

## Overview:

Cross-site scripting (XSS) is a type of vulnerability commonly found in web applications. This vulnerability makes it possible for attackers to inject malicious code (e.g. JavaScript programs) into the victim's web browser. Using this malicious code, attackers can steal a victim's credentials, such as session cookies. The access control policies (i.e., the same origin policy) employed by browsers to protect those credentials can be bypassed by exploiting XSS vulnerabilities.

## Setup instruction:

1. Switch to "seed" user  
`$ sudo su seed`
2. Go to Web\_XSS\_Elgg lab setup folder  
`$ cd ~/seed-labs/category-web/Web_XSS_Elgg/Labsetup`
3. You should be able to see following files  
`$ ls`

```
(seed@kali)-[~/seed-labs/category-web/Web_XSS_Elgg/Labsetup]
$ ls
docker-compose.yml  image_mysql  image_www
```

4. DNS Setup  
We have set up several websites for this lab. They are hosted by the container 10.9.0.5. We need to map the names of the web server to this IP address. Please add the following entries to '/etc/hosts'. You need to use the root privilege to modify this file:  
10.9.0.5 www.seed-server.com  
10.9.0.5 www.example32a.com  
10.9.0.5 www.example32b.com  
10.9.0.5 www.example32c.com  
10.9.0.5 www.example60.com  
10.9.0.5 www.example70.com
5. Container Setup and Commands  
To start both "www" and "mysql":  
`$ docker compose build # Build the container image`  
`$ docker compose up # Start the container`  
To stop both containers:  
`$ docker-compose down # Shut down the container`
6. Once you start both containers, please goto "<http://www.seed-server.com>", you will see:

Elgg For SEED Labs   Blogs   Bookmarks   Files   Groups   Members   More ▾   Search   Log in

Welcome

Welcome to your Elgg site.

Tip: Many sites use the [activity](#) plugin to place a site activity stream on this page.

Log in

Username or email \*

Password \*

☐ Remember me   Log in

[Lost password](#)

## Elgg Web Application:

We use an open-source web application called Elgg in this lab. Elgg is a web-based social-networking application. It is already set up in the provided container images; its URL is <http://www.seed-server.com>. We use two containers, one running the web server (10.9.0.5) , and the other running the MySQL database (10.9.0.6). The IP addresses for these two containers are hardcoded in various places in the configuration, so please do not change them from the docker-compose.yml file.

- **MySQL database.** Containers are usually disposable, so once it is destroyed, all the data inside the containers are lost. For this lab, we do want to keep the data in the MySQL database, so we do not lose our work when we shutdown our container. To achieve this, we have mounted the mysql data folder on the host machine (inside Labsetup, it will be created after the MySQL container runs once) to the /var/lib/mysql folder inside the MySQL container. This folder is where MySQL stores its database. Therefore, even if the container is destroyed, data in the database is still kept. If you do want to start from a clean database, you can remove this folder:

```
$ sudo rm -rf mysql_data
```

- **User accounts.** We have created several user accounts on the Elgg server; the username and passwords are given in the following.

```
-----
UserName | Password
-----
alice    | seedalice
boby     | seedboby
charlie  | seedcharlie
Samy     | seedsamy
-----
```

## Lab Tasks:

### Task 1: Posting a Malicious Message to Display an Alert Window

The objective of this task is to embed a JavaScript program in your Elgg profile, such that when another user views your profile, the JavaScript program will be executed and an alert window will be displayed. The following JavaScript program will display an alert window:

```
<script>alert('XSS');</script>
```

If you embed the above JavaScript code in your profile (e.g. in the brief description field), then any user who views your profile will see the alert window. In this case, the JavaScript code is short enough to be typed into the short description field. If you want to run a long JavaScript, but you are limited by the number of characters you can type in the form, you can store the JavaScript program in a standalone file, save it with the .js extension, and then refer to it using the "src" attribute in the "<script>" tag. See the following example:

```
<script type="text/javascript"
      src="http://www.example.com/myscripts.js">
</script>
```

In the above example, the page will fetch the JavaScript program from <http://www.example.com>, which can be any web server.

### **Task 2: Posting a Malicious Message to Display Cookies**

The objective of this task is to embed a JavaScript program in your Elgg profile, such that when another user views your profile, the user's cookies will be displayed in the alert window. This can be done by adding some additional code to the JavaScript program in the previous task:

```
<script>alert(document.cookie);</script>
```

### **Submission**

You need to submit a detailed lab report, with screenshots, to describe what you have done and what you have displayed. Please also list the important code snippets followed by explanation. Simply attaching code without any explanation will not receive credits.