用 html语言编写一个简单的计算器,跟着crownless学Web之（1）神奇的计算器

转载

清水荷叶粥 于 2021-05-31 15:36:04 发布 309 收藏
文章标签： 用 html语言编写一个简单的计算器
原标题：跟着crownless学Web之(1)神奇的计算器

简介：

本文由Web安全版块的新版主crownless原创讲解。下文将以CTF赛题讲解的形式为大家介绍一系列的Web基础知识，讲解的顺序将是循序渐进，因此不需要读者有任何基础知识。希望能吸引更多人关注看雪的Web安全版块，为版块增加人气和活力。

Web安全版块链接：

https://bbs.pediy.com/forum-151.htm

在这篇文章中，你将学到：

1、一点HTML语言

2、什么是前端和后端

3、HTTP GET请求

4、一点php语言

5、基础的shell命令

话不多说，让我们开始吧。首先打开这次的CTF赛题网站：

http://139.224.220.67:30005

你将看到如下界面：


```
<center>
<h2>calc</h2>

<?php
$str="";
if(!empty($_GET)){
    $str=$_GET["calc"];
}
?>
<form action="./index.php">
input: <input type="text" name="calc" value="<?php echo $str;?>">
  <input type="submit" value="Submit">
</form>

<?php
    echo "result:".shell_exec("echo $str | bc");
?>
</center>


<?php
show_source(__FILE__);
```

对于刚接触Web安全甚至是Web编程的你来说，这个界面或许让你有些困惑。不要紧，请耐心仔细地阅读我的讲解，当你读完这篇文章后一定能对Web安全有一个基本的了解。

首先，页面上方居中部分是一个大标题calc，接着有一个输入框(左边有input:文本)、一个Submit按钮、以及result:文本。

这个界面和我们平时在各种网站上见到的界面很像，可以说是最简单的网页界面之一。那么，这个界面是如何实现的呢？事实上，这是用HTML编写的。HTML即HyperText Markup Language，中文翻译是“超文本标记语言”。

接下来，我就教你怎么编写如上的界面。何谓编写界面？就是通过编写HTML语言源代码来达到我们想要的界面效果。可能细心的你已经发现，网页下方已经有了一堆源代码。正如你猜想的那样，这是我为了方便你的学习而特意将源代码显示在了这里。但是不要急着把这些代码直接拷贝出来，因为里面内嵌了一些php语言的代码。我们要做的是删除彩色的php代码。那么，我们可以得到如下代码：

calc

inpu t:< inputtype= "text"name= "calc"value= "">

< inputtype= "submit"value= "Submit">

center>

将其保存为1.html，并用浏览器打开1.html，你会发现除了最后一行的result:文本消失了之外，其他部分和CTF赛题网站上的界面是相同的。result:文本之所以消失，是因为我们删除了包含result:的彩色的php代码。恭喜你，你已经完成了人生中的第一张网页！但是别高兴得太早，接下来我来讲解一下1.html的结构。

HTML语言的特色是拥有数量众多的“标签”。何谓“标签”？暂时你可以理解为类似于bbb这样的东西。标签的功能多种多样。以1.html为例，

标签可以让其包裹的内容在网页中居中显示。

标签定义了一个大标题。

标签定义了供用户输入的HTML表单(表单的作用在后文中会讲解)。标签定义了输入控件(type="text"代表输入框，type="submit"代表提交表单的按钮)。

好了，现在你已经知道怎么用HTML编写一个简单的网页了，但是你想问：在1.html中，这个Submit按钮点上去似乎没什么用嘛！不像在CTF赛题网站上，在文本框里输入1 + 2，result就会显示为3。(事实上，这就是为什么这道题叫做“神奇的计算器”。)

这里，就涉及到了前端和后端的概念。事实上，你刚才编写的1.html正是前端代码。一般流行的前端技术有HTML、CSS、Java。HTML用来构建网页的基本框架，CSS用于美化网页，Java用于为静态的HTML网页增加动态的交互行为。

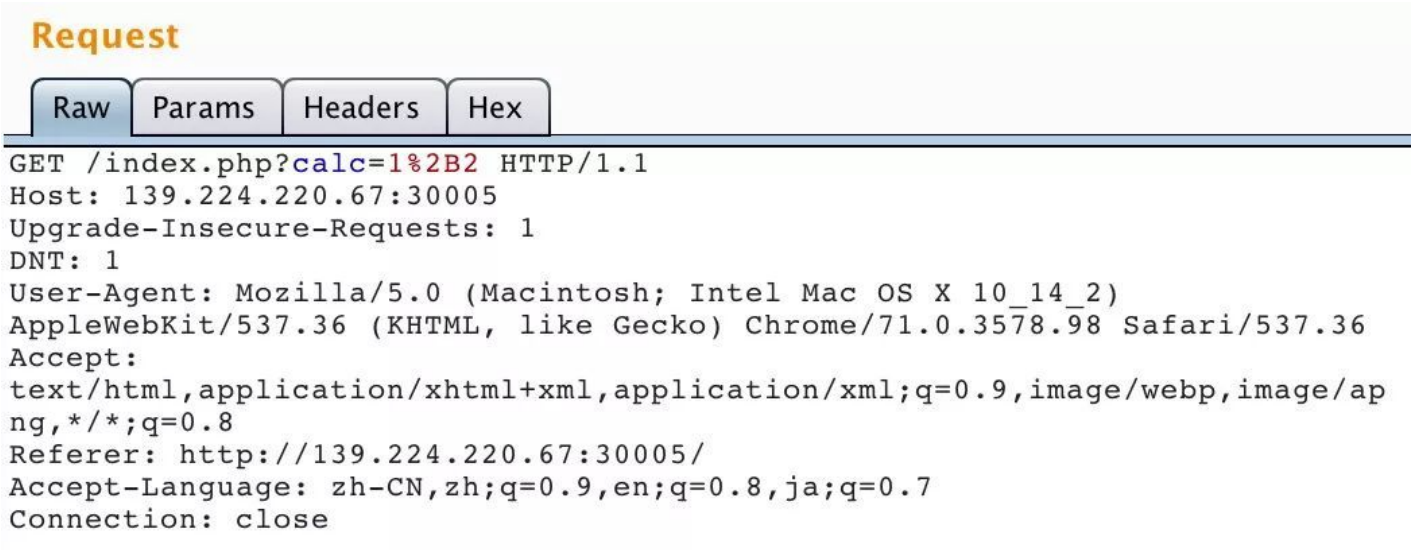
前端代码是运行在浏览器上的。比如说，你打开1.html时，浏览器会读取1.html的内容，并将其转化(行话叫“渲染”)成图像显示在显示器上。而后端代码是运行在服务器上的。比如你刚才在文本框里输入1 + 2并点击Submit，即是将1 + 2这个字符串传递到了./index.php(可以从

标签的action属性看出来)。

./index.php即http://139.224.220.67:30005/index.php的简写。服务器上运行的php程序接受到你的1 + 2字符串，调用shell_exec函数获得执行结果3，再嵌入网页形成“动态网页”发送回你的浏览器，你的浏览器渲染“动态网页”显示在屏幕上，这样你就看到了3。综上所述，我们称./index.php为一个后端，因为它能根据用户在前端的输入生成“动态网页”。

那么用户的输入是怎么从浏览器发送到后端的呢？事实上，这里使用的是HTTP协议，全称HyperText Transfer Protocol(超文本传输协议)。当你点击Submit按钮时，浏览器就会找到包含这个按钮的表单元素，即

，并找到这个表单元素的method属性(如果没有，那么默认为GET)。如果method属性是GET，那么就发送一个HTTP GET请求，其中包含着浏览器传给后端的参数。比如，当我们在输入框里输入1+2并点击Submit时，将会发送以下HTTP请求给后端：



可以看到，calc参数的内容是1%2B2，事实上这是对1+2的URLEncode(URL编码)。为什么是通过calc参数传递1%2B2呢？事实上，这是在输入框标签的name属性约定的。也就是说，在这个输入框中输入的内容都将被作为calc参数发送给后端。

当index.php后端接受到任何HTTP请求后，都会顺序执行index.php这个脚本内的任何内容。如果遇到了HTML代码，那么原封不动地将其返回给浏览器。如果遇到了php代码，那么就执行php代码并将结果返回给浏览器。那么后端是怎么从HTTP GET请求中取得这个参数的呢？接下来，让我们分析一下php代码。

```
$str="";

if(!empty($_GET)){

$str=$_GET["calc"];

}

?>
```

这段代码的意思是，首先将str变量赋值为空字符串，然后检查$_GET变量是否为空(empty函数的文档)，如果不为空，那么将calc参数的内容赋值给str。事实上，$_GET是一个数组变量，$_GET["calc"]中存放着从HTTP GET请求中获得的calc参数。

接着再看如下php代码：

echo的作用是将str变量写入输出。也就是说，你发送给后端的1+2参数会被后端原封不动地嵌入输入框标签的value属性。这样你点击Submit后，输入框里就会出现1+2，而不是空的输入框。

接着再看最后一段php代码：

```
echo"result:".shell_exec("echo $str | bc");

?>
```

首先，输出一个result:，然后这里使用了.运算符。在php中，这代表字符串连接。然后，执行shell_exec函数。这个函数的作用是“通过shell环境执行命令，并且将完整的输出以字符串的方式返回”。那么执行的是什么命令呢？很明显，是echo $str | bc。echo在shell中的作用和在php中的作用一致，比如你在shell中执行echo crownless，你会得到crownless。而bc是一个简易的计算器程序。在你的shell中试试看执行bc指令，再输入1+2并按回车，你会得到3。在echo和bc当中有一个|符号，这是shell中的管道符号。使用管道，就相当于将前一个程序的输出传递给下一个程序。在你的shell中试试看echo 1+2 | bc，你将会得到3。

这里再介绍两个常用的shell命令：

ls可以列出当前目录下的所有文件。

cat可以打印一个文件的所有内容。

好了，聪明的你一定想到了我们可以通过控制str参数来实现任意命令执行。首先，我们要尝试闭合echo语句。那么我们就要用到;符号。好，现在我们可以给str传入;，那么将会执行shell命令echo ; | bc。但是我们还要想个办法让;后的内容失效。

这里我们就要使用shell下的单行注释符号#。也就是说，echo ; # | bc的作用相当于echo ;。那么现在我们就可以在;和#之间插入任意我们想要注入的命令了，如ls;。试试看在输入框中输入; ls; #，并按Submit，我们发现目录下有一个there_1s_4_fl4g文件。我们再输入; cat there_1s_4_fl4g; #，我们就可以得到这道CTF题的flag。

怎么样，通过一道题，我们学会了Web的基础知识，很有成就感吧？敬请期待crownless的后续作品。

作者：周信安

看雪ID:crownless，毕业于复旦大学软件工程专业，看雪WEB安全版块 新版主，研究方向为WEB安全、Android、系统安全。

个人博客地址：

https://zhouxinan.github.io

- End -

看雪ID：crownless

https://bbs.pediy.com/user-833800.htm