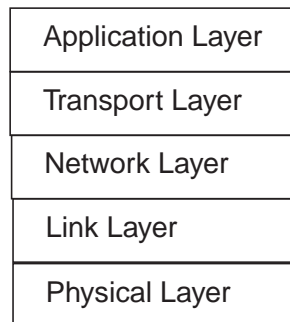


Homework 5

Due date: Nov 30

1. (20 pts) Consider the following security standards, protocols and software: (1) SSL; (2) PGP; (3) IPsec.
 - (a) Please mark these security protocols and software at their corresponding places in the network stack. (10 pts)



- (b) Please identify the ones that are able to support the following security services and briefly explain how. (10 pts)
(For example, PGP provides data confidentiality. It does so by using the symmetric ciphers. To distribute the secret key used in the symmetric cipher, it encrypts the secret key using the receiver's public key.)
 - Data confidentiality.
 - Data content integrity.
 - Source authentication.
 - Defense against replay attack.
2. (10 pts) Suppose Alice sends packets to Bob using TCP over IPsec. If the TCP acknowledgement from Bob is lost, then the TCP sender at Alice's side will assume the corresponding data packet was lost, and thus retransmit the packet. Will the retransmitted TCP packet be regarded as a replay packet by IPsec at Bob's side and be discarded? Please briefly explain your answer.

3. (40 pts) In this problem, you will be using OpenSSL, a toolkit that implements the Secure Socket Layer (SSL) and Transport Layer Security (TLS) protocol as well as a full-strength cryptography library. Information about OpenSSL is available on the homepage

<http://www.openssl.org/>.

OpenSSL is installed in the Linux systems at the instructional lab of EECS department. For each of the machine problems below, please include the *commands you use and the system output* in your answer.

- (a) **openssl** is a command line tool for using the various cryptography functions. Its usage manual can be found at

<http://www.openssl.org/docs/apps/openssl.html>

First generate a pair of RSA keys, encrypt the private key with 3DES cipher, and store the key in file `cs291.key`. Also list the content of `cs291.key` as part of your answer of this question. (5 pts)

(e.g., **openssl genrsa -des3 -out cs291inst.key 2048**)

- (b) You can request a certificate for your public key. Please explain why a certificate would be needed for a public key. (5 pts)

- (c) Generate a X.509 Certificate Signing Request (CSR) and store it in file `cs291.csr` using command **req**. Please also include the content of the certificate request as part of your answer. (5 pts)

- (d) For commercial use, you can send your certificate signing request (CSR) to Verisign to buy a SSL certificate (Refer: <http://www.verisign.com/products-services/security-services/ssl/index.html>). For this homework, please create a self-signed temporary certificate. (5 pts)

- (e) Generate a MD5 digest for the source code package at the course website (`ssl-example.tar.gz`) and compare the result with the following one provided by the instructor

```
MD5(ssl-example.tar.gz)= f828ab1750b281ce7018918e789b6164
```

If the results match, what does it mean? (5 pts)

- (f) Please list the ciphers supported in the system via command **openssl ciphers**. (5 pts)

- (g) The OpenSSL **ssl** library implements the Secure Sockets Layer (SSL v2/v3) and Transport Layer Security (TLS v1) protocols and provides a rich API to the application developers. An example source code is provided for this question in `ssl-example.tar.gz`.

Read the source code, and make the following changes to the code and its supporting files. (10 pts)

- Enable the server to deliver your self-signed temporary certificate to the client upon request.
- Change the port number to 6677. Could you change the port number to 66 and run it on the EECS linux system? If not, why?

As your answer to this question, (1) send the modified code `cs291hw3.tar.gz` (**make clean; tar -zcvf cs291hw3.tar.gz ***) to the CS291 TA (yanchuan.cao@vanderbilt.edu);

(2) compile and run the code. Copy the system output from the client side as part of your answer to this question.