CSE127 — Final Exam

Yee **Winter '02**

Name and Student ID:_____

There are a total of 13 questions on 10 pages. There are 100 points possible. You might not have time to finish the entire exam — don't be discouraged. Wait until the instructor has passed out exams to everybody before you start. Advice: skim through the entire test to determine which of the problems you can solve quickly and work on those first, rather than getting stuck on a hard problem early and wasting too much of your time on it.

When you can start, you should first make sure that you have all the pages, and write your name and your student ID on the first page, and your student ID on the top of *all subsequent pages*. Pages of this exam will be separated and graded separately — if you fail to write your ID at the top of a page, you will not receive credit for answers on that page. **Write clearly**: if we cannot read your handwriting or your pencil smudges, you will not properly get credit for your answers.

This exam is closed book, closed notes.

No electronic computation aids are allowed.

Prob	1	2	3	4	5	6	7	8	9	10	11	12	13	**Total**
Score														
Poss	5	8	6	7	7	12	3	3	8	11	10	10	10	100

1 (Class basics) What does the compute security code of ethics which you signed say? (You do not need to have it memorized; a paraphrasing is okay.)

(5pts)

2 (Expectation Values) Suppose someone asks you to play a game of chance. You pay $1.00 to play each game, and you can win money depending on the roll of a fair tetrahedral die (4 sided die). Here is the payoff table:

Die value	Payoff
1	$0.00
2	$0.50
3	$1.00
4	$2.00

(1) Calculate the expected payoff for each game. Should you play?

(3) Suppose you find out that the tetrahedral die is *not* fair, and that it is weighted so that the probabilities of landing on each face is given by the following table:

Die value	Probability
1	0.35
2	0.30
3	0.20
4	0.15

Again calculate the expected payoff. Should you play?

(8pts)

3 (Network Security)

(1) What are VPNs? (2) What are they good for? (3) How do they work?

(6pts)

4 (Concepts) What is a Trusted Computing Base?

(7pts)

5 (Terminology) In the formula

$$\forall x : x \in \mathbb{Z} \wedge 0 \leq x < y \rightarrow x^2 < y^2$$

identify which variables are *bound* and which are *free*.

(7pts)

Free:

Bound:

6 (Defense/correctness) Find the loop invariant for the following function which will enable us to prove that it satisfies its specification. (1) Mark where in the code the loop invariant should hold, and (2) use the loop invariant to prove that the code correct.

```
double factorial(int n)
{
        double prod = 1.0; int i;
        for (i = n; i > 0; i--)
                prod = prod * i;
        return prod;
}
```

Specifications:

- precondition:

$$n \geq 0$$

- postconditon:

$$prod = n!$$

Hint: does the loop body run for all inputs? (12pts)

Space for previous problem.

7 (Network Programming) Remote Procedure Calls or RPCs are typically used to implement which programming model:

_____ A. Parent/Child
_____ B. Communist/Capitalist
_____ C. Master/Slave
_____ D. Client/Server
_____ E. Sadist/Masochist

(3pts)

8 (Network Programming) What does the function `ntohl` do?

(3pts)

9 (Network Programming) What do the terms *marshalling* and *unmarshalling* refer to? Explain.

(8pts)

10 (Correctness) What is (1) a precondition? (2) postcondition? (3) loop invariant? Explain clearly what they are and how they are used.

(11pts)

11 (Network) Discuss the following two network topologies in terms of protection for the corporate network and for the web server, which must provide marketing and investing information to customers and investors via the Internet.

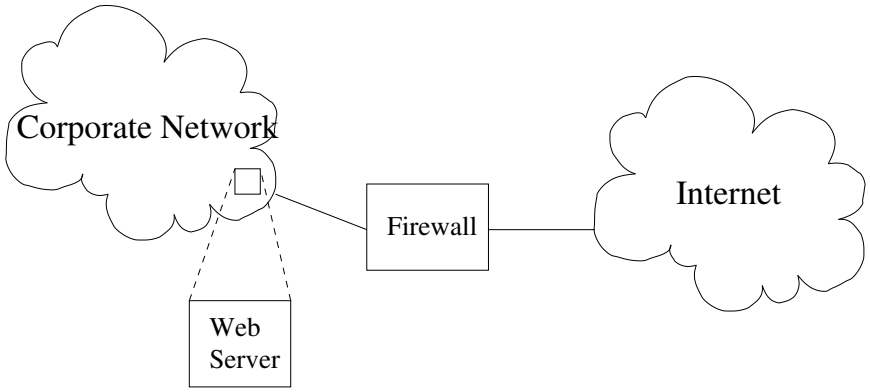

Figure 1: Topology A: The web server is part of the internal network.

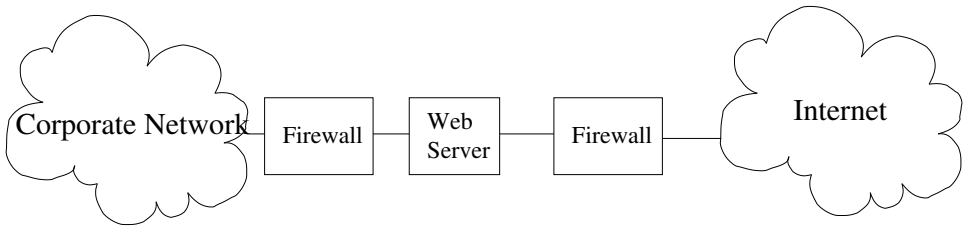

Figure 2: Topology B: The web server is placed on its own network segment between two firewalls.

(10pts)

12 (Terminology/Concepts)

(1) When using virus detection software, what is a "false positive"? Explain what this means.

(2) Do false positives exist in the context of intrusion detection systems? Explain.

(10pts)

13 (Attack) In Differential Timing Analysis (DTA) for the RSA modular exponentiation algorithm, the attacker can determine the value of the hidden exponent value on a bit-by-bit basis. Your security consultant suggests that you should obscure the timing signal by first running the modular exponentiation normally, and then waiting a random amount of time before actually returning a result. Assume that this amount of time is chosen uniformly from 0 to 10 mS, and that the basic modular exponentiation algorithm executes in less than 10 mS.

Explain how DTA works against RSA, and discuss whether or not the consultant's suggestion will help.

```
BigInt modexp(BigInt x, BigInt e, BigInt n)
{
        BigInt y = 1; BigInt z = x;
        while (e != 0) {
                if (e odd) {
                        y = (y * z) mod n;
                }
                z = (z * z) mod n;
                e = e div 2;
        }
        return y;
}
```

(10pts)