Network Programming Exam

May 29th, 2008

This is a closed book exam: no documentation is allowed

---

1 Program Output (1 point)

What will be the output of the following program? In which order will the messages appear on screen?

```
#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 `/bin/echo` program simply outputs all parameters passed to it, plus a new line at the end. The `/bin/does_not_exist` program does not exist.

2 Questions (5 points)

1. Give an example of a race condition. How can it be prevented?
2. Explain how one may write a program that issues read and write operations on several TCP sockets simultaneously, *without using multiple processes or threads.*
3. What are the main differences between a CGI and a servlet?
4. When do you need the help of a certification authority?
5. What is a distributed hash table? What is it useful for?

3 A Road Traffic Information Server (4 points)

We want to build an application that aggregates road traffic information from many different sources, and presents a list of traffic jams to interested drivers. Roads are instrumented with dozens of thousands of sensors installed in different locations, which can measure the average car speed at their location.

As can be seen in Figure 1, each road sensor can be queried through the Internet in order to fetch traffic information from it. As sensors are made by different companies, they do not all have the same interface. Each sensor has one interface out of the following three:

- A Java-RMI interface
- A Sun-RPC interface
- A TCP-socket based interface

The traffic information server is configured so that it knows the list of all road sensors, and which road sensor has which interface.

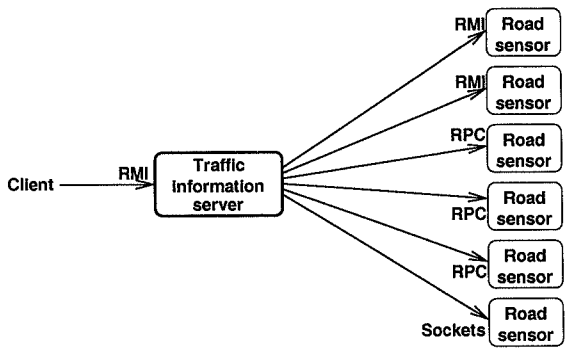

Figure 1: General view of the traffic-jam system

Question 1:

We want to build the traffic information server with a Java-RMI interface which returns the list of current traffic jams. It can be queried frequently, in the order of 10 times per second. It is also important that the server delivers up-to-date information: it should never output information which is more than 5 minutes old.

The traffic information server will need to face two different problems:

1. Be able to fetch information from sensors using different interfaces;
2. Be able to deal with a high number of sensors, while guaranteeing the freshness of the delivered informations.

How should the traffic information server be structured? Present its general architecture and explain why this architecture allows to address the two problems.

Question 2:

We want to allow drivers to report traffic jams using their Internet-enabled mobile phones. When a driver is located inside a traffic jam, he/she can access a Web page and report the location of the traffic jam. The traffic information server considers this as one more source of information to build the full list of traffic jams.

For the moment, we do not take security problems into account.

How should the Web site be designed so that it can be interfaced with the traffic information server? Explain which technology will be used, and how the web server will interact with the traffic information server. If modifications of the traffic information server are required, explain what these modifications are.

Question 3:

Write a sketch of the code of the Web server. Do not waste time writing code to generate nice HTML, but show how the Web server interacts with the traffic jam server.

Question 4:

We want to secure the system to prevent wrong information to enter the system. After careful analysis, we come up with the following potential threats:

1. It is possible for someone to physically disconnect one sensor and replace it with a computer capable of returning wrong information;
2. It is possible to intercept network messages and to modify them to carry wrong information;

3. Drivers can submit wrong information to the Web site. However, if the system has some way to know for sure the identity of drivers who submit information, then we believe that drivers will submit correct information.

We assume that road sensors can use all the security-related tools discussed in class.

Explain which modifications need to be made to secure your system, and why they allow to solve the security problem.

— the end —