

Scripting Languages G22.3033-002 Summer 2008

Example solutions for hw08

Assigned We 7/16/2008, due Fr 7/25 at 9pm. 50 points.

<http://www.cs.nyu.edu/courses/summer08/G22.3033-002/>

Instructions for example solutions

These are example solutions. Please keep in mind that often, there is not just one correct solution to a question. If you come up with different answers, then it may be that both your answers and these answers here are correct. Of course, these answers here may also contain mistakes. If you spot a mistake, please let me know so I can correct it.

Example solutions for Homework 8

solutions-hw08-1 Same Origin Policy

```
result    motivation
1 success same domain+protocol+port
2 success same domain+protocol+port
3 failure different protocol ftp
4 failure different domain www.columbia.edu
5 success same domain+protocol+port (port 80 is default)
6 failure different port 8080
7 failure same domain nyu.edu but different server www2
```

solutions-hw08-2 SQL Injection Vulnerability

- a. Let's assume that the variable `$name` comes from user input, for example, from `$_GET['name']`. Then, the attacker can provide the following input:

```
foo'; drop table custid; --
```

This is the same input as on the slides. If embedded in SQL, it will end the current string and statement, then start a new statement that destroys a table.

- b. PHP already provides a sanitization function for this purpose, called `addslashes`. It is easy to reimplement from first principle:

```
function my_addslashes($str) {
    $badchar = array("\\", "'", '"', "\0");
    $replace = array();
    foreach ($badchar as $c) array_push($replace, "\\\" . $c);
    $result = str_replace($badchar, $replace, $str);
    return $result;
}
```

solutions-hw08-3 Cross-Site Scripting

Somewhere along the way of information flowing from user input to HTML output, the data must be sanitized. PHP already provides the function `htmlspecialchars` for this purpose. If written from first principle, it looks a lot like the SQL sanitization function from the previous question:

```
function my_htmlspecialchars($str) {
    $badchar = array("&", "'", '"', "<", ">");
    $replace = array("&";, """;, "'";, "<";, ">");
    $result = str_replace($badchar, $replace, $str);
    return $result;
}
```