Network Programming Exam

June 25th, 2004

Duration of the exam: 2 hours
This is a closed book exam: no documentation is allowed

---

1 Program Output (2 points)

What will be the output of the following program?

```
#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>

int main() {
  if (fork()==0) {
    fork();
    if (execl("/bin/echo", "/bin/echo", "foo", 0) == -1) {
      fork();
    }
    printf("bar\n");
  }
  else {
    fork();
    if (execl("/bin/does_not_exist", "/bin/does_not_exist", "baz", 0) == -1) {
      fork();
    }
    printf("bat\n");
  }
}
```

The `echo` program simply outputs all parameters passed to it. The `does_not_exist` program does not exist.

2 Questions (5 points)

1. What does the `select()` function do? When is it used?
2. What is the difference between an idempotent and a non-idempotent operation? Why is the distinction important when using Sun-RPC?
3. What is MIME? How is it used in the World-Wide Web?
4. How can one write a CGI in Java? Write a CGI in Java which always returns the same "hello world" document, no matter which input parameters it receives. Explain what needs to be done to get it running.
5. You received an email from an e-commerce site requesting you to transfer them a certain amount in payment of an order you made. The email is signed with GPG. This is the first time you make business with this company. How can you use the signature to make sure that the email really originates from this company?

3 A Chess Server (3 points)

We want to build a server where users can play chess. The server must be able to handle multiple games simultaneously. For each game there are two players and any number of spectators. Player can move pieces, whereas spectators can only watch. When a move is made by one of the player, it is very important that the other player sees it immediately. It is also desirable, but less important, that the spectators see moves as quickly as possible.

1. How can you build such a server with sockets? Which kinds of sockets are you going to use? Why? Explain precisely what the server will do whenever a move is made. Do not write full programs, but explain the major operations that the server must make.
2. Certain spectators want to watch the games via a Web interface. How are you going to link the Web server to the socket-based chess server? Which technology are you going to use? Why?
3. Can you let players use the same Web interface as the spectators? Why?

— the end —