CS110 - Principles of Computer Systems

Final Exam

(Total time = 180 minutes, Total Points = 180)

Name: (please print)_____

In recognition of and in the spirit of the Stanford University Honor Code, I certify that I will neither give nor receive unpermitted aid on this exam.

Signature:_____

This examination is closed book and closed notes. You may not collaborate in any manner on this exam. You have 180 minutes to complete the exam. Please write your answers on the exam. Note there is one problem per page so the amount of space provided does not necessarily provide an indication of the expected length of the answer. In other words, do not feel compelled to fill every nanoacre of the exam with writing. Before starting, please check to make sure that you have all 17 pages.

Question	Points	Score
1	10	
2	10	
3	12	
4	12	
5	10	
6	12	
7	10	
8	12	
9	10	
10	12	
11	12	
12	12	
13	12	
14	10	
15	10	
16	14	
Total	180	

1.

(10 points) Assume we have a secure email system built on top of a universally deployed public key cryptosystem where everyone has a private key and a reliable way of retrieving the public key of someone else. Assume no other cryptographic primitives are available. Which of the following security properties of *Authenticity*, *Integrity*, and *Secrecy* will the email system **not** be able to support. Explain your answer.

2. (10 points) Explain how a website can tell if your password is correct but can honestly say they don't know your password.

3. (12 points) For each of the different security attacks listed below state if prevention would be easy or hard. Describe what would have to be done to prevent it.

a) Timing Attack
b) Replay Attack
c) Denial of Service Attack

4. (12 points) For each of the following protection systems for a web service (e.g. think assignment #4 from the class), classify if the system would be consider as:
   (1) a capability-based system,
   (2) an access control list-based system,
   (3) neither a capability nor access control list system, or
   (4) both a capability and access control list system.

   Briefly justify your answer.

a) The system works by having the user submit an account name and password which are looked up in a database associated with the web service.

b) The system works by having the website URL include a very long and obscure name which is not publicized yet is needed for any request to be processed.

c) The system works by having the user submit an account name and password that is scrambled into a long and obscure string that is part the URL which is submitted to the web service. The web service unscrambles the URL string to extract the name and the password which are looked up in a database.

5. (10 points) Two different end-to-end protocols, Transmission Control Protocol (TCP) and Real-time Transport Protocol (RTP), are used to send streams of bytes across the Internet. While TCP is used when sending streams of normal data, RTP is used for real-time media such as audio and video streams. Explain why although everything is just streams of bytes, a different protocol is needed for real-time media than normal data. Focus on what part of TCP is unsuitable for media.

6. (12 points) The algorithm used by TCP attempts to compute what is called the RTT (round trip time) for each connection. The RTT is the amount of time it takes for a packet to go from the source machine to the destination machine and back to the source.

a) List the possible causes of delay in a network that would factor into the RTT of a connection

b) Explain why TCP needs to compute this RTT value.

7. (10 points) Explain *exponential backoff* and what characteristics of a situation make it applicable. Give an example of such a situation.

8. (12 points) Answer the following two questions about IP fragmentation.

a) What problem does IP fragmentation solve?

b) Explain why it is possible for a TCP implementation to avoid ever triggering IP fragmentation regardless of the workload and network characteristics while a UDP implementation can not always avoid it.

9. (10 points) When doing a request/response protocol over TCP in assignment #4 we had to deal with framing issues. Answer the following two questions:

a) Explain why we wouldn't have framing issues if we had used UDP as a transport in assignment #4.

b) What would have been the disadvantage of using UDP rather than TCP as a transport in assignment #4?

10. (12 points) Most modern operating systems use a disk scheduling algorithm that is similar to the scan (elevator) algorithm rather than first-come-first-served (FCFS) or shortest seek time first (SSTF).

a) List one advantage scan has over FCFS .
b) List one advantage scan has over SSTF.
c) Explain why in practice with normal workloads neither of these advantages is realized.

11. (12 points) The virtual memory pages that make up the address space of a process on a modern operating system can be though of as residing either in memory or on disk. Answer the following two questions.

a) What would cause a process' page that is resident on disk to be moved in to memory?

b) What would cause a page in memory to be moved to disk?

12. (12 points) Assume you have two services A and B with A having latency $latency_A$ and throughput $throughput_A$ and B having latency $latency_B$ and throughput $throughput_B$. Assume that the two services are steps in a pipeline with A being first and B last. We know from the formula the entire system is:
$Thoughput_{A+B} \leq$ **minimum**$(Throughput_A, Throughput_B)$

What is the relationship between $Latency_{A+B}$ and **maximum**$(Latency_A, Latency_B)$? The choices are $\leq$, $\geq$, = , or can't tell. Explain your answer.

13. (12 points) Assume you have a service that produces widgets and takes 2 seconds to produce one. It is clear from the formula:
    Throughput = 1 / (Latency)
    that you can have a throughput of .5 widgets per second (i.e. you produce a widget every two seconds). Explain how it would be possible to get the throughput much higher even if the latency stays the same.

14. (10 points) When implementing virtualization we can use *multiplexing*, *aggregation*, or *emulation*. A virtual machine monitor (VMM) can apply all three to modules with a memory abstraction or a communication channel abstraction but can't apply all three to an interpreter abstraction. Describe what implementation techniques do not work for interpreter abstractions in a VMM.

15. (10 points) Assume you are running a virtual machine monitor on a system that has two identical virtual machines running the same software. Assume somehow the virtual machines are in total synchronization so that the memory is identical and even the page tables of the running processes in the machines are identical. Explain how it is possible that even with identical page tables in the virtual machines the virtual machines are touching distinct parts of the machine's physical memory.

16. (14 points) Assume you have a system with three threads: Thread A calls routine `ThreadA()`, Thread B calls routine `ThreadB()`, and Thread C calls routine `ThreadC()`. Before any of the threads start the routine `Init()` is called and returns. Unfortunately the person that coded these routines didn't understand locks at all. Below are three possible behaviors for the system. Choose the **one** that best describe what would happen if you ran it on a system with an unpredictable but starvation-free thread scheduler.
    a) The system will deadlock. If so, describe the deadlock cycle.
    b) The system will run and print out the same thing every time. If so, describe the output.
    c) The system output will depend on the thread scheduler. Describe why.

```
Init()
{
   Lock(Lock1);
   Lock(Lock2);
   printf("Init() is here\n");
}
ThreadA()
{
   printf("ThreadA() is here\n");
   Unlock(Lock1);
}
ThreadB() {
    Lock(Lock1);
    Printf("ThreadB is here\n");
    Unlock(Lock2);
}
ThreadC() {
    Lock(Lock2);
    printf("ThreadC is here\n");
    Unlock(Lock2);
}
```