Homework 03

Due: 11:59 pm, Monday March 5

Please handin your homework assignment on gradescope as a PDF file with each problem on a separate page.

Web Security

Problem 1

A common way that web application developers check that user inputs are valid (for example, that they adhere to password length requirements, that emails are properly formatted, etc.) is with javascript executing in the browser. When a form is submitted, the javascript first checks to make sure that the inputs are valid, and refuses to submit the form if they are not. Explain why a web site cannot assume that the input it receives from users will be valid as a result of these checks.

Problem 2

For this problem you may want to look up details online, but make sure to cite your sources. See the collaboration policy for details.

In this problem, you'll explore some of the details of HTTP. In order to do this question, you're going to need to be on a Unix machine (either your own Mac or Linux computer, or ssh'd into the department), and you're going to need to have a program listening on a network port for TCP connections. For example, to listen for connections on port 1234, you can do:

```
$ nc -l -p 1234 # Debian (which runs on the CS department machines)
$ nc -l 1234    # OS X or Ubuntu
```

This will wait for new TCP connections on that port, and will quit once the connection has closed - to handle multiple connections, you'll need to run the command multiple times. Everything that is sent over the connection will be printed to the terminal, and you can respond (send data the other way over the connection) by sending data on `nc`'s stdin (e.g., typing in the terminal).

a) For each of the following user agents, connect to this port using HTTP (i.e., `http://localhost:1234`):

- `curl` (command-line utility)
- `wget` (command-line utility)
- normal browser (Chrome/Chromium, Firefox/Iceweasel, etc; specify which one you use)

For each, answer the following questions:

i. Submit the text that the user agent sends to the server as its request as part of your answer to this homework (if you're using LaTeX, consider the `verbatim` environment).

ii. Describe what each line of the request means

b) What differences do you see between the requests performed by the user agents you tested? Why do this differences happen?

Passwords

Problem 3

A website requires users to provide a password to access a secure area of the site. If a user forgets their password, they can reset it by correctly answering a password reset question.

a) Is this more or less secure than a website that only requires the password (and does not allow resetting via a security question)? Why?

b) Is this more or less secure than a website that requires both a password and a security question to log in every time? Why?

Problem 4

Q needs your help in designing the criminal database system in an effort to identify and target criminals who belong to SPECTRE. To gain access to the database, an MI6 Agent would need to provide some sort of authentication. You have come up with two possible authentication systems:

a) Agents provide one password of length 9 (salted and hashed)

b) Agents provide two different passwords, each of length 8 (salted and hashed using two different salts)

Assuming passwords are alphanumeric (upper case and lower case) and chosen uniformly at random, which system is more secure against an attacker who is capable of hashing 100 million passwords per second? Justify your answer analytically.

Problem 5

In practice, when people choose 10-character alphanumeric passwords, they tend to provide much weaker security than when computers generate 60-bit cryptographic keys. There are 62 possible characters (26 lower-case letters, 26 upper-case letters, and 10 digits) in an alphanumeric password, so there are 62^{10} possible 10-character alphanumeric passwords. There are 2^{60} possible 60-bit cryptographic keys. 62^{10} is approximately equal to 2^{60}, so you might expect them to be equally secure. Why, in practice, is this not the case?