Network Programming Exam

May 27th, 2009

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

The `/bin/echo` program simply outputs all parameters passed to it, plus a new line at the end.

2 Questions (5 points)

1. In Sun-RPC, explain what the function `clnt_create()` **does** before one can invoke a remote server.
2. Explain **the reason why** CGIs need to output an empty line before the actual generated content.
3. What is a DOM tree? Show a simple example. What is a DOM tree useful for?
4. Why can't master-slave database replication provide arbitrary scalability to an application, even when given very large numbers of machines?

5. Imagine that a Certification Authority wants to provide its functionality through a client-server interface. Independently of the actual middleware being used for this, what API should it expose to its clients? **Focus only on the most indispensable functionalities**, and ignore all others (such as payment, revocation, etc.)

3 A Travel Agency System (4 points)

We want to build a system so that a travel agency can book flight tickets from any airline company. Each airline company maintains its own reservation system, so that many independent travel agencies can book from them.

The price of a flight can change at any time. When an airline issues a price for a given flight, this price is guaranteed for 30 minutes. After that period, the price may change or the same seat may be sold to another customer.

3.1 Server-side system design

An international airline organization has issued guidelines regarding the way airline companies should expose their functionality to the travel agencies. As a result, a standard Web service-based API has been defined. All airline companies are supposed to implement it. This API comprises two main methods:

- `quote[] getQuote(string originAirport, string destinationAirport, int dateofflight);`
  This method allows one to enter an origin and a destination airport, and the date of the requested flight. The method returns an array of quotes relating to different schedules and/or comfort levels on the plane.

- `confirmation bookFlight(quote selectedQuote, string creditCardNr);`
  This method allows the travel agency to book a flight from a previously requested quote. If payment succeeds, then it receives a confirmation back.

Explain the design of the travel agency's computer system. Which information does the travel agency maintain about each airline company? How is a customer request processed?

Do not describe the interaction between the travel agency and the customer here, this is the topic of the next question.

3.2 Client interaction

A previous programmer has implemented a PHP-based Web interface to allow any Web user to book flights. However, after a few months of operation we notice the following problems:

- Most airlines respond to queries within a few hundreds of milli-seconds. However, some airlines may be temporarily overloaded and be very slow to respond to queries (in the order of 1 minute). Some of them may even never respond.

- Customers do not like waiting for several minutes to get a simple quote. We noticed that many customers leave the site before we finished preparing a quote.

Ideally, we would like to display the quotes to the client as soon as we receive them. After submitting a quote request, the customer should immediately see the list of received quotes, sorted from the cheapest to the most expensive one.

Explain in details how such functionality can be provided: which functionality must be provided at the Web server side? Which functionality must be provided at the Web browser side? How do they interact with each other?

3.3 Load Issues

Thanks to the redesign of the client interface, more and more customers are using our travel agency. This creates additional load at the server side, which in turn means that our system slows down. Additionally, we are planning a major publicity campaign so we expect the system load to be multiplied by a factor 3 in the coming weeks.

How can we accommodate this extra load while maintaining low response times for our customers? Explain which technique(s) can be used for scalability, and how you expect each one to perform in this particular situation.

— the end —