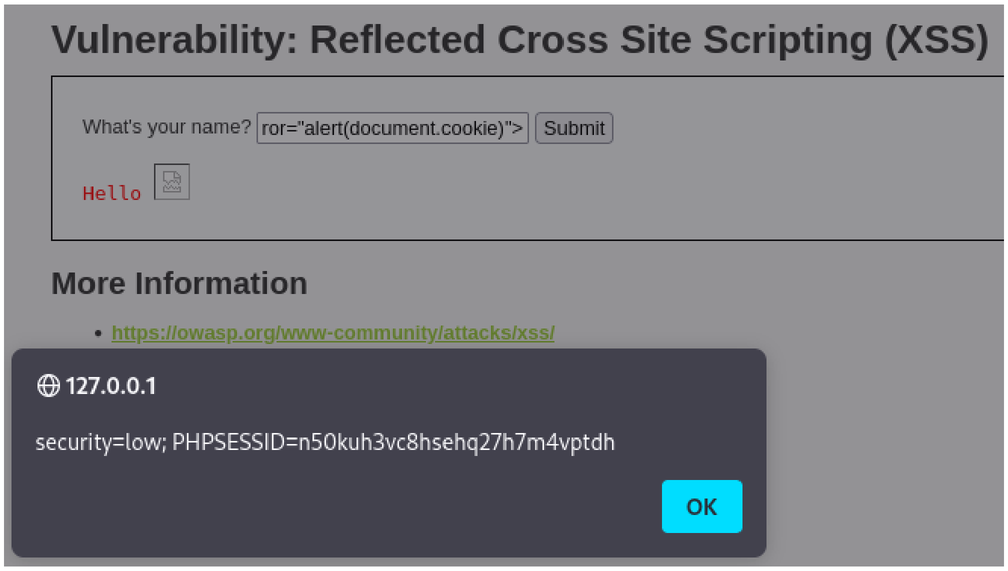
XSS(Reflected)

- The name field which is reflecting on page

payload= `<img src=x onerror="alert(document.cookie)">`

- It triggers an alert pop up with cookie value


Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?
ror="alert(document.cookie)">
Submit
Hello
More Information
https://owasp.org/www-community/attacks/xss/
127.0.0.1
OK

security=low; PHPSESSID=n50kuh3vc8hsehq27h7m4vptdh

- It also worked with low, medium and high level

Vulnerable code

1. **Low**

```
<?php
```

```
header ("X-XSS-Protection: 0");
```

```
// Is there any input?
if( array_key_exists( "name", $_GET ) && $_GET[ 'name' ] != NULL ) {
// Feedback for end user
echo '<pre>Hello ' . $_GET[ 'name' ] . '</pre>';
}
```

```
?>
```

- On the server side, the code does not check if the user is attempting a XSS injection. It simply pastes the user input in the HTML code

- A first line of defense would be removing the script tags

2. **Medium**

- On the server side, the code uses `str_replace` to remove all instances of the string `<script>` but fails to take into account that `<Script>` or `<sCRipt>` or `<scriPt>`,... are also valid tags !
- A better approach would have been using the function `str_ireplace` that is case-insensitive.

```
<?php
header ("X-XSS-Protection: 0");
// Is there any input?
if( array_key_exists( "name", $_GET ) && $_GET[ 'name' ] != NULL ) {
// Get input
$name = str_replace( '<script>', '', $_GET[ 'name' ] );
```

```
// Feedback for end user
echo "<pre>Hello {$name}</pre>";
```

```
}
?>
```

3. **High**

- The code protects itself from `script` tags but fails to take into account other tags. A proper way of sanitizing HTML input is to use a function like `htmlspecialchars`

```
<?php
header ("X-XSS-Protection: 0");
// Is there any input?
if( array_key_exists( "name", $_GET ) && $_GET[ 'name' ] != NULL ) {
// Get input
$name = preg_replace( '/<(.
)s(. )c(. )r(. )i(. )p(. )t/i', '', $_GET[ 'name' ] );
```

```
// Feedback for end user
```

```
  echo "<pre>Hello {$name}</pre>";
```

`}`

`?>`