

choose one: Received Received help from:
 (or more) no help Collaborated with: _____

Homework: /SI110/The Cyber Battlefield/Server-Side Scripting

1. A webpage on www.fropjop.com has a form that allows user input to enter data. Entering data results in a script being executed that depends on this data. The script loops forever (infinite loop) if the user provides bad input. Who gets harmed by the bad input: the user who entered the data, or the webserver owner of the www.fropjop.com, if the script is ...

- a. ... "client-side" ? **Circle one:** user webserver owner 5/4/3/0
- b. ... "server-side" ? **Circle one:** user webserver owner 5/4/3/0

2. Fill in the missing pieces of the HTML below, based on the following: You go to the page, enter **John** in the first box, enter **Doe** in the second box, then press the "Look Up Name" button, which results in the following URL being visited:

<http://rona.cs.usna.edu/pblu.jsx?fn=John&ln=Doe>

```
<html>
  <head></head>
  <body>
    <form name="lookup" onsubmit="return false"
      action="_____ ">
      First Name: <input type="text" name="_____">
      Last Name: <input type="text" name="_____">
      <input type="button" onclick="submit()" value="Look Up Name">
    </form>
  </body>
</html>
```

30/24/18/0

3. Go to the URL: <http://rona.cs.usna.edu/~si110/lec/l14/stfin.html> , enter some numbers, and click the "process" button. Normally, you get a nice message that your input has been processed. It so happens that the server script that the form submits to will crash and cause a server error if it receives -999 for both input values. For that reason, stfin.html does some client-side input validation to ensure that it never submits -999 for both values. Find a way to submit -999 to this server side script none the less!

- a. What is displayed in your browser when this happens. 10/0/0/0
- b. Describe exactly what you did to send this bad input to the server despite stfin.html's input validation. 20/15/10/0

4. Consider this HTML file:

```
<html>
  <head></head>
  <body>
    <form name="login" onsubmit="return false"
      action="http://rona.cs.usna.edu/cgi/login.jsx" method="get">
      Username: <input type="text" name="un" >
      Password: <input type="password" name="pw">
        <input type="button" value="login" onclick="submit()">
    </form>
  </body>
</html>
```

a. You enter username "jonesy", the password "egbdf88", and press the "login" button. What URL gets visited as a result?

15/12/8/0

b. Why is it that submitting a password to a website using forms and the "get" method like this puts users' passwords in danger?

15/12/8/0