Cross Site Scripting Prevention

CYBER SECURITY &
PRIVACY FOUNDATION

HTML entity encoding

Replace the HTML characters with HTML Character Entities, before inserting the Input between HTML tags(Ex: Untrusted data).

Character	HTML Entity
<	<
>	>
"	"
‘	'
/	/

Escaping HTML Characters with php “htmlspecialchars()” function

```php
<?php
$in=$_GET['input'];
$filteredInput=htmlspecialchars($in, ENT_QUOTES, "UTF-8");
…
//process the request
..
?>
```

The Result:

```
<br>
No results found for &lt;script&gt;alert(1)&lt;/script&gt;<br>
```

User Input to HTML attributes

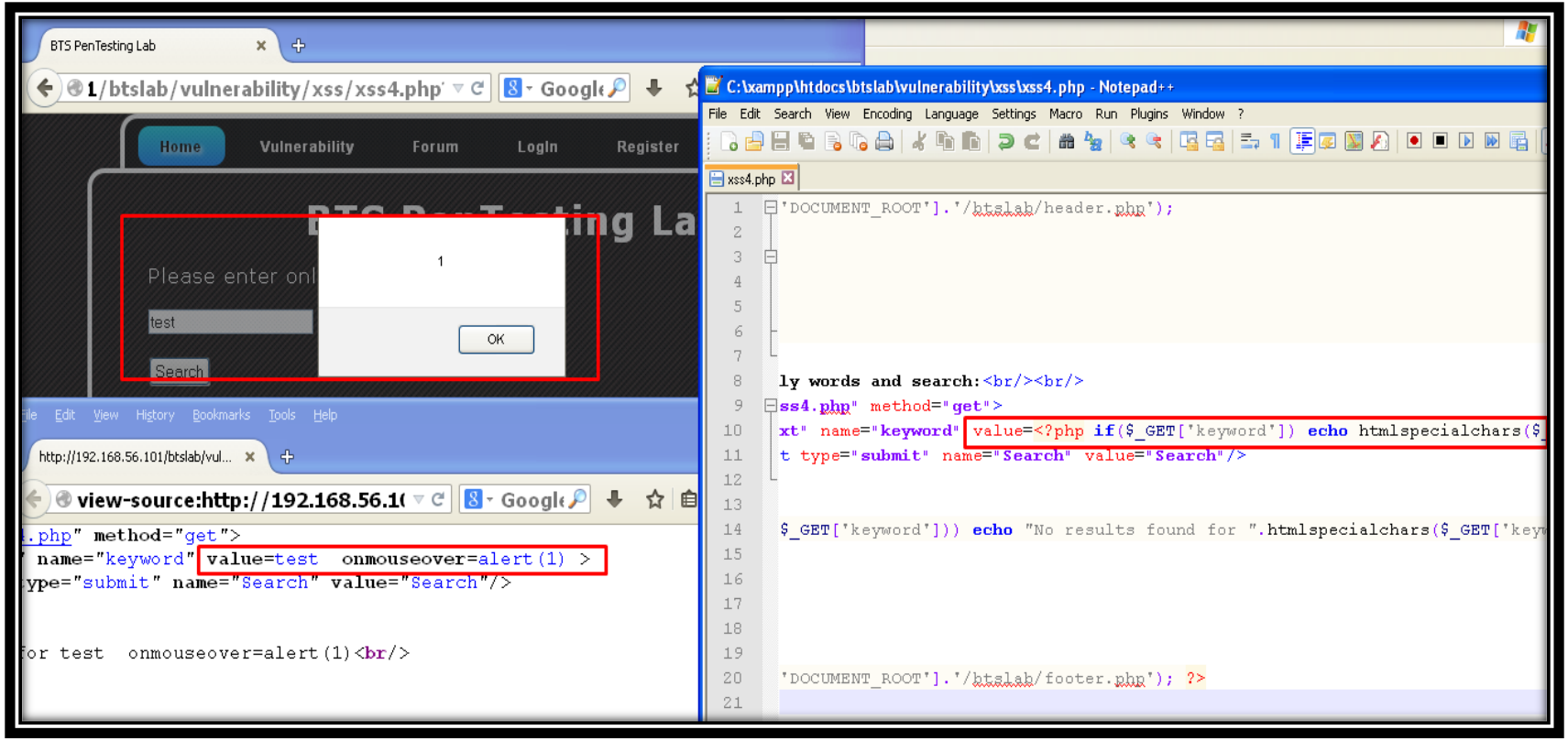
- Filter the User Input before inserting into the HTML attributes such as width, name, value attributes.
- Except for alphanumeric characters, escape all characters with ASCII values less than 256 with the &#xHH; format (or a named entity if available)

```
$userin=$_GET['search'];
$filteredin=htmlspecialchars($userin,ENT_QUOTES,'UTF-8');
?>
<input type='text' name='key' value="<?php echo $filteredin; ?>" />
```

Unquoted HTML attributes

- htmlspecialchars() function escapes only HTML special characters(<,’,”,>,&) .
- It doesn’t escape other special characters(eg: ‘(‘) .
- So, if you are filtering the input with htmlspecialchars passed to **unquoted** html attributes, it will be still vulnerable.
- You have to filter other special characters and do HTML attribute escaping
- **So, It is better to avoid using unquoted HTML attributes.**

Unquoted HTML attributes



HTTP Only Flag

- Restricts Client side scripting(javascript) from accessing the protected cookies.
- HTTPOnly flag tells the browser that the cookie should only be accessed by the server.
- All modern browsers supports HTTPOnly flag.
- Does not prevent XSS, but help to reduce the impact of XSS

Enabling HTTPOnly flag in PHP.ini file:

Find the “session.cookie_httponly =“ setting and change the value to “True”