

在 Web 网页中设置口令

陆慧凤 (北京石油管理干部学院计算中心 100085)

邵庆 (大庆石油学院 151400)

摘要:本文详细介绍了如何在 Web 网页中设置口令。文中首先给出了利用 Javascript 设置口令的一般原理,然后介绍了采用模块化的方式隐藏口令字,最后给出了设置多帐号的方法。

关键词: Javascript 网页 口令 因特网

随着因特网的普及,享受网络上丰富的信息的同时,也在努力将自己掌握的信息从网页上发布出去,与网友们共享。Web 网页是一种开放式的信息窗口,一般来说发布到网页上的内容可被所有上网者阅读。但有时网页发布者并不希望自己的某些信息被所有的人共享,而只提供给那些被授权的用户。一种常见的形式是在网页中向用户提示一个口令输入窗口,用户只有在输入正确的口令后才能打开对应的网页,这样为网页起到了保密的作用。

给网页加口令常用的方法是通过 CGI 编程来实现,但这一方法逐渐被 Javascript 或 Java 编程取代,因此本文只讨论如何用 Javascript 在 Web 网页上加口令。

1. 利用 Javascript 加口令的一般原理

从本质上看, Javascript 是一组编程命令,它们被用于增强对网页的操作,通过将 Javascript 的命令编入一个

网页,可以根据程序改变页面外观,根据系统需求对主页的内容进行动态插入和删除。 Javascript 是一种解释性的语言,它继承了很多 Java 语言的传统,与之非常相似。Java 与其他语言最大区别之处在于它与 WWW 的结合方式,不象一个 CGI 脚本或 Java applet,它们都是单独的文件,而 Javascript 代码则是标准的 HTML 文档的组成部分,就象其他的 HTML 标记和元素一样,是 Web 主页制作者所熟悉的 HTML 语言的扩展。

一般情况下,设计高级 Web 页面并不需要直接进行 Javascript 编程,而用所见即所得的可视化网页制作工具进行设计,如使用 FrontPage 设计的网页基本上体现了 Javascript 的功能所在,但在进行交互处理时 FrontPage 就不行了,而需要直接编写 Javascript 代码,一般的方法是用 FrontPage 进行基本外观制作,再对制成的网页源代码即 Javascript 代码进行细致设计。

下面是一个通过口令访问网页的例子,其中前5行是 FrontPage 自动生成的。

```
// - - - - page1.htm - - - -
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<title>New Page 1</title>
<meta name="GENERATOR" content="Microsoft FrontPage 3.0">
</head>
```

```
Please Input Your Password:<P>
<FORM METHOD=POST ACTION="page3.htm" NAME="PASSWD"
onSubmit="if (document.passwd.password.value!= '1234') {
    alert('invalid password, try again!');
    return false;
} else {
    return true;
}">
<INPUT TYPE="password" NAME="password"
SIZE=10>
<INPUT TYPE="submit" VALUE="OK">
</FORM>
```

在 page1.htm 中, <HEAD></HEAD> 是网页头, 由 FrontPage 自动生成, 标记 <FORM> 用来定义窗体, 一个 FORM 中可以指定多个属性和方法。其中:

属性 METHOD 用来指示如何向服务器提交窗体数据, 这些数据将附于属性 ACTION 所指定的网页上。

属性 ACTION 用来指定要激活的网页。

属性 NAME 用来指定本窗体的名字。

onSubmit 指定当用户单击提交按钮, 要调用的事件处理程序。当用户按下后面定义的提交按钮或在密码域中按下回车后, 开始执行引号中的程序段。该程序段首先检查密码域中的值是否为指定的值"1234", 如果不是则用函数 alert 弹出一个对话框, 并显示警告信息, 返回假值 false, 这样 FORM 将不激活属性 ACTION 指定的网页。如果密码正确, 则返回真值 true, 这样 FORM 将调入 ACTION 指定的网页。

此外, 在 FORM 中还可以包括若干个输入域: 单行

文本域、多行文本域、单选按钮、复选框按钮、隐藏域、密码域和选择菜单。

语句 <INPUT TYPE="password" NAME="password"> 中, TYPE="password" 表示在当前 FORM 中定义的域是一个密码域, NAME="password" 表示该域的名字是 password, 在其他地方的 JavaScript 程序用该域名作为对象名来获取密码值, SIZE=10 表示该域的显示长度为 10 个字符长度。密码域是一个特殊的单行文本域, 当用户在密码域中输入字符串时, 输入的字符并不显示出来, 而是将每个输入字符显示为星号(*)。用户输完字符串并回车后, 程序将转到 FORM 语中 onSubmit 定义的事件处理程序。

语句 <INPUT TYPE="submit" VALUE="OK"> 中, TYPE="submit" 表示在当前 FORM 中定义的域是一个提交按钮, VALUE="OK" 表示在按钮上显示的字符串为"OK"。当用户按下提交按钮后, 程序将转到 FORM 语中 onSubmit 定义的事件处理程序。

可见, 使用 JavaScript 编写网页, 可以轻松解决网页加口令的问题。

2. 隐藏口令码

以上程序我们只是说明给网页加口令的基本原理, 直接使用该程序会有一些问题。这是因为所有的 WWW 浏览器都提供了阅读当前网页源码的功能, 当用户调出本网页后, 用浏览器提供的网页源码阅读功能, 可以看到以上源代码, 从而可以分析出所设的口令和将要调出的网页文件名。为此, 我们将程序可以将 page1.htm 改为 page2.htm, 如下:

```
// - - - - page2.htm - - - -
<HTML>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
<title>New Page 1</title>
<meta name="GENERATOR" content="Microsoft FrontPage 3.0">
</head>

<SCRIPT LANGUAGE="JavaScript" SRC="page2.js">
</SCRIPT>
Please Input Your Password:<P>
<FORM METHOD=POST NAME="PASSWD"
onSubmit="return checkpasswd()">
<INPUT TYPE="password" NAME="pass-
```

```
word" SIZE = 10>
  <INPUT TYPE = "submit" VALUE = " OK ">
</FORM>

</HTML>
```

在 page2. htm 中, 标记 < SCRIPT > 定义了一个 JavaScript 程序 page2. js, 但 page2. js 的代码并不出现在网页中, 因此用户无法阅读密码和要调用的网页文件名。在 page2. js 中, 定义了一个函数 checkpasswd, 该函数用来设置 FORM 的 ACTION 属性值, 即要调用的网页文件名, 并检查密码域中的口令是否正确。page2. js 的代码如下:

```
// - - - - page2. js - - - -
function checkpasswd() {
  this. document. passwd. action = "page3. htm";
  if ( this. document. passwd. password. value! = '
1234') {
    alert('invalid password, try again!');
    return false;
  } else {
    return true;
  }
}
```

3. 设置帐号

上面介绍的口令设置中, 只有对口令的检验, 没有对用户的核对, 也就是说, 要保护的网页对所有用户都设为相同的口令。但在某些情况下, 需要对不同的用户设置不同的口令。如根据不同用户进入的不同的网页, 防止口令传播等。这样就要设置一组帐号, 即设置一组用户, 并为每个用户设置不同的口令。

设置帐号可以采用两种方式, 一种是直接编程方式, 一种是口令库方式。

直接编程方式与前面的方法相似, 但要加上用户名的输入和合法性检查。

将 page2. htm 改为:

```
...
<FORM METHOD = POST NAME = "PASSWD"
  onSubmit = "return checkpasswd()">
  UserName: <P>
  < INPUT TYPE = "text" NAME = "username"
SIZE = 20>
  Password: <P>
```

```
< INPUT TYPE = "password" NAME = "pass-
word" SIZE = 10>
  < INPUT TYPE = "submit" VALUE = " OK ">
</FORM>
```

将 page2. js 改为

```
// - - - - page2. js - - - -
function checkpasswd() {
  var theUserName = this. document. passwd. user-
name. value;
  var thePasswd = this. document. passwd. password.
value;
  this. document. passwd. action = "page3. htm";
  if (theUserName == '用户名 1' && thePasswd =
= '口令 1'
    || theUserName == '用户名 2' && thePasswd
== '口令 2'
    //可以根据需要在此用||联接多组用户名和口令
字
  ) {
    alert('invalid password, try again!');
    return false;
  } else {
    return true;
  }
}
```

用口令库的方式是: 事先将用户名和口令字存入文件或数据库中, 在函数检查时, 从口令库逐读出每个用户名和口令字, 并与输入的用户名和口令字进行比较。这需要使⽤ Java 编程, 或 CGI 编程。由于篇幅有限, 在此不多介绍。

4. 结束

从以上介绍可以看出, JavaScript 为我们提供了简洁方便的网页保密技术。只要将 page2. htm 和 page2. js 中个别参数根据实际情况作一下修改, 把 page2. htm 加到用户网页的适当地方, 就可以直接应用。

参考文献

- [1] 王玉琳译. Web 程序设计教程, 电子工业出版社, 1997, P339396.
- [2] 廖卫东, 陈梅. Java 程序设计, 机械工业出版社, 1996.

(来稿时间: 1998 年 12 月)