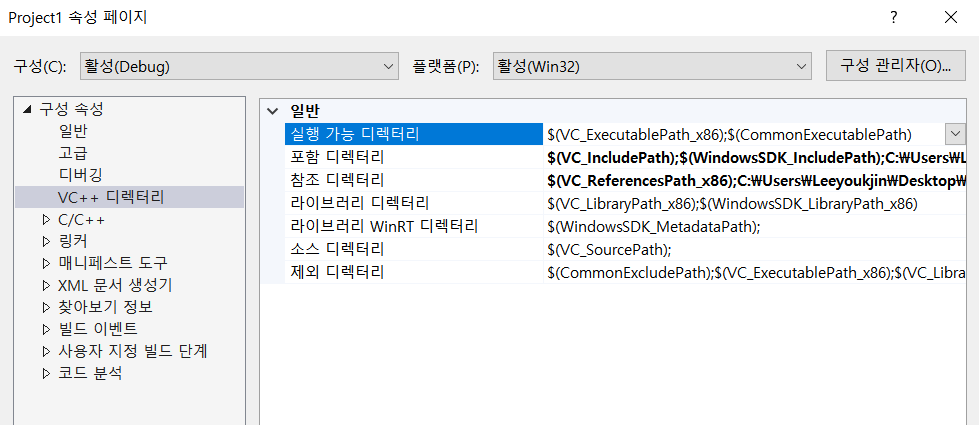
**2020.11.25.(수)**

Homework

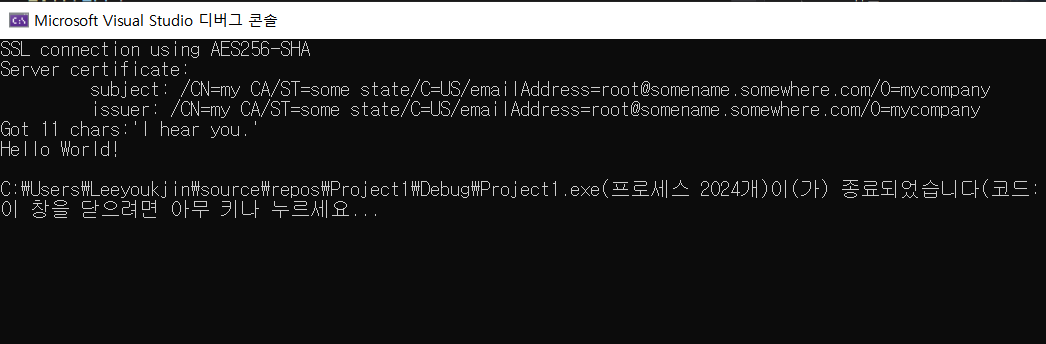
6) Do the steps in Section 3 and 4.

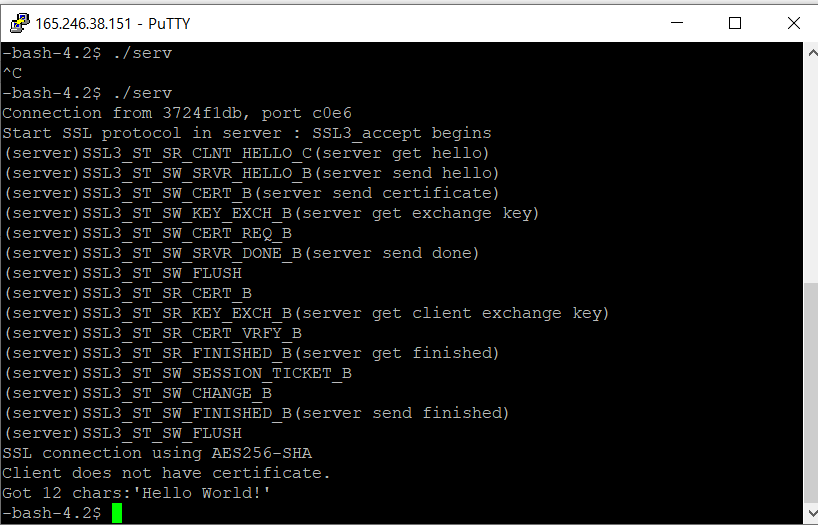
Openssl 정식 홈페이지에 접속하여 2017 버전을 다운받고 압축을 풀었습니다.

그리고 c++파일작성을 위하여 라이브러리와 인클루드 해야 할 파일을 연결시켜주었습니다.



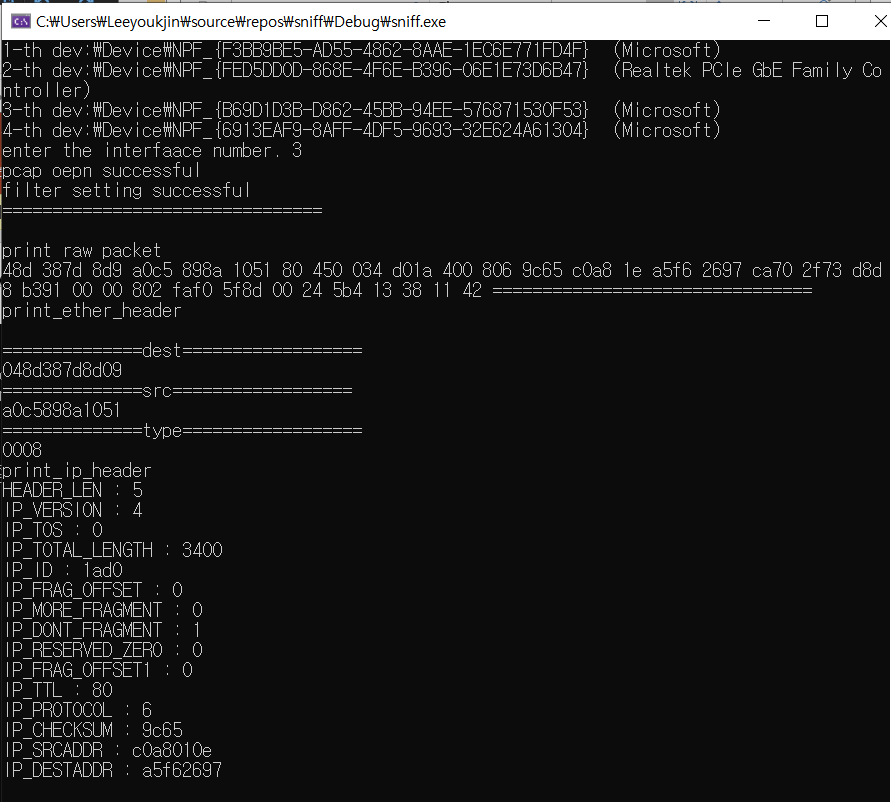
그 후 putty program을 이용하여 랩 서버에서 서버를 실행시켜 연결시켜보았습니다.





정상적으로 연결되는 것을 확인하였습니다.

스니퍼를 실행하여 통신이 이루어질 때 패킷을 살펴보았습니다.



Client hello, server hello만 관찰하기 위하여

3번의 handshake가 진행이후를 살펴보았습니다.



Tcp header 이후 print\_data를 살펴보았습니다.

Client hello protocol message

Message type(1 byte) : 1 (Client Hello)

Message len(3 byte) : 0 0 5d (93)

Message data

Version(2byte) : 3 1

Random(32 byte) : a6 25 23 26 64 45 72 8d 15 5e f0 e2 e0 c1 6 96 81 50 eb b4 45 50 57 c3 10 84 fb 83 f2 35 64 f7

sessionID length(1b byte) : 0

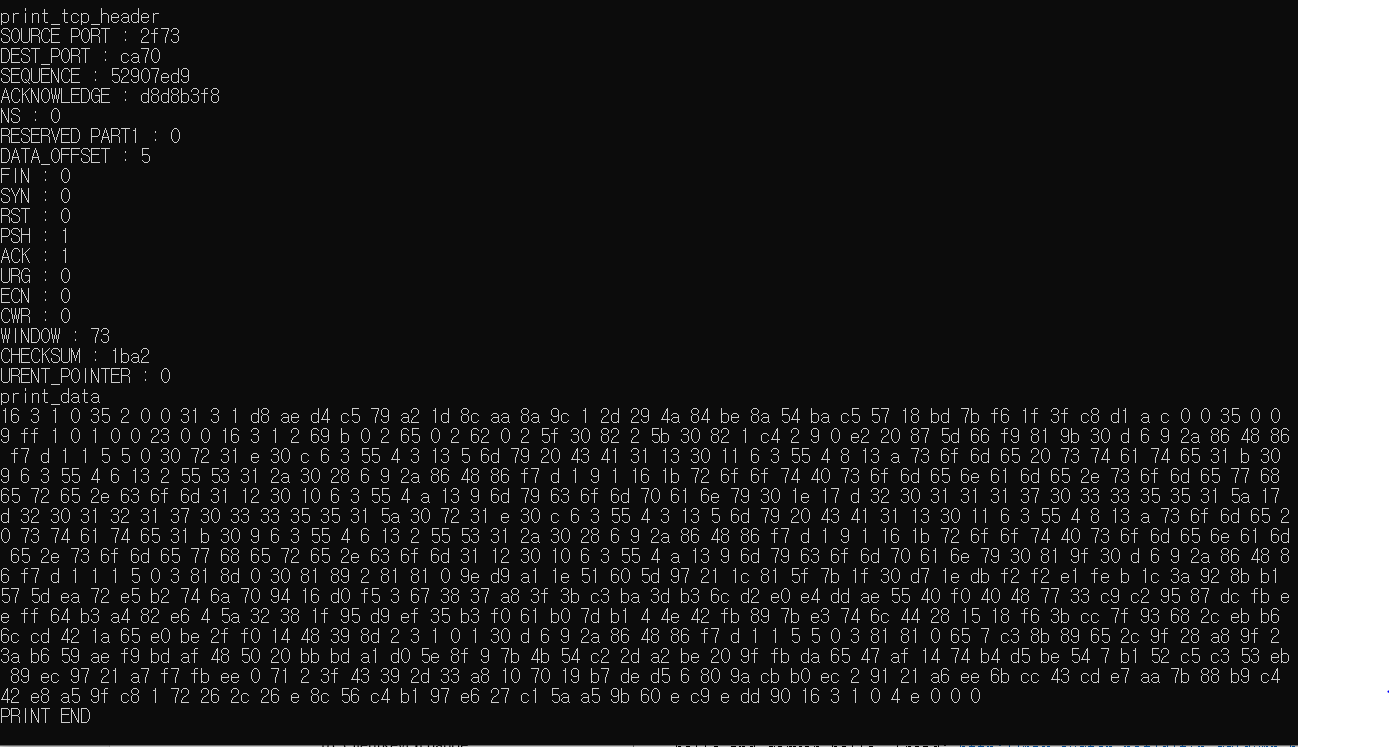
sessionID : 0

CipherSuite length(2 byte) : 0 12 (18)

CipherSuite (18 byte) : c0 a c0 14 0 39 c0 9 c0 13 0 33 0 35 0 2f 0 ff

CompressionMethod length(1 byte) : 1

CompressionMethod (1 byte) : 0



Server hello

위와 동일하게 MT부분을 살펴보아 2로 확인할 수 있으며 이것은 Server hello를 뜻한다.

server hello protocol message

Message type(1 byte) : 2 (Server Hello)

Message len(3 byte) : 0 0 31 (49)

Message data

Version(2byte) : 3 1

Random(32 byte) : d8 ae d4 c5 79 a2 1d 8c aa 8a 9c 1 2d 29 4a 84 be 8a 54 ba c5 57 18 bd 7b f6 1f 3f c8 d1 a c

sessionID length(1b byte) : 0

sessionID : 0

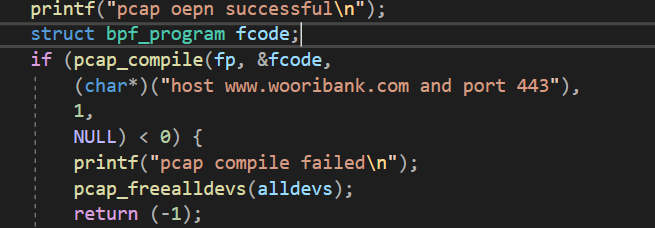
CipherSuite length(2 byte) : 0 35 (53)

CipherSuite (29 byte) : 0 0 9 ff 1 0 1 0 0 23 0 0 16 3 1 2 69 b 0 2 65 0 2 62 0 2 5f 30 82 2 5b 30 82 1 c4

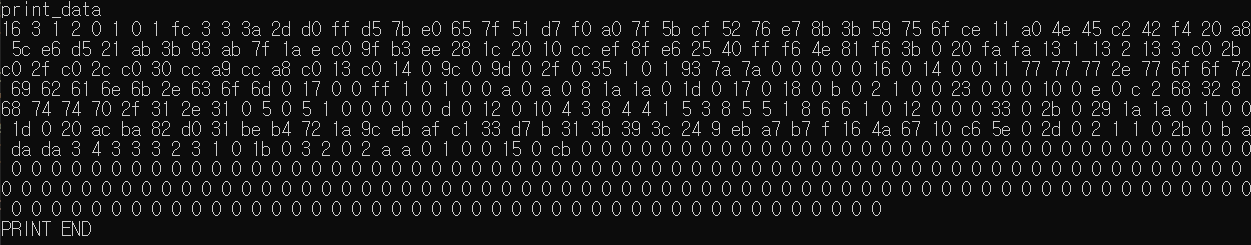
CompressionMethod length(1 byte) : 0 (server hello에서는 존재하지않는다)

CompressionMethod len 이후의 내용.

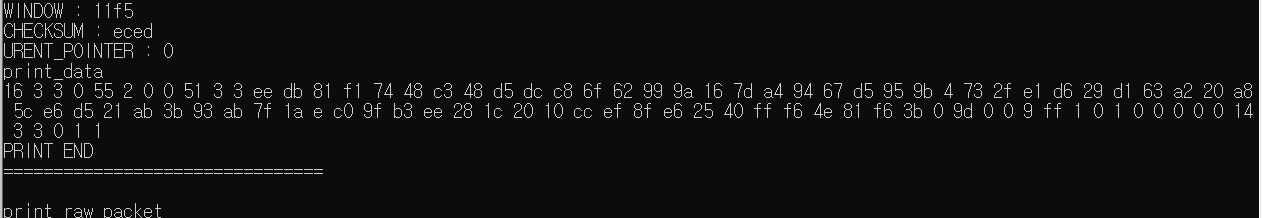
7) Make the sniffer sniff port 443 which is the default ssl port. Use your web browser to connect to www.wooribank.com and let the sniffer analyze the packets. Analyze only client hello and server hello. (read: <http://max.euston.net/d/tip_ssldump.html>)



Client hello



Server hello



위와같이 스니퍼를 통하여 찾았으나 ciphersuit length의 길이가 너무나도 길게 나왔습니다 아마도 문제에서 의도한 패킷이 아닌 것 같았습니다.

10) Read articles about“heartbleed” attack in the Internet and summarize. Find actual attack incidents. Which function in openssl can cause the "heartbleed" attack? Analyze this function and explain why it can allow the "heartbleed" attack.

https://www.bbc.com/news/technology-45472766

위 기사는 Andrei Tyurin러시아 남성이 유선 및 은행 사기와 컴퓨터 해킹 혐의로 조지아에서 미국으로 송환되었습니다. 위 사람 말고도 다른 4명을 포함한 갱단이 미국 금융 기관에서 1억명 이상의 미국인에 대한 정보를 Heartbleed 버그를 포함한 많은 기술로 훔친 내용입니다.

다음 사이트를 참고하여 heartbleed를 알아보았습니다. <https://blog.alyac.co.kr/76>

하트블리드는 OpenSSl 1.0.1 버전에서 발견된 취약점이며 OpenSSL을 구성하고 있는 TLS/DTLS의 HeartBeat 확장 규격에서 발견된 취약점입니다.

클라이언트와 서버가 서로 문제는 없는지 상호 전달받는 HeartBeat 확장규격에서 클라이언트가 서버에게 잘못된 거짓정보를 서버에게 보낼 경우 서버가 검증하지 않는 상황에서 발생하게 됩니다.

클라이언트는 한번에 최대 64KB정보를 요청할 수 있는데 이 작은 정보들을 통해 유의미한 큰 정보를 알아낼 수 있습니다.