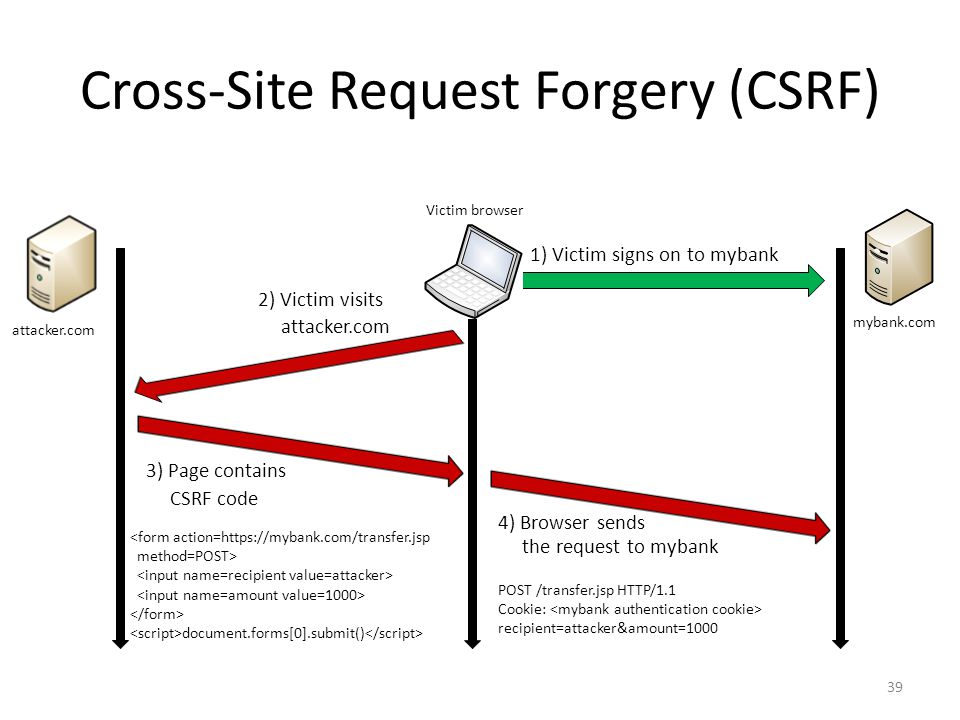
Cross-Site Request Forgery (CSRF) Attack Using GET Request

(Aka: Session Riding)

|  |
| --- |
| Copyrights 2016-2017 Frank Xu, Bowie State University.  The lab manual is developed for teaching cyber-related courses based on http://www.cis.syr.edu/~wedu/seed/Labs\_12.04/Web/Web\_CSRF\_Elgg/Web\_CSRF\_Elgg.pdf. Comments and suggestions can be sent to wxu@bowiestate.edu |

**Introduction**



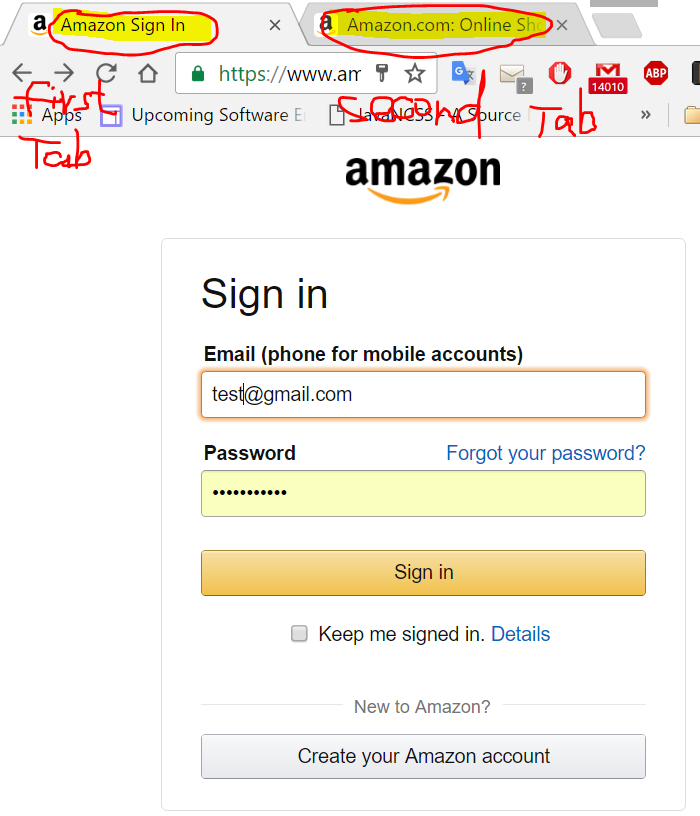
**Lab Environment**

We have created two accounts in the VM. The usernames and passwords are listed in the following:

* + User ID: root, Password: *seedubuntu*.
  + Note: Ubuntu does not allow root to login directly from the login window. You have to login as a normal user, and then use the command **su** to login to the root account.
  + User ID: seed, Password: *dees*

**Task 1: Understand all Tabs of a Browser Share the Same Session ID**

1. Open Amazon.com and view your orders. Amazon asks your user name and password.



1. Questions:
   1. If you open second tab in the same browser and try to check your orders in Amazon, do you have to retype your user name and password? Why?
   2. If you log out in the first tab and try to check your orders in Amazon in the second tab, do you have to retype your user name and password? Why?

**Task 2: Demonstrate CSRF Attack Using GET Request**

<http://www.csrflabelgg.com/> is social website. **Alice** and **Bob** both are registered users. All members of the website can see others’ names and add others as his/her friends. However, the friend relation is unidirectional, e.g., **Bob** added **Alice** as a friend means **Bob** treats **Alice** is a friend, however, it doesn’t mean **Alice** treats **Bob** as a friend. **Bob** wants **Alice** add him as a friend, but **Alice** refuses to do so. **Bob** tricks **Alice** to add him as a friend without **Alice’s** permission using the CSRF attack.

In the task, pretend that you are **Bob**, the task describes how you can construct the content of the web page, so as soon as **Alice** visits the web page, **Bob** is added to the friend list of **Alice** (assuming **Alice** has an active session with Elgg)

Step a: The victim user (**Alice**) holds an active session with a trusted site [**http://www.csrflabelgg.com/**](http://www.csrflabelgg.com/)

Step b: **Bob** injects an HTTP **GET** request in the malicious site [**http://www.csrflabattacker.com**/](http://www.csrflabattacker.com/) owned by **Bob**

Step c: **Bob** lures **Alice** to visit the malicious site [**http://www.csrflabattacker.com**/](http://www.csrflabattacker.com/)

Step d: **Alice** is lured to visit the malicious site.

The results**: Alice’s** visit causes damages (**Bob adds Alice as a friend without Alice’s knowledge**)

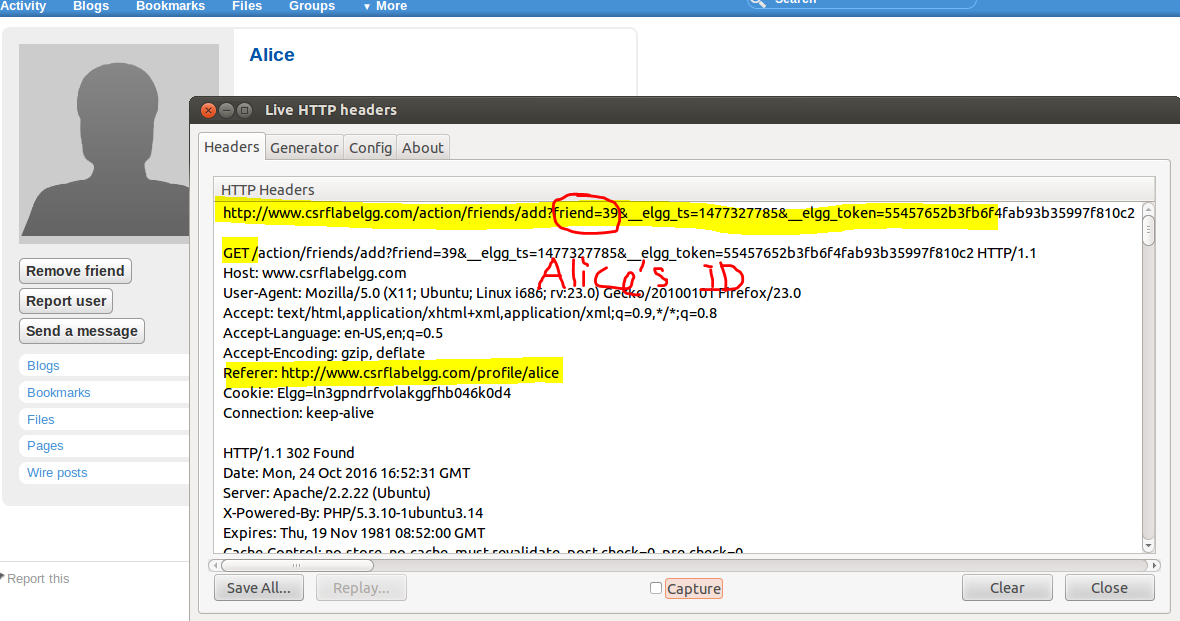
1. Understand a normal friend request HTTP header information. Note that we can use the scenario of either Bob-sends-Alice request or Alice-sends-Bob request to understand friend request’s header information. For the demonstrate purpose, we use the scenario of **Bob-**sends-**Alice** friend request.
   1. Start apache web server



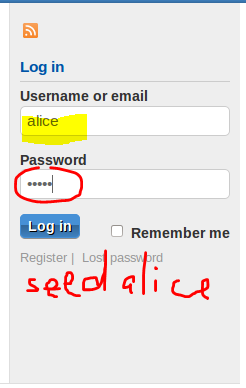
* 1. Bob sends a friend request to Alice

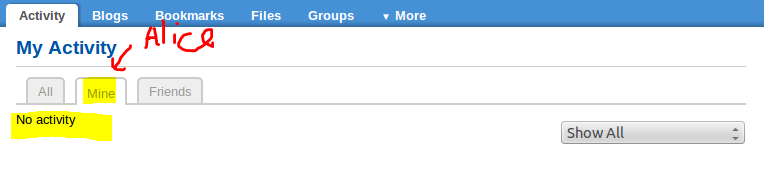
|  |  |
| --- | --- |
|  | |
|  |  |
|  | |
|  | |

* 1. Bob sends normal request is captured using Live HTTP Header

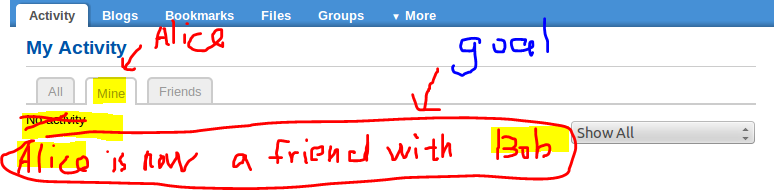


* 1. Question: Use the scenario of Alice-sends-Bob friend request to find ID of Bob

1. Understand the Goal of Bob
   1. Log in as **Alice**



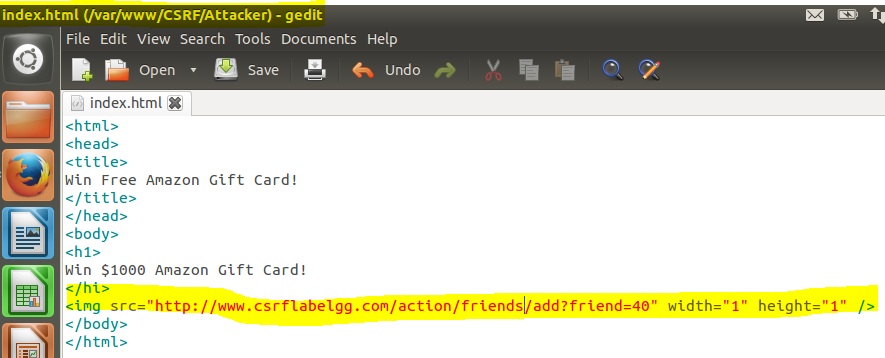
* 1. The goal is to show “Alice is now a friend with Bob”



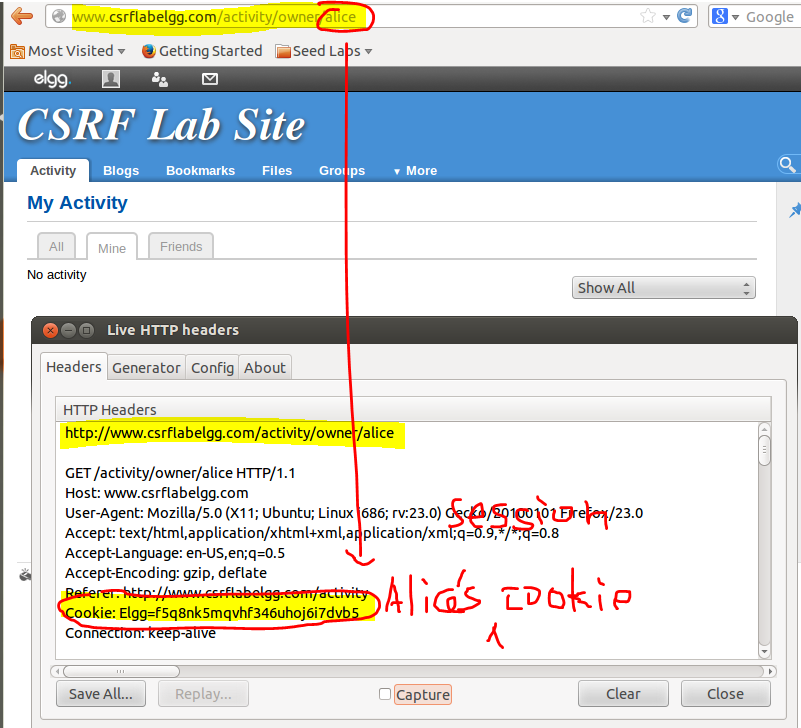
* 1. Technically, to achieve the goal, **Bob** needs to lure **Alice** to click the link <http://www.csrflabelgg.com/action/friend/add?friend=40>

1. Understand how **Bob** lures **Alice** to “click” the link <http://www.csrflabelgg.com/action/friends/add?friend=40>
   1. Edit a malicious page

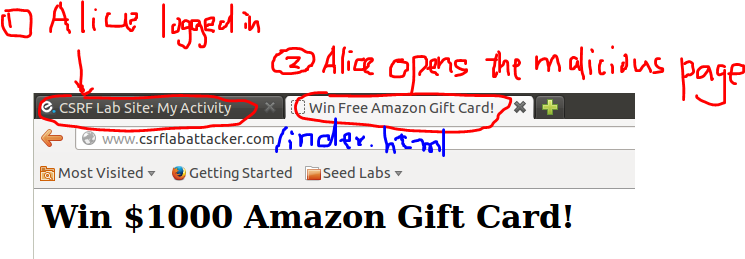




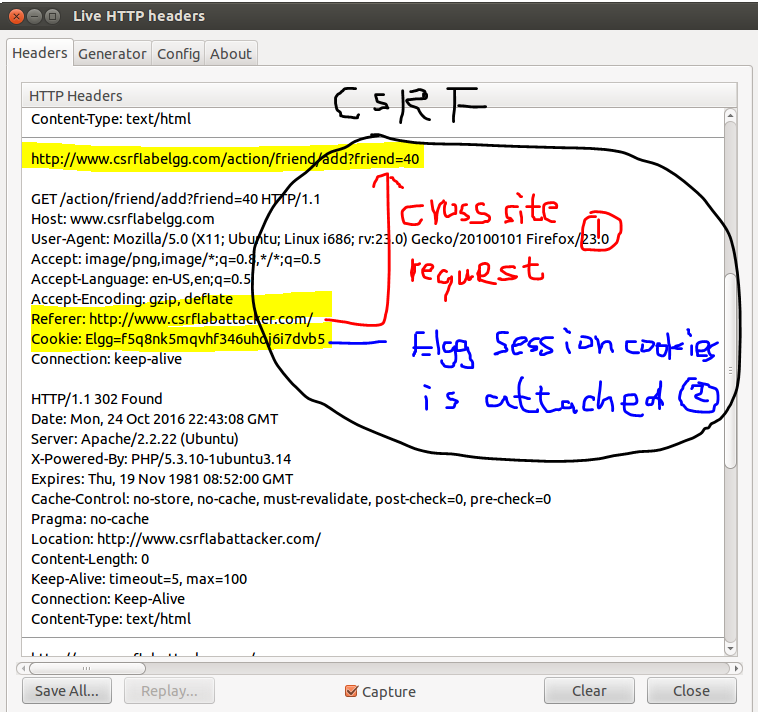
* 1. Find Alice’s session cookie before loading the malicious website



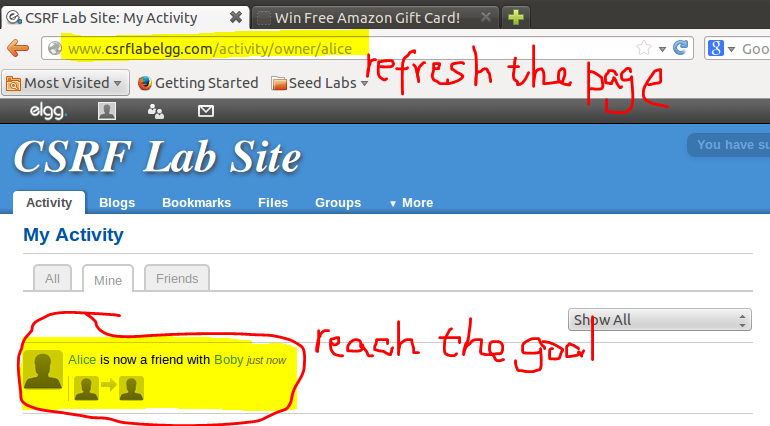
* 1. Hacking! In the same browser, open the second tab and type the following address. You need to use Live HTTP Header to capture the header’s information



* 1. Display the capture index.html header information



* 1. Refresh the page. Alice added Bob as a friend.



**Task 3: Add Charlie as a Friend of Alice Using GET Request**

Assume user name and password of Charlie is as follows:

User Name: charlie Password: seedcharlie

# Reference:

* http://www.cis.syr.edu/~wedu/seed/lab\_env.html
* http://www.cis.syr.edu/~wedu/seed/Labs\_12.04/Web/Web\_CSRF\_Elgg/Web\_CSRF\_Elgg.pdf